



AKADEMIA TECHNICZNO-ROLNICZA  
IM. JANA I JĘDRZEJA ŚNIADECKICH  
W BYDGOSZCZY

**ROZPRAWY NR 119**

Anna Ławrynowicz

**ZARZĄDZANIE PRODUKCJĄ  
W OGNIWIE SIECI DOSTAW  
Z ZASTOSOWANIEM  
SYSTEMU EKSPERCKIEGO  
I ALGORYTMU GENETYCZNEGO**

BYDGOSZCZ – 2006

REDAKTOR NACZELNY  
prof. dr hab. Lucyna Drozdowska

OPINIODAWCY  
prof. dr hab. inż. Wojciech Cellary  
dr hab. inż. Marek Fertsch, prof. nadzw. PP

OPRACOWANIE REDAKCYJNE I TECHNICZNE  
mgr Michał Górecki, mgr inż. Daniel Morzyński

© Copyright  
Wydawnictwa Uczelniane Akademii Techniczno-Rolniczej  
Bydgoszcz 2006

ISSN 0209-0597

Wydawnictwa Uczelniane Akademii Techniczno-Rolniczej  
ul. Ks. A. Kordeckiego 20, 85-225 Bydgoszcz, tel. (052) 3749482, 3749426  
e-mail: [wydawucz@atr.bydgoszcz.pl](mailto:wydawucz@atr.bydgoszcz.pl) <http://www.atr.bydgoszcz.pl/~wyd>

---

Wyd. I. Nakład 150 egz. Ark. aut. 11,2. Ark. druk. 14,1. Zamówienie nr 1/2006  
Oddano do druku i druk ukończono w styczniu 2006 r.  
Uczelniany Zakład Małej Poligrafii ATR Bydgoszcz, ul. Ks. A. Kordeckiego 20

## Spis treści

<b>1. Wstęp</b> .....	7
1.1. Geneza pracy .....	7
1.2. Cel i zakres pracy .....	8
<b>2. Klasyczne metody zarządzania produkcją</b> .....	12
2.1. Wprowadzenie do zarządzania produkcją .....	12
2.1.1. System produkcyjny, przedsiębiorstwo, proces produkcyjny ....	12
2.1.2. Zarządzanie produkcją w przedsiębiorstwie przemysłowym ....	13
2.2. Przedsiębiorstwo produkcyjne ogniwem sieci dostaw .....	17
2.2.1. Łańcuch dostaw .....	17
2.2.2. Zarządzanie łańcuchem dostaw .....	21
2.2.3. Model SCOR .....	22
<b>3. Koncepcje systemów zarządzania produkcją</b> .....	27
3.1. Systemy MRP i ERP .....	27
3.2. Koncepcja JIT .....	29
3.3. Koncepcja LM .....	31
3.4. Koncepcja OPT .....	32
3.5. Koncepcja BOA .....	32
3.6. Koncepcja FZS .....	32
3.7. Zaawansowane systemy planowania i optymalizacji łańcucha dostaw typu APS .....	32
<b>4. Harmonogramowanie produkcji</b> .....	35
4.1. Klasyfikacja zadań harmonogramowania .....	35
4.2. Kryteria oceny harmonogramów .....	41
4.3. Metody poszukujące dokładnych rozwiązań problemów harmono- gramowania.....	45
4.3.1. Przeszukiwanie zupełne i przeszukiwanie losowe .....	45
4.3.2. Programowanie całkowitoliczbowe .....	47
4.3.3. Metoda podziałów i ograniczeń .....	47
4.4. Heurystyczne metody budowy harmonogramów .....	48
4.5. Reguły priorytetu w harmonogramowaniu produkcją .....	51
<b>5. Systemy eksperckie</b> .....	53
5.1. Geneza sztucznej inteligencji .....	53
5.2. Reprezentacja wiedzy w systemach eksperckich .....	53
5.2.1. Wiedza proceduralna i deklaratywna .....	53
5.2.2. Reprezentacja wiedzy w przestrzeni stanów .....	54
5.2.3. Systemy produkcji .....	55

5.2.4. Sieci semantyczne .....	57
5.2.5. Ramy .....	57
5.2.6. Logika .....	59
5.3. Struktura systemu eksperckiego .....	63
5.4. Architektura tablicowa systemu eksperckiego .....	65
5.5. Proces tworzenia systemu eksperckiego .....	67
<b>6. Systemy eksperckie w zarządzaniu produkcją .....</b>	<b>70</b>
6.1. Zastosowanie systemu eksperckiego do podejmowania decyzji wytwórz / kup .....	70
6.2. System ISIS .....	73
6.3. System COSIMA .....	73
6.4. System ekspercki nadzorowania produkcji z architekturą tablicową ...	75
<b>7. Algorytmy ewolucyjne .....</b>	<b>82</b>
7.1. Wprowadzenie .....	82
7.2. Ogólna struktura algorytmu ewolucyjnego .....	83
7.3. Reprezentacja .....	85
7.4. Metody selekcji .....	86
7.5. Operatory krzyżowania i mutacji .....	87
7.6. Funkcja przystosowania .....	89
<b>8. Zastosowanie algorytmów ewolucyjnych do zarządzania produkcją ....</b>	<b>90</b>
8.1. Zastosowanie algorytmów genetycznych do konfigurowania sieci dostaw .....	90
8.1.1. Zastosowanie algorytmu genetycznego do równoważenia alokacji klientów do centrów dystrybucji w sieci dostaw .....	90
8.1.2. Analiza wielopoziomowego logistycznego łańcucha dostaw z zastosowaniem algorytmu genetycznego .....	96
8.2. Zastosowanie algorytmów ewolucyjnych do planowania i harmonogramowania produkcji .....	102
8.2.1. Reprezentacja bazująca na operacjach .....	106
8.2.2. Reprezentacja bazująca na zadaniach .....	107
8.2.3. Reprezentacja bazująca na listach preferencyjnych .....	108
8.2.4. Reprezentacja bazująca na parach zadań .....	109
8.2.5. Reprezentacja bazująca na regułach priorytetu .....	110
8.2.6. Reprezentacja bazująca na grafie dysjunkcyjnym .....	113
8.2.7. Reprezentacja bazująca na terminie zakończenia .....	114
8.2.8. Reprezentacja bazująca na maszynach .....	114
8.2.9. Reprezentacja losowa .....	115
8.2.10. Symbiotyczny algorytm ewolucyjny .....	115
8.2.11. Zaawansowane planowanie i harmonogramowanie z outsourcingiem .....	124

<b>9. Proponowane podejście do zarządzania produkcją w ogniwie sieci dostaw</b> .....	138
9.1. Przesłanki poszukiwania nowych metod zarządzania produkcją .....	138
9.2. Cele i zakres zastosowań proponowanej metody zarządzania produkcją w ogniwie sieci dostaw .....	141
9.3. Prezentacja nowej metody zarządzania produkcją w ogniwie sieci dostaw .....	142
9.4. System ekspercki planowania i sterowania produkcją .....	144
9.4.1. Struktura systemu eksperckiego .....	144
9.4.2. Zewnętrzna baza danych systemu eksperckiego .....	145
9.4.3. Działanie systemu eksperckiego .....	147
9.5. Algorytm genetyczny harmonogramowania produkcji .....	149
9.5.1. Struktura algorytmu genetycznego .....	150
9.5.2. Reprezentacja i tworzenie harmonogramu .....	150
9.5.3. Tworzenie populacji początkowej .....	150
9.5.4. Operatory krzyżowania i mutacji .....	151
9.5.5. Funkcja przystosowania .....	151
9.6. Implementacja nowej metody zarządzania produkcją .....	153
9.6.1. Implementacja systemu eksperckiego .....	153
9.6.2. Implementacja algorytmu genetycznego .....	158
<b>10. Uwagi dotyczące cyklu życia systemu zarządzania produkcją z zastosowaniem systemu eksperckiego i algorytmu genetycznego</b> ....	163
<b>11. Badania eksperymentalne</b> .....	169
11.1. Opis badanych systemów produkcyjnych .....	169
11.2. Strojenie algorytmu genetycznego .....	171
11.3. Wyniki badań .....	180
11.4. Badania porównawcze algorytmu genetycznego z algorytmami bazującymi na kombinacji reguł priorytetu .....	182
11.5. Analiza i ocena końcowa .....	184
<b>12. Wnioski</b> .....	188
<b>Literatura</b> .....	192
<b>Streszczenia</b> .....	205
<b>Aneks. Plany produkcji</b> .....	209

## 1. Wstęp

### 1.1. Geneza pracy

Podjęcie tematu uzasadniają poniżej przedstawione główne przesłanki.

Zarządzanie produkcją obejmuje coraz to szersze obszary podejmowania decyzji i coraz częściej rozszerza się na zarządzania siecią dostaw, w związku z czym w rozważaniach dotyczących planowania i sterowania produkcją przedsiębiorstwo produkcyjne nie stanowi obecnie centralnego podmiotu rozważań, a jedynie stanowi ogniwo sieci dostaw. W takim podejściu do zarządzania produkcją, szczególną rolę odgrywa integracja uczestników sieci dostaw, a w przypadku systemów produkcyjnych wydzielanie części ich wewnętrznych procesów i realizowanie w zewnętrznych systemach. Należy więc poszukiwać takich metod usprawniania zarządzania produkcją, które obejmują outsourcing.

Wraz z nowymi tendencjami w zarządzaniu produkcją powstała potrzeba opracowania nowych podstaw metodologicznych dla zarządzania produkcją w ogniwie sieci dostaw. W szczególności, jak wykazano w niniejszej pracy konieczne jest uściślenie definiowania pojęcia sieci dostaw, ponieważ aktualne modele sieci dostaw proponowane w literaturze przedmiotu tylko wybiórczo odwzorowują rzeczywiste procesy w sieci dostaw. W związku z tym, istnieje potrzeba opracowania nowych metod zarządzania produkcją w ogniwie sieci dostaw, ponieważ znane i stosowane dotychczas metody nie znajdują zastosowania w wielu obiektach rzeczywistych i nie przynoszą oczekiwanych korzyści.

Kluczowym problemem w zarządzaniu produkcją jest opracowanie metod optymalizujących planowanie i harmonogramowanie produkcji. Ze względu na konkurencję kryteria optymalizacji powinny z jednej strony satysfakcjonować klienta, a z drugiej pozwolić przedsiębiorstwu zdobywać przewagę rynkową. Z tego powodu najczęściej przyjmowanym kryterium optymalizacji planowania i harmonogramowania produkcji jest dotrzymanie terminu realizacji zleceń klientów i minimalizacja całkowitego czasu zakończenia wszystkich realizowanych w systemie produkcyjnym zadań. Takie podejście do problemu przyjęto w niniejszej rozprawie, ponieważ daje ono wiele korzyści. Pozwala na skrócenie czasu oczekiwania klienta na realizację zamówienia oraz obniża koszty produkcji, na skutek zmniejszenia stanu robót w toku.

Dążenie człowieka do perfekcji znajduje swój wyraz w optymalizacji. Oceniając wartość metod i procedur optymalizacyjnych z zakresu zarządzania produkcją nie można koncentrować się jedynie na tym, czy metoda osiąga optimum, należy brać także pod uwagę efektywność procesu jego poszukiwania. Algorytmy dokładne – ze względu na dużą liczbę zmiennych, nierówności i równań oraz stopień skomplikowania formalnego zapisu – nie nadają się do wykorzystania w praktyce, gdyż wymagają długiego czasu obliczeń i korzystania z superszybkich komputerów, szczególnie dla problemów o dużych rozmiarach. Z tego powodu zdecydowana większość opisanych w literaturze metod nie

nadaje się do zastosowania w praktyce. Innym powodem małej przydatności opublikowanych metod planowania i harmonogramowania produkcji jest przyjmowanie przez ich autorów wielu ograniczeń, co powoduje, że metody te mogą być zastosowane tylko w wybranych przypadkach, na ogół nie wymagających stosowania złożonych procedur obliczeniowych.

Z doniesień literaturowych i doświadczeń autorki można wysnuć wniosek, że wdrożenia systemów ze sztuczną inteligencją do zarządzania produkcją są nieliczne. Badania w tym zakresie dotyczą głównie eksperymentów z wykorzystaniem danych testowych. Powodem nielicznych wdrożeń jest złożoność aplikacji, wynikająca z interdyscyplinarności zagadnienia. Wymagana jest bowiem w takich przypadkach dobra znajomość problematyki zarządzania produkcją i inżynierii produkcji, przygotowanie informatyczne, a także śledzenie – szybkiego w ostatnich latach – rozwoju metod sztucznej inteligencji. Ważne znaczenie szczególnie w tworzeniu systemów eksperckich ma doświadczenie praktyczne.

Mimo, że doniesienia literaturowe o zastosowaniach sztucznej inteligencji już od dawna motywowały do podejmowania badań z tego zakresu, to największą inspiracją prezentowanych w niniejszej rozprawie badań było wyzwanie Dawida E. Goldberga, które można znaleźć w pracy [73]: „Konstruktorzy systemów – inżynierskich, komputerowych, jak też ekonomicznych – mogą tylko podziwiać odporność, wydajność i łatwość przystosowania się systemów biologicznych. Zdolność do regeneracji, samosterowania i reprodukcji, będąca regułą w świecie biologii, jest prawie nieobecna w świecie najbardziej nawet zaawansowanej techniki”. Takie podejście do problemów optymalizacji ostatecznie skłoniło autorkę do zastosowania metod sztucznej inteligencji w zarządzaniu produkcją.

## 1.2. Cel i zakres pracy

Rozprawa poświęcona jest udoskonaleniu zarządzania produkcją. Przegląd powyższej tematyki w zakresie perspektyw przedstawiony w rozprawie wskazuje, że problemy dotyczące optymalizacji planowania i sterowania produkcją wymagają dalszych systematycznych badań. Potrzeba ta wynika zarówno z luk metodologicznych, jak i z zastosowań praktycznych.

Celem rozprawy było opracowanie nowych metod i procedur zarządzania produkcją w ogniwie sieci dostaw o szybkozmiennym asortymencie produkcji, z outsourcingiem i z zastosowaniem metod sztucznej inteligencji. Zaproponowane przez autorkę metody i procedury wraz z przeprowadzoną krytyczną dyskusją wykorzystania istniejących algorytmów tworzą spójne metodyczne podstawy dla zintegrowanego podejścia do zarządzania produkcją, w dyskretnym systemie produkcyjnym przedsiębiorstwa przemysłowego. W związku z tym, sformułowano następujące cele szczegółowe rozprawy:

1. Uogólnienie teorii zarządzania produkcją.
2. Dokonanie krytycznego przeglądu zastosowań sztucznej inteligencji w zarządzaniu produkcją.
3. Zaproponowanie nowych metod i procedur zarządzania produkcją z zastosowaniem sztucznej inteligencji. Z metodologicznego i praktycznego punktu widzenia istotnym rozszerzeniem modeli zarządzania produkcją jest uwzględnienie zadań podzielonych i outsourcingu.
4. Sformułowanie zasad budowy zintegrowanych systemów zarządzania produkcją ze sztuczną inteligencją.

Praca składa się z dwunastu rozdziałów. Rozdziały od drugiego do jedenastego dotyczą kolejnych celów szczegółowych rozprawy.

W drugim rozdziale zawarto krytyczny przegląd podstawowych (z punktu widzenia problematyki dyskutowanej w rozprawie) pojęć związanych z zarządzaniem produkcją. Wkraczając w obszar wiedzy związanej z zarządzaniem produkcją trzeba było określić, czego ono dotyczy i na czym polegają jego związki z innymi obszarami działalności przedsiębiorstwa. Równocześnie krótko omówiono uwzględnianie niespójności opisu. Ostatecznie stwierdzono, że pojęcie *zarządzanie produkcją* obejmuje coraz to szersze obszary podejmowania decyzji i coraz częściej rozszerza się na zarządzania siecią dostaw. W takim podejściu do zarządzania produkcją szczególną rolę odgrywa integracja uczestników sieci dostaw, a w przypadku systemów produkcyjnych (przedsiębiorstw) outsourcing.

Rozdział trzeci to krytyczna dyskusja nad koncepcją zarządzania produkcją. Rozważania tego rozdziału wykazały, że nie opracowano dotychczas takiej koncepcji modelu zarządzania produkcją, która mogłaby znaleźć zastosowanie w systemach produkcyjnych o szybkozmiennym asortymencie produkcji i mającej na celu optymalizację dwóch kryteriów: zaspokojenie zmiennych oczekiwań klienta na coraz to nowsze, lepsze produkty, w możliwie najkrótszym czasie i przy możliwie najniższych kosztach. Przy budowie takiego systemu należy wziąć pod uwagę zarówno dotrzymywanie terminów zleceń klienta, jak i wyrównywanie obciążeń maszyn. Z takiego punktu widzenia można przyjąć, że koncepcja OPT (*Optimized Production Technology*), w której za nadrzędne kryterium sterowania zleceniami produkcyjnymi przyjmuje się możliwie maksymalne wykorzystanie tych stanowisk wytwórczych, na których występuje spiętrzenie zadań wytwórczych może stanowić pewną fazę rozwiązywania problemu zarządzania produkcją w systemach o niestabilizowanej produkcji.

W rozdziale czwartym przedstawiono klasyfikację problemów szeregowania zadań z punktu widzenia rodzaju zadań, zasobów oraz kryteriów optymalizacji, wprowadzono niezbędne definicje i pojęcia teorii szeregowania zadań. Na tym tle przedstawiono analizę złożoności obliczeniowej wybranych problemów dyskretnych. Przegląd dyskretnych modeli szeregowania zadań przedstawiony w tym rozdziale wykazał, że problemy opisane za pomocą metod dokładnych nie znajdują zastosowań praktycznych ze względu na zbyt długie czasy



obliczeń. Natomiast w modelach przybliżonych harmonogramowania produkcji przyjmuje się zbyt wiele ograniczeń, w związku z czym nie nadają się one do zastosowania w praktyce.

W rozdziale piątym przeanalizowano nowe sposoby widzenia problemów kładących nacisk na wiedzę, w przeciwieństwie do formalnych metod rozumowania. Jest to konsekwencją podejść, w których „... paradygmat systemów eksperckich pochodzi z wiedzy jaką one posiadają, a nie z formalizmów i schematów wnioskowania, jakie stosują. Wiedza eksperta jest kluczem do systemu, podczas gdy reprezentacja wiedzy i schematy wnioskowania dostarczają mechanizmów do jej użycia” [81]. Duży nacisk położono na sposoby reprezentacji wiedzy w sztucznej inteligencji, którymi różnią się one od innych modeli systemów.

W rozdziale szóstym przedstawiono syntetyczny przegląd niektórych osiągniętych rezultatów dotyczących zastosowań systemów eksperckich w zarządzaniu produkcją oraz dyskusję aspektów metodycznych wynikających z zastosowań praktycznych. W rozdziale tym – na tle przeglądu dotychczasowych propozycji – zaprezentowano niektóre możliwości zastosowań systemów eksperckich w zarządzaniu produkcją.

Rozdział siódmy to syntetyczne wprowadzenie do problematyki algorytmów ewolucyjnych, które w dalszej części rozważano pod kątem ich adaptacji do zastosowań praktycznych.

W literaturze poświęconej algorytmom ewolucyjnym dominują rozważania natury teoretycznej. Zdecydowani rzadziej są przedstawiane przykłady praktycznych zastosowań. Głównym celem rozdziału ósmego jest prezentacja tych rozwiązań w zakresie algorytmów genetycznych, które mogą mieć zastosowania praktyczne. Pozwoliło to uzyskać lepszy wgląd w typowe dla algorytmów genetycznych metody reprezentacji i struktury powstałych systemów.

W rozdziale dziewiątym, korzystając z rozwiązań przedstawionych w rozdziałach szóstym i ósmym zaproponowano nową metodę zarządzania nieustabilizowaną produkcją w ogniwie sieci dostaw z zastosowaniem sztucznej inteligencji. Z metodologicznego i praktycznego punktu widzenia istotnym rozszerzeniem podejść do problemu planowania i sterowania produkcją jest uwzględnienie przez autorkę zadań podzielonych i outsourcingu. Proponując nowe metody zarządzania produkcją zwrócono uwagę na to, że z matematycznego punktu widzenia używane powszechnie, a więc i w tej pracy, pojęcie *sieci dostaw* nie jest tożsame z pojęciem *sieci* rozumianym w matematyce jako skierowany, skończony graf bez pętli i łuków równoległych (por. [23]). Uwzględniając aktualne tendencje rozwojowe sieci dostaw, dotyczące wyrównywania obciążeń zasobów, do modelowania *sieci dostaw* zaproponowano graf. W związku z tym, że aktualnie znane metody zarządzania produkcją oparte na modelach *sieci* (w rozumieniu matematycznym) nie odpowiadają tendencjom rozwojowym *sieci dostaw*, do zarządzania produkcją w ogniwie sieci dostaw zaproponowano system ekspercki i algorytm genetyczny.

Rozdział dziesiąty zawiera uwagi dotyczące metodyki wdrażania nowego systemu zarządzania produkcją w ogniwie sieci dostaw z zastosowaniem systemu eksperckiego i algorytmu genetycznego.

Rozdział jedenasty zawiera dyskusję nad zastosowaniem (przedstawionej w rozdziale dziewiątym) nowej metody zarządzania produkcją do problemów praktycznych. W celu ilustracji takich możliwości rozważa się dane z rzeczywistych systemów produkcyjnych i wyniki odpowiednich eksperymentów obliczeniowych. Ponadto rozdział ten zawiera także dyskusję aspektów metodycznych wynikających z zastosowań praktycznych systemów eksperckich i algorytmów genetycznych.

Osiągnięte wyniki podsumowano w postaci wniosków w rozdziale dwunastym. W tym rozdziale przedstawiono przede wszystkim wkład autorki niniejszej rozprawy w rozwój podstaw metodologicznych zarządzania produkcją. Ponadto pokazano korzyści z zastosowania nowego – opracowanego w ramach prezentowanych tu badań – podejścia do zarządzania produkcją. Wskazano także dalsze kierunki badań w zakresie zarządzania siecią dostaw z zastosowaniem systemów eksperckich i algorytmów genetycznych.

## 2. Klasyczne metody zarządzania produkcją

### 2.1. Wprowadzenie do zarządzania produkcją

#### 2.1.1. System produkcyjny, przedsiębiorstwo, proces produkcyjny

Obiektem rozważań w niniejszej pracy jest system produkcyjny przedsiębiorstwa przemysłowego. *System produkcyjny* stanowi celowo zaprojektowany i zorganizowany układ materialny, energetyczny oraz informacyjny eksploatowany przez człowieka i służący produkowaniu określonych produktów (wrobów lub usług) w celu zaspokojenia różnorodnych potrzeb konsumentów [31, 55]. System produkcyjny składa się z następujących, podstawowych elementów [57]:

- 1) wektora wejścia X, w skład którego wchodzi wszystkie czynniki produkcji,
- 2) wektora wyjścia Y, w skład którego wchodzi wyroby, usługi, a także szkodliwe odpady produkcyjne, zanieczyszczające środowisko,
- 3) procesów przetwarzania wektora wejścia w wektor wyjścia T, nazywanych zwykle procesami produkcyjnymi,
- 4) procesów zarządzania systemem,
- 5) sprzężeń materialnych, energetycznych oraz informacyjnych pomiędzy wymienionymi elementami składowymi systemu produkcyjnego.

Pierwsze trzy elementy wraz ze sprzężeniami materialnymi, energetycznymi i informacyjnymi są nazywane *podsystemem przetwarzania* lub *produkcji*; natomiast proces zarządzania wraz ze sprzężeniami informacyjnymi określa się jako *podsystem zarządzania* [57].

Proces transformacji, czyli przekształcanie wektora wejścia systemu produkcyjnego w wektor wyjścia nazywa się *procesem produkcyjnym*. Proces produkcyjny definiuje się także jako uporządkowany ciąg działań (lub zbiór powiązanych działań), w wyniku którego konsument (użytkownik) otrzymuje produkty (wroby lub usługi) [55]. Elementem procesu produkcyjnego jest proces wytwórczy. *Proces wytwórczy* dotyczy wytwarzania produktu, czyli przetwarzania czynników produkcji w produkty (wroby lub usługi) [55].

System produkcyjny, rozpatrywany w rzeczowym ujęciu, obejmuje [58]: techniczne środki produkcji, przedmioty pracy, czynniki energetyczne, czynnik ludzki, zbiory informacji, środki finansowe, wroby. Systemem produkcyjnym jest więc *przedsiębiorstwo* jako podstawowy podmiot gospodarczy, którego zadaniem jest produkcja określonych dóbr i świadczenie usług (por. [96]). Przedsiębiorstwo stanowi wyodrębniony prawnie i finansowo podmiot gospodarczy działający co najmniej w siedmiu sferach [57]:

- badań i rozwoju,
- marketingu, przygotowania i zakupu materiałów oraz półwrobów do wytwarzania,
- przetwarzania czynników produkcji w wroby gotowe i usługi zwane powszechnie produktami,

- dystrybucji sprzedaży i obsługi serwisowej klientów,
- utylizacji i recyklingu produktów wytwarzanych przez siebie lub przez innych wytwórców,
- szkolenia i przygotowania personelu do zarządzania zmianami,
- zarządzania kapitałem dla osiągnięcia najwyższej jego produktywności.

Za wyodrębniony system produkcyjny w skrajnym przypadku można uważać nawet pojedyncze stanowisko robocze [55].

### 2.1.2. Zarządzanie produkcją w przedsiębiorstwie przemysłowym

Wkraczając w obszar wiedzy związanej z zarządzaniem produkcją trzeba określić, czego dotyczy zarządzanie produkcją, na czym ono polega i jakie są jego związki z innymi pojęciami.

Pojęcie *zarządzanie produkcją* mimo częstego używania w literaturze ([np. 41, 53, 58, 61, 103, 144, 156]) podobnie jak terminy *planowanie i sterowanie przepływem produkcji* [26] (lub jego synonim *planowanie i sterowanie przebiegiem produkcji*) nie zostały jeszcze jednoznacznie zdefiniowane. W literaturze funkcjonuje wiele podobnych pojęć, tj. *planowanie i sterowanie produkcją* [56, 60, 194], *planowanie i harmonogramowanie produkcji* [146].

Zarządzanie produkcją (*production management*) to problematyka bardzo obszerna, różnorodna oraz na ogół różnie określana.

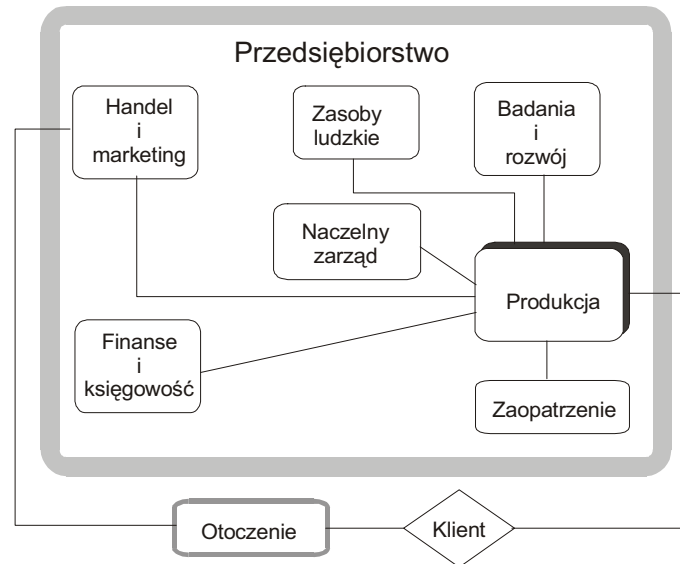
Za *American Production and Inventory Control Society Dictionary* – APICS<sup>1</sup> *zarządzanie produkcją* definiuje się jako: wiedzę i działalność praktyczną związaną z planowaniem, bieżącym sterowaniem i kontrolą ilości wyrobów gotowych, robót w toku i surowców oraz rozmiarem wykorzystywanych zasobów dla zaspokojenia zapotrzebowania klientów, minimalizacji kosztów, opóźnień, zapasów oraz maksymalizacji produktywności i pośrednio – maksymalizacji zysków i zwrotu z zainwestowanego kapitału.

Tradycyjne funkcje zarządzania produkcją, według G. Doumeingts'a [53] to: planowanie obejmujące także planowanie biznesowe, harmonogramowanie główne, planowanie zdolności produkcyjnych, harmonogramowanie szczegółowe oraz powiązania z planowaniem zapotrzebowania na materiały, zaopatrzeniem, sprzedażą i zarządzaniem zasobami.

Funkcje produkcji są zintegrowane z wieloma innymi funkcjami przedsiębiorstwa. Powiązania te zilustrowano na rysunku 2.1.

---

<sup>1</sup> VIII<sup>th</sup> edition, American Production and Control Society, Inc., Falls Church, VA 1995 w: [62].



Rys. 2.1. Funkcje przedsiębiorstwa oraz ich powiązanie z funkcjami produkcji [53]

Według R. Knosali i jego zespołu [104] zarządzanie produkcją obejmuje szereg działań podejmowanych przed oraz w czasie wytwarzania określonych wyrobów. Do działań tych należą: prognozowanie zbytu, prognozowanie zapotrzebowania na materiały, prognozowanie kosztów, projektowanie organizacji procesu produkcyjnego, tworzenie harmonogramu produkcji, kontrolowanie wykonania planu, korygowanie odchyłeń.

W zarządzaniu produkcją można wyróżnić – ze względu na zakres rzeczowy i horyzont czasowy – trzy poziomy (por. [61]): strategiczny, taktyczny, operatywny.

Pierwszy poziom zarządzania produkcją – strategiczny – dotyczy długiego horyzontu czasowego i obejmuje zagadnienia wyboru celów strategicznych przedsiębiorstwa w zakresie produkcji, określanie strategii produkcyjnych i wyznaczanie długoterminowych kierunków zmian w potencjale przedsiębiorstwa. Podstawowym celem na tym poziomie jest osiągnięcie i utrzymanie konkurencyjności produkowanych wyrobów i procesów produkcji.

Jako drugi wyodrębniony został poziom taktycznego zarządzania produkcją. Dotyczy on krótszego niż poziom strategiczny horyzontu czasowego, a rozstrzygane w jego obszarze zagadnienia to dobór szczegółowych rozwiązań w zakresie technologii i organizacji produkcji oraz takie kształtowanie struktury procesu produkcyjnego, aby zapewnić realizację zadań wyznaczonych na poziomie strategicznym

Trzecim poziomem jest operatywne zarządzanie produkcją. Horyzont jego działania zmienia się w zależności od cech wytwarzania wyrobów i seryjności produkcji. Operatywne zarządzanie produkcją koncentruje się na maksymalizacji

efektywności procesu produkcji. Odbywa się to przez planowanie zadań, bieżące zasilanie w potrzebne do produkcji surowce, materiały i podzespoły oraz sterowanie wszystkimi elementami systemu produkcyjnego dla zapewnienia realizacji zaplanowanych zadań.

W literaturze dotyczącej zarządzania produkcją operuje się także pojęciem systemu zarządzania produkcją. Według L. Dwilińskiego [58] system zarządzania produkcją jest to „...zbiór hierarchicznie powiązanych i wyposażonych w środki techniczne oraz posiadających uprawnienia kontrolne i decyzyjne osób lub komórek organizacyjnych, powołanych do prognozowania i wdrażania zmian w systemie produkcyjnym, planowania produkcji, jej przygotowania i organizowania zespołów pracowniczych, przewodzenia im, kierowania pracownikami i motywowania ich do twórczej pracy, sterowania produkcją, kontrolowania uzyskiwanych efektów i porównywania ich z planem oraz harmonogramem, koordynowania prac i, ewentualnie, dokonywania zmian w planach i harmonogramach produkcji. Chodzi także o ocenę sprawności i skuteczności zarządzania, doskonalenie systemu i metod zarządzania”.

Za REFA<sup>2</sup> (w [173]) można określić podstawowe funkcje planowania i sterowania produkcją:

„*Planowanie* jest to systematyczne poszukiwanie, klasyfikacja oraz ustalanie zadań produkcyjnych i środków do osiągnięcia postawionych celów”.

„*Sterowanie* jest to uruchamianie, nadzorowanie i zapewnianie realizacji zadań produkcyjnych z uwagi na ilość, termin, jakość, koszt i warunki pracy”.

Planowanie jest jedną z podstawowych funkcji zarządzania, występującą na wszystkich jego szczeblach. Proces planowania składa się z pięciu podstawowych etapów [55]:

- określenie celów,
- dokonanie analizy sytuacji, co sprzyja, a co przeszkadza w realizacji celu (planowanie właściwe),
- opracowanie sekwencji działań zmierzających do realizacji celu (planowanie właściwe),
- opracowanie harmonogramów realizacji,
- podjęcie decyzji dotyczących realizacji.

W polskiej literaturze można znaleźć inne rozumienia pojęcia *sterowania produkcją*, w których przez sterowanie produkcją rozumie się kompleks operacji procesów informacyjno-decyzyjnych, mających zapewnić realizację ustalonych dla danej komórki zadań, np. ujętych w formie planu produkcji wyrobów, zazwyczaj odniesionego do pewnego okresu (por. [179]). Procesy te obejmują realizację następujących funkcji [103, 179]:

- planowanie przebiegu procesu produkcyjnego,
- ewidencja produkcji,

---

<sup>2</sup> The REFA – Association for Work Design/Work, Structure, Industrial Organization and Corporate Development. [www.refa-international.de](http://www.refa-international.de)

- kontrolowanie, tzn. porównywanie zaplanowanego przebiegu procesu produkcyjnego z rzeczywistym, a także weryfikowanie dostępności zasobów warunkujących wykonanie procesu produkcyjnego zgodnie z założeniami,
- korygowanie operatywnego planu produkcji w trakcie realizacji.

W literaturze występuje także pojęcie zarządzania materiałami (*materials management*), które powstało w rezultacie podejść logistycznych do problemu „... i tradycyjnie ogranicza się do określonej firmy i związane jest to z usprawnieniami przepływu materiałów w procesie produkcji, redukcją zapasu surowca oraz części zapasowych i wybiórczo dotyczy magazynowania i transportu” [76].

W ogólnym ujęciu można przyjąć, że systemy planowania i sterowania produkcją (*Production Planning and Control*) – PPC obejmują wszystkie obszary funkcjonalne przedsiębiorstwa, związane z przepływem materiałów i informacji w produkcji. Systemy PPC są uważane za systemy nadrzędne w obiegu i wymianie danych nie tylko w planowaniu i sterowaniu produkcją. Systemy PPC pełnią też rolę integratorów danych dla technicznego przygotowania produkcji, gdyż dane przygotowane w środowisku CAD/CAP/CAM stanowią podstawowe parametry planowania i sterowania produkcją. Systemy PPC odgrywają również istotną rolę w organizacji produkcji według koncepcji CIM, gdzie są „nerwem” systemu wytwórczego we wszystkich obszarach przygotowania, planowania i realizacji produkcji [41]. Poprzez aktualizację zadań w realizacji poszczególnych zleceń, jak i aktualizację danych o stanie ich zaawansowania (meldunki zwrotne), system PPC steruje procesami informacyjnymi i technicznymi, które mają zasadnicze znaczenie dla podejmowania decyzji. Podstawowym zadaniem systemu PPC jest zatem takie oddziaływanie na proces produkcyjny (podstawowy i pomocniczy), które zapewniłoby realizację zadań ujętych w planie produkcji, w ustalonym przedziale czasu. Uwarunkowanie te są pochodną realizowanej strategii przedsiębiorstwa oraz rodzaju i skali produkcji.

Podsumowując można stwierdzić, że planowanie i sterowanie produkcją obejmuje całość działań, których celem jest ustalanie programów zbytu i produkcji, a także określenie wykonania tego programu (z punktu widzenia ilości produkowanych wyrobów i terminów realizacji). Zgodnie z tym określeniem w ramach planowania i sterowania produkcją należy określić następujące zmienne decyzyjne i ich wartości [71]:

- zapotrzebowanie na gotowe wyroby i odpowiednie wielkości produkcji,
- zlecenia produkcyjne,
- zamówienia u kooperantów i poddostawców,
- terminy realizacji zleceń i operacji produkcyjnych.

Posługiwanie się w literaturze różnymi pojęciami dotyczącymi zarządzania produkcją świadczy o dużym skomplikowaniu tego problemu i trudnościach w jego formalizacji. Rozważając problem zarządzania produkcją należy zwrócić szczególną uwagę na jego otoczenie. Współczesne zarządzanie produkcją przekracza granice przedsiębiorstwa, ponieważ przedsiębiorstwo na ogół jest tylko ogniwem łańcuchów dostaw.

## 2.2. Przedsiębiorstwo produkcyjne ogniwo sieci dostaw

### 2.2.1. Łańcuch dostaw

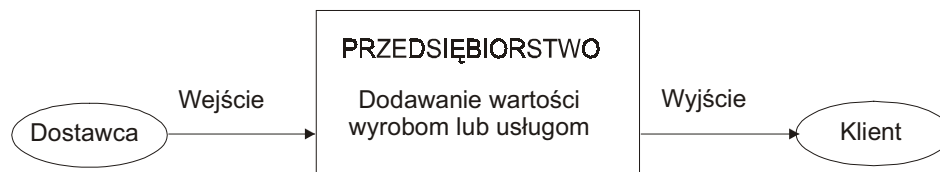
W systemach produkcyjnych realizowane są różnego rodzaju procesy. Najogólniej, proces jest ciągiem powiązanych ze sobą działań, które doprowadzają do przekształcenia wszelkich nakładów w produkt procesu [141].

Celem każdego procesu jest dostarczenie korzyści, a więc dodanie wartości wyrobowi lub usłudze. Dlatego wyodrębnić można łańcuch wartości, tj. ciąg procesów zmierzających bezpośrednio do klienta zewnętrznego. Pozostałe procesy nie tworzące wartości nazywa się *wspierającymi*.

Podstawowe cechy procesu (inaczej wymiary opisu) określają [141]:

- Kto jest wykonawcą? (wymiar podmiotowy)
- Co jest wynikiem procesu? (wymiar przedmiotowy)
- Jaka jest kolejność działań? (wymiar dynamiczny)
- Jakie są zależności pomiędzy procesami? (wymiar strukturalny)
- Jaki jest cel procesu? (wymiar użyteczności)

Ze względu na cel procesu i wykonawcę proces zarządzania produkcją można przedstawić jak na rysunku 2.2.



Rys. 2.2. Ogniwo łańcucha procesu [opracowanie własne]

Na wstępnym etapie identyfikacji procesu należy określić, gdzie się zaczyna proces zarządzania produkcją i gdzie kończy, z punktu widzenia użyteczności i podmiotu.

Struktura procesu nie jest obojętna dla efektywności ekonomicznej firmy. Istnieją następujące sytuacje dotyczące układu dostawca-odbiorca (klient końcowy):

- jeden dostawca – jeden odbiorca,
- jeden dostawca – wielu odbiorców,
- wielu dostawców – jeden odbiorca,
- wielu dostawców – wielu odbiorców – taką sytuację można sprowadzić do któregoś z powyższych układów.

Na procesy gospodarcze można patrzeć przez pryzmat przepływu dóbr od źródeł ich pozyskania z przyrody, aż do finalnych odbiorców. W całym przepływie można wyodrębnić wiele faz, a w nich z kolei poszczególne ogniwa. Podstawowe fazy mogą być następujące: pozyskanie surowców z przyrody, przetwórstwo surowców w materiały i półfabrykaty, produkcja półfabrykatów,



części i elementów, produkcja dóbr finalnych, konsumpcyjnych i produkcyjnych, handel środkami produkcji i konsumpcji, procesy usługowe, naprawy, remonty, pozyskanie odpadów i ich przepływ do sfer przetwórstwa.

Całość tych procesów w gospodarce tworzy sferę produkcji materialnej, a przepływ między poszczególnymi podmiotami tej sfery następuje w wyniku aktów kupna-sprzedaży na rynku, czyli ma charakter towarowo-pieniężny. Infrastruktura techniczna, a więc środki transportu, budynki i budowle magazynowe, urządzenia do składowania i manipulacji oraz opakowania transportowe umożliwiają przepływ.

Przepływ strumieni rzeczowych w całej gospodarce, a także w pojedynczym przedsiębiorstwie nie zawsze może być ciągły ze względów technicznych, organizacyjnych oraz ekonomicznych. W większości przypadków nie jest bowiem możliwe takie zsynchronizowanie tych procesów w czasie i przestrzeni, a także pod względem intensywności i rozmiarów, aby przerwy i zahamowania nie występowały. A to z kolei jest przyczyną zjawiska tworzenia zapasów, które występują w różnych ogniwach procesów przepływu i pełnią rozmaite funkcje. Przede wszystkim jednak powinny one tworzyć warunki stabilizacji procesów gospodarczych, tj. nie zakłócanego rytmu zaspokajania potrzeb konsumpcyjnych i produkcyjnych poszczególnych podmiotów. Tak więc powstawanie i utrzymywanie zapasów to także integralna część procesów logistycznych.

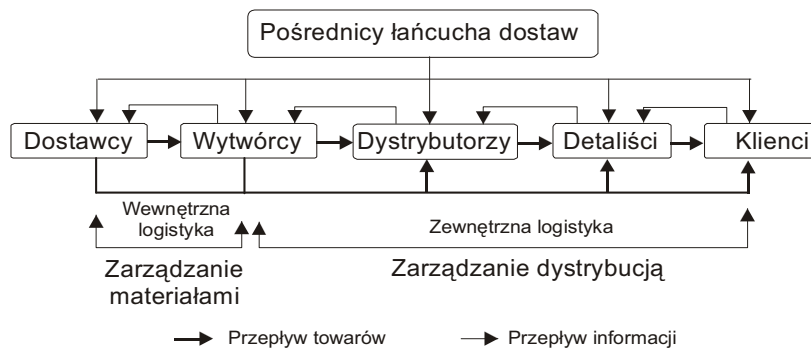
W ostatnich latach coraz częściej pojawia się w literaturze pojęcie *łańcucha dostaw* (*supply chain*) [16, 39, 59, 111, 170, 184, 193, 197]. Przez łańcuch dostaw rozumie się na ogół działalność związaną z przepływem materiału (towaru) od jego oryginalnego źródła – poprzez wszystkie formy pośrednie – aż do postaci, w której jest konsumowany przez ostatecznego klienta [2, 77].

*Łańcuch dostaw* jest referowany jako zintegrowany system, który synchronizuje cykle powiązanych procesów biznesowych żeby: nabyć surowe materiały i części, przekształcić te materiały i części w produkt finalny, dodać wartość tym produktom, rozprowadzić (dystrybucja) i wypromować te produkty detalistom albo innym klientom, ułatwić wymianę informacji pomiędzy różnymi biznesowymi obiektami, jak np. dostawcami, wytwórcami, dystrybutorami, pośrednikami i detalistami. Głównym celem tych procesów jest podnoszenie efektywności działań, opłacalności i pozycji konkurencyjnej firmy, a także jego partnerów w łańcuchu dostaw [48, 109, 152]. Łańcuch dostaw jest charakteryzowany przez przepływ towaru w przód i przepływ informacji wstecz [107, 152] (rys.2.3).

Optymalizacja łańcucha dostaw polega na optymalizacji zarówno funkcjonowania poszczególnych jego ogniw, jak i współdziałania pomiędzy nimi [22, 35, 63, 94, 114, 178, 201].

Zgodne z tym rozumieniem jest podejście J. Witkowskiego [202], który w oparciu o analizę podmiotowego, przedmiotowego i czynnościowego zakresu współpracy uczestników łańcucha dostaw zaproponował jego definicję i zwrócił uwagę na to, że łańcuch dostaw jest pojęciem szerszym niż łańcuch logistyczny.

Łańcuch dostaw to współdziałające ze sobą w różnych obszarach funkcjonalnych firmy wydobywcze, produkcyjne, handlowe, usługowe oraz ich klienci, między którymi przepływają strumienie produktów, informacji i środków finansowych.



Rys. 2.3. Proces łańcucha dostaw [152]

Typowy łańcuch dostaw składa się z dwóch głównych procesów biznesowych:

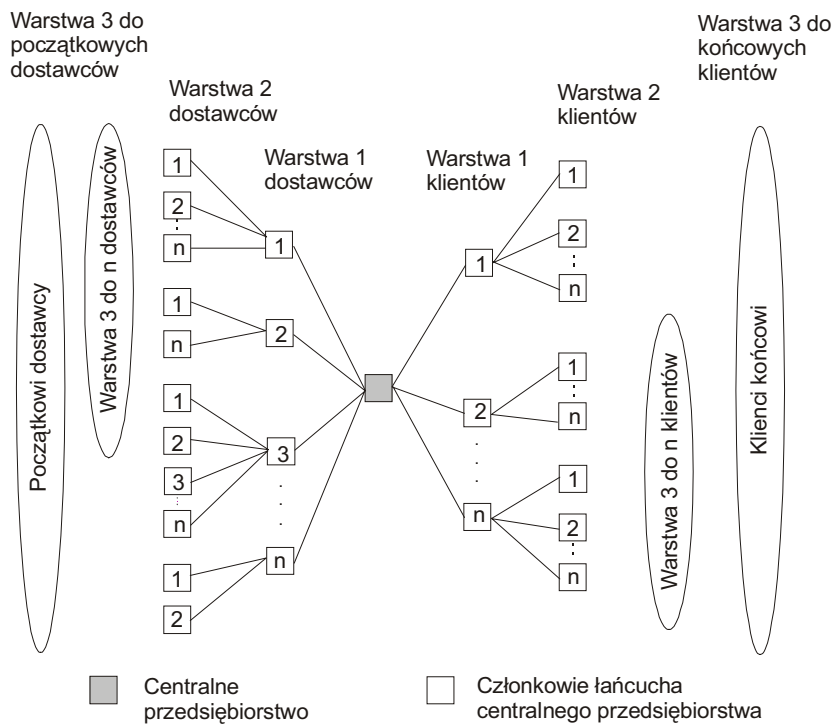
- zarządzania materiałami (wewnętrzna logistyka),
- dystrybucji fizycznej (zewnętrzna logistyka).

Zarządzanie materiałami dotyczy nabywania i magazynowania surowego materiału i części od dostawców. Obejmuje ono wspomaganie zarządzania materiałami w całym cyklu przepływu materiałów – od zakupu i sterowania produkcją materiałów, planowania i sterowania produkcją w toku, aż do hurtowni, transportu morskiego i dystrybucji finalnych produktów.

Fizyczna dystrybucja obejmuje wszystkie zewnętrzne czynności logistyczne związane z obsługą klientów. Te czynności obejmują przyjmowanie i przetwarzanie zleceń, rozmieszczanie zapasów, składowanie i manipulację, zewnętrzny transport, konsolidację, wycenę, wspomaganie promocji, manipulowanie zwracanymi produktami i podtrzymywanie cyklu życia. Wzdłuż łańcucha dostaw może być wielu uczestników – dostawców, wytwórców, dystrybutorów, pośredników, detalistów i klientów. Połączenie czynności zarządzania materiałami i fizyczną dystrybucją w pewien łańcuch dostaw nie daje jedynie reprezentacji liniowej powiązań łańcucha typu jeden-do-jeden, ale splot wielokrotnych powiązań sieci. W zasadzie pojęcie *sieci dostaw* jest szersze od tradycyjnie interpretowanych łańcuchów dostaw, w których następuje centralna koordynacja przepływu wraz z inicjowanym przez podmiot dominujący procesem integracji pionowej dostawców i odbiorców [45, 202]. Przez sieć logistyczną można rozumieć grupę niezależnych firm konkurujących i kooperujących w celu poprawy sprawności i efektywności przepływów produktów oraz towarzyszących im informacji zgodnie z oczekiwaniami klientów. Jeżeli do tych przepływów dodamy strumienie pieniężne, to bardziej zasadne będzie używanie terminu

*sieci dostaw* [202]. Zgodnie z teorią sieci tak rozumiane łańcuchy dostaw można uznać za szczególny przypadek sieci, składającej się z części centralnej i peryferyjnej. Ogólnie, koncepcja zarządzania łańcuchem dostaw jest więc rozwinięciem wizji klienta wokół centralnej korporacji, która kieruje zmianami więzi poprzez wewnątrz i na zewnątrz firmy oraz wychwytuje synergię międzyorganizacyjną i ją koordynuje. Dlatego sukces integracji procesów łańcucha dostaw zależy w dużym stopniu od dostępności dokładnych informacji „akurat na czas”, dzielonych na ogół przez członków łańcucha dostaw.

Konfiguracje sieci dostaw wskazują na występowanie bardzo zróżnicowanych układów hierarchicznych i rodzajów więzi między ich uczestnikami (por. [27]). Zrozumienie topologii (wymiaru struktury) łańcucha dostaw jest wstępnym etapem analizy i modelowania więzi łańcucha dostaw. Na ogół wyróżnia się dwa wymiary struktury łańcucha dostaw: poziomy (*horizontal*) i pionowy (*vertical*). Struktura pozioma referowana jest liczbą warstw (*tiers*) wzdłuż łańcucha dostaw. Łańcuch dostaw może być długi, z licznymi warstwami lub krótki, z niewielką liczbą warstw. Pionowa struktura łańcucha dostaw oznaczona jest liczbą dostawców i klientów reprezentowaną przez każdą warstwę jak pokazano na rysunku 2.4.



Rys. 2.4. Struktura sieci łańcucha dostaw [92, 109]

Tak więc, wzrost lub redukcja liczby dostawców i/lub klientów będzie zmieniać wymiary łańcucha dostaw. Outsourcing (włączający pośredników) lub funkcjonalność obrotu wyłączono także będzie zmieniać wymiar łańcucha dostaw przez jego wydłużanie i poszerzanie. Mimo że wymiar łańcucha dostaw jest w pewnym stopniu arbitralny i niejasny, to bardzo ważne jest dla budowanego modelu zrozumienie klucza istnienia granic sieci łańcucha dostaw i określenia, które aspekty (lub zasięg) sieci łańcucha dostaw muszą być modelowane.

### 2.2.2. Zarządzanie łańcuchem dostaw

Koncepcja zarządzania łańcucha dostaw (*Supply Chain Management*) – SCM powstała jako alternatywa wobec tradycyjnego sposobu pojmowania relacji między dostawcami a odbiorcami w kategoriach ciągłych antagonizmów i dążenia do wykorzystania siły przetargowej [202]. Wzrost znaczenia zarządzania łańcuchami dostaw jako źródła sukcesów powoduje rosnące zainteresowanie tą problematyką ze strony środowisk naukowych i gospodarczych. Naukowcy zajmujący się zarządzaniem łańcuchem dostaw odwołują się nawet do publikacji z końca lat pięćdziesiątych i wczesnych lat sześćdziesiątych ubiegłego stulecia [33].

Koncepcja zarządzaniem łańcuchem dostaw jest szersza, aniżeli tradycyjne zarządzanie materiałami ograniczające się na ogół do określonej firmy i uwzględnia: źródła surowców i dostawców, politykę zakupu i gromadzenie surowców, przepływ materiałów wewnątrz firmy, a także przechowywanie końcowych produktów, dystrybucję, magazynowanie i transport. W tym więc rozumieniu firma integruje swoich dostawców i klientów stając się jednocześnie ogniwem w łańcuchu logistycznym [7, 76, 82, 191, 198].

Pierwsze definicje zarządzania łańcuchem dostaw były formułowane w kontekście logistycznym i w takim rozumieniu zarządzanie łańcuchem dostaw to planowanie, koordynowanie i kontrolowanie przepływu materiału, części i wyrobów gotowych od dostawców do odbiorców, które obejmuje dwa oddzielne strumienie przepływów: materiałów i informacji (por. [42, 100, 101, 116]). Poza interpretacją zarządzania łańcuchem dostaw – z punktu widzenia logistyki – coraz liczniejsze grono zwolenników zyskuje rozważania akcentujące jego aspekty integracyjne [175]. Ważnym aspektem w tych interpretacjach jest przestrzenno-czasowa synchronizacja zarządzania strumieniem popytu z fizycznym strumieniem podaży produktów.

The Global Supply Chain Forum (GSCF)<sup>3</sup> definiuje Supply Chain Management (SCM) jako: integrację kluczowych procesów biznesowych obejmujących końcowego użytkownika z jego oryginalnymi dostawcami, którzy dostarczają produkty, usługi i informacje, a także wartość dodaną wszystkim uczestnikom łańcucha dostaw. W tym rozumieniu, przedsiębiorstwo może być postrzegane

---

<sup>3</sup> The Global Supply Chain Forum (GSCF) jest grupą nie konkurujących firm i zespołu naukowców, która powstała w styczniu 1999 r. w celu rozwijania teorii i praktyki SCM, związana z Uniwersytetem Stanowym w Ohio (OSU) [110].

jako centrum w sieci dostawców i klientów, jak pokazano na rysunku 2.4. Popyt i dostawca są rdzennymi elementami w tym podejściu [92, 110].

Ważnymi kwestiami w SCM są:

- dzielenie informacji łańcucha dostaw i monitorowanie systemów,
- koordynowanie wielokrotnych poziomów systemów w celu redukcji zapasów,
- wspólne planowanie na różnych poziomach zarządzania,
- redukcja baz dostawcy dla lepszej koordynacji,
- zwiększanie szybkości przepływu operacji, informacji, zapasów,
- koszt i korzyści uczestników systemu.

Większość badań dotyczących SCM koncentruje się na jednolitym przepływie produktów przez łańcuch do końcowego klienta. Można jednak różnicować rodzaje współpracy w łańcuchu dostaw i zdefiniować trzy rodzaje współpracy pomiędzy prawnie niezależnymi przedsiębiorstwami:

- łańcuch dostaw – rodzaj współpracy, w którym węzły łańcucha muszą synchronicznie obsługiwać potrzeby klienta,
- rozszerzone przedsiębiorstwo – rodzaj współpracy, w którym informacje i decyzje oraz procesy produkcyjne uczestników łańcucha są zintegrowane,
- wirtualne przedsiębiorstwo – rodzaj współpracy, w którym występują luźne relacje przedsiębiorstw, wiążące ich fachowość z pomocą technik informatycznych dla zaspokojenia potrzeb klientów (por. [212]).

Najbardziej zintegrowanym rodzajem współpracy jest rozszerzone przedsiębiorstwo, w którym systemy informacyjne i procesy produkcyjne są zintegrowane. Łańcuch dostaw bazuje na długoterminowej współpracy, ale jest mniej zintegrowany. Przedsiębiorstwo wirtualne może być krótkookresową współpracą pomiędzy uczestniczącymi przedsiębiorstwami bez integracji.

### 2.2.3. Model SCOR

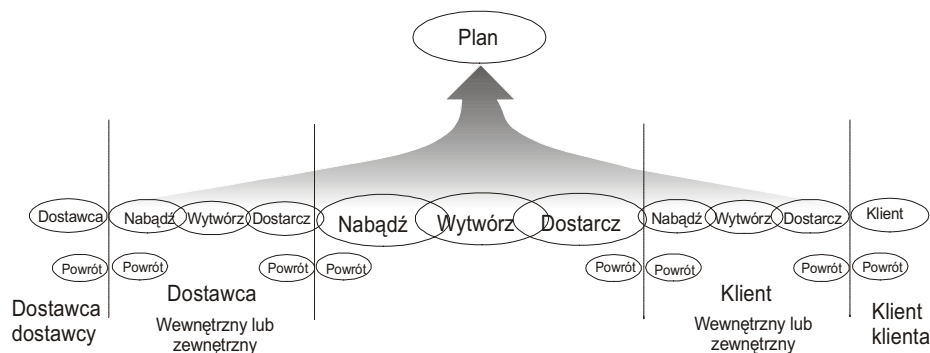
W nurcie logistycznych interpretacji łańcucha dostaw, pod auspicjami organizacji *Supply-Chain Council Inc.* – SCC<sup>4</sup>, powstał model referencyjny zawierający zbiory zasad i elementów wykorzystywanych w tworzeniu i optymalizacji łańcucha dostaw, zwany w skrócie SCOR (*Supply Chain Operations Reference*).

Model SCOR łączy dobrze znane koncepcje: reorganizacji procesów biznesowych – BPR (*Business Process Reengineering*), porównywania się z innymi (*Benchmarking*), analizy najlepszych praktyk biznesowych (*Best Practices Analysis*) oraz pomiar procesu w funkcjonalną strukturę. Poziomą integrację łańcucha zależy przede wszystkim od jakości prezentowanej przez jego uczestników. Wyższa jakość tworzy bowiem warunki i przesłanki do pogłębiania integracji, a to z kolei odkrywa nowe możliwości odnoszenia sukcesu na rynku.

<sup>4</sup> Organizacja SCC została powołana w 1996 roku przez Pittigilo Rabin Toda i McGratha. Członkostwo w tej Radzie jest otwarte dla wszystkich firm i organizacji zainteresowanych stosowaniem oraz doradztwem w zakresie zarządzania łańcuchem dostaw.

Model SCOR jest próbą stworzenia standardu SCM poprzez określenie ogólnego zbioru elementów składowych łańcucha dostaw, kategoryzację tych elementów oraz szczegółową ich definicję wraz z miarami efektywności [135].

Wszelkie działania w ramach SCOR można podzielić na pięć podstawowych procesów, nazywanych: planowanie, nabywanie, wytwarzanie, dostarczanie, powracanie (*plan, source, make, deliver, return*). Procesy te są realizowane wewnątrz przez wszystkich uczestników łańcucha. Podstawową koncepcję modelu SCOR przedstawiono na rysunku 2.5.



Rys. 2.5. Podstawowa koncepcja modelu SCOR [za SCC SCOR Overview 2000]

Model referencyjny zawiera: standardowe opisy procesów zarządzania, strukturę wzajemnych stosunków wśród znormalizowanych procesów, wzorce do pomiaru osiągnięć procesu, najlepsze dla danej kategorii praktyki zarządzania (tj. praktyki dające najlepsze efekty), standardy cech i funkcji.

Definicje procesów w modelu SCOR są następujące:

1. Planowanie – procesy, które równoważą popyt i podaż, a tym samym określają działania: nabywania, produkcji i dostarczania. Proces planowania obejmuje planowanie popytu na wyroby oraz planowanie ich pozyskiwania. W jego ramach są agregowane i priorytetyzowane zapotrzebowania niezależne, planowane zapasy i przesunięcia dystrybucyjne oraz przeprowadzane jest zgrubne planowanie zdolności w podziale na grupy i kanały zbytu.
2. Nabywanie – procesy, które zaopatrują ogniwo łańcucha w dobra i usługi, w celu zaspokojenia popytu. Proces nabywania odnosi się do działalności zaopatrzeniowej. Są to takie działania, jak: zaopatrywanie, potwierdzanie i kontrolowanie przyjmowanego materiału, składowanie i wydawanie do zużycia materiału oraz zarządzanie infrastrukturą zaopatrzenia obejmującą: certyfikowanie i ocenę jakościową dostawcy, frachtowanie, inżynierię komponentów, kontraktację sprzedaży, inicjowanie płatności za sprzedane towary.
3. Wytwarzanie – procesy, które przekształcają materiały lub produkty w ostateczny stan w celu zaspokojenia popytu. Proces wytwarzania wiąże się z działalnością produkcyjną. Obejmuje zgłaszanie zapotrzebowań i pobieranie

materiału, wytwarzanie i testowanie wyrobów, pakowanie, składowanie i wydawanie wyrobów gotowych z magazynu zbytu oraz zarządzanie infrastrukturą wytwarzania w zakresie: zmian projektowych, urządzeń i wyposażenia, standardów jakości produkcji, planowania warsztatowego, krótkoterminowego planowania zdolności produkcyjnych i harmonogramowania operacji.

4. Dostarczanie – procesy, które dostarczają finalne dobra i usługi, w celu zaspokojenia popytu i obejmują: zarządzanie zamówieniami, transport i dystrybucję. Proces dostarczania jest najszerszej zdefiniowany spośród omawianych tutaj czterech podstawowych procesów SCM. Jego zakres obejmuje takie działania, jak: tworzenie ofert, wprowadzanie i obsługiwanie zamówień, notowanie cen i kursów walut, tworzenie i obsługę bazy danych klientów, alokację zapasów, zarządzanie bazą danych produktów i cen, fakturowanie, zarządzanie należnościami, kredytowanie klientów i prowadzenie księgowości dotyczącej klientów. Na proces ten składa się również gospodarka magazynowa – sortowanie, pakowanie, etykietowanie i konfigurowanie produktów oraz wysyłka produktów. Dostarczanie to także zarządzanie transportem i infrastrukturą dostaw – zarządzanie ruchem, frachtowanie, zarządzanie eksportem i importem, harmonogramowanie czynności transportowych i załadunku, realizacja harmonogramów i kontrola. W ramach tego procesu następuje również określanie zasad biznesowych odnośnie funkcjonowania kanałów sprzedaży oraz ustalanie reguł dotyczących zamówień, zarządzanie zapasami dostaw i zarządzanie jakością dostaw.
5. Powracanie – procesy skojarzone ze zwracaniem lub otrzymywaniem zwróconych produktów z różnych powodów. Proces powracania (*return*) obejmuje wszystkie działania związane z uprawnionym zwrotem produktów. Dotyczy on zarówno zwrotu wadliwych produktów, jak i nadwyżkowych produktów.

Zgodnie z założeniami modelu SCOR analiza procesów w łańcuchu dostaw jest przeprowadzana na hierarchicznie ułożonych poziomach, którymi są: strategia operacyjna, przepływ materiału, informacja i workflow.

Poziom pierwszy – to podstawowa analiza konkurencji i definiowanie podstaw współzawodnictwa w osiąganiu celów. Bazą dla tych prac są cele osiągnięte przez konkurencję.

Na poziomie drugim następuje konfigurowanie łańcucha dostaw zarówno wewnątrz, jak i na zewnątrz firmy. W ramach konfigurowania łańcucha następuje rozróżnienie tzw. kategorii procesu. Wyróżnia się siedemnaście kategorii procesów o charakterze planistycznym, wykonawczym lub zabezpieczająco-kontrolnym. Zgodnie z koncepcją modelu SCOR proces „wytwórz” może być określony np. przez następujące kategorie: produkcja na zlecenie (*make-to-order*), produkcja na magazyn (*make-to-stock*), projektowanie na zlecenie (*engineer-to-order*). Podobnie tworzy się kategorie w odniesieniu do procesu nabywania i dostarczania. Pomiędzy pierwszym i drugim etapem istnieje sprzężenie zwrotne. Mogą więc zaistnieć takie przypadki, w których w trakcie konfigurowania zaistnieje konieczność zmiany strategii.

Poziom trzeci modelu to dekompozycja każdej z kategorii. Na tym etapie opracowuje się wewnętrzne i zewnętrzne procesy, procedury oraz konfiguruje się elementy systemu i wyrównuje poziomy wydajności.

Implementacja to usprawnianie łańcucha dostaw wewnątrz i na zewnątrz firmy. Pomiedzy trzecim i czwartym poziomem może wystąpić sprzężenie zwrotne, tj. powrót do konfigurowania elementów systemu, opracowywania procedur czy też procesów.

Przestrzenne uwarunkowania pomiędzy ogniwami łańcucha dostaw w dużej mierze sprowadzają się do określenia rodzaju i struktury powiązań pomiędzy poziomem zagospodarowania przestrzennego regionów a lokalizacją produkcji, transportu i magazynowania. Zachodzi wobec tego konieczność konfigurowania sieci logistycznej. Konfiguracja sieci logistycznej to taka struktura punktów i dróg, przez którą produkty przepływają od źródeł wydobycia surowców do miejsc podaży tych produktów (por. [77]). Konfiguracja wymaga określenia, ile wystąpi tych punktów i dróg, gdzie będą zlokalizowane, jaki rodzaj transportu będzie użyty, jakie magazyny zostaną wykorzystane oraz które produkty będą przemieszczane. Mogą przy tym wystąpić różne konfiguracje sieci, nawet w odniesieniu tylko do jednego producenta różnych produktów.

Przestrzenne konfigurowanie to tworzenie geograficznej mapy łańcucha dostaw – wyznaczonych miejsc lokalizacji fabryk, magazynów, detalu, a aspekt czasowy oznacza czas dostawy produktu logistycznego do odbiorcy.

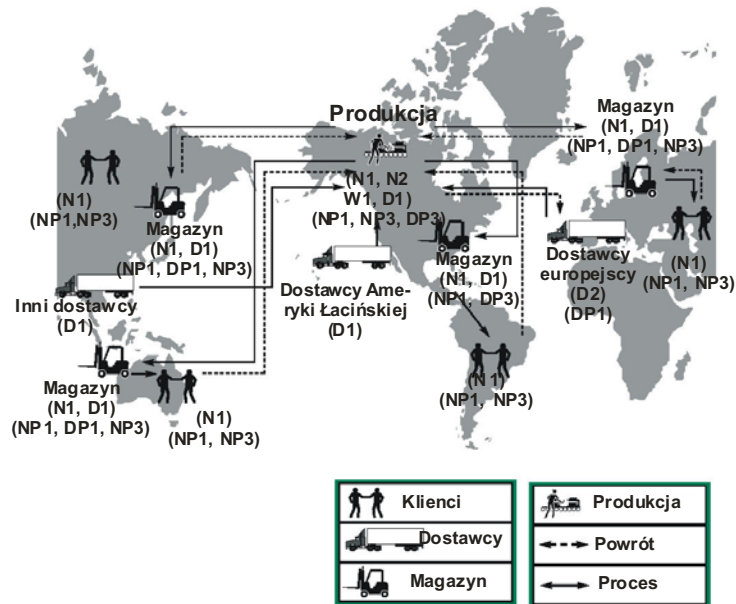
Celami konfiguracji sieci są [77]:

- minimalizowanie kosztów logistycznych, zwłaszcza w sytuacji, w której ich poziom wynika z uwarunkowań rozwoju regionu,
- maksymalizacja poziomu obsługi klientów, gdy ocenie podlegają całkowite koszty logistyczne, a nie ich poszczególne składniki,
- maksymalizacja zysku w ogniwach łańcucha logistycznego.

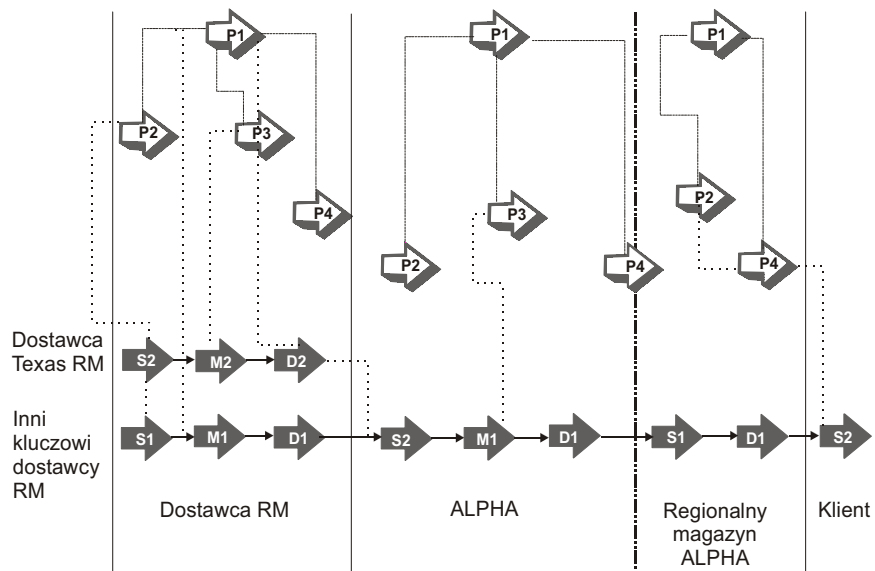
Przykład geograficznej mapy łańcucha dostaw pokazano na rysunku 2.6. Konfigurowanie łańcucha dostaw w modelu SCOR realizowane jest za pomocą diagramu zależności, którego przykład pokazano na rysunku 2.7.

Podsumowując rozważania dotyczące zarządzania produkcją można stwierdzić, że pojęcie *zarządzania produkcją* cały czas ewoluuje. Zarządzanie produkcją rozszerza się na zarządzanie siecią dostaw i obejmuje planowanie, organizowanie i kontrolowanie działań w sieci dostaw.





Rys. 2.6. Przykład konfigurowania geograficznej mapy łańcucha dostaw [za SCC SCOR Overview 2000]



Rys. 2.7. Konfigurowanie diagramu łańcucha dostaw [za SCC SCOR Overview 2000]

### 3. Konceptcje systemów zarządzania produkcją

W ostatnich latach zastosowanie znajdują następujące koncepcje komputerowo wspomaganego zarządzania przedsiębiorstwem i jego sferą produkcyjną:

- koncepcja zintegrowanego zarządzania zasobami produkcyjnymi MRP/ ERP,
- koncepcja sterowania produkcją i logistyką zgodnie z zasadą JIT,
- koncepcja LM,
- koncepcja OPT,
- koncepcja BOA,
- koncepcja FZS.

Poniżej przedstawiono syntetyczny przegląd najpopularniejszych komputerowych systemów zarządzania produkcją.

#### 3.1. Systemy MRP i ERP

Postęp w technologiach informatycznych rozszerzył skalę wdrożeń w zakresie technik, znanych od wielu lat pod akronimem MRP [31, 62, 84, 88, 144, 156]. Pierwsze generacje systemów MRP funkcjonowały pod angielską nazwą *Material Requirements Planning*, co można przetłumaczyć jako „planowanie potrzeb materiałowych”, a kolejne generacje pod nazwą *Manufacturing Resource Planning*, tłumaczoną jako „planowanie zasobów produkcyjnych” lub „planowanie i sterowanie zasobami materiałowymi”. Mimo że techniki MRP opracowane zostały w latach 50-tych ubiegłego stulecia, nie mogły znaleźć od razu szerszego zastosowania. Powodem tego był zbyt długi czas obliczeń niezbędnych do modyfikacji harmonogramów produkcji. Dopiero wsparcie szybkimi komputerami umożliwiło rozwój wdrożeń systemów MRP.

Główną funkcją MRP jest zagwarantowanie dostępności materiałów, tj. określenie jaki materiał, w jakiej ilości oraz na kiedy jest potrzebny, a następnie wygenerowanie odpowiednich pozycji zamówień. Proces ten obejmuje monitorowanie zapasów magazynowych, a w szczególności tworzenie propozycji zleceń zakupów i zleceń produkcji. System MRP bazuje na rozróżnieniu dwóch popytów na wyroby i materiały przewidziane do wytworzenia lub do innych celów w przedsiębiorstwie: popyt niezależny i popyt zależny. Pierwszy dotyczy wyrobów, które są przedmiotem zbytu. Popyt ten jest określany centralnie w planie głównym (asortymentowo-ilościowym) na podstawie zamówień i prognoz sprzedaży. Popyt zależny dotyczy elementów składowych wyrobów gotowych lub materiałów niezbędnych do ich wytworzenia. Popyt ten jest funkcją popytu niezależnego i może być na jego podstawie wyliczony. Kierunek planowania potrzeb MRP odnosi się do planowania popytu zależnego. Dane o potrzebach materiałowych i zapasach tworzone są dla poszczególnych okresów horyzontu planistycznego. Wyliczanie potrzeb zależnych realizowane jest w oparciu o tzw. struktury rozwinięcia wyrobów.

MRP do prawidłowego funkcjonowania wymaga danych o dwóch wielkościach:

- zapotrzebowaniu – określanym przez zakładaną wielkość sprzedaży, przyjęte zamówienia lub ustaloną produkcję wyrobów (zarówno produktów finalnych, jak i podzespołów potrzebnych do ich wytworzenia),
- zapasach i planowanych przyjęciach – wyznaczanych przez zapasy (gospodarka zapasami), zaplanowaną i zatwierdzoną produkcję (planowanie produkcji) oraz zaopatrzenie (złożone zamówienia, otrzymane potwierdzenia o dostawach itp.)

Rozwinięciem systemów MRP są systemy MRP II, w których dodatkowo bilansuje się plan produkcji z różnymi wymaganymi do jego realizacji zasobami (zdolności produkcyjne, środki finansowe, możliwości kadrowe itp.).

Z formalnego punktu widzenia MRPII obejmuje funkcje systemu oraz relacje pomiędzy nimi. Każda z funkcji jest związana z odmiennym zakresem symulacji rzeczywistości produkcyjnej. Funkcje systemu są najpełniej przedstawione w normie opracowanej przez stowarzyszenie APICS (*American Production and Inventory Control Society*), gdzie wyróżnia się następujące z nich:

- Planowanie biznesowe (*Business Planning*),
- Planowanie produkcji i sprzedaży (*Sales and Operation Planning – SOP*),
- Harmonogramowanie planu produkcji (*Master Production Scheduling – MPS*),
- Zarządzanie popytem (*Demand Management – DEM*),
- Planowanie potrzeb materiałowych (*Material Requirement Planning – MRP*),
- Podsystem struktur wyrobów (*Bill of Materials Subsystem – BOM*),
- Podsystem transakcji materiałowych (*Inventory Transaction Subsystem – INV*),
- Podsystem harmonogramów spływu (*Scheduled Receipts Subsystem – SRS*),
- Sterowanie produkcją (*Shop Floor Control – SFC*),
- Planowanie zdolności produkcyjnych (*Capacity Requirements Planning – CRP*),
- Zarządzanie stanowiskiem roboczym (*Input/Output Control*),
- Zaopatrzenie (*Purchasing – PUR*),
- Planowanie dystrybucji (*Distribution Resource Planning – DRP*),
- Pomoce warsztatowe (*Tooling*),
- Interfejs do planowania finansowego (*Financial Planing Interfaces*),
- Symulacje (*Simulation*),
- Pomiar działania systemu (*Performance Measurement*).

Podział ten oraz jego omówienie można znaleźć w pracy D. Landvater i C. Gray „MRP2 Standard System – A handbook for manufacturing software survival”, wydanej przez Oliver Wight Company (obecnie wydawcą jest John Wiley and Sons) i zaprezentowanej w artykule A. Popończyka [173].

System MRP II uzupełniony o moduły finansowe, takie jak: rachunek kosztów i analiza zarządcza określany jest jako MRP II + lub MRP III (*Money Resource Planning*) – planowanie zasobów finansowych. System taki jest także określany mianem systemu ERP (*Enterprise Resource Planning*) – planowanie zasobów przedsiębiorstwa. Ten właśnie termin jest prawie powszechnie stosowany dla określania systemu tego typu.

Systemy klasy ERP zdobywają coraz większą popularność. Najistotniejszą cechą systemu ERP wzbogacającą jego funkcjonalność jest wprowadzenie do systemu wspomnianych już modułów finansowych pozwalających planować i sterować produkcją nie tylko na podstawie wskaźników ilościowych, lecz także wartościowych. Oprócz tego system ERP stanowi uzupełnienie systemu MRP II o wspomaganie dodatkowych funkcji (realizowanych przez oddzielne aplikacje), np. kontakty z dostawcami, dystrybutorami oraz klientami. Jądrem całego systemu jest centralna baza danych, z którą poszczególne aplikacje wymieniają dane i na której opiera się działanie całego systemu. Baza danych gromadzi i przechowuje dane pochodzące z różnych obszarów działalności przedsiębiorstwa. Użycie jednej bazy danych usprawnia przepływ informacji pomiędzy wszystkimi obszarami działalności biznesowej przedsiębiorstwa. Źródłem danych dla bazy oraz korzystającymi z niej są komórki organizacyjne (tj. działające w nich aplikacje) w różnych pionach funkcjonalnych przedsiębiorstwa. W bazie danych ERP nowe dane wprowadzane są tylko raz w jednym miejscu. Ich wprowadzenie powoduje automatyczne uaktualnienie wszystkich powiązanych z nimi danych. W praktyce zbudowanie takiej uniwersalnej, integrującej dane z różnych aplikacji bazy danych w przypadku dużego przedsiębiorstwa okazuje się trudne. Wynika to w dużej mierze ze znacznego stopnia defragmentacji informacji generowanej w przedsiębiorstwie oraz różnego stopnia ich szczegółowości. Dlatego też w systemach ERP duży nacisk kładzie się na właściwe zdefiniowanie interfejsów wymiany informacji między aplikacjami a bazą danych.

### 3.2. Koncepcja JIT

Przykładem techniki zarządzania produkcją opartej na koncepcji „ssania” jest system (*Just in Time*) – JIT – „akurat na czas, który traktuje czas cyklu produkcyjnego jako najważniejszy wskaźnik potencjału produkcyjnego (por. [55, 62]). Technika produkcji „akurat na czas” polega na tym, że na zamówienia klientów oczekują nie zapasy wyrobów gotowych, lecz zdolności produkcyjne do ich wytworzenia. Takie podejście powoduje duży wzrost konkurencyjności na rynku.

Technika JIT zakłada, że w łańcuchu procesów technologicznych i kooperacyjnych każdy element wyrobu finalnego wyprodukowany będzie *dokładnie na czas* jego montażu i jego ilość będzie odpowiadała zamówieniom. W tym systemie postuluje się zwiększenie majątku produkcyjnego przy jednoczesnym

zmniejszaniu partii produkcyjnych oraz produkcji w toku (zmniejszenie środków obrotowych), co podwyższa elastyczność i prowadzi do skrócenia czasu realizacji zamówienia klienta (por. [31, 84]).

JIT to system, który traktuje proces produkcji jako łańcuch logistyczny, obrazujący przebieg strumienia materiałów od dostawców do odbiorców, poprzez kolejne stadia wytwarzania. W konwencjonalnym zarządzaniu produkcją łańcuch logistyczny rozpada się na trzy rozłączne części: zaopatrzenie, produkcja i dystrybucja. W systemie „akurat na czas” wszystkie elementy tworzą integralną całość zorganizowaną z punktu widzenia końcowego elementu łańcucha logistycznego, tzn. odbiorcy produktów.

W przeciwieństwie do konwencjonalnych metod sterowania, w których centralne ogniwo planowania produkcji wydaje zlecenia produkcyjne do wszystkich komórek produkcyjnych, w metodzie JIT planowanie centralne dotyczy jedynie ostatniego stadium łańcucha logistycznego, tzn. magazynu wyrobów gotowych. Wszelkie inne komórki produkcyjne i zaopatrzeniowe otrzymują zlecenie bezpośrednio z kolejnego, dalszego (tzn. znajdującego się bliżej końca) ogniwa łańcucha logistycznego. Przykładowo, magazyn wyrobów gotowych zgłasza zapotrzebowanie (co jest równoznaczne z wystawieniem zlecenia produkcyjnego) na określoną liczbę wyrobów do wydziału montażu, wydział montażu zleca wykonanie podzespołów wydziałom obróbki oraz kooperacji itd. Oznacza to, że zlecenie produkcyjne wystawiane jest zawsze w komórce zużywającej (lub obrabiającej) dany przedmiot. W ten sposób między dwiema kolejnymi komórkami produkcyjnymi tworzą się ogniwa międzystadialne, w których następuje wzajemny przepływ materiałów i informacji. Przepływ materiałów od „źródła” do „odbiornika” poprzedzony jest przepływem informacji w odwrotnym kierunku, tzn. produkcja „akurat na czas” poprzedzona jest informacją „akurat na czas”.

System JIT nie jest łatwy do implementacji. Nie tylko wymaga długotrwałego specjalnego wykształcenia, nowego oprzyrządowania, reorganizacji rozplanowania i korekt procedur, zrewidowania powiązań z dostawcami, podniesienia „morale” i uświadomienia pracowników, ale również ogromnego zaangażowania zarządu firmy oraz wielkiej cierpliwości.

Technika zarządzania „akurat na czas” z zerowymi zapasami znana jest też pod nazwą KANBAN [144] (w Japonii) lub ZIP (*Zero Inventory Production*). Głównym zadaniem organizacji produkcji wg KANBAN jest minimalizacja kosztów poprzez eliminowanie wszelkich strat, a więc realizacja haseł: żadne braki, żadne opóźnienia, żadne zapasy, żadne kolejki – gdziekolwiek i po cokolwiek, żadna bezczynność, żadne zbędne operacje technologiczne i kontrolne, żadne zbędne pomieszczenia. W KANBAN tylko stanowiska montażu finalnego otrzymują harmonogram produkcji z biura ekspedycji. Pozostali operatorzy maszyn oraz dostawcy otrzymują zamówienia produkcyjne w postaci kart KANBAN z następnego, dalszego w procesie w stosunku do danego, stanowiska pracy. W systemie JIT zmieniają się relacje pomiędzy dostawcami

a producentami. KANBAN używa tylko dwóch kart zwanych „KANBAN” dla sterowania ilością i czasem przepływu materiału. Są to:

- *karta ruchu* (przepływu) – upoważniająca do transferu jednego standardowego zasobnika określonych części ze stanowiska, gdzie były te części wyprodukowane, do stanowiska, gdzie powinny być zużyte,
- *karta produkcji* – upoważniająca do obróbki, czyli wytworzenia jednego standardowego zasobnika określonych części na stanowisku roboczym, z którego te części mają być przekazane na inne stanowisko na podstawie karty ruchu.

Specjalne karty KANBAN używane są również dla poddostawców, a inne specjalne karty są wydawane w sytuacji krytycznego zachwiania ustalonego rytmu produkcyjnego. Karty KANBAN sprzyjają koordynacji pomiędzy stanowiskami roboczymi a modułami produkcyjnymi szczególnie w warunkach produkcji seryjnej i powtarzalnej. Ilość części i materiałów w produkcji jest sterowana przez liczbę wypuszczonych kart. W tej sytuacji każde stanowisko chcąc wykonać swoje zadanie stara się „ciągnąć” kontenery z poprzedzających stanowisk pracy za pomocą właśnie uzyskanej karty ruchu. Odpowiednio żadne poprzedzające stanowisko nie może „pchać” kontenera z wyrobem na następne stanowisko, ponieważ żaden ruch nie jest możliwy, dopóki stanowisko następne nie jest gotowe, na co wskazuje dostarczenie karty ruchu. Co więcej, żadne stanowisko nie może podjąć obróbki, dopóki nie otrzyma zwolnienia w postaci karty produkcji. Ponieważ karta porusza się wraz z kontenerami zawierającymi części – wyszczególniając co ma być robione kolejno z tymi częściami – koniecznością jest, aby każda zmiana w przepływie była rejestrowana odpowiadającą jej zmianą w karcie ruchu.

### 3.3. Koncepcja LM

Koncepcja zarządzania LM (*Lean Management*) [199] często też określana jako LP (*Lean Production*), oznacza rozwój przedsiębiorstwa produkcyjnego w kierunku przedsiębiorstwa racjonalnie gospodarującego środkami. Przynosi ona znaczne zmiany w sposobach organizacji, zarządzania, a także w dziedzinie przygotowania zawodowego oraz kształtowania postaw pracowników. Koncepcję LM można zdefiniować, jako: „współdziałania stosunkowo niezależnych zespołów roboczych, składających się z wszechstronnie przygotowanych pracowników, pracujących w ramach elastycznych i zautomatyzowanych systemów organizacji pracy, w celu optymalnej realizacji zadań produkcyjnych, we współpracy z klientami i poddostawcami” (por. [41]).

Zarządzanie według koncepcji LM jest niekończącym się procesem polegającym na ograniczaniu zasobów ludzi, maszyn, powierzchni itd. potrzebnych do produkcji (por. [212]).

### 3.4. Koncepcja OPT

Istotą koncepcji zarządzania wąskimi przekrojami OPT (*Optimized Production Technology*) jest optymalne obciążanie stanowisk o krytycznych możliwościach [31, 41, 55]. W koncepcji OPT za nadrzędne kryterium sterowania zleceniami produkcyjnymi przyjmuje się możliwie maksymalne wykorzystanie tych stanowisk wytwórczych, na których występuje spiętrzenie zadań wytwórczych. Taka koncepcja sterowania produkcją jest najczęściej stosowana w:

- produkcji mało- i średnioseryjnej technologicznie zorientowanej,
- zleceniach niewrażliwych na zmiany terminów ich realizacji,
- przypadku, gdy tylko jedno „wąskie gardło” steruje procesami produkcji.

### 3.5. Koncepcja BOA

Koncepcja średniego stałego obciążenia stanowisk wytwórczych BOA (*niem. Belastungsorientierten Auftragsfreigabeverfahren*) polega na sterowaniu zleceniami produkcyjnymi według średniej wielkości [41]. Metoda ta wspomaga reguły MRP II w produkcji jednostkowej i seryjnej. Stosowana jest wtedy, gdy:

- terminy dostaw są stałe,
- zdolności produkcyjne są znane i do dyspozycji,
- stosowane są terminale zbierające dane wytwórcze do określania produkcji w toku i nadzorowania realizacji zleceń produkcyjnych,
- jest zapewniona dostępność materiałów.

### 3.6. Koncepcja FZS

Istotą koncepcji FZS (*niem. Fortschrittszahlenkonzept*) jest ciągłe uzupełnianie (zbilansowanych stanów) potrzeb materiałowych w produkcji seryjnej i masowej [41]. Ta koncepcja nadaje się w szczególności do:

- produkcji w przeważającej części seryjnej ciągłej lub powtarzalnej,
- produkcji płynnej, ze stałym zapotrzebowaniem materiałów zarówno dla średnich, jak i dużych serii,
- sterowania produkcją przy wykorzystaniu terminali zbierających dane wytwórcze, służących do nadzorowania zleceń produkcyjnych.

### 3.7. Zaawansowane systemy planowania i optymalizacji łańcucha dostaw typu APS

Rozwiązania typu APS (*Advanced Planning and Scheduling Systems*) tłumaczone jako zaawansowane systemy planowania i optymalizacji łańcucha dostaw są indywidualnymi, niezależnymi rozwiązaniami koncepcyjnymi

(por. [116, 174]). Systemy te powstały w ramach poszukiwań systemów, które nie miałyby ograniczeń wynikających z sekwencyjności planowania. Za podstawową cechę wspólną systemów APS należy uznać możliwość uwzględnienia w czasie planowania wielu ograniczeń, czyli możliwość planowania równoległego. Systemy te wspomagają podejmowanie decyzji głównie na poziomie taktycznym. Systemy klasy APS realizują generowanie planów z uwzględnieniem warunków biznesowych, opierając się na ogół na regułach priorytetu. Plany powstają najczęściej metodą „z góry na dół” i są korygowane przez planowanie „w przód” i „w tył” na wielu poziomach struktury wyrobu BOM (*Bill of Materials*) w oparciu o ograniczenia.

Cechą wspólną systemów tej klasy jest rozbudowanie planowania symulacyjnego, obejmującego zarówno generowanie gotowych planów, jak i ich przechowywanie. Do tego celu wykorzystuje się różnego rodzaju, opisane na ogół w literaturze algorytmy planowania, m.in.:

- algorytmy, które powstały w ramach teorii ograniczeń (*Theory of constraints*); w algorytmach tych przyjmuje się założenie, że istotne jest jedynie dokładne planowanie „wąskich gardeł”, nawet kosztem dotrzymania terminów dla zleceń,
- algorytmy simplex,
- algorytmy genetyczne.

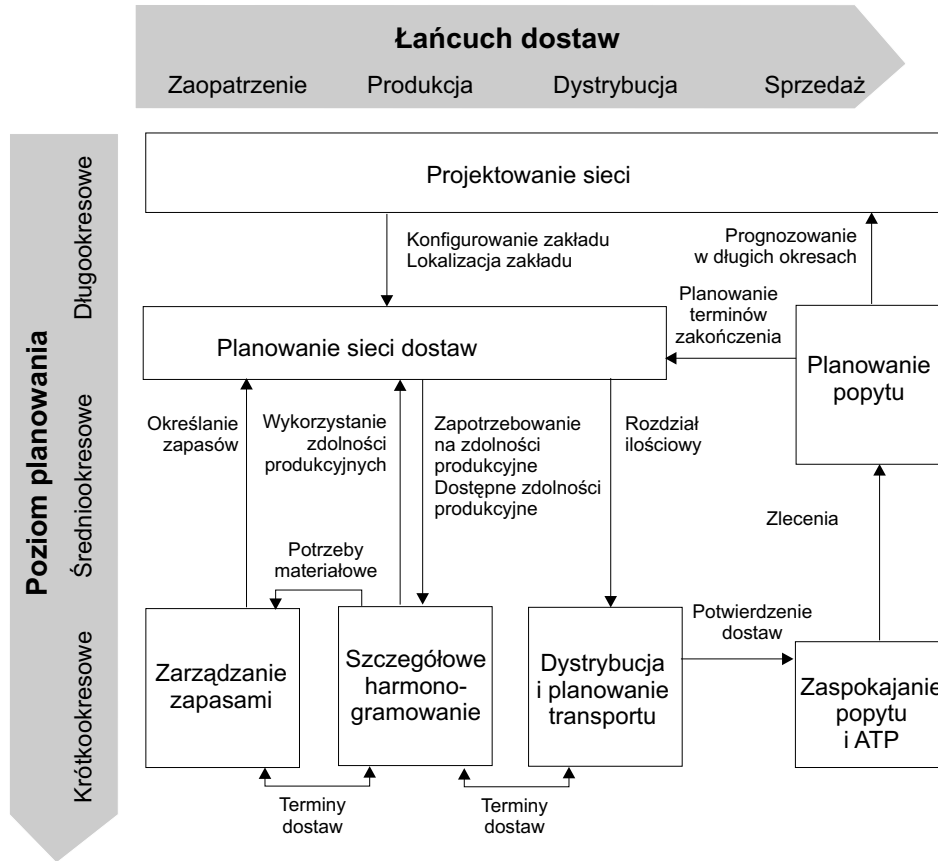
Najczęściej spotykane moduły w systemach APS to:

1. Planowanie popytu – wspomagający alokację popytu między miejsca uzupełnień.
2. Planowanie sieci dystrybucji – ze względu na zadane parametry, np. koszty transportu i magazynowania.
3. Planowanie transportu – optymalizujący trasy przewozu i kolejność załadunku.
4. Planowanie zakładowe – wspomagający szczegółowe harmonogramowanie „wąskich gardeł” w trybie równoległym, podejmowanie decyzji w ramach scenariuszy „*make or buy*” oraz optymalizujące zapasy w magazynach (typu *VMI*).

Przykład struktury APS pokazano na rysunku 2.8.

W oparciu o powyższe rozważania można stwierdzić, że zarządzanie produkcją w przedsiębiorstwie obejmuje coraz to szersze obszary podejmowania decyzji i coraz częściej rozszerza się na zarządzanie łańcuchami dostaw. W związku z tym, w rozważaniach dotyczących sterowania produkcją przedsiębiorstwo nie stanowi obecnie centralnego obiektu, a jedynie stanowi ogniwo sieci dostaw. Należy więc poszukiwać nowych metod zarządzania produkcją, które powinny uwzględniać aspekty integracyjne procesów realizowanych w sieci dostaw.





Rys. 3.1. Struktura APS [159]

Rozważania tego rozdziału wykazały również, że nie opracowano dotychczas takiej koncepcji modelu zarządzania produkcją, która mogłaby znaleźć zastosowanie w systemach produkcyjnych o szybkozmiennym asortymencie produkcji i mającej na celu optymalizację dwóch kryteriów: zaspokojenie zmiennych oczekiwań klienta na coraz to nowsze, lepsze produkty, w możliwie najkrótszym czasie i jednocześnie przy możliwie najniższych kosztach. Przy budowie takiego systemu należy wziąć pod uwagę zarówno dotrzymywanie terminów zleceń klienta, jak i wyrównywanie obciążeń maszyn. Z takiego punktu widzenia można przyjąć, że koncepcja OPT (*Optimized Production Technology*), w której za nadrzędne kryterium sterowania zleceniami produkcyjnymi przyjmuje się możliwie maksymalne wykorzystanie tych stanowisk wytwórczych, na których występuje spiętrzenie zadań wytwórczych może stanowić pewną fazę rozwiązywania problemu zarządzania produkcją w systemach o niestabilizowanej produkcji.

## 4. Harmonogramowanie produkcji

### 4.1. Klasyfikacja zadań harmonogramowania

Na ogół w problemach harmonogramowania (inaczej szeregowania zadań) zakłada się istnienie zbioru zasobów i ustalonego systemu zadań, które mają być wykonane z wykorzystaniem tych zasobów. Korzystając z przyjętych właściwości systemu oraz nałożonych ograniczeń należy skonstruować efektywny algorytm znajdowania takiego uszeregowania zadań, które pozwala zoptymalizować lub zmierza do optymalizacji przyjętego kryterium (por. [9, 99, 122, 142, 143]).

W klasycznej teorii sformułowanie *harmonogramowanie* ma następującą postać [11, 97]. Na obróbkę oczekuje  $N$  różnych przedmiotów (inaczej zadań lub prac). Dla realizacji operacji przydzielonych jest  $M$  maszyn. Zadane są marszruty technologiczne poszczególnych przedmiotów z ustalonym porządkiem wykonywania operacji oraz czasy wykonania tych operacji na poszczególnych maszynach  $t_{ij}$ ,  $i = 1, 2, \dots, N$ ;  $j = 1, 2, \dots, M$ . Należy wyznaczyć taką kolejność wprowadzania przedmiotów na poszczególne maszyny, która minimalizuje całkowity czas niezbędny do wykonania  $N$  przedmiotów. Ilustrację graficzną harmonogramu stanowi wykres Gantta.

Przytoczenia w tym miejscu wymaga także definicja *sekwencjonowania* (*sequencing*). Ustalanie sekwencji polega na tym, że dla każdej maszyny ustalona jest kolejność, w jakiej będą na niej wykonywane poszczególne operacje. Budowa harmonogramu polega na tym, że na podstawie sekwencji wykonania operacji na maszynach określone są ich terminy rozpoczęcia i zakończenia (proces określania terminów rozpoczęcia i zakończenia operacji określa się mianem *timetabling* [169]).

W literaturze przedmiotu stosowana jest zazwyczaj notacja zadań harmonogramowania wykorzystująca cztery parametry:  $n/m/A/B$ , przy czym:

$n$  – określa liczbę zrealizowanych prac (inaczej zadań lub zleceń),

$m$  – określa liczbę zaangażowanych maszyn,

$A$  – określa sposób przepływu detali w systemie produkcyjnym.

Jeżeli  $m = 1$ , to miejsce wartości  $A$  pozostaje puste. W innym przypadku  $A$  może przyjmować następujące wartości:

$F$  – dla przypadku *flow-shop*; oznacza to, że porządek wykonania operacji na poszczególnych maszynach jest we wszystkich pracach taki sam,

$P$  – dla przypadku określanego mianem *permutation flow-shop*. W przypadku tym nie tylko porządek operacji dla wszystkich prac jest taki sam, ale także poszukiwania są ograniczone do harmonogramów, w których porządek prac jest taki sam dla każdej maszyny. W tej sytuacji harmonogram jest całkowicie określony przez prostą permutację liczb  $1, 2, \dots, n$ , określającą porządek w jakim prace są wykonywane na każdej maszynie,

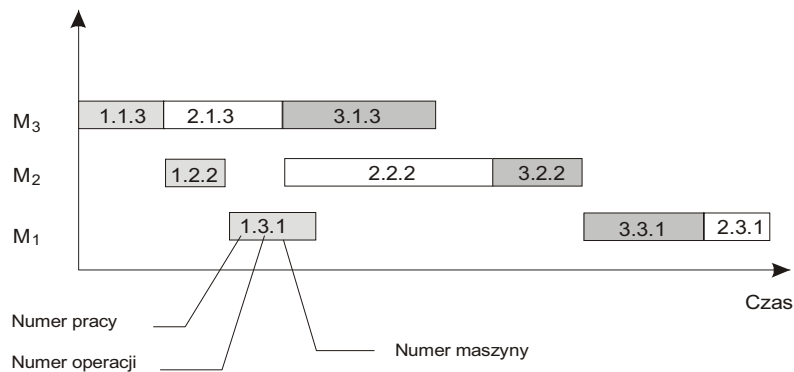
$G$  – dla ogólnego przypadku określanego mianem *general job-shop*, gdzie nie ma żadnych ograniczeń formy kolejności technologicznej.

$B$  – opisuje miarę oceny jakości harmonogramu. Najczęściej stosowaną miarą jest termin realizacji zleceń.

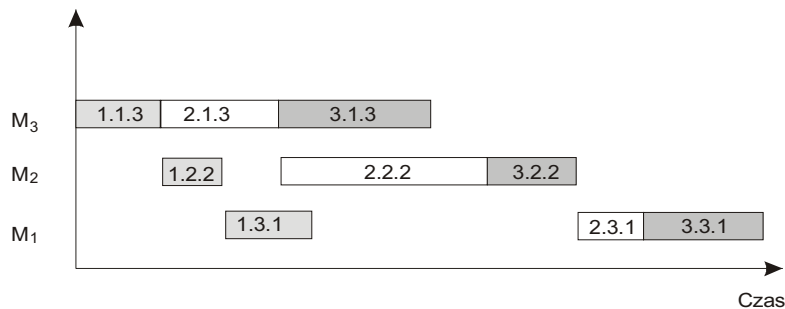
Najprostszym systemem spośród wymienionych jest system jednej maszyny i  $n$  prac. W tym przypadku uwzględniane są zazwyczaj czasy przebrojeń maszyny pomiędzy kolejnymi pracami. Przy formułowaniu problemu dla jednej maszyny podawane są często także terminy, w jakich poszczególne prace powinny być zakończone, a także terminy, w jakich prace są dostępne i można rozpocząć ich realizację. Zbyt późne lub zbyt wczesne zakończenie prac może być powiązane z zapłaceniem odpowiedniej kary.

Istnieje także problem harmonogramowania, w którym każda praca ma tylko jedną operację i mogą być one wykonane na jednej z równoległe ustawionych maszyn. Każda maszyna może wykonać każdą pracę, ale czasy wykonania są różne. Problem polega na ustaleniu kolejności w jakiej prace powinny być wykonane oraz przypisaniu prac do maszyn.

Na rysunkach 4.1 i 4.2 przedstawiono przykłady ilustrujące różnicę między dwoma wymienionymi odmianami problemu *flow-shop*. Na rysunku 4.1 operacje ze wszystkich zleceń są wykonywane w takiej samej kolejności, tzn. najpierw na maszynie  $M_3$ , następnie na maszynie  $M_2$  i w końcu na maszynie  $M_1$ . Kolejność wykonywania zleceń na poszczególnych maszynach nie musi być identyczna. W tym przypadku można powiedzieć, że zadania mają identyczną marszrutę technologiczną. Jest to przypadek linii produkcyjnej. Na przykład na maszynie  $M_3$  wykonywane jest najpierw zlecenie 1, później zlecenie 2 i w końcu 3; natomiast na maszynie  $M_1$  wykonywane jest najpierw zlecenie 1, później 3, a na końcu 2. Na rysunku 4.2 podano przykład harmonogramowania *permutation flow-shop*. Wszystkie prace są wykonywane najpierw na maszynie  $M_3$ , następnie  $M_2$  i w końcu  $M_1$ . Także na każdej maszynie kolejność jest taka sama. Najpierw jest wykonywana praca pierwsza, później druga i na końcu trzecia. Jest to przypadek permutacyjnej linii produkcyjnej.



Rys. 4.1. Harmonogram dla systemu typu *flow-shop* [169]



Rys. 4.2. Harmonogram dla systemu typu *permutation flow-shop* [169]

Problem wyboru sposobu realizacji zadań produkcyjnych w systemie przepływowym, może być przedstawiony następująco [23, 83, 144, 180]:

- 1) zbiór  $n$  wielooperacyjnych zadań produkcyjnych oczekuje na realizację począwszy od momentu  $t = 0$ . Każde zadanie wymaga  $O$  operacji, każda zaś operacja jest realizowana na innym urządzeniu spośród  $M$  stale dostępnych,
- 2) kolejność realizacji poszczególnych operacji jest *identyczna* dla wszystkich zadań,
- 3) poszczególne operacje nie mogą być przerywane (zadania niepodzielne),
- 4) czasy przygotowawczo-zakończeniowe dla poszczególnych urządzeń nie są zależne od kolejności zadań.

Tak zdefiniowane zagadnienie jest procesem deterministycznym, bowiem wszystkie parametry są z góry znane, i statycznym, gdyż zapotrzebowanie produkcyjne nie zmienia się w przyjętym przedziale czasowym [23]. W rezultacie – wybór optymalnego rozwiązania w tej klasie problemów wymaga przeliczenia  $n!$  sekwencji. Dla  $n > 10$  jest to nierealne i/lub nieopłacalne ekonomicznie. Z tych względów opracowano wiele metod wyznaczania rozwiązań suboptymalnych i nadal pojawiają się nowe propozycje [19, 38, 50, 132, 210].

Jako kryterium uszeregowania w procesie przepływowym stosowana jest powszechnie *długość uszeregowania (makespan)*, czyli czas zakończenia zadań w systemie. Termin ten odpowiada pojęciu *długość cyklu* znanemu z teorii organizacji produkcji. Dolną granicę długości uszeregowania można obliczyć według zasady podanej przez E. Taillarda [188, 189].

Dla przypadku określanego mianem *general job-shop* formułowane są zazwyczaj następujące założenia. Mamy  $n$  prac (inaczej zleceń)  $\{J_1, J_2, \dots, J_n\}$ , które mają być wykonane za pomocą  $m$  maszyn  $\{M_1, M_2, \dots, M_m\}$ . Każda praca jest zbiorem  $o$  operacji  $\{O_1, O_2, \dots, O_z\}$ . Operację pracy  $i$  wykonaną na maszynie  $j$  określa się  $o_{ij}$ . Ograniczenia technologiczne określają, w jakim porządku prace mają być wykonane na określonych maszynach. Dla ogólnego problemu harmonogramowania (*general job-shop*) nie ma żadnych ograniczeń porządku technologicznego (por. [14]). Każda praca ma swój własny porządek wykonania i nie muszą tu występować powiązania między porządkiem wykonania poszczególnych prac, tak jak ma to miejsce w przypadku *flow-shop*. Każda

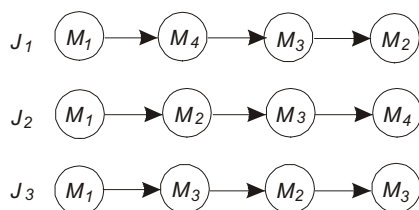
operacja  $o_{ij}$  ma określony czas wykonania oznaczany jako  $t_{ij}$ , w którym zazwyczaj zawarty jest czas przygotowawczo-zakończeniowy, tzn. czas wymagany do przygotowania stanowiska do wykonywania operacji i czas, jaki musi być poświęcony na przywrócenie go do stanu normalnego po zakończeniu wykonania operacji. W czasie  $t_{ij}$  zawarty jest często także czas wymagany na transport zlecenia do danej maszyny. Jest to przypadek gniazda produkcyjnego.

Przykład sformułowania ogólnego problemu harmonogramowania podano w tabeli 4.1.

Tabela 4.1. Przykład problemu harmonogramowania

Prace	Operacje			
	1	2	3	4
$J_1$	1(6)	4(8)	3(9)	2(4)
$J_2$	1(1)	2(3)	3(9)	4(6)
$J_3$	1(5)	3(5)	2(3)	3(6)

W tabeli 4.1 zaprezentowane są trzy prace  $J_1, J_2, J_3$ , z których każda składa się z 4 operacji. Na przykład praca  $J_1$  wykonywana jest najpierw na maszynie  $M_1$ , później na maszynie  $M_4$ , następnie na maszynie  $M_3$  i wreszcie  $M_2$ . Czasy wykonania wynoszą odpowiednio 6, 8, 9, 4 jednostki. Problem ten można sformułować wykorzystując formę graficzną (rys. 4.3).

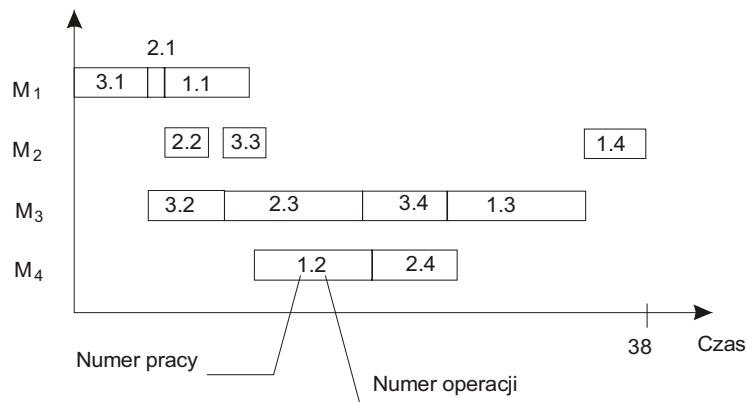
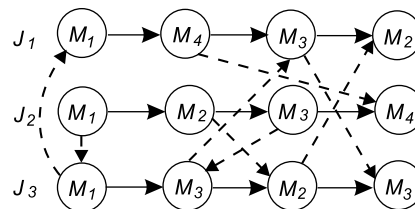


Rys. 4.3. Graf problemu *general job-shop* [169]

W sformułowaniu tym wierzchołki grafu odpowiadają poszczególnym operacjom. Łuki określają porządek technologiczny wykonania operacji w ramach poszczególnych prac. Na rysunku 4.3 pokazano ponadto, na jakich maszynach wykonywane są poszczególne operacje.

Dla tak sformułowanego problemu harmonogram może mieć postać jak na rysunku 4.4.

Dla przykładu z tabeli 4.1 można ustalić inną kolejność wykonywania prac. Na przykład na maszynie  $M_1$  wykonywana jest najpierw praca  $J_2$ , następnie praca  $J_3$  i na końcu  $J_1$ . Takie rozwiązanie można także przedstawić za pomocą grafu (rys. 4.5), w którym łuki zaznaczone linią przerywaną pokazują kolejność wykonywania operacji na poszczególnych maszynach.

Rys. 4.4. Prezentacja problemu *general job-shop* [169]

Rys. 4.5. Problem harmonogramowania w formie graficznej [169]

Niekiedy zakłada się, że nie wszystkie prace są dostępne w momencie rozpoczęcia produkcji. Moment czasu, w jakim praca  $J_i$  jest dostępna i można rozpocząć jej obróbkę określa się mianem *terminu gotowości* lub *daty uruchomienia* (*ready time or release date*) i oznacza  $r_i$ . Niekiedy formułowane jest także wymaganie dotyczące terminu zakończenia pracy; termin ten oznacza się zazwyczaj jako  $d_i$  (*due date*).

Kolejnym nieco mniej znanym w literaturze problemem harmonogramowania jest *open-shop scheduling problem*. W tym przypadku nie są określone relacje następstwa pomiędzy operacjami wchodzącymi w skład prac, czyli operacje mogą być wykonywane w (prawie) dowolnej kolejności.

Można wyróżnić trzy rodzaje harmonogramów [37, 169]:

- harmonogram pseudoaktywny (*semi-active schedule*),
- harmonogram aktywny (*active schedule*),
- harmonogram bez opóźnień (*non-delay schedule*).

Harmonogram zbudowany na podstawie sekwencji nazywa się *harmonogramem pseudoaktywnym*, jeżeli wszystkie operacje rozpoczynają się w najwcześniejszych możliwych terminach i ponadto nie są naruszone ograniczenia technologiczne, a także terminy, w jakich prace są gotowe do umieszczenia ich na harmonogramie. Innymi słowy, w pseudoaktywnym harmonogramie każda operacja rozpoczyna się tak wcześnie, jak jest to możliwe. Każdy dopuszczalny dla danych zadań harmonogram można przekształcić w harmonogram pseudoaktywny przez przesunięcie operacji na harmonogramie maksymalnie w lewo.

W *harmonogramie aktywnym* sekwencja wykonania operacji na maszynach jest taka, że żadna operacja nie może się rozpocząć wcześniej bez naruszenia ograniczeń technologicznych lub opóźnienia jakiejś innej operacji. Harmonogramy aktywne stanowią podklasę harmonogramów pseudoaktywnych, to znaczy każdy harmonogram aktywny jest harmonogramem pseudoaktywnym. Przy budowie harmonogramu aktywnego dopuszcza się także możliwość „przeskakiwania” niektórych operacji i umieszczania operacji (właśnie wstawianych) w lukach, jakie istnieją w harmonogramie, jeżeli nie naruszy to ograniczeń technologicznych. Oczywiście „przeskakiwanie” to zmienia sekwencje wykonywania operacji na maszynach. Zbiór harmonogramów aktywnych jest mniejszy od zbioru harmonogramów pseudoaktywnych. Na przykład na rysunku 4.6 harmonogramy 1, 2 i 3 są pseudoaktywnymi, natomiast tylko harmonogram 3 jest harmonogramem aktywnym, ponieważ jest możliwość przeniesienia operacji 2.1.2 i wykonania jej przed operacją 1.2.2 bez zmiany momentu rozpoczęcia tej ostatniej.

Zadanie 1	1.1.1	1.2.2
Zadanie 2	2.1.2	2.2.1

Maszyna 1	1.1.1			2.2.1				1.1.1	2.2.1
Maszyna 2		1.2.2	2.1.2		2.1.2			1.2.2	1.2.2
		1			2			3	

Rys. 4.6. Harmonogramy pseudoaktywne [169]

Liczba harmonogramów aktywnych dla danego zbioru zadań jest znacznie mniejsza od liczby harmonogramów pseudoaktywnych, jednak w dalszym ciągu jest to liczba bardzo duża. Harmonogram optymalny należy do klasy harmonogramów pseudoaktywnych, a także do klasy harmonogramów aktywnych. Ponieważ klasa harmonogramów aktywnych jest zawarta w klasie harmonogramów pseudoaktywnych, szukając rozwiązania optymalnego można ograniczyć się do rozpatrzenia harmonogramów aktywnych.

Zdefiniowanie pojęcia *harmonogramu bez opóźnień* wymaga zdefiniowania pojęcia *operacji możliwej do umieszczenia na harmonogramie*. Mówi się, że operacja jest możliwa do umieszczenia jej na harmonogramie maszyny  $k$  w momencie  $t$ , jeżeli w momencie tym zakończono wykonywanie wszystkich operacji poprzedzających rozpatrywaną i maszyna na której ma być ona wykonywana jest wolna. Innymi słowy, w harmonogramie tym maszyna nie pozostaje wolna, jeżeli może rozpocząć jakąś operację. W harmonogramie bez opóźnień nie występują momenty czasu, w których zatrzymuje się wykonywanie jakiejś pracy, jeśli maszyna, na której operacja ma być wykonana, jest wolna.

Harmonogramy bez opóźnień stanowią podklasę harmonogramów aktywnych. Rozwiązanie optymalne nie musi należeć do klasy harmonogramów bez opóźnień, to znaczy może istnieć harmonogram optymalny, który nie jest harmonogramem bez opóźnień. Przykład takiej sytuacji pokazano na rysunku 4.7.

Mamy tu zadanie typu 2x2 (dwa zadania i dwie maszyny), dla którego istnieją tylko dwa harmonogramy aktywne.

Harmonogram 1 na rysunku 4.7 nie jest harmonogramem bez opóźnień, ponieważ jednocześnie z rozpoczęciem wykonywania operacji 1.1.1 można tu było rozpocząć wykonywanie operacji 2.1.2, dla której była wolna maszyna. Harmonogram 2 na rysunku 4.7 jest harmonogramem bez opóźnień, jednak czas jego realizacji jest dłuższy niż harmonogramu 1.

Zadanie 1	1.1.1	1.2.2	1.3.1		
Zadanie 2	2.1.2				

Maszyna 1	1.1.1		1.3.1	1.1.1		1.3.1
Maszyna 2		1.2.2	2.1.2	2.1.2	1.2.2	
		1		2		

Rys. 4.7. Harmonogramy aktywne i harmonogramy bez opóźnień [169]

Należy podkreślić, że znacznie trudniej jest przekształcić dowolny harmonogram w harmonogram bez opóźnień, niż harmonogram pseudoaktywny w harmonogram aktywny. Niemniej harmonogramy bez opóźnień stanowią ważną klasę, ponieważ łatwo jest je zbudować od początku. Zdaniem R.W. Conwaya, W.L. Maxwella i L.W. Millera [47] istnieją przekonujące dowody mówiące o tym, że zbiór harmonogramów bez opóźnień, który może nie zawierać rozwiązania optymalnego, jako całość ma lepsze charakterystyki niż pozostałe harmonogramy aktywne.

Harmonogram optymalny zawarty jest w zbiorze harmonogramów aktywnych. Natomiast nie ma gwarancji na to, że zbiór harmonogramów bez opóźnień będzie zawierać harmonogram optymalny [169].

Podsumowując, można stwierdzić, że ogólny problem harmonogramowania polega na tym, aby przy uwzględnieniu ograniczeń technologicznych – kolejności wykonania poszczególnych operacji w ramach zleceń – ustalić terminy rozpoczęcia i zakończenia wykonywania operacji na poszczególnych maszynach tak, aby minimalizować lub maksymalizować określoną funkcję będącą miarą jakości harmonogramu.

## 4.2. Kryteria oceny harmonogramów

Ocena harmonogramu jest sprawą trudną, ponieważ w praktyce produkcyjnej dąży się przy jego budowie do jednoczesnego spełnienia wielu, niekiedy sprzecznych celów. Często stosowanymi kryteriami optymalności szeregowania są, m.in., czas wykonania wszystkich zadań z pewnego zbioru, średni czas wykonania zadania i maksymalne opóźnienie.



Przed szczegółowym omówieniem najczęściej spotykanych w literaturze kryteriów oceny należy określić następujące wielkości [67]:

$r_i$  – termin, w którym praca  $J_i$  jest gotowa do umieszczenia jej w harmonogramie (*ready time*),

$t_{ij}$  – czas wykonania operacji  $o_{ij}$ ,

$d_i$  – wymagany termin zakończenia pracy  $J_i$  (*due date*),

$a_i$  – fundusz (*allowance*) czasu dla pracy  $J_i$  (okres, w jakim praca powinna być wykonana, czyli czas między terminem gotowości i terminem zakończenia:  $a_i = d_i - r_i$ ),

$W_{ik}$  – czas oczekiwania pracy  $J_i$  przed wykonaniem jej  $k$ -tej operacji. Jeżeli ograniczenia technologiczne wymagają, aby praca  $J_i$  była wykonywana na maszynach w porządku  $M_{j(1)}, M_{j(2)}, M_{j(3)}, \dots$ , to operacja  $k$  ma symbol  $o_{ij(k)}$  i jest wykonywana na maszynie  $M_{j(k)}$ .  $W_{ik}$  jest to czas, jaki upływa między zakończeniem wykonywania pracy  $J_i$  na maszynie  $M_{j(k-1)}$  (lub  $r_i$ , gdy  $k = 1$ ) a początkiem obróbki na maszynie  $M_k$ ,

$W_i$  – całkowity czas oczekiwania pracy  $J_i$ . Oczywiście  $W_i = \sum_{k=1}^m W_{ik}$ ,

$C_i$  – termin zakończenia (*completion time*) pracy  $J_i$ , to znaczy moment, w którym zakończy się realizacja pracy  $J_i$ . Można tu napisać następujące równanie:

$$C_i = r_i + \sum_{k=1}^m (W_{ik} + p_{ij(k)}); \quad (4.1)$$

$F_i$  – czas przepływu (*flow time*), jest to okres, w jakim praca  $J_i$  przebywa w produkcji. Można tu napisać:  $F_i = C_i - r_i$ ,

$L_i$  – opóźnienie (*lateness*) pracy  $J_i$ , czyli różnica między rzeczywistym terminem zakończenia i wymaganym terminem zakończenia:  $L_i = C_i - d_i$ . Należy zauważyć, że jeżeli praca jest wykonana wcześniej, to znaczy przed wymaganym terminem zakończenia, to  $L_i$  ma wartość ujemną. Często przyjmuje się, że jeżeli praca jest wykonana przed wymaganym terminem zakończenia, to jej opóźnienie przyjmuje wartość równą zero. Wartość większa od zera występuje wtedy, gdy praca ma opóźnienie,

$T_i$  – opóźnienie pracy  $J_i$ :  $T_i = \max \{L_i, 0\}$ ,

$U_i$  – kara za opóźnienie pracy  $J_i$ :  $U_i = \begin{cases} 1 & \text{jeżeli } C_i > d_i \\ 0 & \text{inaczej} \end{cases}$ ,

$E_i$  – wyprzedzenie (przedwczesność) pracy  $J_i$ :  $E_i = \max \{-L_i, 0\}$ .

Często używaną wielkością jest czas bezczynności (*idle time*) maszyny  $M_k$ .

Określa się go jako:

$$I_k = C_{max} - \sum_{i=1}^n p_{ij(k)} \quad (4.2)$$

gdzie:

$C_{max}$  – termin, w którym kończy się realizacja wszystkich prac,  
a  $\sum_{i=1}^n p_{ij(k)}$  jest to całkowity czas, w jakim wykonywane są operacje na maszynie  $M_k$ . Różnica tych dwóch wielkości daje czas, w jakim maszyna  $M_k$  jest nieobciążona.

Można także wprowadzić wielkości charakteryzujące liczbę prac w różnych stanach w danym momencie czasu:

$N_w(t)$  – liczba prac oczekujących pomiędzy maszynami lub nie gotowych do obróbki w momencie  $t$ ,

$N_p(t)$  – liczba prac będących w obróbce w momencie  $t$ ,

$N_c(t)$  – liczba prac zakończonych w momencie  $t$ ,

$N_u(t)$  – liczba prac, które w momencie  $t$  nie zostały jeszcze zakończone.

Między tymi wielkościami zachodzą następujące zależności:

$$N_w(t) + N_p(t) + N_c(t) = n \quad (\text{dla wszystkich } t) \quad (4.3)$$

$$N_w(t) + N_p(t) = N_u(t) \quad (\text{dla wszystkich } t) \quad (4.4)$$

$$N_u(0) = n \quad (4.5)$$

$$N_u(C_{max}) = 0. \quad (4.6)$$

Dla każdej z tych wielkości można zdefiniować wartość średnią. Na przykład średnia liczba prac, będących w obróbce wynosi:

$$\bar{N}_p = \frac{1}{C_{max}} \int_0^{C_{max}} N_p(t) dt \quad (4.7)$$

Kryteria oceny harmonogramu można podzielić na trzy grupy (por. [67, 169]):

#### 1. Kryteria bazujące na czasie wykonania

Głównymi kryteriami w tej grupie są  $F_{max}$ ,  $C_{max}$ ,  $\bar{F}$  i  $\bar{C}$ . Minimalizacja  $F_{max}$  (maksymalnego czasu przepływu) oznacza, że koszt harmonogramu jest bezpośrednio związany z czasem trwania najdłuższej pracy. Minimalizacja  $C_{max}$  (maksymalnego terminu zakończenia – inaczej długości uszeregowania) oznacza, że koszt harmonogramu zależy od tego, jak długo system będzie zaangażowany w realizację wszystkich prac. Należy zauważyć, że w przypadku, gdy wszystkie terminy gotowości  $r_i$  są równe zero,  $C_{max}$  i  $F_{max}$  są identyczne. W przypadku, gdy niektóre czasy gotowości są różne od zera,  $C_{max}$  i  $F_{max}$  mogą się znacznie różnić. Rzeczywiście, jeżeli jedna praca ma ekstremalnie późny

termin gotowości, łatwo może dojść do sytuacji, w której praca z najkrótszym czasem przepływu kończy się w momencie  $C_{max}$ . Maksymalny termin zakończenia prac jest określony także w literaturze jako całkowity czas produkcji (*total production time*) lub okres wykonania (*makespan*).

Minimalizacja średniego czasu przepływu  $\bar{F}$  oznacza, że koszt harmonogramu jest ściśle związany ze średnim czasem wykonania pojedynczej pracy. Można także zauważyć, że minimalizacja średniego czasu wykonania  $\bar{C}$  jest równoważna minimalizacji  $\bar{F}$ , tj. harmonogram, który osiąga minimalne  $\bar{C}$ , osiąga także minimalne  $\bar{F}$ , i na odwrót. Może się wydać dziwne, że  $F_{max}$  i  $C_{max}$  są z zasady różnymi miarami jakości, podczas gdy  $\bar{F}$  i  $\bar{C}$  są w zasadzie takie same.

## 2. Kryteria bazujące na wymaganych terminach zakończenia

Ponieważ koszt realizacji harmonogramu jest zwykle związany ze stopniem niedotrzymania ustalonych terminów, to miarami jakości mogą być: średnie opóźnienie  $\bar{L}$ , maksymalne opóźnienie  $L_{max}$ , średnie opóźnienie  $\bar{T}$  i maksymalne opóźnienie  $T_{max}$ . Minimalizacja  $\bar{L}$  lub  $L_{max}$  jest odpowiednia wtedy, gdy można otrzymać nagrodę za wcześniejsze wykonanie pracy, i nagroda ta jest tym większa, im wcześniej praca jest zrealizowana. Minimalizacja  $\bar{T}$  lub  $T_{max}$  jest odpowiednia wtedy, gdy wcześniejsze wykonanie pracy nie daje żadnej nagrody, są natomiast kary nakładane za opóźnienie wykonania prac.

Czasem kary spowodowane przez opóźnienie realizacji prac nie są uzależnione od tego, jak duże jest to opóźnienie.

## 3. Kryteria bazujące na kosztach zapasów i wykorzystania maszyn

Można tu minimalizować średnią liczbę prac oczekujących na maszynie  $\bar{N}_w$  lub średnią liczbę nie zakończonych prac  $\bar{N}_u$ . Obie te wielkości są z grubsza związane z kosztami magazynowania produkcji w toku. Można także minimalizować  $\bar{N}_c$  – średnią liczbę zakończonych prac, ponieważ realizacja tego celu pozwoliłaby zredukować koszty magazynowania zakończonych prac. Jeżeli celem jest zapewnienie najbardziej efektywnego wykorzystania maszyn, to należy wybrać minimalizację  $\bar{N}_p$ , czyli liczbę prac, które są aktualnie realizowane. Innym kryterium jest minimalizacja  $\bar{I}$  lub  $I_{max}$  – średniego lub maksymalnego czasu bezczynności maszyny.

Kryteria oceny jakości harmonogramów można podzielić na regularne i nieregularne [169]. Miara regularna jest niemalejącą funkcją terminów zakończenia. Jeżeli mamy dwa harmonogramy  $H$  i  $H'$ , i  $R$  jest funkcją  $C_1, C_2, \dots, C_n$ , taką, że:

$$C_1 \leq C_1', C_2 \leq C_2', \dots, C_n \leq C_n' \quad (4.8)$$

$$R(C_1, C_2, \dots, C_n) \leq R(C_1', C_2', \dots, C_n') \quad (4.9)$$

to wyjaśnienie tej definicji jest następujące: Jeżeli w pierwszym harmonogramie wszystkie prace kończą się nie później niż w harmonogramie drugim, to wartość regularnej miary jakości dla harmonogramu pierwszego jest co najmniej tak samo dobra, jak dla drugiego. Należy zauważyć, że dąży się do minimalizacji regularnej miary jakości.

Następujące wielkości są regularnymi miarami jakości harmonogramu:  $\bar{C}$ ,  $C_{max}$ ,  $\bar{F}$ ,  $F_{max}$ ,  $\bar{L}$ ,  $L_{max}$ ,  $\bar{T}$ ,  $T_{max}$ ,  $n_T$  (por. [67]).

Omówione miary jakości harmonogramu są miarami podstawowymi. W większości przypadków dąży się do ich minimalizacji. Jednakże w algorytmach genetycznych wykorzystywana jest zazwyczaj funkcja przystosowania, która ma być maksymalizowana. W takich przypadkach tworzy się odpowiednie przekształcenia funkcji oceny do funkcji przystosowania. Niekiedy, dążąc do wyeliminowania konieczności przekształcania funkcji oceny do funkcji przystosowania, tworzy się bardziej złożone funkcje oceny, które są maksymalizowane. Na przykład autorzy pracy [18] proponują następujące wyrażenie oceniające ogólne wykorzystanie maszyn:

$$\sum_{i=1}^n \sum_{j=1}^{l_i} p_{ij} / m C_{max} \quad (4.10)$$

gdzie:  $l_i$  – liczba operacji w pracy  $i$ ,

$m$  – liczba maszyn,

$n$  – liczba prac.

Licznik to suma czasów trwania wszystkich operacji we wszystkich pracach, a mianownik iloczyn liczby maszyn i terminu zakończenia wszystkich prac  $C_{max}$ . Harmonogram, który rozdzieli operacje pomiędzy maszyny tak, że czas bezczynności wyniesie zero, otrzyma największą wartość przystosowania równą 1. Inne podejścia do tego problemu można znaleźć w pracy [169].

### 4.3. Metody poszukujące dokładnych rozwiązań problemów harmonogramowania

Metody poszukujące dokładnych rozwiązań problemów harmonogramowania to metody, w których obliczenia przerywa się wtedy, gdy osiągnięto ekstremum funkcji celu. Trudności związane ze złożonością obliczeniową stanowią ważną przyczynę rezygnowania z tych metod. W związku z tym mają one raczej wartość teoretyczną niż praktyczną. Zostaną tu jednak krótko przedstawione, aby pokazać sposób sformułowania zadania i stopień skomplikowania tego sformułowania.

#### 4.3.1. Przeszukiwanie zupełne i przeszukiwanie losowe

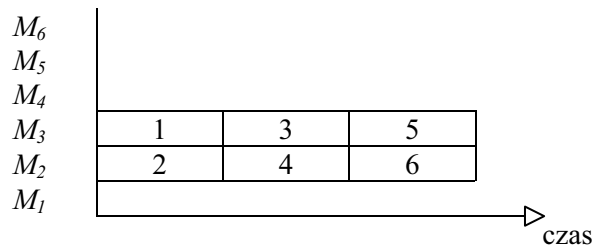
W celu zilustrowania metody *przeszukiwania zupełnego* rozważa się następujący przykład podany przez B. Gifflera, G. Thompsona i V.V. Nessa [70, 169]. Przyjmuje się 6 prac (zleceń), z których każda zawiera 6 operacji (tab. 4.2).

Tabela 4.2. Przykład problemu harmonogramowania

Praca	Kolejność maszyn					
$J_1$	3	1	2	4	6	5
$J_2$	2	3	5	6	1	4
$J_3$	3	4	6	1	2	5
$J_4$	2	1	3	4	5	6
$J_5$	3	2	5	6	1	4
$J_6$	2	4	6	1	5	3

Każda operacja jest wykonywana na innej maszynie. W celu uproszczenia badań przyjęto ponadto, że czas trwania każdej operacji wynosi 1. Wielkość problemu kombinatorycznego została istotnie ograniczona dzięki temu, że rozpatruje się jedynie dopuszczalne harmonogramy aktywne, których zbiór jest znacznie mniejszy niż zbiór harmonogramów dopuszczalnych.

W pierwszym kroku założono, że każda praca zawiera tylko jedną (pierwszą) operację. Przykład harmonogramu przedstawiono na rysunku 4.8. Pierwsze operacje wszystkich prac są wykonywane na maszynie  $M_2$  lub  $M_3$ .



Rys. 4.8. Przykład harmonogramu przy założeniu, że każda praca zawiera tylko jedną operację [70]

Liczba możliwych harmonogramów wynosi 36, ponieważ na maszynie  $M_2$  może wystąpić kombinacja 3 prac, i taka sama sytuacja jest na maszynie  $M_3$ , co daje w sumie  $3! * 3! = 36$ . W tabeli 4.3 podano liczbę harmonogramów przy stopniowym rozbudowywaniu problemu. Na przykład w pierwszym wierszu tabeli 4.3 zapis  $6 \times 6 * 1$  oznacza, że pojawia się problem składający się z sześciu prac realizowanych na sześciu maszynach, ale w harmonogramie umieszczane są tylko pierwsze operacje każdej pracy (tak jak na rysunku 4.8).

Tabela 4.3. Liczba harmonogramów aktywnych

Problem	Liczba aktywnych harmonogramów
$6 \times 6 * 1$	36
$6 \times 6 * 2$	290
$6 \times 6 * 3$	914
$6 \times 6 * 4$	7546
$6 \times 6 * 5$	84802

Jak można było oczekiwać, liczba aktywnych harmonogramów w każdym z tych problemów wzrasta bardzo szybko. Wariantu  $6 \times 6 \times 6$  nie testowano ze względu na jego rozmiary i czas obliczeń [169]. Dla każdego harmonogramu aktywnego można zbudować około 100 harmonogramów dopuszczalnych, które nie są aktywne. Tak więc całkowita liczba dopuszczalnych harmonogramów dla przypadku  $6 \times 6 \times 5$  wynosi około 9 milionów.

*Przeszukiwanie losowe* [169] polega na tym, że zamiast generowania wszystkich dopuszczalnych harmonogramów aktywnych lub nieaktywnych generowane są jedynie niektóre z nich w sposób losowy, czyli że przy budowaniu harmonogramu podejmowane są losowe decyzje w każdym momencie wystąpienia konfliktu. Nie jest to zatem to samo, co wybieranie losowe z takim samym prawdopodobieństwem ze skończonego zbioru wcześniej wygenerowanych wszystkich dopuszczalnych aktywnych harmonogramów. Przy porównywaniu rozwiązań otrzymanych w sposób losowy z rozwiązaniami otrzymanymi za pomocą przeszukiwania zupełnego zauważono, że krótkie harmonogramy mają większe prawdopodobieństwo wystąpienia przy tworzeniu próbki w sposób losowy niż w przeszukiwaniu zupełnym. Efekt ten nazwano *czynnikiem wzmocnienia (amplification factor)* [70].

#### 4.3.2. Programowanie całkowitoliczbowe

Metoda programowania całkowitoliczbowego nie nadaje się do praktycznego zastosowania. Nie nadaje się także do zastosowania w przypadku bardziej złożonych problemów teoretycznych [73, 169]. Metoda ta została opisana w wielu pracach [23, 24, 69, 169], zostanie tu jednak krótko przedstawiona, aby pokazać sposób sformułowania zadania.

Zdanie programowania całkowitoliczbowego formułuje się w następującej postaci:

zminimalizować (bądź zmaksymalizować) funkcje  $f(x)$ , (4.11)

przy ograniczeniach:

$$g_i(x) \leq b_i \quad i = 1, \dots, m \quad (4.12)$$

$$x \in N^n, \quad (4.13)$$

gdzie:  $N^n$  – jest zbiorem wszystkich  $n$ -wymiarowych wektorów o składowych całkowitoliczbowych,  $g_i$  jest funkcją rzeczywistą  $n$ -wymiarowego wektora zmiennych  $x$ , a  $f$  rzeczywistą funkcją celu określoną na zbiorze ograniczeń.

#### 4.3.3. Metoda podziałów i ograniczeń

Metodę podziału i ograniczeń stosuje się do szeregowania zadań niepodzielnych, tj. zadań, które nie mogą być przerwane przed zakończeniem wykonywania. Opis tej metody można znaleźć w wielu pracach [23, 44, 69].

Metoda podziału i ograniczeń polega na uporządkowanym przeszukiwaniu zbioru rozwiązań zadania optymalizacyjnego (4.11) ÷ (4.13), gdy zakłada się,

że jest skończony. Zwykle zbiór wszystkich rozwiązań dzieli się sukcesywnie na coraz mniejsze podzbiory, a dla każdego podzbioru oblicza się dolną granicę (w przypadku minimalizacji) wartości funkcji celu. Po każdym podziale te podzbiory, dla których dolne ograniczenie jest większe od wartości funkcji celu dla najlepszego, znanego rozwiązania dopuszczalnego, są eliminowane z dalszego podziału. Dzięki temu unika się bezpośredniego sprawdzania rozwiązań należących do eliminowanych podzbiorów. Podział trwa do chwili znalezienia rozwiązania dopuszczalnego, dla którego wartość funkcji celu jest nie większa niż najmniejsze dolne ograniczenie wszystkich nie podzielonych podzbiorów. W najgorszym razie trzeba sprawdzić wszystkie rozwiązania, co przesądza o wykładniczej złożoności obliczeniowej metody podziału i ograniczeń.

#### 4.4. Heurystyczne metody budowy harmonogramów

Algorytmy heurystyczne ogólnie można podzielić na *jednokrotne* i *korygujące* (inaczej: wielokrotne lub przeszukiwania sąsiedztwa lub poprawiające rozwiązanie bazowe). W algorytmach jednokrotnych, jeżeli raz określony zostanie moment rozpoczęcia operacji, to nie ulega on później zmianie. Algorytmy jednokrotne działają w ten sposób, że kolejno umieszczają operacje w harmonogramie, uwzględniając i nie zmieniając terminów rozpoczęcia operacji wcześniej włączonych. W algorytmach korygujących moment rozpoczęcia operacji wcześniej włączonej do harmonogramu może się zmieniać. Zmiana ta może polegać na opóźnieniu lub przyspieszeniu terminu rozpoczęcia jednej lub wielu operacji [169].

We wszystkich jednokrotnych algorytmach budowy harmonogramów określa się pewien zbiór operacji, które mają być umieszczone na harmonogramie. W każdym kroku algorytmu z tego zbioru wybierana jest jedna operacja i przypisywany jest jej moment rozpoczęcia, co oznacza umieszczenie operacji na harmonogramie. Sposób wyboru operacji i terminu jej rozpoczęcia określony jest przez charakter algorytmu. Fakt, że w algorytmach jednokrotnych moment rozpoczęcia operacji jest ustalany tylko raz i później nie może być zmieniony, nie powoduje, że są one gorsze od algorytmów, w których możliwa jest wielokrotna korekta terminów rozpoczęcia wszystkich operacji. Według autorów pracy [47] dla każdego zadania harmonogramowania istnieje algorytm jednokrotny dający rozwiązanie optymalne, jednak algorytm ten nie musi dać rozwiązania optymalnego dla innego zadania.

W ramach jednokrotnych algorytmów należy wyróżnić ważną ich klasę, określaną mianem *systemów dyspozytorskich*. Charakterystyczną cechą tej klasy jest to, że w porządku ściśle rosnącym określone są kolejno momenty rozpoczęcia operacji dla każdej maszyny. Innymi słowy, decyzje o włączeniu operacji do harmonogramu podejmowane są w takim porządku, w jakim będą one wykonywane na poszczególnych maszynach. Proces budowy harmonogramu można tu rozciągnąć w czasie, podejmując kolejne decyzje bezpośrednio przed ich

wykonaniem, a więc przed realizacją operacji. Zatem w systemach dyspozytorskich operacje są wstawiane na końcu kolejki prac realizowanych na maszynach. Jeżeli natomiast operacje są wstawiane do harmonogramu w najwcześniejszych możliwych terminach ich rozpoczęcia, przy uwzględnieniu ograniczeń technologicznych i bez przesuwania operacji wcześniej wstawionych do harmonogramu, to systemu takiego nie można określić mianem dyspozytorskiego. W takim systemie operacje mogą być bowiem wstawiane w „luki” pomiędzy już wstawionymi operacjami, jeżeli nie naruszy to porządku technologicznego. Kolejność wstawiania nie odpowiada jednak kolejności późniejszego wykonywania operacji.

Działanie algorytmów przeszukiwania sąsiedztwa (*neighbourhood search techniques*) polega na wyborze sekwencji bazowej (rozwiązania dopuszczalnego) przy użyciu dowolnego algorytmu, a następnie poprawianiu tak otrzymanego rozwiązania różnymi metodami.

S. French [67] zaproponował następujące ogólne kroki algorytmu przeszukiwania sąsiedztwa, inaczej zwanego procedurą *wspinania na szczyt* (*hill climbing*):

1. Zbuduj początkowy harmonogram.
2. Wygeneruj sąsiedztwo początkowego harmonogramu  $X$  i sprawdź, czy któryś z harmonogramów należących do sąsiedztwa jest lepszy od początkowego. Jeżeli żaden nie jest lepszy, to należy zakończyć procedurę i harmonogram początkowy jest najlepszym harmonogramem znalezionym za pomocą tej metody. W przeciwnym razie należy przejść do kroku 3.
3. Wybierz z sąsiedztwa  $X$  jeden z harmonogramów lepszych od początkowego i wstaw go w miejsce początkowego. Przejdź do kroku 2.

W symulowanym wyżarzaniu używa się procesów losowych, w celu określenia kierunków poszukiwań stanów o minimalnej energii [73].

Idea symulowanego wyżarzania (*simulated annealing*) [97, 106, 144, 169, 172] pochodzi z metalurgii i nawiązuje do podgrzewania metalu do określonej temperatury, a następnie powolnym chłodzeniu go. Symulowane wyżarzanie użyto po raz pierwszy do opisu ruchów termicznych atomów cieczy w ustalonej temperaturze  $T$ . Algorytm ten działa na założonej konfiguracji cząstek. Krok obliczeniowy polega na losowym przesunięciu jednej cząstki i wyliczeniu zmiany energii całkowitej systemu  $dE$  spowodowanej tym przesunięciem. Przesunięcie zmniejszające wartość energii całkowitej jest bezwzględnie akceptowane, natomiast przesunięcie zwiększające tę energię może zostać zaakceptowane z prawdopodobieństwem określonym wzorem:

$$P(dE) = \exp(-dE/T) \quad (4.14)$$

Takie podejście znalazło szerokie zastosowanie do rozwiązywania problemów optymalizacji kombinatorycznej. W takim przypadku rozwiązanie w przestrzeni rozwiązań dopuszczalnych jest odpowiednikiem stanu układu fizycznego  $S$ , funkcja celu odpowiada energii systemu  $E$ , a temperatura  $T$  jest parametrem



kontrolującym. Algorytm symulowanego wyżarzania można zapisać następująco [106, 144]:

```

BEGIN
  R := R0 {rozwiązanie początkowe};
  T := T0 {temperatura początkowa};
  WHILE (nie spełniony warunek zatrzymania) DO
  BEGIN
    WHILE (nie osiągnięty stan równowagi) DO
    BEGIN
      R' := F(R) {nowe, sąsiednie rozwiązanie generowane losowo}
      ΔE := E(R') - E(R);
      IF (ΔE < 0) OR (RANDOM < exp(-ΔE/T)) THEN R := R'
    END;
    Aktualizacja T;
  END;
END;

```

Warunkiem poprawnego działania przedstawionej metody jest prawidłowy dobór czterech czynników:

- 1) warunku zatrzymania obliczeń,
- 2) prawdopodobieństwa akceptacji „gorszego” rozwiązania,
- 3) sposobu generowania nowych rozwiązań,
- 4) liczby generowanych rozwiązań dla danej temperatury.

Przykłady implementacji tego algorytmu dla szeregowania zadań w systemach przepływowych można znaleźć w pracach: H. Ishibuchi, S. Misaki, H. Tanaka [95], F. A. Ogbu i D. K. Smitha [162] oraz I. H. Osmana i C. N. Potts [165], S. H. Zegordi, K. Itoba, T. Enklawa [207].

Do algorytmów korygujących należy również *przeszukiwanie z zabronionymi ruchami* – TS (*Tabu Search*) [90]. Algorytm ten został zaproponowany przez F. Glovera [72] i służy do rozwiązywania ogólnych problemów optymalizacji kombinatorycznej typu: zminimalizuj  $C(x)$  dla  $x \in X$ . W skład tego algorytmu wchodzi następujące elementy: ruch (*move*), sąsiedztwo (*neighbourhood*), lista ruchów zabronionych (*tabu list*), funkcja aspiracji (*aspiration function*) i warunek zakończenia działania (*stopping rule*). Ruch jest funkcją transformującą dane rozwiązanie w inne rozwiązanie. Podzbiór ruchów generuje dla danego rozwiązania podzbiór rozwiązań dany sąsiedztwem. Na każdym stopniu iteracji przeszukiwane jest sąsiedztwo w celu wyboru najlepszego rozwiązania, które staje się bazowym dla następnego kroku. Ostatnio wykonany ruch jest wprowadzany na listę zabronionych ruchów, które nie mogą być wykonane w ciągu najbliższych iteracji. Mechanizm ten umożliwia wyjście z lokalnego minimum i chroni przed cyklicznością ruchów. Niekiedy możliwe jest wykonanie zabronionego ruchu, o ile zdefiniowana funkcja aspiracji określi zyskowność takiego ruchu. Koniec obliczeń następuje w momencie przekroczenia limitu czasu, po wykonaniu dopuszczalnej liczby iteracji albo po wykonaniu dopuszczalnej liczby ruchów bez

poprawy wartości funkcji kryterium. Udana implementacje TS dla problemu szeregowania zadań w systemie przepływowym stworzyli M. Widmer i A. Hertz [200], E. Taillard [189], E. Nowicki i Cz. Smutnicki [161].

Trzy ostatnie omawiane metody można traktować jako strategię lokalnego przeszukiwania, którego istota polega na przechodzeniu z pewnego rozwiązania bieżącego do innego rozwiązania z jego sąsiedztwa z zachowaniem określonych reguł [1, 97]. Wadą tych systemów jest to, że w każdej iteracji rozważa się sąsiedztwo tylko pojedynczego rozwiązania.

#### 4.5. Reguły priorytetu w harmonogramowaniu produkcją

Reguły priorytetu są powszechnie wykorzystywane w przedsiębiorstwie – w komputerowo zintegrowanych systemach zarządzania produkcją [21, 80]. Każda decyzja dotycząca wyboru jakiegoś wyrobu lub operacji spośród ich zbioru oczekującego na zaplanowanie do wykonania lub wykonanie, bez względu na przyjęte kryterium, jest nadaniem temu wyrobowi lub operacji najwyższego stopnia pilności, czyli *priorytetu*. Można powiedzieć, że planowanie obciążeń stanowisk (harmonogramowanie) i dysponowanie prac odbywa się wyłącznie przez bardziej lub mniej świadome sukcesywne nadawanie priorytetu (inaczej pierwszeństwa) wyrobom i operacjom.

Problem wyboru, testowania i zastosowania różnych reguł priorytetów przy budowie harmonogramów był rozpatrywany przez wielu badaczy. Najbardziej ogólnie można powiedzieć, że reguły priorytetu mogą być wykorzystywane do wyboru jednej operacji z pewnego ich zbioru (tj. zbioru operacji oczekujących na umieszczenie ich na harmonogramie) i wstawieniu do harmonogramu. W węższym ujęciu reguły priorytetu mogą być wykorzystywane do wyboru operacji, która ma być wykonana na danej maszynie. W literaturze używa się także terminu *wskaźnik priorytetu*, który oznacza numeryczną cechę każdej operacji oczekującej na wykonanie w kolejce przed stanowiskiem [179, 203]. Wskaźniki priorytetu mogą być ustalane na podstawie parametrów charakteryzujących zadania produkcyjne, stany systemu produkcyjnego lub poszczególne zasoby itp. Wielkości te mogą mieć charakter statyczny lub dynamiczny i wówczas ich wartość zależy od aktualnych parametrów charakteryzujących zaawansowanie zlecenia (np. pozostały czas zadania).

W literaturze przedmiotu znanych jest ponad 100 reguł priorytetu [169]. Nie wszystkie z nich działają z wymaganą skutecznością. Dlatego istotną sprawą jest kwestia doboru odpowiedniej reguły, która spełniałaby określony cel.

W przypadku, kiedy obliczone wartości wskaźników priorytetu byłyby jednakowe dla kilku operacji równocześnie, wówczas o wyborze jednej operacji spośród kilku o tym samym priorytecie można rozstrzygnąć w różny sposób, np. stosując kolejną regułę z najsukuteczniejszych dla danego kryterium, stosując priorytet dla operacji, która pierwsza przybyła do kolejki lub też dokonując wyboru losowego. Procedura postępowania w takich przypadkach powinna być

jednoznacznie określona w algorytmie sterowania przepływem produkcji z zastosowaniem reguł priorytetu.

Najczęściej testowane i stosowane reguły priorytetów przy budowie harmonogramów przedstawiono w tabeli 4.4.

Tabela 4.4. Lista reguł priorytetów [38]

Reguła	Opis
SPT ( <i>Shortest Processing Time</i> ) – najkrótszy czas wykonania	Wybierz operację z najkrótszym czasem wykonania
LPT ( <i>Longest Processing Time</i> ) – najdłuższy czas wykonania	Wybierz operację z najdłuższym czasem wykonania
MWR ( <i>Most Work Remaining</i> ) – najwięcej pracy do wykonania	Wybierz operację, która należy do zadania, w którym pozostało jeszcze najwięcej (w sensie czasu, a nie operacji) do wykonania
LWR ( <i>Least Work Remaining</i> ) – najmniej pracy do wykonania	Wybierz operację, która należy do zadania, w którym pozostało jeszcze najmniej (w sensie czasu) do wykonania
MOR ( <i>Most Operations Remaining</i> ) – najwięcej operacji do wykonania	Wybierz operację, która należy do zadania, w którym pozostało jeszcze najwięcej operacji do wykonania
LOR ( <i>Least Operations Remaining</i> ) – najmniej operacji do wykonania	Wybierz operację, która należy do zadania, w którym pozostało jeszcze najmniej operacji do wykonania
EDD ( <i>Earliest Due Date</i> ) – najwcześniejszy termin zakończenia	Wybierz zadania z najwcześniejszym terminem zakończenia
FCFS ( <i>First Come First Served</i> ) – pierwsza przyszła, pierwsza wykonana	Spośród operacji oczekujących na wykonanie w kolejce przed stanowiskiem wybierz operację zadania, które pierwsze przybyło do komórki produkcyjnej
Random – losowo	Wybierz operację losowo

Obszerniejszy przegląd literatury dotyczącej problematyki szeregowania zadań i złożoności tego problemu można znaleźć w pracy J. Błażewicz, W. Domschke i E. Pesch [25] i innych, jak np. [9, 18, 65, 105, 180, 185, 186].

## 5. Systemy eksperckie

### 5.1. Geneza sztucznej inteligencji

Złożoność obliczeniowa różnych problemów występujących w zautomatyzowanej produkcji o szybkozmiennym asortymencie oraz wymagania związane z podejmowaniem decyzji w czasie rzeczywistym powodują ciągłe poszukiwania nowych metod komputerowego wspomaganie działalności produkcyjnej człowieka. Tym wysokim wymaganiom w chwili obecnej sprostać mogą jedynie metody heurystyczne rozwijane w ramach badań nad sztuczną inteligencją.

Przez wiele dziesiątków lat próbowano na bazie komputerów stworzyć sztuczne systemy, które w jakimś stopniu byłyby zdolne odtwarzać w komputerze przynajmniej niektóre z cech intelektu naturalnego. Mimo znacznego postępu i przeprowadzenia wielu badań w dziedzinie *sztucznej inteligencji*, termin ten nie jest definiowany jednolicie. Poniżej podane są dwie definicje uwypuklające różne aspekty badań w zakresie tej dziedziny:

- według M. Minskyego (w: [155]) sztuczna inteligencja jest nauką o maszynach realizujących zadania, które wymagają inteligencji wówczas, gdy są wykonywane przez człowieka,
- według E. A. Feigenbauma (w: [155]) *sztuczna inteligencja* stanowi dziedzinę informatyki dotyczącą metod i technik wnioskowania symbolicznego przez komputer oraz symbolicznej reprezentacji wiedzy stosowanej podczas takiego wnioskowania. Podobną definicję sztucznej inteligencji można znaleźć w literaturze polskiej [64, 176, 211].

Powstaje więc pytanie, kiedy system komputerowy wykazuje cechy inteligencji. Turing (w: [155]) zaproponował prosty test dający odpowiedź na to pytanie: „Maszyna jest nazywana inteligentną, jeśli zewnętrzny obserwator nie jest w stanie odróżnić jej odpowiedzi od odpowiedzi człowieka mogącego zastępować maszynę”.

*Sztuczna inteligencja* jest to nauka, która zajmuje się konstruowaniem systemów komputerowych, imitujących zachowanie tradycyjnie przypisywane posiadanej przez człowieka inteligencji [176].

*Systemami sztucznej inteligencji* nazywane są systemy spełniające funkcje, które przyjęto uważać za charakterystyczne dla intelektualnej działalności człowieka [158].

### 5.2. Reprezentacja wiedzy w systemach eksperckich

#### 5.2.1. Wiedza proceduralna i deklaratywna

Specyfika systemów eksperckich nierozdzielnie wiąże się z metodami reprezentacji wiedzy, ponieważ fundamentalnym problemem w sztucznej inteligencji jest *wiedza*. Obejmuje ona fakty i wiedzę heurystyczną. *Fakty*, to część ogólnie dostępnej wiedzy, dotyczącej danej dziedziny. Wiedza heurystyczna

obejmuje wiedzę zdobytą poprzez wieloletnie doświadczenie, pozwalającą na podstawie intuicji i tzw. zdrowego rozsądku badacza wykrywać nowe fakty i związki między nimi.

Wiedza w sztucznej inteligencji jest więc nie tylko gromadzeniem danych, lecz także kojarzeniem lub uogólnianiem doświadczeń na podstawie ustalonych przez człowieka zasad uświadamiania i rozumienia faktów oraz zjawisk rzeczywistości. Dlatego gromadzenie wiedzy jest procesem intelektualnym, który obok zwykłych funkcji przechowywania, wyszukiwania i proceduralnej obróbki danych obejmuje funkcję najważniejszą – skojarzonego logicznego wnioskowania.

Ważnym problemem w procesie tworzenia baz wiedzy jest wybór sposobu reprezentacji (tj. opisu) wiedzy [133]. Spośród stosowanych obecnie sposobów reprezentacji wiedzy wyróżnić można grupy: deklaratywnego, proceduralnego oraz mieszanego sposobu reprezentacji wiedzy. W reprezentacji typu *deklaratywnego* zakłada się, że wiedzę można przedstawić w postaci twierdzeń pewnego ustalonego języka. Modelowanie obszaru przedmiotowego za pomocą wiedzy deklaratywnej wymaga utworzenia pełnego opisu jego stanów koniecznych do wnioskowania. Proceduralna reprezentacja wiedzy polega na określeniu zbioru procedur, które mogą zastąpić proces wnioskowania. W reprezentacji wiedzy typu *proceduralnego* nie jest konieczne przechowywanie wszystkich możliwych stanów niezbędnych do wnioskowania. Wystarczy bowiem znać stan początkowy i odpowiednią procedurę, za pomocą której można generować opisy późniejszych stanów. Podział typów reprezentacji wiedzy na deklaratywne i proceduralne ma charakter względny, ponieważ konkretne modele w różnym stopniu wykorzystują oba te typy reprezentacji wiedzy.

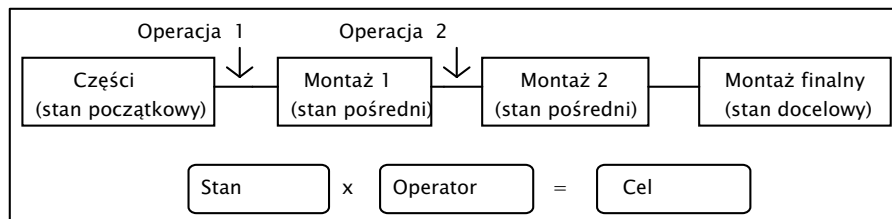
Niezależnie od sposobu reprezentacji wiedzy w inteligentnym systemie muszą być prezentowane jej dwa rodzaje: wiedza o dziedzinie danego problemu i wiedza o tym, jak rozwiązywać problem. Na przykład system wizyjny musi wykorzystywać wiedzę o rodzaju obiektów, które widzi oraz wiedzieć, jak analizować różne kształty czy kolory w zadaniach klasyfikacji tych obiektów.

### **5.2.2. Reprezentacja wiedzy w przestrzeni stanów**

Oryginalne problemy, które mają być rozwiązywane przez inteligentny system komputerowy, muszą być reprezentowane w postaci takich modeli, na których łatwo komputer może manipulować. Jednocześnie reprezentacja sytuacji problemowej w postaci modelu musi być analogiczna do oryginalnego problemu. Takie warunki spełnia reprezentacja problemów w przestrzeni stanów z komputerowej bazy danych.

Reprezentacja wiedzy w przestrzeni stanów jest szczególnie dogodna dla reprezentacji zadań. Stany problemów w postaci zadań są przekształcane z użyciem operatorów, które umożliwiają poszukiwanie rozwiązań problemów, czyli osiągnięcie stanu docelowego. To, kiedy i gdzie stosowane są operatory zależy od strategii sterowania. Jako przykład reprezentacji zadania w przestrzeni stanów można przytoczyć zadanie montażu. Rysunek 5.1 ilustruje kolejne stany

procesu montażu. Wiele problemów może być określonych poprzez wyszczególnione na tym rysunku trzy składniki: *stany*, *operatory* i *cele*. Rozwiązaniem zadań reprezentowanych w przestrzeni stanów jest sekwencja operatorów, która umożliwia przekształcenia od stanu początkowego do stanu końcowego.



Rys. 5.1. Reprezentacja zadania montażu w przestrzeni stanów [opracowanie własne]

Operatory są zbiorami procedur lub reguł, które są wykorzystywane do „produkcji” pożądanego stanu. Operatory w postaci reguł zawierają dwie części: warunkową i akcję. Akcja nie może nastąpić, dopóki dane warunki nie zostaną napotkane. Taka strategia poszukiwania rozwiązania oparta na idei tzw. systemów „produkcji” jest szeroko implementowana w charakteryzowanych dalej systemach eksperckich.

Do opisu przestrzeni stanów w sztucznej inteligencji bardzo często wykorzystuje się jeden z rodzajów grafu nazywany *drzewem*. Przykładem mogą tu być problemy planowania działań. Problem planowania działań jest określony poprzez stan początkowy i końcowy (lub zbiór stanów końcowych) oraz sposób tworzenia grafu rozwiązań. Rozwiązaniem problemu jest droga łącząca stan początkowy ze stanem końcowym w grafie. Węzły grafu to stany (sytuacje), a łuki to operatory przekształcania stanów.

*Graf* jest podstawą sieciowych modeli reprezentacji wiedzy. Najczęściej stosowane modele sieciowe to: sieci semantyczne (*semantic networks*) i ramy (*frames*). Do bardzo popularnych modeli reprezentacji wiedzy należą również systemy „produkcji” (*production systems*) oraz modele logiczne (*logic models*).

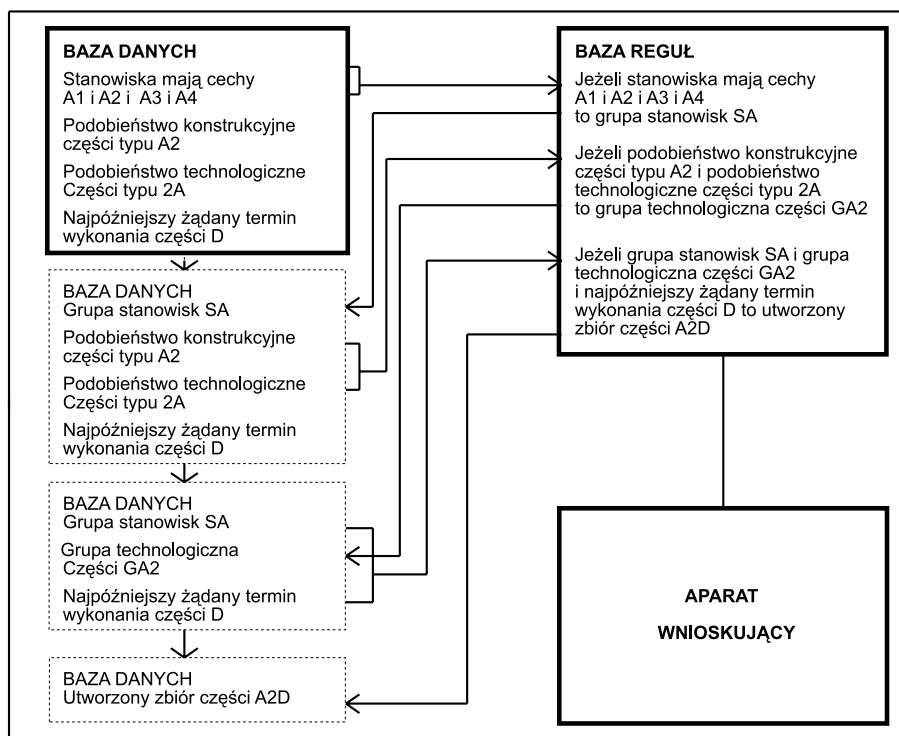
### 5.2.3. Systemy produkcji

*Systemy produkcji (production systems)* są szeroko stosowane w systemach eksperckich. Typowy system „produkcji” funkcjonuje w trybie dwufazowym obejmując fazy decyzji i działania. Faza decyzji obejmuje etapy:

- wyboru zbioru reguł decyzyjnych, tzn. reguł postaci „Jeżeli/to”. Mogą to być powiązania typu: warunek – akcja, sytuacja – akcja, przesłanka – konkluzja. Zarówno lewa, jak i prawa strona reguły może zawierać jedno lub więcej zdań zespolonych łącznikiem „i”,
- uzgodnienia wzorców występujących w warunkach reguł z faktami bazy danych,
- wyboru spośród zbioru uzgodnionych reguł tej, której następnik (np. akcja) ma zostać wykonany.

W fazie działania wprowadzane są odpowiednie zmiany w bazie danych (dopisywanie lub skreślanie pewnych faktów), jak i bazie reguł (dopisywanie lub skreślanie pewnych reguł). Ideę rozumowania w systemie „produkcji” przedstawiono na rysunku 5.2.

Po inicjacji pierwszej „produkcji” baza danych jest aktualizowana, tak jak pokazano to na rysunku 5.2 – element „*stanowiska mają cechy A1 i A2 i A3 i A4*” może być zastąpiony przez nowy element „*grupa stanowisk SA*”. Tak więc, inicjowanie „produkcji” może tworzyć nową bazę danych. Ten nowy element „*grupa stanowisk SA*” jest włączony do nowej bazy danych, a stary element jest pomijany, co uniemożliwia powtórzenie inicjacji tej samej „produkcji”.



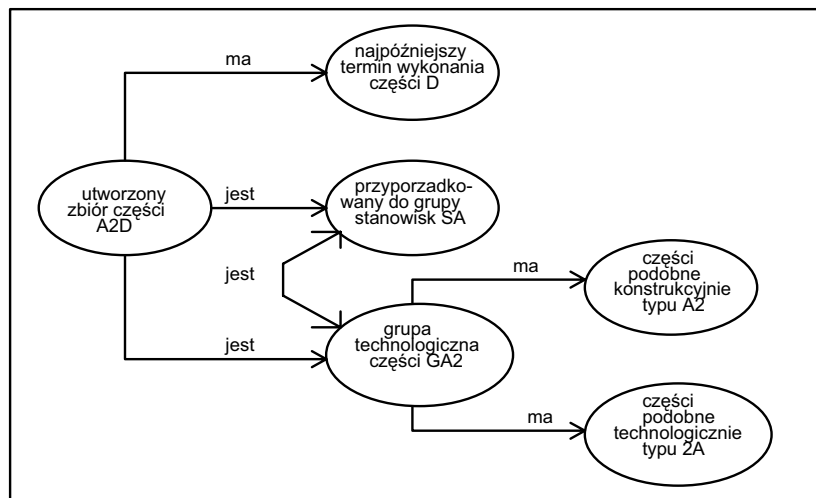
Rys. 5.2. Operacje systemu „produkcji” [133]

Następnie aparat wnioskujący kieruje badaniem dla elementów „*podobieństwo technologiczne części typu 2A*”. Te elementy zostają dopasowane do drugiej „produkcji” i ponownie aktualizowana jest baza danych. Ostatecznie aparat wnioskujący bada elementy „*grupa stanowisk SA*”, „*grupa technologiczna części GA2*”, „*najpóźniejszy żądany termin wykonania części D*”. Po dopasowaniu tych elementów inicjowana jest trzecia „produkcja” i na końcu uzyskiwana jest konkluzja „*utworzony zbiór części A2D*”.

#### 5.2.4. Sieci semantyczne

Problemy cechujące się dobrze zdefiniowaną hierarchiczną strukturą mogą być – z korzyścią dla użytkownika systemu – dogodnie formułowane w postaci sieci semantycznej lub ramy.

*Sieć semantyczna* jest reprezentacją graficzną wiedzy, opisującą za pomocą symboli związek między elementami wiedzy. Typowa sieć semantyczna składa się z obiektów reprezentowanych przez wierzchołki grafu i relacji między obiektami reprezentowanymi przez łuki (rys. 5.3).



Rys. 5.3. Sieć semantyczna [133]

#### 5.2.5. Ramy

Ramy (*frames*) uogólniają ideę sieci semantycznych. Rama umożliwia deklaracyjną i proceduralną reprezentację wiedzy, tzn. zawiera opis obiektu lub zjawiska oraz związane z nim procedury. Ramę można nieformalnie określić jako pewną strukturę, posiadającą nazwę, reprezentującą wiedzę o atrybutach danego obiektu (pojęcia). W strukturze tej wyróżnia się substruktury zwane szczelinami (*slots*). Każda szczelina, odpowiadając pewnej własności obiektu, należy do określonej dziedziny i może dalej dzielić się na fasety (*facets*). Każda rama, szczelina oraz fasetka identyfikowane są przez swoją nazwę.

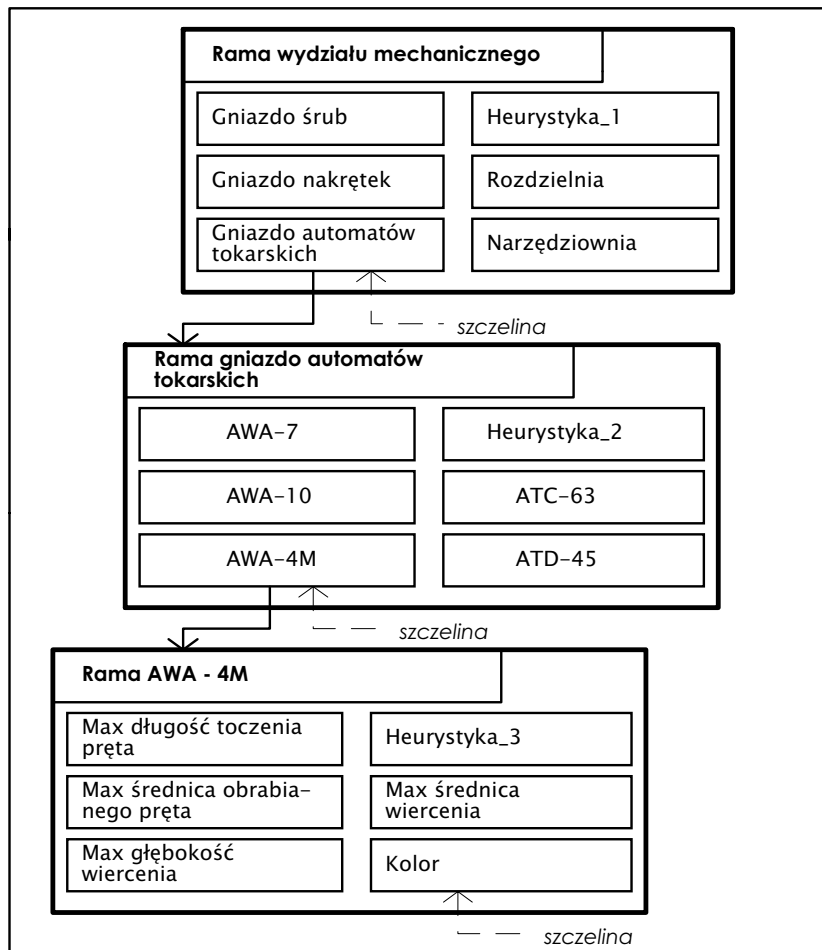
Wartością szczeliny może być pewna procedura. Procedury te dzieli się na dwie grupy: procedury demony i procedury sługi. Demon (*demon*) to procedura uruchamiana automatycznie przy określonych operacjach, niewidzialna dla użytkownika. Sługa (*servant*) to procedura uruchamiana tylko na żądanie użytkownika.

Zawartością szczeliny może być nazwa innej ramy, co pozwala na tworzenie między ramami połączeń łańcuchowych lub hierarchicznych.

Interesującym zagadnieniem jest wykorzystanie ram do rozumowania. Ideę rozumowania robota – w oparciu o ramy – przedstawiono na przykładzie ramy



wydziału mechanicznego (rys. 5.4). Na przykład, jeżeli robot znajdzie się w gnieździe automatów tokarskich, to w jaki sposób może on określić, że jest aktualnie w tym, a nie innym miejscu? Przypuśćmy, że robot jest wyposażony w system wizyjny i zaprogramowany do przemieszczania się na wydziale mechanicznym przy użyciu wiedzy magazynowanej w ramach. W takim przypadku rozumowanie może przebiegać w następujący sposób. Załóżmy, że robot „wszedł” na wydział mechaniczny i ładowana jest rama wydziału mechanicznego do pamięci roboczej robota. Zrozumienie, w którym miejscu robot się znajduje, następuje przez dopasowanie odpowiedniej szczeliny ramy wydziału mechanicznego. Każda szczelina, jak wynika z rysunku 5.4 zawiera ramę opisującą dany obiekt. Problemem dla robota jest więc zdecydowanie, która szczelina ramy wydziału mechanicznego ma być aktywowana? Do tego celu wykorzystuje się szczeliny o nazwie heurystyka.



Rys. 5.4. Rama wydziału mechanicznego [opracowanie własne]

Szczeliny heurystyk, wykorzystywane do sterowania procesem rozumowania, mogą zawierać reguły produkcji „Jeżeli/to”. W analizowanym przykładzie robot może aktywować szczelinę heurystyka\_1 ramy wydziału mechanicznego przedstawioną na rysunku 5.5.

Heurystyka_1
<p>Jeżeli zielony to aktywować szczelinę gniazdo śrub.          Jeżeli niebieski to aktywować szczelinę gniazdo nakrętek.          Jeżeli żółty to aktywować szczelinę gniazd automatów tokarskich.          Jeżeli biały to aktywować szczelinę rozdzielnia.          Jeżeli pomarańczowy to aktywować szczelinę narzędziownia.</p>

Rys. 5.5. Heurystyka [opracowanie własne]

Według tej szczeliny inicjowane są reguły produkcji stosownie do koloru pomieszczenia. W przykładzie przyjęto założenie, że system wizyjny robota jest wyposażony w filtry koloru. Dany kolor inicjuje regułę produkcji, która aktywuje odpowiednią szczelinę ramy wydziału mechanicznego. Załóżmy, że kolor pomieszczenia, do którego wszedł robot, jest żółty. Wtedy aktywowana jest szczelina o nazwie gniazdo automatów tokarskich. Konsekwentnie rama gniazdo automatów tokarskich ładowana jest do pamięci roboczej robota itd. Wypełnione szczeliny ram mogą więc być kluczem do rozumienia na podstawie ram. W prezentowanym przykładzie wykorzystywana jest wiedza o rodzaju obiektów, które robot „widzi” oraz wiedza o tym, jak analizować różne obiekty. Rama wydziału mechanicznego zawiera więc opis obiektu (w innych przypadkach może to być zjawisko) i związane z nim procedury.

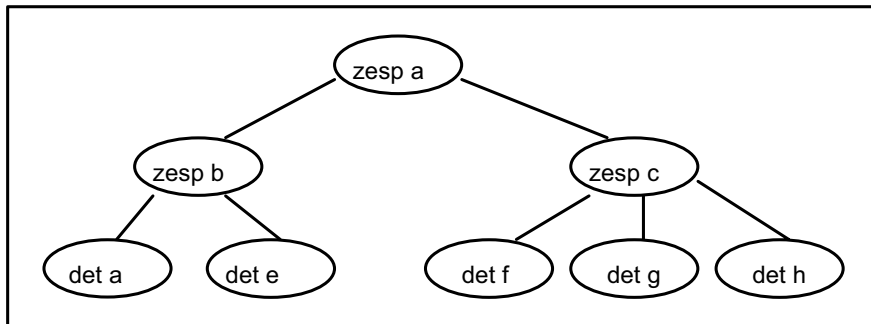
### 5.2.6. Logika

Do reprezentacji wiedzy z użyciem *logiki* wykorzystuje się rachunek zdań i rachunek predykatów. Rachunek zdań to ogólnie znany dział logiki matematycznej. Predykatem nazywana jest funkcja dowolnej liczby zmiennych, przyjmująca wartości logiczne T i N (tak i nie, prawda i fałsz, 1 i 0). Predykaty są często stosowanym efektywnym narzędziem modelowania wiedzy, a w szczególności mogą służyć do formalizowania faktów, zdarzeń, procesów, opisów strukturalno-funkcjonalnych oraz reguł wnioskowania.

Na logice predykatów pierwszego rzędu jest oparty m.in. język Prolog. Należy on do grupy języków deklaratywnych, tj. opisowych, zasadniczo różniących się od języków proceduralnych, takich jak C czy Pascal. Program napisany w języku proceduralnym zawiera szczegółowy opis kolejnych kroków postępowania zgodnie z przyjętym algorytmem, dzięki czemu z wprowadzonych danych otrzymuje się żądane wyniki. W Prologu programista deklaruje jedynie obiekty konieczne do rozwiązania danego problemu oraz relacje logiczne, jakie pomiędzy obiektami zachodzą w postaci tzw. *faktów* oraz *regul*. Zadanie do

rozwiązania przedstawione jest w postaci tzw. *celu* (*goal*). Fakty, reguły oraz podany cel tworzą program w języku Prolog. W celu przedstawienia istoty programowania w języku Prolog przedstawiono poniżej dwa przykłady zawierające podstawowe sekwencje programu napisanego w tym języku <sup>1</sup>.

Pierwszy przykład to program opisujący wyrób, który przedstawić można za pomocą struktury zwanej drzewem:



Rys. 5.6. Struktura wyrobu w postaci drzewa [opracowanie własne]

#### Przykład 1

DOMAINS  
nazwa=symbol

PREDICATES  
zesp(nazwa,nazwa)

CLAUSES  
zesp(zesp\_a,zesp\_b). /\* zespół a zawiera zespół b \*/  
zesp(zesp\_a,zesp\_c).  
zesp(zesp\_b,det\_d). /\* zespół b zawiera detal d \*/  
zesp(zesp\_b,det\_e).  
zesp(zesp\_c,det\_f).  
zesp(zesp\_c,det\_g).  
zesp(zesp\_c,det\_h).

PREDICATES  
wyrob(nazwa,nazwa)

CLAUSES  
wyrob(A,B) :- /\* A jest wyrobem z B \*/  
zesp(A,X),  
zesp(X,B).

<sup>1</sup> Zawarte w niniejszym opracowaniu programy napisano w języku Turbo Prolog.

W przytoczonym programie można wyróżnić trzy sekcje: DOMAINS, PREDICATES i CLAUSES. Każda z nich może wystąpić wielokrotnie. Sekcja DOMAINS (dziedziny) służy do określenia typów obiektów, które występują w programie. W prezentowanym przykładzie zadeklarowano własną dziedzinę danych *nazwa*, która odpowiada standardowej dla Turbo Prologu dziedzinie *symbol* (ciąg znaków). Sekcja PREDICATES (predykaty) zawiera deklaracje predykatów użytkownika (stanowi zapowiedź relacji zachodzących między określonymi obiektami). Sekcja CLAUSES (klauzule) służy do zdefiniowania relacji między poszczególnymi obiektami. Każda z klauzul może mieć postać faktu lub reguły. Fakty są stwierdzeniami, że zachodzą określone relacje między obiektami. W ich zapisie występuje jedynie nazwa relacji oraz zestaw argumentów (w nawiasach). Reguły mają charakter warunkowy – relacja zapisana po lewej stronie „:-” (if) jest prawdziwa jedynie wówczas, gdy są prawdziwe wszystkie warunki zapisane po prawej stronie tego znaku. Na przykład ostatnią klauzulę podaną w przykładzie można odczytać: *dla każdego wyrobu A i każdego wyrobu B: A jest wyrobem z B, jeśli istnieje takie X, że A zawiera X i X zawiera B*. Między znakiem „/\* „, a znakiem „\*/ „, zawarty jest komentarz.

Integralną część języka Prolog stanowi tzw. *aparatus (mechanizm) wnioskowania*, którego działanie jest odpowiedzialne za cały proces sterowania, tj. wnioskowania na podstawie zadeklarowanych związków.

Zadanie programisty w przykładzie 1 polegało na utworzeniu bazy danych opisujących strukturę wyrobu. Jeżeli chcemy na przykład sprawdzić, jakie detale zawiera wyrób, to odpowiednie pytanie powinno mieć następującą postać:

```
GOAL: wyrob(zesp_a, X)
```

Na tak sformułowane pytanie uzyskamy odpowiedź:

```
X=det_d      X=det_e      X=det_f      X=det_g      X=det_h
5 SOLUTIONS
```

Można zadać wiele różnych pytań do programu przedstawionego w przykładzie 1. Mogą to być m.in. następujące pytania:

```
GOAL:                Pytanie:
zesp(X, det_g)        Do jakiego zespołu należy detal g?
                       Odpowiedź:
```

```
X=zesp_c      1 SOLUTION
```

```
GOAL:                Pytanie:
zesp(zesp_c, det_i)  Czy zespół c zawiera detal i?
                       Odpowiedź:
```

No.

Język Prolog umożliwia również tworzenie baz danych. Za pomocą sekcji CLAUSES można reprezentować zawartość informacyjną bazy danych. Poniżej podano przykład programu zawierający relację „detaloperacja”, której postać tablicowa jest następująca:

Tabela 5.1. Opis detaloperacji

Kod detaloperacji	Czas przygotowawczo-zakończeniowy	Czas jednostkowy	Kod detalu	Nazwa detalu	Priorytet	Stanowisko
2530	1	50	268	walek	11	AWA-7
3143	2	30	352	walek	21	ATC-63
3534	7	85	343	walek	31	AWA-7
1175	2	20	123	walek	18	ATD-45

## Przykład 2

DOMAINS

Nazwa, stanowisko=symbol

tpz,czas\_tj,priorytet=real

Kod\_det\_oper,detal\_kod=integer

PREDICATES

det\_oper(kod\_det\_oper,tpz,czas\_tj,detal\_kod,nazwa,priorytet,stanowisko)

CLAUSES

det\_oper(2530,1,50,268,walek,11,AWA-70).

det\_oper(3143,2,30,352,walek,21,ATC-63).

det\_oper(3534,7,85,343,walek,31,AWA-7).

det\_oper(1175,2,20,123,walek,18,ATD-45).

PREDICATES

kandydat(kod\_det\_oper,tpz,czas\_tj,detal\_kod,nazwa,priorytet,stanowisko)

CLAUSES

kandydat(K,Tpz,Ctj,D,N,Priorytet,S):-

det\_oper((K,Tpz,Ctj,D,N,Priorytet,S),

Priorytet.19.

Przedstawiony w przykładzie 2 program umożliwia typowanie detaloperacji do wykonania (spośród ich zbioru przedstawionego w tabeli 5.1) w oparciu o wskaźnik priorytetu. Do wykonania kandydują te detaloperacje, których wskaźnik priorytetu jest wyższy niż 19. Po zadaniu pytania:

GOAL kandydat(K,Tpz,Ctj,D,N,Priorytet,S)

uzyskamy odpowiedź:

$K=3143$   $Tpz=2$   $Ctj=30$   $D=352$   $N=walek$   $Priorytet=21$   $S=ATC-63$

$K=3534$   $Tpz=7$   $Ctj=85$   $D=343$   $N=walek$   $Priorytet=31$   $S=AWA-7$

2 SOLUTIONS

Nie jest konieczne umieszczanie w programie z przykładu 2 ostatniego predykatu i klauzuli, ponieważ w Prologu można formułować pytania złożone. Dla programu z przykładu 2, w którym pominięto ostatni predykat i klauzulę można sformułować następujące pytanie:

*GOAL det\_oper(K,T,C,D,N,P,S), P>25*

W rezultacie otrzymamy odpowiedź:

*K=3534 T=7 C=85 D=343 N=walek P=31 S=AWA-7  
1 SOLUTION*

### 5.3. Struktura systemu eksperckiego

Jedną z możliwych form technicznej realizacji sztucznej inteligencji jest system ekspercki. Podobnie jak w przypadku sztucznej inteligencji, nie ma jednej definicji systemu eksperckiego. W literaturze można spotkać m.in. następujące określenia:

*Systemy ekspertowe (expert systems)* są to technologie oparte na sztucznej inteligencji, operujące na wyspecjalizowanej wiedzy w zakresie specyficznego obszaru ludzkiej działalności. Wiedza ta zorganizowana jest w ten sposób, że umożliwia systemowi ekspertowemu wejść w interakcyjny dialog z użytkownikiem, w wyniku czego system może oferować rozsądne rady lub proponować rozsądne decyzje oraz objaśniać sposób rozumowania, leżący u podstaw tych rad i propozycji [64].

*System ekspertowy* jest programem komputerowym, który wykonuje złożone zadania o dużych wymaganiach intelektualnych i robi to tak dobrze jak człowiek, będący ekspertem w tej dziedzinie [155].

Powyższe określenia eksponują na pierwszym miejscu wiedzę zgromadzoną w *bazie wiedzy*, umiejętność posługiwania się nią za pomocą *podsystemu wnioskującego* oraz dominującą rolę eksperta w tworzeniu systemu eksperckiego (por. [182, 211]).

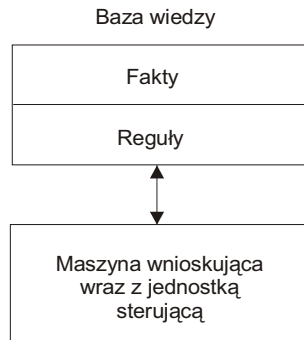
*System ekspertowy* jest programem, który może być implementowany na określonym sprzęcie komputerowym z odpowiednim oprogramowaniem narzędziowym (por. [119]).

W literaturze polskiej system ekspertowy nazywany jest także systemem eksperckim [137].

Podstawowe obszary wykorzystania systemów eksperckich w systemach produkcyjnych wiążą się z funkcjami, jakie mogą one spełniać, tzn. funkcjami: konsultanta, narzędzia modelowania, eksperta w poszczególnych specjalistycznych dziedzinach.

Realizację programową funkcji użytkowych systemu eksperckiego zapewnia jego zorganizowanie w ten sposób, że wiedza dotycząca danej dziedziny, zwana *bazą wiedzy (knowledge base)* jest odseparowana od reszty systemu, zawierającej m.in. sterowanie wiedzą. Ta część systemu, zawierająca mechanizmy

wnioskowania, nazywana jest *maszyną wnioskującą* (*inference engine*) [15, 155]. Podstawowe człony systemu eksperckiego pokazano na rysunku 5.7.



Rys. 5.7. Podstawowe człony systemu eksperckiego [155]

Systemy eksperckie używają reguł typu: „IF (jeżeli) ..., THEN (to) ...”. Baza danych zawiera określone fakty z przedmiotowej dziedziny, z których korzysta człon bazy wiedzy. Przed zastosowaniem reguły elementy jej części warunkowej muszą być prezentowane w bazie danych. Zastosowanie reguły kreuje nowe fakty, które aktualizują bazę danych. Maszyna wnioskująca kieruje procesem rozumowania poprzez dopasowywanie faktów z bazy danych do warunków reguł, a następnie rozpatrywanie konkluzji reguły jako przesłanki do następnej reguły. Jest to tzw. „łańcuch do przodu” (*forward chaining*). Może także występować „łańcuch wstecz” (*backward chaining*) polegający na wyznaczaniu przesłanek z zakładanych konkluzji – od celów do faktów elementarnych.

Rozważając bardziej szczegółowo strukturę systemu eksperckiego można wyróżnić następujące podstawowe jej elementy (por. [15, 32, 155]):

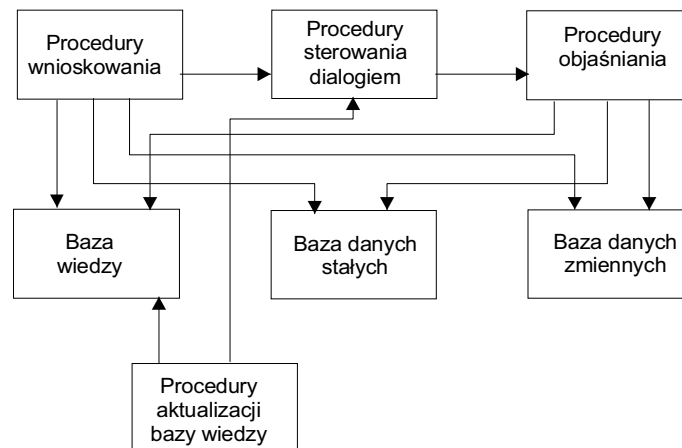
- bazę wiedzy – np. zbiór reguł,
- bazę danych – np. dane o obiekcie,
- procedury wnioskowania – maszyna wnioskująca,
- procedury objaśniania – objaśniają strategię wnioskowania,
- procedury sterowania dialogiem – procedury wejścia/wyjścia umożliwiają formułowanie zadań przez użytkownika i przekazywanie rozwiązania przez program,
- procedury umożliwiające rozszerzanie oraz modyfikację wiedzy – pozyskiwanie wiedzy.

Uwzględniając te elementy strukturę systemu eksperckiego można przedstawić w postaci schematu jak na rysunku 5.8.

Rozważania na temat implikacji ogólnej architektury systemów eksperckich można znaleźć w pracy J. Liebowitza i W.E. Pottera [120].

Budowę systemu eksperckiego ułatwiają specjalnie skonstruowane *szkieletowe systemy eksperckie* (*expert system shell*) [112]. Takie systemy posiadają pustą bazę wiedzy oraz pustą bazę danych i kompletne pozostałe elementy.

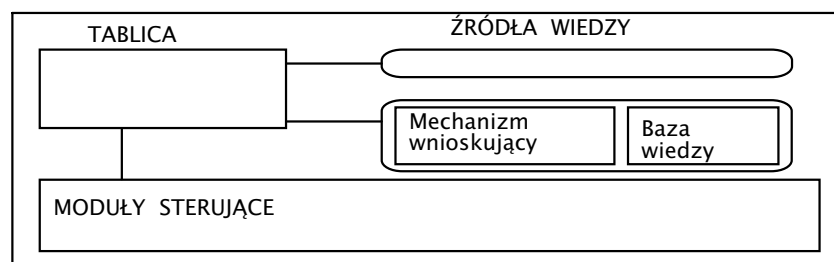
Możliwość tworzenia szkieletowych systemów eksperckich wynika z opisanej struktury systemu eksperckiego, w którym – jako wyraźnie oddzielone – występują: zbiór informacji o dziedzinie (fakty, reguły) oraz niezależny od dziedziny mechanizm wnioskowania.



Rys. 5.8. Elementy systemu eksperckiego [155]

#### 5.4. Architektura tablicowa systemu eksperckiego

Działanie systemu o architekturze tablicowej sterowane jest zdarzeniami. Przez zdarzenie rozumie się zmianę stanu pamięci systemu, reprezentowanej jako zbiór obiektów strukturalnych, przechowujących w swoich polach dane. Architektura tablicową tworzą: źródła wiedzy (moduły wiedzy), tablica, moduły sterujące (rys. 5.9) [81, 89, 136, 149].



Rys. 5.9. Architektura tablicowa [149]

W modelu tablicowym cała niezbędna wiedza o tym jak rozwiązywać problem zawarta jest w tzw. źródłach wiedzy (*knowledge source*). Są one niezależne od siebie i posiadają własne mechanizmy wnioskujące. Każde źródło wiedzy może używać innej reprezentacji wiedzy. Źródła wiedzy tworzą, modyfikują



i usuwają obiekty w tablicy. Każdy moduł wiedzy zawiera warunek i akcję. Warunki mogą obejmować: warunek pobudzenia, dotyczący zmian obiektów w tablicy oraz warunki uaktywnienia, dotyczące stanu tablicy. Tylko uaktywnione moduły wiedzy, tj. takie, których warunki są spełnione mogą spowodować akcję. Z każdym źródłem wiedzy skojarzony jest więc warunek aktywacji.

Tablica (*blackboard*) jest pewną globalną bazą danych obiektów, w której przechowywany jest bieżący stan procesu rozwiązywania problemu. Tablica może zatem zawierać dane wejściowe, rozwiązania częściowe i alternatywne oraz rozwiązania końcowe. Może być podzielona na poziomy, odpowiadające wyróżnionym w procesie rozwiązywania problemu szczeblom abstrakcji problemu oraz może uwzględniać przedziały czasowe procesu rozwiązywania problemu. Zmiany w tablicy powodowane przez źródło wiedzy doprowadzają do rozwiązania. Współpraca i komunikacja między modułami wiedzy odbywa się tylko za pośrednictwem tablicy.

Przetwarzanie w modelu tablicowym oparte jest na źródłach wiedzy (reprezentowanych przez niezależnych ekspertów). A ponadto opiera się na założeniu, że informacja niezbędna do rozwiązania problemu znajdzie się w najdogodniejszym miejscu i czasie w tablicy oraz będzie dostarczona przez jakiegoś eksperta (*opportunistic reasoning*). Współpracę źródeł koordynują moduły sterowania (*control unit*). Moduł sterowania może podejmować decyzje na podstawie kryteriów określonych przez użytkownika, użyteczności akcji możliwych do wykonania, informacji o stanie tablicy, charakterystyk problemu lub innych dostarczonych informacji.

Działanie całego systemu można opisać następującym ciągiem operacji:

1. Wywołane źródło wiedzy dokonuje zmian w tablicy (konkretnie w jakimś obiekcie lub kilku obiektach).
2. Każde źródło sygnalizuje, jakie działania może podjąć na rzecz współpracy w nowej sytuacji zaistniałej na tablicy.
3. Moduły sterujące dokonują wyboru elementu koncentracji uwagi w oparciu o dane z dwóch poprzednich punktów.
4. Element koncentracji uwagi określa następny krok w procesie rozwiązywania:
  - A. Jeżeli elementem koncentracji uwagi jest źródło wiedzy, to wybierany jest obiekt w tablicy, który posłuży jako kontekst wywołania tego źródła (*knowledge-scheduling approach*).
  - B. Jeżeli elementem koncentracji uwagi jest obiekt tablicy, to wybierane jest źródło wiedzy, które będzie przetwarzać ten obiekt (*event-scheduling approach*).
  - C. Jeżeli elementem koncentracji uwagi jest źródło wiedzy i obiekt tablicy, to obiekt ten jest przetwarzany przez to źródło.

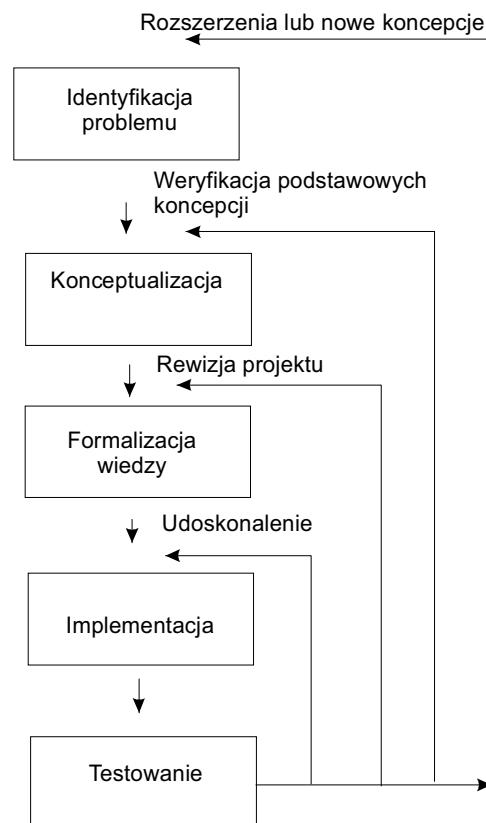
Podstawową zaletą architektury tablicowej jest możliwość wydzielenia proceduralnych składowych systemu. Każdy moduł jest wyspecjalizowany w wąskim zakresie działania i jego zadaniem jest konkretna reakcja na zdarzenie odpowiedzialne za jego aktywację. Po wykonaniu odpowiednich akcji moduł przestaje być aktywny, co umożliwi uruchomienie innych modułów.

W zakresie modelowania problemu architektura tablicowa poprzez zastosowanie różnych metod reprezentacji wiedzy (np. reguły, ramy) umożliwia przedstawienie modelu problemu w ujęciu statycznym. Dynamikę problemu uwzględniają takie elementy architektury tablicowej, jak: zdarzenia (zmiany stanów obiektu), warunki i akcje. Architektura tablicowa, w porównaniu z innymi modelami reprezentacji wiedzy, jest więc najbardziej zbliżona do naturalnego sposobu rozwiązywania problemów przez człowieka.

### 5.5. Proces tworzenia systemu eksperckiego

Całość problematyki związanej z budową systemów wiedzy implementowanych na sprzęcie komputerowym, w tym systemów eksperckich, jest jednym z działów sztucznej inteligencji nazywanym inżynierią wiedzy (*knowledge engineering*). Proces wdrażania systemu ze sztuczną inteligencją jest pewnym szczególnym przypadkiem procesu wdrażania oprogramowania.

Na rysunku 5.10 przedstawiono podstawowe etapy budowy systemu eksperckiego.



Rys. 5.10. Tworzenie systemu eksperckiego [15]

Należy podkreślić, że w praktyce pomiędzy wyszczególnionymi etapami następuje zacieranie się granic, a ponadto między nimi istnieją sprzężenia zwrotne, co znacznie utrudnia sformalizowanie procesu oraz wymaga od inżyniera wiedzy inwencji twórczej.

Odpowiednio etap identyfikacji i nabywania wiedzy odpowiada tzw. specyfikacji, czyli określaniu opisu problemu. Formalizacji wiedzy podporządkować można proces projektowania, czyli zapisania programu w konkretnym języku. Etap implementacji jest równoważny z procesem programowania. Ostatni etap – testowania systemu – odpowiada procesowi uruchomienia i testowania programu.

Na pierwszym etapie następuje *identyfikacja* celów użytkowych systemu, tzn. określa się, co powinno być osiągnięte za pomocą nowego systemu. W rezultacie tych działań specyfikowane są zadania. Identyfikacja problemu to takie rozpoznanie sytuacji problemowej użytkownika lub środowiska danej branży, które umożliwi scharakteryzowanie przedmiotów projektowania w sposób wyczerpujący i w skali określonej granicami kompetencji ekspertów. Inżynier wiedzy musi poznać znaczeniowy i logistyczny obszar sytuacji problemowej i uzyskać możliwość jego oceny. Często w toku takiego postępowania konieczne są liczne iteracje, co zmusza inżyniera wiedzy do modyfikacji jego wcześniejszych opisów lub modeli. Na tym etapie następuje także identyfikacja uczestników projektu. Na samym początku prac powinny zostać również określone źródła wiedzy. Z reguły podstawowym źródłem wiedzy jest ekspert dziedzinowy (*domain expert*) – specjalista lub zespół specjalistów z dziedziny, w której ma pracować budowany system ekspercki. Do podstawowych źródeł wiedzy należą również wszelkiego rodzaju publikacje z rozważanej dziedziny, np. książki, artykuły. Źródłem wiedzy może być także Internet. W działaniach związanych z nabywaniem wiedzy ważna jest umiejętność jej znajdowania i ukazywania, ograniczania i ujmowania (rejestrowania, pojmowania, uchwycenia), formalizowania i przedstawiania systemowi w przynależnej formie.

Zadaniem *konceptualizacji* jest dokładne wypracowanie ważniejszych koncepcji, które są znaczące i niezbędne w praktycznej akwizycji wiedzy. Inżynier wiedzy określa więc sposób, w jaki ekspert będzie postępował podczas rozwiązywania problemu. Zarówno inżynier wiedzy, jak i ekspert rozstrzygają i orzekają wspólnie, jakie są jeszcze wymagane koncepcje, aby opis przyjętego sposobu rozwiązania problemu stał się całościowy dla danego obszaru zastosowań technologii informacyjnych.

Proces zapisywania zgromadzonej wiedzy w wybranej metodzie reprezentacji wiedzy nazywa się *formalizacją wiedzy*. Inżynier wiedzy dokonuje selekcji reprezentacji wiedzy poprzez wybór odpowiedniego narzędzia. Wybór spośród narzędzi dostępnych na rynku następuje po analizie ich następujących cech:

- metod reprezentacji wiedzy,
- strategii wnioskowania,
- reprezentacji wiedzy niepewnej,
- realizacji modułu we/wy systemu,

- realizacji modułu wyjaśniającego „zachowanie” systemu,
- możliwości połączenia z zewnętrznymi źródłami wiedzy,
- istnienia modułu nabywania wiedzy,
- wymagań co do sprzętu komputerowego.

Wyboru odpowiedniego narzędzia dokonuje inżynier wiedzy. Decydującym kryterium jest sposób reprezentacji wiedzy.

Panuje powszechna opinia, że formalizowanie wiedzy jest zdecydowanie najtrudniejszym etapem budowy systemu eksperckiego [166]. Wynika to głównie z faktu, że metody formalizowania wiedzy opierają się na heurystycznych metodach bazujących na intuicji i doświadczeniu inżyniera wiedzy. Zarówno ekspert, jak i inżynier wiedzy dysponują pewnymi zasobami wiedzy, na bazie których poprzez wzajemną wymianę kontrolowaną przez inżyniera wiedzy dochodzi do jej określenia. Ekspert niejednokrotnie dopiero w trakcie pracy z inżynierem wiedzy uświadamia sobie, jakie faktycznie stosuje reguły. Na ogół trudno opisać formalne związki, które są uchwytnie intuicyjnie [166].

Zbudowanie pracującego systemu eksperckiego z wykorzystaniem wybranego narzędzia na określonym sprzęcie komputerowym nazywa się *implementacją* systemu eksperckiego. Ten etap prac całkowicie podlega inżynierowi wiedzy. Wynikiem etapu implementacji jest tzw. system prototypowy, który po odpowiednim przetestowaniu jeszcze wielokrotnie będzie podlegał przebudowie, czyli powrotowi do wcześniejszych faz. W wyniku realizacji poprzednich etapów budowy otrzymany prototypowy system jest na ogół niekompletny i niesprawdzony oraz wymagający przetestowania i poprawek. Metody *testowania* bazują na testowaniu klasycznych systemów informatycznych.

Podsumowując można zauważyć, że nowe sposoby widzenia problemów zarządzania produkcją kładą nacisk na wiedzę w przeciwieństwie do formalnych metod rozumowania. Jest to konsekwencją podejść, w których „paradygmat systemów ekspertowych pochodzi z wiedzy jaką one posiadają, a nie z formalizmów i schematów wnioskowania, jakie stosują. Wiedza eksperta jest kluczem do systemu, podczas gdy reprezentacja wiedzy i schematy wnioskowania dostarczają mechanizmów do jej użycia” [81]. „Ten sposób widzenia wiedzy kładzie nacisk na wiedzę w przeciwieństwie do formalnych metod rozumowania” [102]. B. Hayes-Roth [81] objaśnia to podając trzy powody:

1. Większość trudnych, a interesujących problemów, wywodzących się ze złożonych środowisk fizycznych czy społecznych, nie posiada prostych rozwiązań algorytmicznych.
2. Jak już wspomniano, eksperci osiągają dobre efekty dzięki swej wiedzy w specyficznej dziedzinie.
3. Istnieje potrzeba posiadania wiedzy w coraz bardziej złożonym świecie.

W opisanych wyżej metodach następuje przejście od przetwarzania danych do przetwarzania wiedzy. Systemy te określa się więc także jako systemy oparte na wiedzy (*knowledge-based systems*). Metody algorytmiczne – charakterystyczne dla przetwarzania proceduralnego – zostają w sztucznej inteligencji zastąpione *przeszukiwaniem inteligentnym* (*intelligent search*).

## 6. Systemy eksperckie w zarządzaniu produkcją

Systemy eksperckie znajdują zastosowanie w różnych obszarach wytwórczej działalności człowieka. Wykorzystywane są przede wszystkim wtedy, kiedy sytuacje rzeczywiste w przedsiębiorstwie są trudne do zdefiniowania i sformalizowania poprzez budowę czysto matematycznych modeli. Systemy eksperckie są także odpowiednimi środkami dla planowania i sterowania przebiegiem produkcji, zdobywającymi coraz większe uznanie i popularność [3, 40, 98, 115, 121, 132, 164, 168]. Stanowią one problemowo zorientowane programy komputerowe wspomagające podejmowanie decyzji. W wielu przypadkach, celem szeroko zakrojonych i intensywnie prowadzonych w tym obszarze badań jest integracja znanych technik wspomagania komputerowego z systemami eksperckimi. Największe zastosowanie znajdują obecnie systemy komputerowego wspomagania planowania i sterowania produkcją, które wykorzystują osiągnięcia trzech dyscyplin naukowych: sztucznej inteligencji, badań operacyjnych i sterowania. Przykładem mogą tu być hybrydowe modele łączące modele zamkniętej sieci kolejek z systemami eksperckimi [30] oraz systemy będące kombinacją sieci Petri i systemu eksperckiego [195].

Pewną reprezentację wiedzy dla harmonogramowania produkcji zaproponowali m.in. S.J. Park, J.W. Kim i H.W. Kang [168].

Analizę reguł stosowanych w systemach wytwórczych zawierają publikacje [79, 121, 153, 163]. Przykłady zastosowania systemów eksperckich w zarządzaniu produkcją można znaleźć także w polskich opracowaniach, jak np. w książce pod redakcją R. Knosali [104] lub pracach [126, 127, 128, 130].

W literaturze opisano wiele przykładów zastosowań systemów eksperckich w zarządzaniu produkcją. Poniżej zaprezentowano wybrane przykłady zastosowań systemów eksperckich w zarządzaniu produkcją na różnych jego poziomach.

### 6.1. Zastosowanie systemu eksperckiego do podejmowania decyzji wytwórz / kup

Systemy eksperckie mogą wspomagać strategiczne zarządzanie przedsiębiorstwem. Przykładem takiego systemu jest system ekspercki wspomagania decyzji *make or buy* (wytwórz/kup), którego twórcami są: R.T. McIvor, M.D. Mulvenna oraz P. Humphreys [147].

Decyzja wytwórz/kup stanowi fundament strategicznego zarządzania produkcją, ponieważ bardzo często determinuje rentowność przedsiębiorstwa. Problem wytwórz/kup jest analizowany z punktu widzenia poziomu integracji i rozpiętości procesów zarządzania w gospodarce oraz sposobów oceny sprzedaży. W podejmowaniu decyzji wytwórz/kup szczególnie ważne są kwestie projektowania urządzeń i zdolności produkcyjnych oraz ich wpływu na rozwój produktu.

Model konceptualny systemu powstał nie tylko w oparciu o przegląd literatury. Początkowe fazy tworzenia systemu obejmowały także liczne wywiady z menedżerami działów zaopatrzenia.

Celem tworzenia systemu eksperckiego było dostarczenie przedsiębiorstwom formalnej metody do analizy decyzji wytwórz/kup, która może być referowana do:

- analizy technicznych możliwości i profili organizacyjnych potencjalnych dostawców,
- porównania technicznych możliwości jego wewnętrznych operacji w relacji do najlepszych potencjalnych dostawców w przemyśle,
- umożliwienia grupie użytkowników systemu przeprowadzenia pełnej oceny czynników, które muszą być rozważane, kiedy rezygnuje się z partnerskich powiązań z dostawcami (dotyczy to długookresowych powiązań z dostawcami),
- magazynowania informacji niezbędnych do podejmowania decyzji o zleceniach wytwórz/kup,
- analizy kosztów związanych z zastosowaniem innych strategii wytwórz/kup,
- zespołowego stylu interakcji, w którym dialog pomiędzy zespołem menedżerskim a systemem będzie w postaci pytań z menu dla wybieranych opcji,
- analizy „co-jeżeli?” w zleceniach kontroli wpływu danych wejściowych na rezultaty.

Model systemu eksperckiego do wspomagania decyzji wytwórz/kup prezentuje rysunek 6.1.

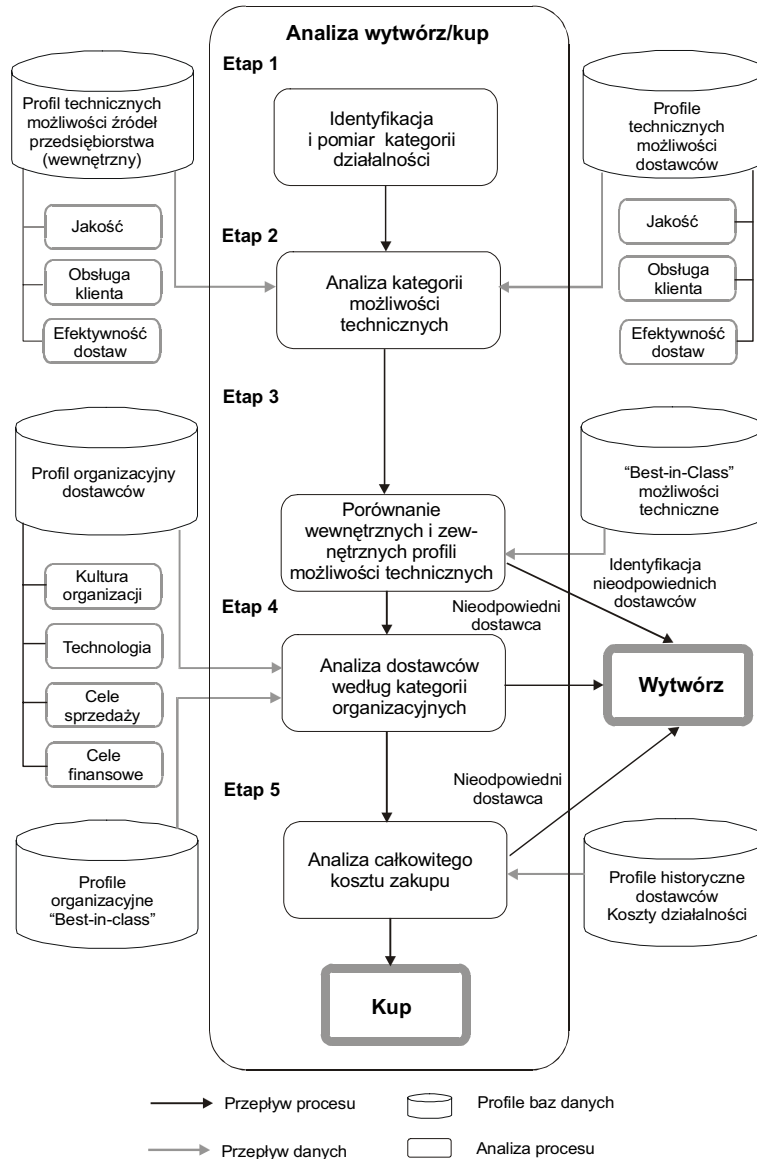
Przedstawiony na rysunku 6.1 system ekspercki wykorzystuje bazy danych jako końcowe magazyny danych, dzięki czemu firmy używające go będą miały system oceniający sprzedaż z następującymi informacjami o:

- działalności dostawców, z którymi zawarły umowę dotyczącą jakości oraz dostaw i obsługi klienta,
- ocenie „best-in-class” dla danego przemysłu,
- ocenie kosztów załamania dostaw,
- dostawcach i ich wynikach finansowych oraz kulturze organizacyjnej.

Podczas procesu interakcji system będzie określał dostawców o idealnej charakterystyce dla aktualnego kontraktu opierając się na powyższej bazie danych.

Analizowany model obejmuje pięć etapów podejmowania decyzji wytwórz/kup:

- identyfikację kategorii i pomiar osiągnięć,
- analizę technicznych możliwości,
- porównywanie wewnętrznych i zewnętrznych technicznych możliwości,
- analizę możliwości organizacyjnych,
- analizę całkowitego kosztu zakupu.



Rys. 6.1. Model podejmowania decyzji wytwórz/kup [147]

W pierwszym etapie procesu identyfikuje się kluczowe kategorie oceny. Do podstawowych kategorii zalicza się: kategorie technicznych możliwości oraz kategorie organizacyjnych możliwości. Są one niezbędne do identyfikacji, projektowania i wytwarzania. Celem tego etapu jest klasyfikowanie dostawców w łańcuchu dostaw. Wydajność potencjalnych źródeł dostaw (wewnętrznych i zewnętrznych) jest oceniana według tych kategorii i kryteriów identyfikacji w następnych etapach. Ważną kwestią, którą rozpatrują menedżerowie jest oce-

na możliwości technicznych podobnych przedsiębiorstw za granicą. Etap porównywania wewnętrznych i zewnętrznych profili technicznych możliwości rozwija wcześniejszą ocenę źródeł dostaw opierając się na kryteriach identyfikacji „best-in-class”. System będzie „filtrował” każdego potencjalnego dostawcę i porównywał całkowity wynik otrzymany ze zbioru ocen z wartością bazową. Jeżeli wynik dla danego dostawcy będzie poniżej wartości bazowej, wtedy system oceni dostawcę jako nieodpowiedniego. Celem etapu analizy kategorii organizacyjnych jest ocena profilu organizacyjnego potencjalnego dostawcy, który pozytywnie został oceniony w poprzednich etapach. Taka analiza rozwijana jest w oparciu o cztery kategorie: kulturę organizacyjną, technologię, realizację celów sprzedaży oraz realizację celów finansowych. W ten sposób system będzie wyszukiwał nieodpowiednich dostawców. Końcowy etap to analiza całkowitych kosztów zakupu. Na tym etapie system będzie oceniał dostawców dotychczas potraktowanych jako odpowiednich i kwalifikował jako dostawców, u których można dokonać zakupu.

## 6.2. System ISIS

System ISIS (*Intelligent Scheduling and Information System*) określany jest mianem bazującego na wiedzy systemu wspierającego funkcje kierownicze [66, 169]. Cechą charakterystyczną tego systemu jest uwzględnianie przy podejmowaniu decyzji wielu ograniczeń. System ISIS umożliwia:

- planowanie asortymentu i wielkości produkcji,
- planowanie wymagań zasobowych,
- planowanie i harmonogramowanie przebiegu produkcji,
- wykrywania rozbieżności w realizacji harmonogramów.

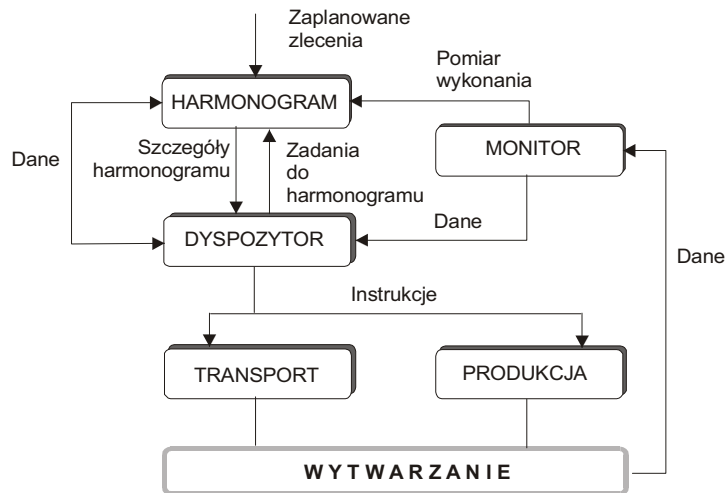
Harmonogramowanie realizowane jest na dwóch poziomach. Pierwszy jest związany z analizą powstawania „wąskich gardeł”, drugi zaś z wyznaczaniem harmonogramów szczegółowych. ISIS jest systemem interakcyjnym, w którym dialog z użytkownikiem jest prowadzony z wykorzystaniem standardowego zestawu opcji. System ten umożliwia bieżącą edycję danych, w tym również modelowanie uszkodzeń urządzeń technologicznych. Pozwala ponadto na prowadzenie analizy przebiegu procesów technologicznych na różnych poziomach szczegółowości, od poziomu elastycznego systemu produkcyjnego do pojedynczych obrabiarek. Stanowi przykład systemu eksperckiego wyposażonego w programy symulacji komputerowej, które umożliwiają graficzną ilustrację przebiegu dynamiki modelowanych procesów oraz różnych charakterystyk pracy systemu.

## 6.3. System COSIMA

Jednym z osiągnięć z zakresu komputerowego wspomaganie sterowania produkcją, z wykorzystaniem technik sztucznej inteligencji, jest system COSIMA zrealizowany w ramach międzynarodowego programu badań



ESPRIT-PROJECT 477 przez Digital Equipment Corporation, Renault Automobiles oraz University College Galway [140]. System COSIMA (*Control Systems for Integrated Manufacturing*) realizuje sterowanie opierając się na pięciu podstawowych modułach: *harmonogram*, *dyspozytor*, *transport*, *produkcja* i *monitor* (rys. 6.2).



Rys. 6.2. Moduły systemu COSIMA [140]

Moduł *harmonogram* ustala kolejkę zadań do realizacji w systemie produkcyjnym, z uwzględnieniem relacji czasowych. Moduł *dyspozytor* wysyła polecenia realizacji operacji zaplanowanych przez moduł *harmonogram* do modułu *transport* oraz modułu *produkcja*. Dysponowanie operacji odbywa się w czasie rzeczywistym, z uwzględnieniem zakłóceń w procesie produkcyjnym. Moduł *transport* w oparciu o żądania modułu *dyspozytor* zarządza wszystkimi urządzeniami transportowymi. Moduł *produkcja* steruje specyficznymi typami wyposażenia, takimi jak CNC maszyn i robotów (realizuje translację ogólnych instrukcji z modułu *dyspozytor* na specyficzne instrukcje urządzeń). Ponadto realizuje przekazywanie danych do modułu *monitor*. Moduł *monitor* gromadzi dane o materiałach, wyposażeniu, stanie zarządzania oraz realizuje funkcje sprzężeń zwrotnych do odpowiednich modułów lub interfejsów innych użytkowników na wyższych szczeblach zarządzania. Według koncepcji oprogramowania systemu COSIMA przedstawionego na rysunku 6.2, na wejściu inżynier systemu poprzez interfejs profilu wytwarzania podaje m.in. następujące informacje źródłowe: o centrach obróbkowych, buforach, urządzeniach transportowych, wykazach materiałowych, procesach technologicznych i zleceniach produkcji. Te informacje przechowywane są w bazie danych profilu wytwarzania. Właściwe strategie harmonogramowania wskazywane są przez bazę reguł. Prezentowany system umożliwia nie tylko maszynowe generowanie harmonogramu. Możliwe jest także interaktywne tworzenie harmonogramu z udziałem

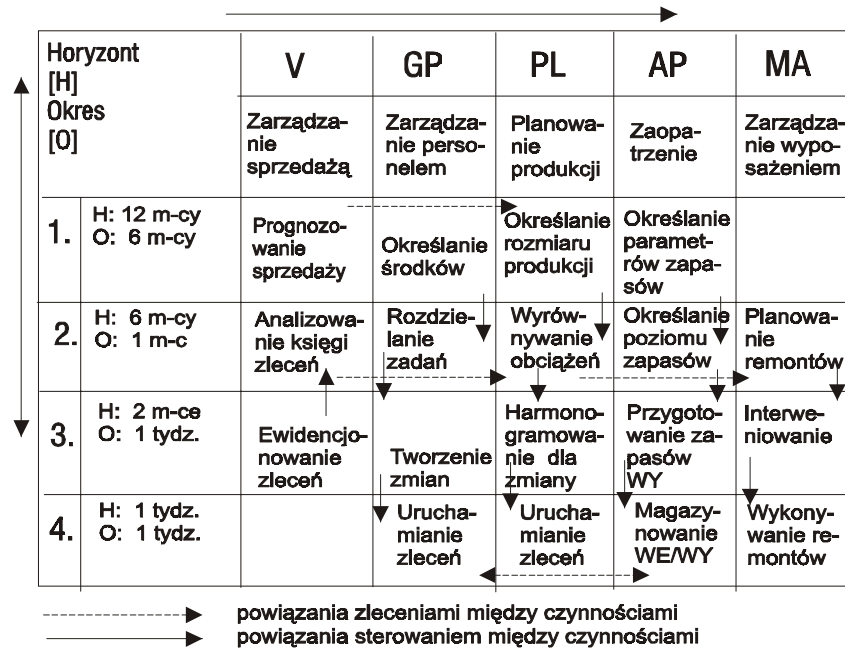
inżyniera systemu. Moduł dyspozytor implementuje harmonogram z ograniczeniami środowiska w czasie rzeczywistym. Emulator komórki produkcyjnej bazuje na sieci Petri oraz sztucznej inteligencji. Monitor gromadzi dane w czasie rzeczywistym i przekazuje je użytkownikowi (np. o stanach maszyn, zapasach robót w toku, stanach materiału). Istnieje możliwość dołączenia do systemu dodatkowego modułu (*monitor reporting*) w celu wykonywania sprawozdań do analiz w dłuższych okresach. Zastosowanie sieci typu magistrala umożliwia bezpośrednią komunikację między modułami.

#### 6.4. System ekspercki nadzorowania produkcji z architekturą tablicową

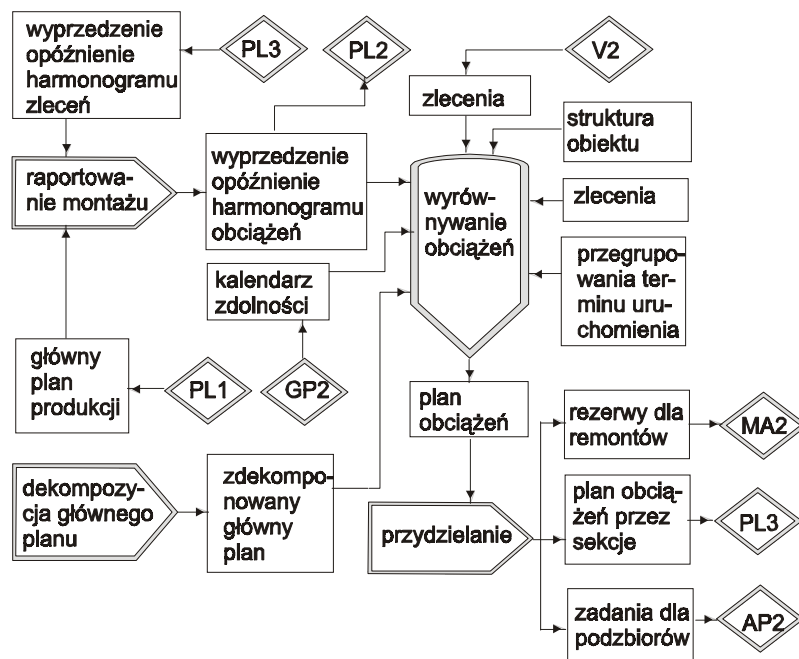
Nowatorską koncepcję w zakresie rozproszonych systemów sterowania, z wykorzystaniem technik sztucznej inteligencji opracowano w ramach międzynarodowego programu badań ESPRIT-PROJECT 932 [5]. Celem tych badań było zaprojektowanie systemu zarządzania produkcją w przedsiębiorstwie. Autorzy projektu opracowali oryginalną koncepcję modułu nadzorowania CMS (*Conceptual Supervision Model*), według której nadzorowanie w przedsiębiorstwie może być realizowane poprzez rozproszone lokalne jednostki nadzorowania (*supervision unit*), w różny sposób wzajemnie ze sobą powiązane. Organizacyjny wybór CSM przedstawiono na rysunku 6.3. Funkcję nadzorowania zdefiniowano jako:

- planowanie czynności zarządzania produkcją,
- aktywowanie czynności zarządzania produkcją,
- sterowanie akcją czynności zarządzania produkcją,
- reagowanie na zakłócenia w środowisku (np. awarie wyposażenia technicznego, braki magazynowe, absencje pracowników).

Przyjęto, że nadzorowanie powinno odbywać się w czasie rzeczywistym. Projektowanie systemu nadzorowania rozpoczęto od analizy zarządzania produkcją według metodologii GRAI, która dostarcza opisu systemu podejmowania decyzji. Autorzy metodologii GRAI zaproponowali przedstawiony na rysunku 6.3 model, który wyraża hierarchiczną strukturę podejmowania decyzji. Na każdym poziomie wyróżniono centrum decyzyjne, z którym związany jest zawsze pewien zbiór czynności. Wyodrębnienie poziomu następuje w związku z horyzontem i okresem, w jakim jest podejmowana decyzja. Krótki horyzont związany jest z decyzjami na niskim poziomie (np. centrum obróbkowego), dłuższy – dotyczy decyzji na poziomach wyższych. W myśl metodologii GRAI [53] dokonano więc rozbioru zarządzania produkcją wg dwóch kryteriów: funkcji zarządzania produkcją i czasu. Na podstawie tego rozbioru, w celu maksymalizacji wykorzystania zdolności produkcyjnych, zbudowano makrosieć czynności związanych z wyrównaniem obciążeń w systemie produkcyjnym, którą przedstawiono na rysunku 6.4.



Rys. 6.3. Przykład siatki GRAI [5]



Rys. 6.4. Makrosieć czynności związanych z „wyrównywaniem obciążeń” [5]

Zgodnie z organizacyjnym wyborem CSM przedstawionym na rysunku 6.3 oraz makrosiecią czynności związanych z wyrównywaniem obciążeń dokonano grupowania w zbiory czynności biorących udział w zarządzaniu produkcją. Metodologia GRAI narzuciła przy tym pewne ograniczenia, zgodnie z którymi grupowane mogą być tylko:

- czynności realizujące te same funkcje,
- czynności biorące udział w podejmowaniu tych samych decyzji,
- czynności odnoszące się do tych samych fizycznych konfiguracji,
- czynności, które są realizowane w tym samym czasie.

Zbiory czynności przyporządkowano odpowiednio do poszczególnych jednostek nadzorowania. Wyróżniony zbiór czynności wraz ze swoim blokiem sterującym tworzy jednostkę nadzorowania, co można przedstawić w postaci następującego symbolicznego zapisu:

$$JNi = \{\{czyn_k\}, BSi\} \quad (6.1)$$

gdzie:

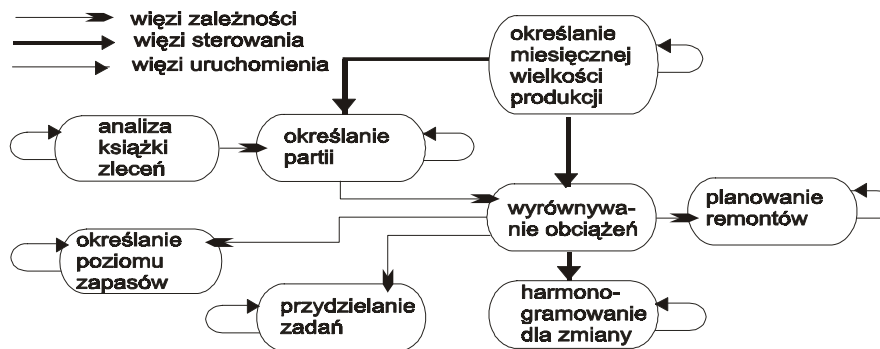
- $JNi$  – jednostka nadzorowania  $i$ ,
- $czyn_k$  – zbiór czynności,
- $BSi$  – blok sterujący jednostki nadzorowania  $i$ .

Każda jednostka nadzorowania realizuje określone cele poprzez podejmowanie decyzji w ramach pewnych ograniczeń narzuconych jej, m.in. przez pozostałe jednostki nadzorowania. I tak np. zadaniem  $JN2$  może być harmonogramowanie produkcji w danej komórce produkcyjnej. Pewne ograniczenia narzuca w takim przypadku  $JN1$ , której zadaniem jest planowanie produkcji dla wszystkich komórek produkcyjnych. W związku z tym blok sterujący  $BSi$  musi równocześnie pełnić następujące role w czasie rzeczywistym:

- scentralizowane nadzorowanie czynności (w danej jednostce nadzorowania  $JNi$ ),
- udział w nadzorowaniu rozproszonym.

Funkcja nadzorowania jest więc spełniana przez wszystkie bloki sterujące  $BSi$  razem.

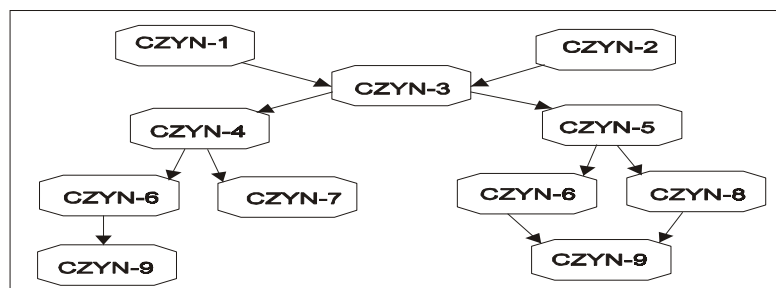
Interakcje między jednostkami nadzorowania wyrażają trzy typy więzi: sterowania, zależności i uruchomienia. Na rysunku 6.5 przedstawiono „zewnątrzną” sieć nadzorowania, w której węzły reprezentują jednostki nadzorowania, a łuki różne typy więzi między tymi jednostkami (sieć ta koresponduje z siecią przedstawiającą organizacyjny wybór CSM – rys. 6.4).



Rys. 6.5. Zewnętrzna sieć nadzorowania [5]

Jak wspomniano wyżej, decyzje podejmowane są na różnych poziomach obiektu, czego odbiciem są więzi sterowania wyrażające strukturę hierarchiczną procesu podejmowania decyzji. Więzy zależności pokazane na rysunku 6.5 wyrażają powiązania pomiędzy funkcjami, a więzi uruchomienia wyrażają zależności czasowe między jednostkami nadzorowania.

Realizację czynności nadzorowania wewnątrz każdej jednostki nadzorowania opisano grafem (rys. 6.6). Wierzchołki grafu to moduły, które realizują czynności wewnętrzne jednostki nadzorowania, a łuki pokazują możliwe serie czynności nadzorowania. Dla danego problemu blok sterujący ma zdolność podejmowania decyzji o wyborze „ścieżki rozwiązań problemu” na podstawie grafu.



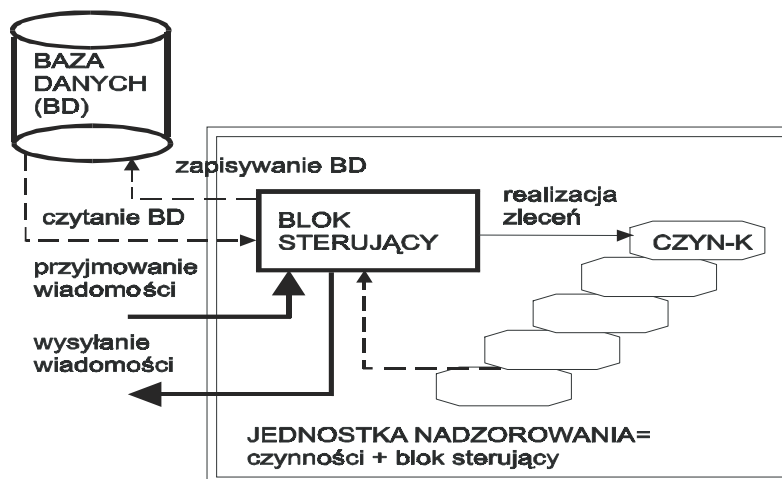
Rys. 6.6. Graf strategii jednostki nadzorowania [5]

Każdy blok sterujący jest oprogramowanym modulem, którego zadaniem jest aktualizowanie decyzji dotyczącej danej jednostki nadzorowania oraz interakcje z innymi jednostkami nadzorowania (rys. 6.7). Zatem rolą bloku sterującego jest:

- odpowiednie uruchamianie czynności jednostki nadzorowania (poprzez dokonanie wyboru ścieżki rozwiązań problemu w grafie),
- uwzględnienie uzupełniających raportów z wnętrza jednostki nadzorowania po to, aby aktywnie i szczegółowo opracować plan rozwiązania problemu,

- analizowanie wiadomości przychodzących z innych bloków sterujących do określenia ich wpływu na opracowanie swojej własnej decyzji,
- identyfikowanie sytuacji, które będą wymagały wysyłania wiadomości do innych jednostek nadzorowania,
- tworzenie i wysyłanie wiadomości do innych jednostek nadzorowania,
- aktualizacja wydzielonych danych, które są modyfikowane przez jednostkę nadzorowania.

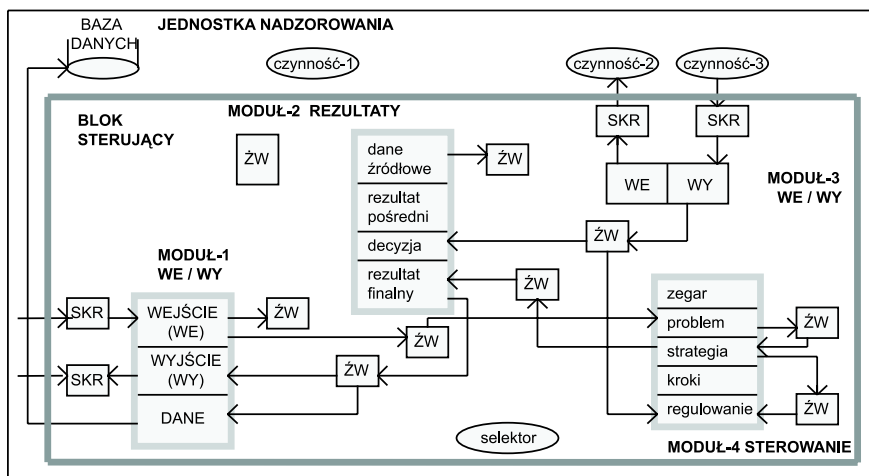
Każdy blok sterujący realizuje scentralizowane podejmowanie decyzji dotyczących czynności swojej jednostki nadzorowania i bierze udział w rozproszonym sterowaniu interakcji między jednostkami nadzorowania. Te dwie role się przeplatają, co oznacza, że każdy blok sterujący musi być systemem zdolnym do modyfikacji swoich zachowań w kategoriach wewnętrznych zdarzeń (bieżącego stanu swojego procesu decyzyjnego) i zewnętrznych zdarzeń (wiadomości przyjmowanych z innych jednostek nadzorowania, stanu środowiska).



Rys. 6.7. Wewnętrzna struktura jednostki sterowania [5]

Do oprogramowania bloku sterującego wykorzystano osiągnięcia z zakresu sztucznej inteligencji. Architekturę bloku sterującego pokazano na rysunku 6.8. Tworzą ją: źródła wiedzy – ŻW, moduły pamięci roboczej zorganizowanej w poziomy hierarchiczny i mechanizm sterowania (selektor).

Źródła wiedzy zawierają wiedzę, która umożliwia rozwiązywanie częściowych problemów. Do reprezentacji wiedzy w źródle wiedzy wykorzystuje się rachunek predykatów. Każde źródło wiedzy zawiera część warunkową i część akcji. Część warunkowa jest wyrażona w kategoriach faktów zawartych w pamięci roboczej.



Rys. 6.8. Architektura bloku sterującego jednostki nadzorowania [5]

Mechanizm sterowania kieruje procesem rozwiązywania problemu poprzez próbowanie dopasowania faktów z pamięci roboczej do warunków reguł, a następnie akcji polegającej na zmianie pamięci roboczej, a więc występuje tzw. łańcuch do przodu.

W architekturze bloku sterującego można wyróżnić cztery moduły pamięci roboczej (*blackboard*): MODUŁ-1 WE/WY, MODUŁ-2 REZULTATY, MODUŁ-3 WE/WY, MODUŁ-4 STEROWANIE. Każdy z tych modułów zawiera fakty opisujące wewnętrzne i zewnętrzne sytuacje jednostki nadzorowania. MODUŁ-1 WE/WY zawiera dwa poziomy. Pierwszy do magazynowania wiadomości otrzymywanych z innych jednostek nadzorowania, drugi do magazynowania wiadomości tworzonych przez blok sterujący przeznaczonych dla innych jednostek nadzorowania. MODUŁ-2 REZULTATY wykorzystywany jest do lokalnego magazynowania danych źródłowych, rezultatów pośrednich i rezultatów finalnych procesu rozwiązywania problemu. MODUŁ-3 WE/WY ma strukturę dwupoziomową. Pierwszy poziom przeznaczony jest do magazynowania zleceń uruchamiania wewnętrznych czynności jednostki nadzorowania, drugi – do magazynowania uzupełniających raportów dotyczących wykonywania tych czynności. MODUŁ-4 STEROWANIE magazynuje rozwiązywane problemy, wybór strategii, bieżący stan procesu i zaakceptowane ograniczenia podejmowania decyzji, a także informację wykorzystywaną do synchronizacji czynności.

W bloku sterującym funkcjonują ponadto cztery skrytki (SKR), które zawierają:

- komunikaty z innych bloków sterujących,
- komunikaty do wysłania do innych jednostek nadzorowania,
- zlecenia do wewnętrznych czynności JN,
- uzupełnienia raportów o tych czynnościach.

Wymianę komunikatów pomiędzy jednostkami nadzorowania umożliwia specjalny protokół.

Na podstawie zaprezentowanej wyżej koncepcji oprogramowano jednostkę nadzorowania i dokonano próbnej eksploatacji, symulując pracę innych jednostek nadzorowania przez przyjmowanie różnego rodzaju wiadomości dotyczących wyrównywania obciążeń. Tę implementację zrealizowano w Common Lisp na komputerze SUN3. Próby zastosowań przemysłowych prowadzono w zakładach: PHILIPS (produkujących radia samochodowe) i PIRELLI (produkujących opony).



## 7. Algorytmy ewolucyjne

### 7.1. Wprowadzenie

W ramach kierunku badań pod pojęciem *algorytmy ewolucyjne* są rozwijane różne metody: algorytmy genetyczne, strategie ewolucyjne, programowanie ewolucyjne [148]. Jednakże ze względu na zacieranie się różnic między nimi zazwyczaj tych badań nie klasyfikuje się [54]. Najczęściej używa się pojęć *algorytm genetyczny* – GA (*Genetic Algorithm*) lub *algorytm ewolucyjny* – EA (*Evolutionary Algorithm*) [65, 177]. Takie podejście przyjęto również w niniejszej pracy.

Początku algorytmów genetycznych można szukać we wczesnych latach pięćdziesiątych ubiegłego stulecia. Jednakże dopiero prace wykonane w późnych latach sześćdziesiątych i wczesnych siedemdziesiątych w Uniwersytecie w Michigan pod kierownictwem J.H. Hollanda doprowadziły do takiego pojmowania algorytmów genetycznych, jak się je rozumie współcześnie [85]. Dynamiczny w ostatnich latach rozwój algorytmów genetycznych, jest rezultatem poszukiwań efektywnych sposobów rozwiązania problemów optymalizacyjnych, zapoczątkowanych w latach osiemdziesiątych ubiegłego stulecia przez D.E. Goldberga [74].

*Algorytmy genetyczne* są to algorytmy poszukiwania, oparte na mechanizmach doboru naturalnego oraz dziedziczności. Łącząc w sobie ewolucyjną zasadę przeżycia najlepiej przystosowanych osobników z systematyczną, choć zrandomizowaną wymianą informacji, tworzą metodę poszukiwania obdarzoną jakąś dozą pomysłowości właściwą umysłowi ludzkiemu [73]. Pomimo elementu losowości, algorytmy genetyczne nie sprowadzają się do zwykłego błędzenia przypadkowego. Wykorzystują one efektywnie przeszłe doświadczenia do określenia nowego obszaru poszukiwań o spodziewanej podwyższonej wydajności [73].

W naturze *osobnikami* są żywe organizmy. W algorytmie ewolucyjnym *osobnikami populacji* są zakodowane w postaci *chromosomów* zbiory *parametrów zadania*, czyli potencjalne rozwiązania, określane też jako *punkty przestrzeni poszukiwań* (*search points*). *Chromosomy* – inaczej *łańcuchy* lub *ciągi kodowe* – to uporządkowane *ciągi genów*.

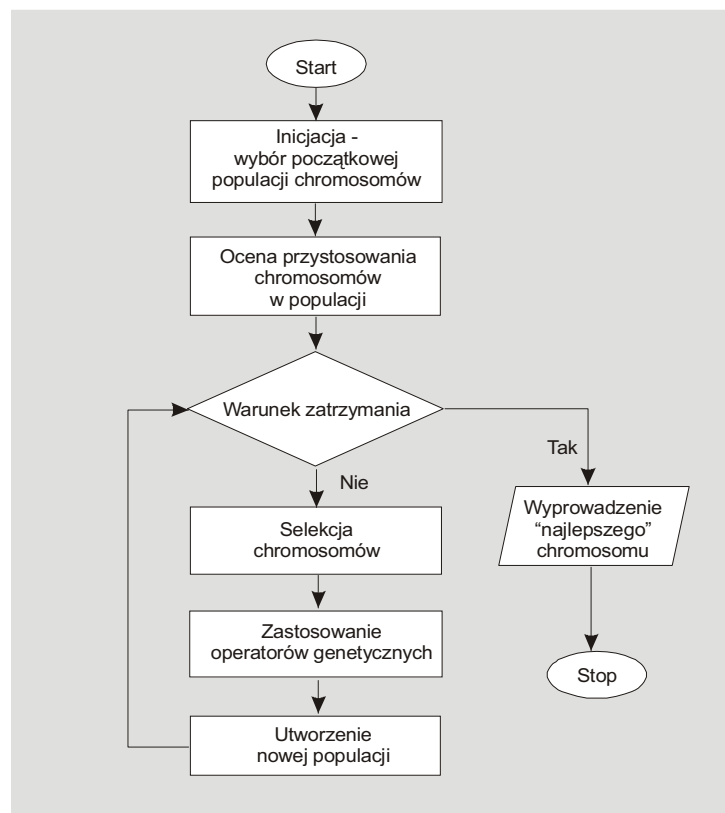
Wybór populacji początkowej wiąże się z przedstawieniem parametrów zadania (problemu) w postaci chromosomów. Dotyczy to sposobu kodowania. W klasycznym algorytmie genetycznym przyjmuje się kodowanie binarne. Długość chromosomu zależy od warunków zadania, a ściślej mówiąc – od ilości punktów przestrzeni poszukiwań. Chromosomami mogą być takie struktury, jak np.: wektory liczb rzeczywistych, macierze, czy harmonogramy produkcji.

W naturze w życiu gatunków można rozróżnić pokolenia. Podobnie w algorytmie ewolucyjnym istnieją momenty, w których z danej populacji tworzona jest nowa, określana mianem następnej generacji. Zazwyczaj osobniki lepiej przystosowane mają większe prawdopodobieństwo utworzenia osobników potomnych

wchodzących do następnej generacji. Czasem bywa także tak, że osobniki najlepiej przystosowane ze starej generacji mogą się znaleźć w niezmienionej postaci w nowej generacji. Proces tworzenia nowych osobników odbywa się zazwyczaj przez *rekombinację* (krzyżowanie) chromosomów, w których zapisane są informacje o strukturze osobników będących rodzicami. Tworzenie nowych osobników nie polega jednak tylko na kombinowaniu informacji o strukturze rodziców. Mogą tu także wystąpić pewne niewielkie losowe zmiany w strukturze chromosomów potomnych, określane mianem *mutacji*. Mimo, że takie podejście jest znacznym uproszczeniem procesów ewolucyjnych z biologicznego punktu widzenia, to jak wykazano w dalszych rozdziałach niniejszej pracy algorytmy ewolucyjne są wystarczająco mocnym narzędziem gwarantującym wysoką jakość procesu poszukiwania rozwiązań.

## 7.2. Ogólna struktura algorytmu ewolucyjnego

Na klasyczny algorytm genetyczny składają się kroki (rys. 7.1):



Rys. 7.1. Schemat blokowy algorytmu genetycznego [177]

*Inicjacja*, czyli utworzenie populacji początkowej, to losowy wybór żądanej liczby chromosomów (osobników) reprezentowanych przez ciągi binarne o określonej długości.

*Ocena przystosowania chromosomów w populacji* polega na obliczeniu funkcji przystosowania dla każdego chromosomu z tej populacji. Im większa wartość tej funkcji, tym lepsza „jakość” chromosomu.

*Sprawdzenie warunków zatrzymania.* Określenie warunku zatrzymania algorytmu genetycznego zależy od konkretnego zastosowania tego algorytmu, jak np.:

- a) w zagadnieniach optymalizacji, jeśli znana jest wartość maksymalna (lub minimalna) funkcji przystosowania, zatrzymanie algorytmu może nastąpić po uzyskaniu żądanej wartości optymalnej, ewentualnie z określoną dokładnością,
- b) zatrzymanie algorytmu może również nastąpić, jeśli dalsze jego działanie nie poprawia już uzyskanej wartości,
- c) algorytm może także zostać zatrzymany po upływie określonego czasu działania lub po określonej ilości iteracji.

Jeśli warunek zatrzymania jest spełniony, następuje przejście do ostatniego kroku, czyli wyprowadzenia „najlepszego” chromosomu. Jeśli nie, to następnym krokiem jest selekcja.

*Selekcja chromosomów* polega na wybraniu, na podstawie obliczonych wartości funkcji przystosowania tych chromosomów, które będą brały udział w tworzeniu potomków do następnego pokolenia, czyli następnej generacji. Wybór ten odbywa się zgodnie z zasadą naturalnej selekcji, tzn. największe „szanse” na udział w tworzeniu nowych osobników mają chromosomy o największej wartości funkcji przystosowania.

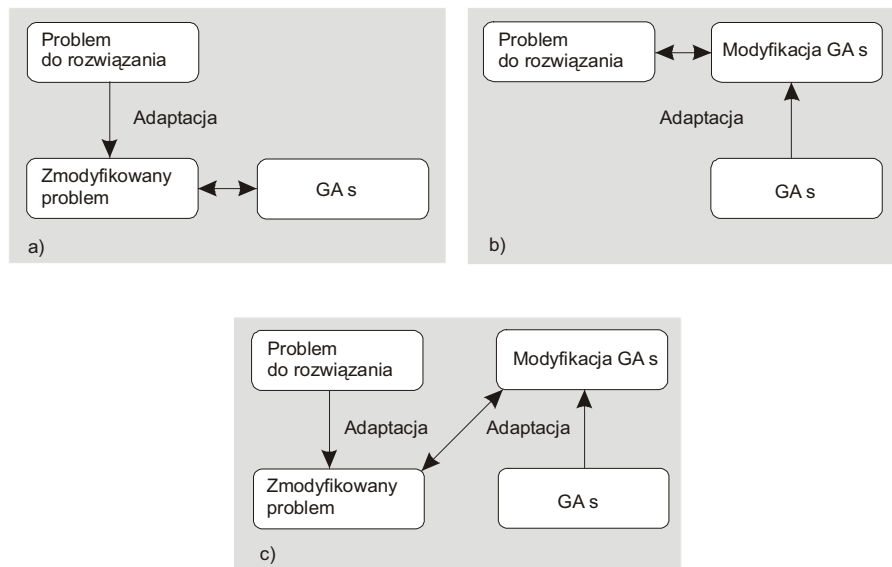
*Zastosowanie operatorów genetycznych* do chromosomów wybranych metodą selekcji prowadzi do utworzenia nowej populacji (potomków), powstałej z wybranej metodą selekcji populacji rodziców. W klasycznym algorytmie genetycznym stosuje się dwa podstawowe operatory genetyczne: *operator krzyżowania (crossover)* oraz *operator mutacji (mutation)*.

Należy jednak zaznaczyć, że operator mutacji odgrywa zdecydowanie drugoplanową rolę w stosunku do operatora krzyżowania. W algorytmie genetycznym mutacja chromosomu może być dokonywana na populacji rodziców przed operacją krzyżowania lub na populacji potomków utworzonych w wyniku krzyżowania.

*Utworzenie nowej populacji.* Chromosomy otrzymane w wyniku działania operatorów genetycznych na chromosomy tymczasowej populacji rodzicielskiej wchodzi w skład nowej populacji. Populacja ta staje się tzw. populacją bieżącą dla danej iteracji algorytmu genetycznego.

*Wyprowadzenie „najlepszego” chromosomu.* Jeżeli spełniony jest warunek zatrzymania algorytmu genetycznego, należy wyprowadzić wynik działania algorytmu, czyli podać rozwiązanie problemu. Najlepszym rozwiązaniem jest chromosom o największej wartości funkcji przystosowania.

W nurcie badań nad zastosowaniem algorytmów genetycznych można wyróżnić trzy podstawowe podejścia zaprezentowane na rysunku 7.2 [37].



Rys. 7.2. Trzy podejścia do zastosowań algorytmów genetycznych [37]

W pierwszym podejściu klasyczny algorytm genetyczny operuje na chromosomach w postaci ciągów binarnych (kodowaniu binarnym). Algorytmy genetyczne operujące na takich ciągach wymagają modyfikacji oryginalnego problemu do problemu w postaci odpowiedniej dla algorytmu genetycznego (rys. 7.2a).

W drugim podejściu pozostawia się rzeczywisty problem niezmieniony, a reprezentacja chromosomów odpowiada potencjalnym rozwiązaniom z użyciem naturalnych struktur danych. W takim przypadku odpowiednio modyfikowane są operatory genetyczne (rys. 7.2b).

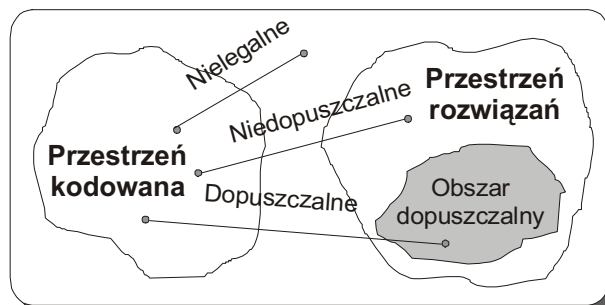
Trzecie podejście to adaptacja zarówno problemu do algorytmu genetycznego, jak i algorytmu genetycznego do problemu (rys. 7.2c).

### 7.3. Reprezentacja

Wybór populacji początkowej wiąże się z przedstawieniem parametrów zadania w postaci chromosomów. Dotyczy to sposobu kodowania. W klasycznym algorytmie genetycznym przyjmuje się kodowanie binarne, tzn. allele wszystkich genów w chromosomie są równe 0 lub 1.

Specyfikacja przestrzeni rozwiązań, jaka będzie przeszukiwana przez algorytm ewolucyjny jest jedną z najbardziej istotnych decyzji przy stosowaniu podejść ewolucyjnych do rozwiązania określonych problemów. Istnieją różne

możliwości odwzorowania między punktem w przestrzeni reprezentacji (*coding space*) – chromosomem, a punktem w przestrzeni rozwiązań (*solution space*) – np. harmonogramem. Na rysunku 7.3 przedstawiono odwzorowanie punktów w algorytmach ewolucyjnych. Jak z tego rysunku wynika, niektóre zbiory przestrzeni rozwiązań mogą stanowić zbiory znajdujące się poza przestrzenią zbiorów rozwiązań dopuszczalnych. Należy unikać takich reprezentacji, w których możliwe jest uzyskanie rozwiązań nie spełniających ograniczeń zadania. Ponadto od stworzenia odpowiedniej reprezentacji zależy szybkość i łatwość wykonania etapów GA – krzyżowania, mutacji, selekcji. Uzasadnione jest więc intensywne rozwijanie badań mających na celu określenie najlepszej reprezentacji dla problemów planowania produkcji.



Rys. 7.3. Kodowanie w algorytmach genetycznych [37]

#### 7.4. Metody selekcji

*Metoda selekcji oparta na kole ruletki (roulette wheel selector)* stanowi w algorytmach genetycznych podstawową metodę wyboru osobników do populacji rodzicielskiej w celu dalszego przetwarzania za pomocą operatorów genetycznych. Mimo, że jest to losowa procedura selekcji to osobniki rodzicielskie wybierane są proporcjonalnie do wartości ich funkcji przystosowania, tzn. zgodnie z prawdopodobieństwem selekcji. Wadą metody jest możliwość jej stosowania do jednej klasy zadań, tzn. tylko maksymalizacji. Drugą słabą stroną jest fakt, że osobniki o bardzo małej wartości funkcji przystosowania są zbyt wcześnie eliminowane z populacji, co może doprowadzić do przedwczesnej zbieżności algorytmu genetycznego [177].

W *metodzie selekcji turniejowej (tournament selection)* dzieli się osobniki populacji na podgrupy, a następnie z każdej z nich wybiera się osobniki o najlepszym przystosowaniu. Rozróżnia się dwa sposoby takiego wyboru: deterministyczny (z prawdopodobieństwem równym 1) i losowy (z prawdopodobieństwem mniejszym niż 1). Podgrupy mogą być dowolnego rozmiaru, najczęściej dzieli się populację na podgrupy złożone z 2 lub 3 osobników. Metoda turniejowa nadaje się zarówno do maksymalizacji, jak i minimalizacji funkcji. Oprócz tego może być łatwo rozszerzona na zadania dotyczące optymalizacji

wielokryterialnej. W metodzie turniejowej można zmieniać rozmiar podgrup, na jakie dzielona jest populacja.

W *selekcji rankingowej* (*ranking selection*) osobniki populacji są ustawiane kolejno w zależności od wartości funkcji przystosowania. Można to sobie wyobrazić jako listę rankingową osobników uszeregowanych od najlepiej do najgorzej przystosowanych (lub odwrotnie), gdzie każdemu osobnikowi przypisana jest liczba określająca jego kolejność na liście i nazywana rangą (*rank*). Liczba kopii każdego osobnika, wprowadzonego do populacji rodzicielskiej  $M(k)$ , jest ustalana zgodnie z wcześniej zdefiniowaną funkcją, zależną od rangi osobnika. Zaletą metody rankingowej jest możliwość wykorzystania jej zarówno do maksymalizacji, jak i minimalizacji funkcji. Nie napotyka ona również na konieczność skalowania w związku z problemem przedwczesnej zbieżności, co może wystąpić przy zastosowaniu metody ruletki.

## 7.5. Operatory krzyżowania i mutacji

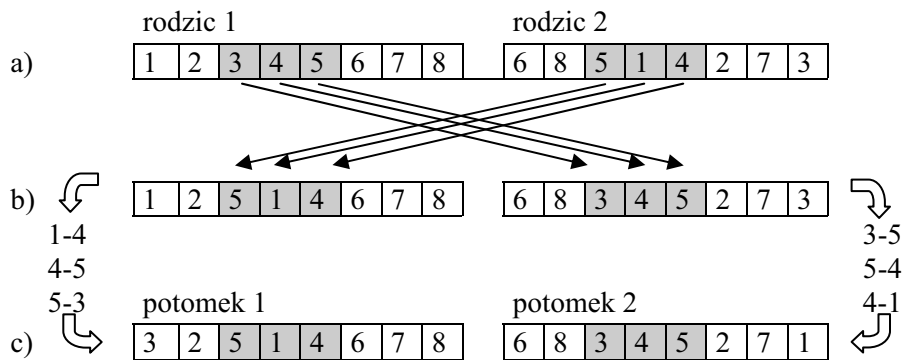
Ewolucyjny rozwój populacji chromosomów odbywa się przez mechanizm reprodukcji, na który składają się procesy krzyżowania, mutacji i inwersji. W procesie krzyżowania z dwóch chromosomów rodzicielskich wybierane są geny, które po zespoleniu tworzą jeden lub więcej chromosomów potomnych. W procesie mutacji dochodzi do przekłamania kodu w efekcie zmiany jednego genu lub ich ciągu, natomiast inwersja odwraca fragment chromosomu. Za pomocą tych mechanizmów tworzone są kolejne pokolenia (populacje chromosomów), zawierające coraz to „doskonalsze” osobniki.

W klasycznym algorytmie genetycznym pierwszym etapem krzyżowania jest wybór par chromosomów z populacji rodzicielskiej. Dokonuje się tego w sposób losowy, zgodnie z prawdopodobieństwem krzyżowania  $p_c$ . Następnie dla każdej pary rodziców losuje się pozycję genu (*locus*) w chromosomie, określającą tzw. *punkt krzyżowania*. Jeżeli chromosom każdego z rodziców składa się z  $L$  genów, to punkt krzyżowania  $l_k$  jest liczbą naturalną mniejszą od  $L$ . Zatem wybór punktu krzyżowania sprowadza się do wylosowania liczby z przedziału  $[1, L-1]$ . W wyniku krzyżowania pary chromosomów rodzicielskich otrzymuje się następującą parę potomków:

- 1) potomek, którego chromosom składa się z genów na pozycjach od 1 do  $l_k$ , pochodzących od pierwszego rodzica, i następnych genów, od pozycji  $l_k+1$  do  $L$ , pochodzących od drugiego rodzica,
- 2) potomek, którego chromosom składa się z genów na pozycjach od 1 do  $l_k$ , pochodzących od drugiego rodzica, i następnych genów, od pozycji  $l_k+1$  do  $L$ , pochodzących od pierwszego rodzica.

Jednym z najbardziej popularnych operatorów krzyżowania jest operator *Partially Mapped Crossover* – PMX zaproponowany przez D. Goldberga i R. Lingle [75]. To krzyżowanie może być postrzegane jako odmiana *two-cut-point* obejmująca procedurę naprawiania. W przypadku zastosowania operatora PMX

najpierw należy wybrać dwa punkty przecięcia chromosomu. Te dwa punkty określają podłańcuch, który następnie zostanie wymieniony. Załóżmy, że jest to podłańcuch obejmujący pozycje od 3 do 5 (rys. 7.4a).



Rys. 7.4. Ilustracja operatora krzyżowania PMX [opracowanie własne]

W drugim kroku należy wymienić między chromosomami zaznaczone podłańcuchy. W wyniku takiego przekształcenia otrzymuje się dwa rozwiązania niedopuszczalne – rysunek 7.4b. W trzecim kroku określa się powiązania między elementami, jakie występują w wymienionych podłańcuchach. W analizowanym przykładzie występują powiązania 3-5, 4-1, 5-4. W czwartym kroku rozpatruje się tylko te elementy, które leżą poza wymienionymi podłańcuchami i postępując iteracyjnie, wymienia się w pierwszym chromosomie 1 na 4, 4 na 5, 5 na 3. Analogicznie postępuje się z drugim chromosomem i wymienia się: 3 na 5, 5 na 4, 4 na 1. W rezultacie otrzymuje się dwa chromosomy potomne pokazane na rysunku 7.4c, przedstawiające rozwiązania dopuszczalne.

W klasycznym algorytmie genetycznym operator mutacji, zgodnie z prawdopodobieństwem mutacji  $p_m$ , dokonuje zmiany wartości genu w chromosomie na przeciwną (tzn. z 0 na 1 lub z 1 na 0). Jak już wcześniej wspomniano, prawdopodobieństwo zaistnienia mutacji jest zwykle bardzo małe i oczywiście od niego zależy, czy dany gen w chromosomie podlega mutacji, czy też nie. Dokonanie mutacji, zgodnie z prawdopodobieństwem  $p_m$ , polega np. na losowaniu liczby z przedziału  $[0,1]$  dla każdego genu i wybraniu do mutacji tych genów, dla których wylosowana liczba jest mniejsza lub równa prawdopodobieństwu  $p_m$ .

Operator mutacji może być też losowym wyborem dwóch pozycji i zamianą genów z tych pozycji miejscami. Tego rodzaju mutację ilustruje następujący przykład. Przypuśćmy, że chromosom, tj. harmonogram jest następujący:

1	3	7	2	4	5	6	8	9	10	11	12
---	---	---	---	---	---	---	---	---	----	----	----

Niech losowo wybranymi pozycjami będą 3 i 6, co zaznaczono na chromosomie poniżej.

1	3	7	2	4	5	6	8	9	10	11	12	
pozycje		3	6									

Wtedy wartości pozycji 3 i 6, którymi są 7 i 5 zamienia się miejscami. Nowy harmonogram zaprezentowano poniżej.

1	3	5	2	4	7	6	8	9	10	11	12	
pozycje		3	6									

Obszerny przegląd stosowanych operatorów genetycznych można znaleźć w pracy [38] oraz polskich książkach [169, 177].

## 7.6. Funkcja przystosowania

Bardzo ważnym pojęciem w algorytmach genetycznych jest *funkcja przystosowania* lub inaczej *funkcja dopasowania* (*fitness function*), która stanowi miarę przystosowania (dopasowania) danego osobnika do populacji. Funkcja ta pozwala wybrać osobniki najlepiej przystosowane (czyli o największej wartości funkcji przystosowania), zgodnie z ewolucyjną zasadą przetrwania „najsilniejszych”. W zagadnieniach optymalizacji funkcją przystosowania jest zwykle optymalizowana funkcja, nazywana *funkcją celu*. Jeżeli osobnikiem jest harmonogram produkcji, a przyjętym kryterium jest termin zakończenia wszystkich zadań  $C_{max}$  (*makespan*), to osobniki o mniejszej wartości  $C_{max}$  mają większą wartość miary przystosowania.

Podsumowując należy podkreślić, że algorytmy genetyczne są wzorowane na ewolucji gatunków. Podstawowa różnica między opisanymi wcześniej metodami lokalnego przeszukiwania a algorytmami genetycznymi polega na tym, że w algorytmach genetycznych operuje się na populacjach rozwiązań, a nie na pojedynczych rozwiązaniach. Algorytmy te zalicza się zatem do metod lokalnego przeszukiwania w szerokim sensie [97, 172]. Nie przeszukuje się bowiem sąsiedztwa pojedynczego rozwiązania, ale sąsiedztwo całej populacji rozwiązań, którego nie można utożsamiać z sumą mnogościową sąsiedztw pojedynczych rozwiązań.

Algorytm genetyczny jest ewolucyjną metodą przeszukiwania, która może dostarczyć optymalnych lub bliskich optymalnemu rozwiązań dla optymalizacyjnych problemów kombinatorycznych. Metoda ta może mieć zastosowanie do liczbowych problemów, włączając inżynierię, biologię, informatykę i nauki społeczne. Najważniejszą cechą algorytmów genetycznych jest ich elastyczność w manipulowaniu różnymi funkcjami celu.

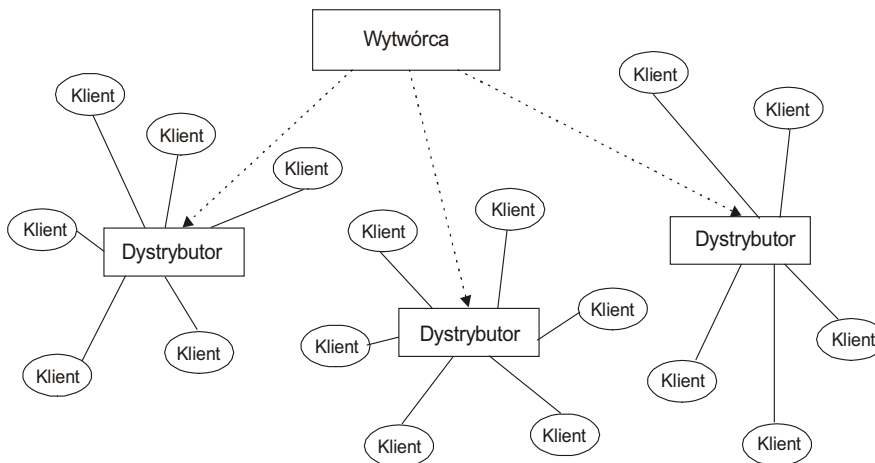


## 8. Zastosowanie algorytmów ewolucyjnych do zarządzania produkcją

### 8.1. Zastosowanie algorytmów genetycznych do konfigurowania sieci dostaw

#### 8.1.1. Zastosowanie algorytmu genetycznego do równoważenia alokacji klientów do centrów dystrybucji w sieci dostaw

W ostatnich latach zarządzanie łańcuchem dostaw stanowi centrum zainteresowań wielu menedżerów. Powodem tego są rosnące oczekiwania klientów oraz rozwój technologii informacyjnych. Doskonalenie łańcucha dostaw w coraz większym stopniu jest uzależnione od rozwoju technologii inteligentnych. W tym nurcie badań coraz większe znaczenie ma rozwój zastosowań algorytmów genetycznych do przestrzennego konfigurowania sieci dostaw [154]. Celem konfiguracji fizycznego przepływu materiałów w sieci jest na ogół minimalizacja kosztu lub maksymalizacja zysku. Projektowanie sieci dostaw, w której mają być utrzymywane optymalnie zrównoważone koszty transportu i obsługi klientów jest bardzo trudnym kombinatorycznym problemem (*NP – hard*). Na sukcesy firm mają wpływ powiązania poszczególnych członów łańcuchów dostaw, do których należą: wytwórcy, dystrybutorzy, dostawcy i klienci. Problem równoważenia alokacji jest kluczowy dla optymalnego skonfigurowania fizycznego przepływu produktów pomiędzy wielokrotnymi centrami dystrybucji i zbiorem klientów. Sieć łańcuchów dostaw (*supply chain network*) – SCN można przyrównać do gwiazdzistego drzewa częściowego (*star-spanning tree*), w którym korzenie (*root nodes*) są dystrybutorami, a liście (*leaf nodes*) klientami (rys. 8.1) (por. [206]). Problem alokacji klientów można więc odnieść do problemu równoważenia gwiazdzistego lasu częściowego (*star-spanning forest*) (por. [210]).



Rys. 8.1. Sieć łańcuchów dostaw z ich dystrybutorami [210]

W podejściu zaproponowanym przez G. Zhou, H. Mina i M. Gena [210] zastosowano algorytm genetyczny do optymalizacji łańcucha dostaw. Główny nacisk w tym podejściu położono na kodowaniu problemu w chromosom, ponieważ kluczowym problemem w zastosowaniu algorytmów genetycznych są techniki kodowania. Poniżej pokazano możliwości zastosowania kodowania Prüfera do algorytmu genetycznego. Podstawy matematyczne prezentowanego tu podejścia są jak następuje poniżej [210].

Dany jest zbiór korzeni  $U = \{u_1, u_2, \dots, u_r\}$  w nie skierowanym grafie  $G = (V, E)$ . Pewien  $U$  – zakorzeniony gwiazdzisty las częściowy  $F$  jest gwiazdzistym lasem częściowym  $G$  zawierającym  $r$  rozłącznych gwiazdzistych drzew  $T_1, T_2, \dots, T_r$ , takich, że  $u_i$  jest korzeniem  $T_i$  ( $i = 1, 2, \dots, r$ ). Jeżeli każda krawędź  $e \in E$  ma przypisaną wagę – liczbę całkowitą (np. koszt transportu lub odległość)  $w(e) > 0$ , to waga  $w(T)$  gwiazdzistego drzewa  $T$  może być definiowana jako suma wag jego składowych krawędzi. Dlatego, w najprostszy sposób oszacowanie gwiazdzistego lasu częściowego może być wyrażone jako:

$$w(F) \square \min \sum_{i \square 1}^r w(T_i) \quad (8.1)$$

$U$  – korzeni gwiazdzistego lasu częściowego  $F^*$  będzie znajdować w ten sposób nie zrównoważone wszystkie rozłączne gwiazdziste drzewa, np. segmenty rynku mające zarówno wielu klientów do obsłużenia, jak i segmenty rynku mające niewielu klientów do obsłużenia. Do utrzymywania wszystkich rozłącznych gwiazdzistych drzew jako możliwie wyrównanych adoptuje się operator min-max do formułowania problemu gwiazdzistego lasu częściowego. Wartość szukana gwiazdzistego lasu jest wyrażona jako:

$$w(F) \square \min_{1 \square i \square r} \max \{w(T_i)\} \quad (8.2)$$

Problemem jest znalezienie  $U$  – zakorzenionego gwiazdzistego lasu częściowego  $F^*$ , który minimalizuje  $w(F)$  wszystkich innych  $U$  – zakorzenionych gwiazdzistych lasów częściowych  $G$  zabezpieczając wyrównywanie przepływu produktów od wszystkich dystrybutorów.

Niech  $F^0 = (T_1^0, T_2^0)$  oznacza rezultat gwiazdzistego lasu częściowego w przypadku, w którym  $r = 2$ . Niech  $F^* = (T_1^*, T_2^*)$  będzie rozwiązaniem optymalnym problemu wyrównywania gwiazdzistego lasu częściowego i niech  $F^\dagger = (T_1^\dagger, T_2^\dagger)$  będzie przybliżonym rozwiązaniem otrzymanym przez zastosowanie algorytmu genetycznego – GA. Dla dowolnego gwiazdzistego lasu częściowego  $F = (T_1, T_2)$  definiuje się:

$$\underline{w}(F) = (w(T_1) + w(T_2))/2 \quad (8.3)$$

dla następującej relacji:

$$w(F^\dagger) \geq w(F^*) \geq \underline{w}(F^0) \quad (8.4)$$

Można zauważyć, że  $F^0$  jest wyprowadzone z  $F^\dagger$ , z wykorzystaniem algorytmu Prima dla początkowego gwiazdzonego lasu częściowego. Odpowiednio do równania (8.4) błąd względny  $(w(F^\dagger) - w(F^0))/w(F^0)$  heurystycznego rozwiązania lub rozwiązania z zastosowaniem GA ostatecznie wynosi:

$$w(F^\dagger) - \underline{w}(F^0)/w(F^0) \quad (8.5)$$

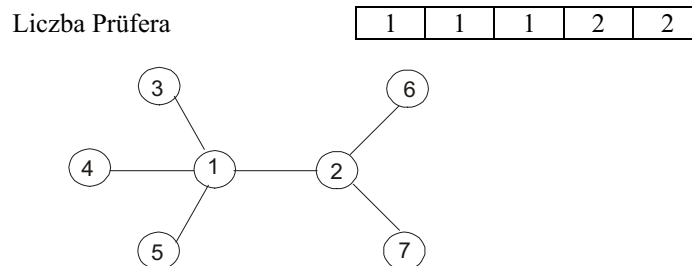
Dla każdego drzewa, które zawsze ma przynajmniej dwa liście, można zauważyć, że liście mają tylko jedną krawędź powiazaną z drzewem. Jeżeli wszystkie wierzchołki są definiowane przez etykiety, to powiazania lub zawartości tych wierzchołków będą określone. Zgodnie z podejściem Prüfera można zastosować permutację  $n-2$  liczb do unikalnego reprezentowania drzewa z  $n$  wierzchołkami, gdzie każda liczba jest całkowita i zawiera się w przedziale od 1 do  $n$ . Prüfer skonstruował następującą procedurę kodowania:

- Krok 1:* Niech węzeł  $j$  będzie najmniejszą etykietą liścia spośród etykiet drzewa  $T$ .
- Krok 2:* Zbiór  $k$  będzie pierwszą liczbą w permutacji, jeżeli wierzchołek  $k$  jest przyległy do wierzchołka  $j$ . W tym kroku buduje się permutację przez dodawanie na końcu liczb do prawej, tak że permutacja może być budowana i czytana z lewej do prawej.
- Krok 3:* Usuń wierzchołek  $j$  i krawędź z  $j$  do  $k$ , a wtedy następuje utworzenie drzewa z  $n-1$  wierzchołków.
- Krok 4:* Powtarzaj kroki 1-3 tak długo, aż jedna krawędź jest z lewej i produkcja liczby Prüfera lub permutacja z liczbą  $n-2$  jest uporządkowana.

Możliwe jest również generowanie unikalnego drzewa z liczbą Prüfera poprzez następującą procedurę dekodowania:

- Krok 1:* Niech  $A$  będzie oryginalną liczbą Prüfera i  $\bar{A}$  będzie zbiorem wszystkich wierzchołków nie należących do  $A$ .
- Krok 2:* Niech  $j$  będzie wierzchołkiem z najmniejszą etykietą w  $A$  i niech  $k$  będzie najbardziej na lewo położoną liczbą w  $A$ . Dodaj krawędź z  $j$  do  $k$  w drzewie. Usuń  $j$  z  $\bar{A}$  i  $k$  z  $A$ . Jeżeli  $k$  nie występuje w jakiegokolwiek pozostałej części  $A$ , wstaw go do  $\bar{A}$ . Powtarzaj proces tak długo, aż nie będzie liczb na lewo w  $A$ .
- Krok 3:* Jeżeli nie ma liczb pozostających w  $A$ , wtedy są dokładnie dwa wierzchołki  $r$  i  $s$  w  $\bar{A}$ . Dodaj krawędź z  $r$  do  $s$  w drzewie i utwórz drzewo z  $n-1$  krawędziami.

Na przykład niech zawartość liczby Prüfera [1 1 1 2 2] odpowiada siedmiu wierzchołkom drzewa częściowego (rys. 8.2) (por.[108, 206]).



Rys. 8.2. Drzewo i liczba Prüfera [210]

Konstrukcja liczby Prüfera rozpoczyna się lokowaniem liści drzewa mających najmniejszą etykietę. W tym przypadku jest to wierzchołek 3. Wierzchołek 1 (unikalny wierzchołek) jest przyległy do wierzchołka 3 w drzewie, przyjmuje się więc 1 jako pierwszą liczbę w permutacji, a następnie usuwa się wierzchołek 3 i krawędź (3,1). Aktualnie wierzchołek 4 ma najmniejszą etykietę liścia drzewa, a wierzchołek 1 jest przyległy do niego. Przyjmuje się więc 1 jako drugą liczbę w permutacji i wtedy usuwa się wierzchołek 4 i krawędź (4,1). Proces ten powtarza się tak długo w subdrzewie, aż krawędź (2,7) jest na lewo i liczba Prüfera tego drzewa z piątą liczbą jest produkcją finalną.

Odwrotnie można konstruować odpowiednie drzewo przez zastosowanie liczb Prüfera  $A = [1\ 1\ 1\ 2\ 2]$ . Ponieważ wierzchołki 3-7 są wyłączone z  $A$ , to są w zbiorze  $\bar{A} = \{3,4,5,6,7\}$ . Po pierwsze, wierzchołek 3 jest najmniejszą liczbą w  $\bar{A}$  i wierzchołek 1 jest liczbą położoną najbardziej na lewo w  $A$ . Następuje więc dodanie krawędzi (3,1) do drzewa, a następnie usunięcie wierzchołka 3 z  $\bar{A}$ . Liczba położona najbardziej na lewo w  $A$ , tj. 1, opuszcza zbiór i wtedy  $A = [1\ 1\ 2\ 2]$ . Po drugie, wierzchołek 4 jest nową najmniejszą liczbą w  $\bar{A}$  i wierzchołek 1 jest liczbą najbardziej na lewo w pozostałym  $A$ , dodaje się więc krawędź (4,1) do drzewa i usuwa wierzchołek 4 z  $\bar{A}$ ; liczba najbardziej na lewo z  $A$  opuszcza ten zbiór, wtedy  $A = [1\ 2\ 2]$ . Dalej kontynuuje się ten proces, dodając krawędź (5,1) do drzewa i usuwając ostatnią liczbę tj. 1 z  $A$  i wierzchołek 5 z  $\bar{A}$ . Ponieważ wierzchołek 1 nie pozostaje teraz w  $A$ , to następuje wstawienie go do  $\bar{A}$  i opuszczenie  $\bar{A} = \{1\ 6\ 7\}$ . Aktualnie wierzchołek 1 jest najmniejszą liczbą w  $\bar{A}$  i wierzchołek 2 jest liczbą najbardziej na lewo w  $A$ . Następuje więc dodanie krawędzi (1,2) do drzewa. Proces powtarza się tak długo, aż  $A$  jest puste i tylko wierzchołki 2 i 7 są w  $\bar{A}$ . Na końcu dodaje się krawędź (2,7) do drzewa. W ten sposób drzewo na rysunku 8.2 zostaje ukształtowane.

Można zauważyć, że liczba Prüfera „zręcznie” reprezentuje kodowanie gwiazdowego drzewa. Długość kodowania jest tylko  $n-2$ , rozmiar przestrzeni poszukiwań jest  $n^{(n-2)}$  i powiązanie pomiędzy liczbą Prüfera a drzewem częściowym jest odwzorowaniem jeden-do-jeden. Dlatego prawdopodobieństwo produkcji losowej (*randomly producing*) drzewa częściowego jest równe jedności.

Takie kodowanie umożliwia konstruowanie wszystkich gwiazdzistych drzew tylko wtedy, kiedy wszystkie liście drzewa nie ukazują się w kodowaniu. Ponadto konstruowanie wszystkich gwiazdzistych drzew lasu jest możliwe, jeżeli pominię się powiązania pomiędzy korzeniami lub wyłączy się wagi pomiędzy korzeniami. Dlatego adoptuje się liczbę Prüfera do reprezentacji każdego kandydata rozwiązania dla problemu wyrównywania alokacji.

Zastosowane w tym podejściu operatory genetyczne to: krzyżowanie równomierne nazywane też jednolitym lub jednostajnym (*uniform crossover*) i mutacja. Krzyżowanie równomierne odbywa się zgodnie z wylosowanym wzorcem wskazującym, które geny dziedziczone są od pierwszego z rodziców (pozostałe pochodzą od drugiego). Wzorec jest prostym łańcuchem binarnym z tą samą długością co chromosom, w którym 0 oznacza przejęcie genu na odpowiedniej pozycji (*locus*) od rodzica 1, a 1 od rodzica 2. W ten sposób powstaje potomek 1. Dla potomka 2 wzorec należy odczytać podobnie, przy czym 1 oznacza przejęcie genu na danej pozycji od rodzica 1, a 0 od rodzica 2. Rysunek 8.3 prezentuje operację z siedmioma liśćmi i trzema korzeniami.

Wzorec	1	0	1	1	0	1	0
Rodzic 1	1	2	1	3	2	3	1
Rodzic 2	1	1	2	2	3	3	3
	⇩						
Potomek 1	1	2	2	2	2	3	1
Potomek 2	1	1	1	3	3	3	3

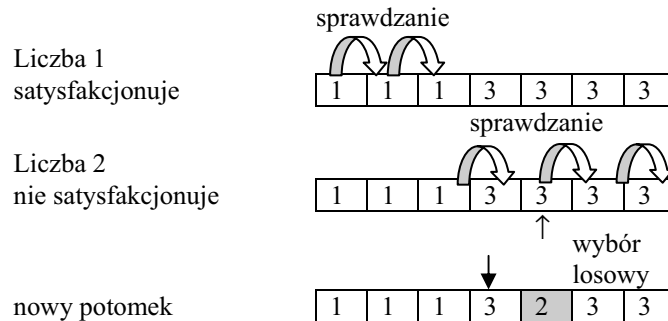
Rys. 8.3. Przykład krzyżowania równomiernego [210]

Zmiany mutacji (*single swap mutation*) dokonują losowego wyboru dwóch pozycji i ich liczby są wzajemnie zamienione (rys. 8.4).

Rodzic	1	1	2	2	3	3	3
Potomek	1	1	3	2	3	2	3

Rys. 8.4. Przykład zmian mutacji [210]

Rysunek 8.3 pokazuje wygenerowanego krzyżowaniem potomka, gdzie brakuje powiązania liści z korzeniem 2, ponieważ liczba 2 nie ukazuje się w permutacji. Istnieją różne sposoby pokonania takich trudności. Dla kombinatorycznego problemu, takiego jak wyrównywanie gwiazdzistego lasu częściowego, najefektywniejszym sposobem jest naprawianie niedopuszczalnego rozwiązania [209, 210]. Można sprawdzać każdy korzeń (liczbę w permutacji) i jeśli okaże się, że jakaś liczba nie ukazuje się w permutacji, to można zakłócać porządek i dokonać wstawienia brakującej liczby. Rysunek 8.5 pokazuje proces modyfikacji niedopuszczalnego rozwiązania z rysunku 8.3.



Rys. 8.5. Przykład modyfikacji [210]

Procedura oceny może być opisana następująco:

- konwersja osobnika lub liczby Prüfera w drzewo przez procedurę dekodowania i pominięcie wszystkich krawędzi pomiędzy korzeniami,
- obliczenie całkowitego kosztu lasu z zastosowaniem funkcji celu (8.2) i przyjęcie wielkości odwrotnej jako wartości dopasowania osobnika,
- powtarzanie procedury dla wszystkich osobników.

Ogólna procedura algorytmu genetycznego dla problemu wyrównywania obciążeń jest następująca:

Begin

```

 $t \leftarrow 0$ 
initialize  $P(t)$ ;
modify  $P(t)$ ;
evaluate  $P(t)$ ;
while (not terminate condition) do
begin
recombine  $P(t)$  to yield  $C(t)$ ;
modify  $C(t)$ ;
evaluate  $C(t)$ ;
select  $P(t+1)$  z  $P(t)$  and  $C(t)$ ;
 $t \leftarrow 0 + 1$ 
end

```

end

gdzie  $P(t)$  i  $C(t)$  są populacją rodziców i potomków w aktualnej generacji  $t$ . Warunek zatrzymania algorytmu jest kontrolowany przez wcześniej zdefiniowaną liczbę generacji.

G. Zhou, H. Min, M. Gen [210] przeprowadzili badania testowe proponowanego wyżej algorytmu genetycznego. W badaniach posłużono się wskaźnikiem BD obliczanym według następującej formuły:

$$[1 - (w_{\max}(T) - w_{\min}(T))/w_{\text{ave}}(T)] \times 100\% \quad (8.6)$$

gdzie:  $w_{\max}(T)$  jest największym kosztem przewozu pomiędzy dystrybutorami,  $w_{\min}(T)$  jest najmniejszym kosztem przewozu pomiędzy dystrybutorami i  $w_{\text{ave}}(T)$  jest średnim kosztem przewozu pomiędzy dystrybutorami.

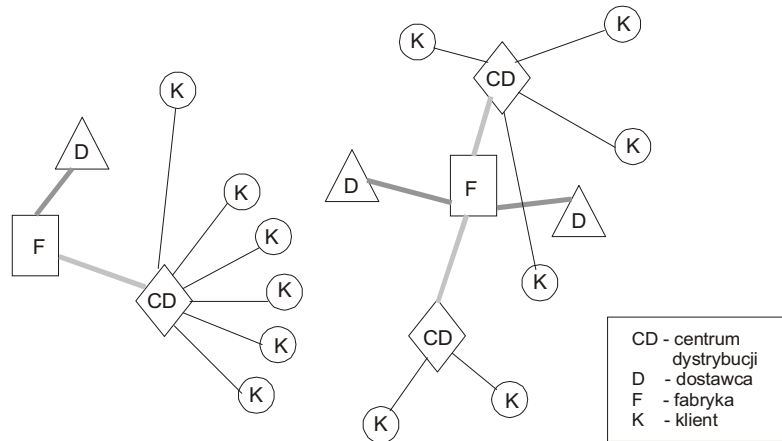
Wskaźnik BD dla różnych kombinacji dystrybutorów i klientów (maksimum 10 dystrybutorów i 100 klientów) bez równoważenia wahał się w granicach od 17,03% do 67,85%, natomiast dla tych samych kombinacji z wyrównaniem alokacji wynosił w granicach od 98,58% do 100%. Świadczy to o tym, że proponowany wyżej algorytm genetyczny można wykorzystywać do równoważenia alokacji klientów do wielokrotnych centrów dystrybucji w sieci łańcucha dostaw.

### 8.1.2. Analiza wielopoziomowego logistycznego łańcucha dostaw z zastosowaniem algorytmu genetycznego

Logistyka jest często definiowana jako sztuka dostarczania właściwych ilości, właściwego produktu we właściwym czasie i na ogół referuje model łańcucha dostaw, w którym pomiędzy dostawcami a klientami są umiejscowione różne ogniwa (obiekty) łańcucha tworzące wielopoziomowy model. Dlatego w większości przypadków łańcucha dostaw problemem jest konfigurowanie łańcucha, przy założeniu, że obiekty łańcucha mają ograniczone możliwości. W problemach alokacji zakłada się, że liczba potencjalnych obiektów z pewną wydajnością (takich jak centra obsługi, fabryki, centra dystrybucji – CD) jest znana. Problemem jest przyporządkowanie obiektów do miejsc w taki sposób, że suma kosztów stałych otwartych obiektów i kosztów zmiennych transportu jest minimalizowana.

W tym podejściu rozważany jest rozbudowany wielopoziomowy logistyczny problem dany maksymalną liczbą obiektów (fabryk i CD) do otwarcia oraz ograniczeniami. Projektowanie zadań tego problemu dotyczy projektowania sieci dystrybucji i wyboru obiektów (fabryk i CD) do otwarcia dla zaspokojenia żądań klientów przy jak najniższym koszcie.

W prezentowanym tu podejściu przyjmuje się, że logistyczny proces jest zorganizowany odpowiednio do porządku poziomów drzewa, jak pokazano na rysunku 8.6 [187]. Proponowana metoda rozwiązania problemu jest nowym sposobem, w którym bazuje się na algorytmie genetycznym. W metodzie tej adoptuje się liczbę Prüfera. Jest to opisany wyżej sposób reprezentacji problemu różnych sieci i tworzenia dopuszczalnych chromosomów. Manipulując dopuszczalnym chromosomem rozwija się w tym podejściu procedurę naprawiania. A. Syarif, Y.S. Yun, M. Gen [187] zaproponowali wszechstronny matematyczny model, który uwzględnia czynniki rzeczywistego świata tj. różne ograniczenia.



Rys. 8.6. Sieć dystrybucji [187]

W prezentowanym tu modelu przyjmuje się, że liczba klientów i ich potrzeby oraz liczba dostawców i ich możliwości są odpowiednio znane. Liczba potencjalnych fabryk i centrów dystrybucji, tak jak ich maksimum możliwości zaspokojenia potrzeb są znane. Przyjmuje się następującą nomenklaturę:

indeksy:

- $I$  – liczba dostawców ( $i = 1, 2, \dots, I$ ),
- $J$  – liczba fabryk ( $j = 1, 2, \dots, J$ ),
- $K$  – liczba centrów dystrybucji ( $k = 1, 2, \dots, K$ ),
- $L$  – liczba klientów ( $l = 1, 2, \dots, L$ ),

parametry:

- $a_i$  – możliwości dostawcy  $i$ ,
- $b_j$  – możliwości fabryki  $j$ ,
- $c_k$  – możliwości centrum dystrybucji  $k$ ,
- $d_l$  – zapotrzebowanie klienta  $l$ ,
- $s_{ij}$  – jednostkowy koszt produkcji w fabryce  $j$  ze zużyciem materiału od dostawcy  $i$ ,
- $t_{jk}$  – jednostkowy koszt transportu z fabryki  $j$  do centrum dystrybucji  $k$ ,
- $u_{kl}$  – jednostkowy koszt transportu z centrum dystrybucji  $k$  do klienta  $l$ ,
- $f_j$  – koszty stałe dla operacji fabryki  $j$ ,
- $g_k$  – koszty stałe dla operacji centrum dystrybucji  $k$ ,
- $W$  – górna granica sumarycznej liczby centrów dystrybucji, które mogą być otwarte,
- $P$  – górna granica sumarycznej liczby fabryk, które mogą być otwarte,

zmienne:

- $x_{ij}$  – wielkość produkcji fabryki  $j$  ze zużyciem materiału od dostawcy  $i$ ,
- $y_{jk}$  – wielkość ładunku z fabryki  $j$  do centrum dystrybucji  $k$ ,
- $z_{kl}$  – wielkość ładunku z centrum dystrybucji  $k$  do klienta  $l$ .



$w_j \begin{cases} \square 1, & \text{jeżeli produkcja ma miejsce w fabryce } j \\ \square 0, & \text{w przeciwnym przypadku} \end{cases}$

$z_k \begin{cases} \square 1, & \text{jeżeli centrum dystrybucji } k \text{ jest otwarte} \\ \square 0, & \text{w przeciwnym przypadku} \end{cases}$

Rozważany problem to wybór podzbioru fabryk i centrów dystrybucji do otwarcia z takim projektem sieci dystrybucji, który będzie satysfakcjonować wszystkie wymagane – przy minimalnym koszcie – zapotrzebowania narzucone przez klientów. Ten problem został sformułowany z wykorzystaniem modelu mieszanego liniowego programowania całkowitoliczbowego – MILP (*mixed integer linear programming*).

$$\min \sum_i \sum_j s_{ij} x_{ij} \square \sum_j \sum_k t_{jk} y_{jk} \square \sum_k \sum_l u_{kl} z_{kl} \square \sum_j f_j w_j \square \sum_k g_k z_k \quad (8.7)$$

$$\sum_j x_{ij} \square a_i \quad \text{dla każdego } i \quad (8.8)$$

$$\sum_k y_{jk} \square b_j w_j \quad \text{dla każdego } j \quad (8.9)$$

$$\sum_j w_j \square P \quad (8.10)$$

$$\sum_l z_{kl} \square c_k z_k \quad \text{dla każdego } k \quad (8.11)$$

$$\sum_k z_k \square W \quad (8.12)$$

$$\sum_k z_{kl} \square d_l \quad \text{dla każdego } l \quad (8.13)$$

$$w_j, z_k = \{0, 1\} \quad \text{dla każdego } j, k \quad (8.14)$$

$$x_{ij}, y_{jk}, z_{kl} \geq 0 \quad \text{dla każdego } i, j, k, l \quad (8.15)$$

Reprezentacja bazująca na drzewie jest dobrze znanym sposobem reprezentacji problemów sieciowych. W tych reprezentacjach można wyróżnić trzy sposoby kodowania:

- kodowanie krawędzi,
- kodowanie wierzchołków,
- kodowanie wierzchołków i krawędzi.

Do kodowania można stosować kodowanie wierzchołków z kodem Prüfera. Dla tego problemu można użyć subdrzewa  $I$ - $J$ , subdrzewa  $J$ - $K$  i subdrzewa  $K$ - $L$  do reprezentacji wzorca dystrybucji odpowiednio dla pierwszego stopnia, drugiego stopnia, trzeciego stopnia. Każdy chromosom w tym problemie składa się z pięciu części. Pierwsza część  $J$  jest liczbą binarną do reprezentacji otwartych/zamkniętych fabryk. Druga część  $K$  jest liczbą binarną do reprezentacji otwartych/zamkniętych CD. Ostatnie trzy części są trzema kodami Prüfera reprezentującymi wzorce dystrybucji odpowiednio dla trzech stopni.

W pierwszym kroku trzeba generować dwie zmienne 0-1 i sprawdzać liczbę otwartych fabryk lub otwartych CD dla górnej granicy ograniczeń. Jeżeli nastąpi przekroczenie danej górnej granicy, to zamyka się jedną z otwartych fabryk lub otwartych CD, które mają minimum możliwości. Ponadto trzeba sprawdzać sumaryczne możliwości otwartych fabryk i CD dla zapewnienia zapotrzebowania klienta. Jeżeli sumaryczne możliwości są mniejsze niż sumaryczne zapotrzebowanie, wtedy zamyka się obiekty z minimum możliwości i otwiera zamknięte obiekty z maksimum możliwości. Ten proces trwa tak długo, aż sumaryczne możliwości obiektów są wystarczające dla zapotrzebowania klientów. Kody Prüfera są generowane po tym kroku.

Dla subdrzewa  $I$ - $J$ , oznaczonego dostawcami  $1, 2, \dots, I$  jako elementami zbioru  $S = \{1, 2, \dots, I\}$  i definiowaniu fabryk  $1, 2, \dots, J$  jako elementy zbioru  $D = \{I + 1, I + 2, \dots, I + J\}$  graf dystrybucji ma  $I + J$  wierzchołków, które oznaczają, że potrzeba  $I + J - 2$  cyfr kodu Prüfera w zakresie  $[1, I + J]$  do unikalnego reprezentowania subdrzewa  $I$ - $J$ . Podczas generowania kodu Prüfera jest możliwe generowanie niedopuszczalnego kodu Prüfera, którego nie można adoptować do generowania subdrzewa transportowego. Dopuszczalność tego kodu można sprawdzać przez zastosowanie poniżej opisanych warunków.

Niech  $R_i$  oznacza liczbę ukazującą wierzchołek  $i$  w kodzie Prüfera  $A(T)$  i  $L_i$  oznacza, że liczba połączeń wierzchołków  $i \in S \cup D$ . Wyżej wymieniony kod Prüfera  $A(T)$  jest dopuszczalny jeżeli:

$$\sum_{i=1}^m L_i \leq \sum_{i=m+1}^{m+n} L_i \quad (8.16)$$

Kod Prüfera, który nie zadowala powyższych kryteriów dopuszczalności jest chromosomem niedopuszczalnym. Istnieją dwa możliwe sposoby postępowania z niedopuszczalnym chromosomem: pierwszy, to odrzucenie tego chromosomu i generowanie nowego, aż spełnione zostanie powyższe kryterium. Taka technika wymaga długich czasów obliczeń, szczególnie wtedy, kiedy liczby dostawców i liczby punktów przeznaczenia (odbiorców) są bardzo duże. Inny sposób to rozwijanie procedur naprawiania dla niedopuszczalnych chromosomów. W takim przypadku tworzone jest sprawdzanie dopuszczalności i procedura naprawiania dla kodu Prüfera jest stosowana do dekodowania w drzewo częściowe, jak niżej.

Procedura: sprawdzanie dopuszczalności i procedura naprawiania dla kodu Prüfera.

Powtarzaj poniższe kroki, aż  $\sum_{i \in S} L_i \leq \sum_{i \in D} L_i$  (warunek dopuszczalności).

*Krok 1:* Określ  $R_i$  dla  $i \in S \cup D$  z  $A(T)$ .

*Krok 2:*  $L_i = R_i + 1$ .

*Krok 3:* Jeżeli  $\sum_{i \in S} L_i > \sum_{i \in D} L_i$ , wtedy wybierz jedną liczbę z  $A(T)$ , która zawiera wierzchołek  $i$  ( $i \in S$ ) i powtórz to z liczbą  $j$  ( $j \in D$ ). W przeciwnym przypadku wybierz jedną liczbę z  $A(T)$ , która zawiera wierzchołek  $i$  ( $i \in S$ ) i powtórz to z liczbą  $j$  ( $j \in S$ ), następnie przejdź do kroku 1.

Po sprawdzeniu dopuszczalności kodu Prüfera może nastąpić dekodowanie w drzewo częściowe w celu określenia wzorca dystrybucji na każdym poziomie. Procedury kodowania problemu sieci w kod Prüfera i dekodowanie kodu Prüfera w sieć-graf są jak poniżej:

Procedura kodowania  $I$ - $J$ .

*Krok 1:* Niech wierzchołek  $i$  będzie najmniejszą etykietą liścia spośród etykiet subdrzewa  $I$ - $J$ .

*Krok 2:* Niech  $j$  będzie pierwszą liczbą w kodzie, jako że wierzchołek  $j$  przylega do wierzchołka  $i$  i jest unikalnie określony. W tej procedurze tworzy się kod przez dostawianie liczby do prawej. Kod jest czytany z lewej do prawej.

*Krok 3:* Usunięcie wierzchołka  $i$  i powiązania z  $i$  do  $j$ , zatem subdrzewo ma  $I + J - 1$  wierzchołków.

*Krok 4:* Powyższe kroki powtarza się, aż jest jedno powiązanie na lewo. Kod Prüfera produkuje się kodując liczby  $I + J - 2$  zawarte pomiędzy 1 i  $I + J$ .

Procedura: dekodowanie  $I$ - $J$ .

*Krok 1:* Niech  $A$  jest oryginalnym kodem Prüfera i niech  $\bar{A}$  będzie zbiorem wszystkich wierzchołków niezawartych w  $A$ , które oznaczają możliwe do przyjęcia wierzchołki dla rozważań w tworzonym drzewie.

*Krok 2:* Powtarzaj proces (2.1) do (2.5) tak długo, aż nie ma liczb na lewo w  $A$ .

(2.1) Niech  $i$  będzie najmniejszą etykietą wierzchołka  $\bar{A}$ . Niech  $j$  będzie najbardziej na lewo liczbą w  $A$ .

(2.2) Jeżeli  $i$  i  $j$  nie są w tym samym zbiorze  $S$  lub  $D$ , dodaj krawędź z  $i$  do  $j$  w drzewie. W przeciwnym przypadku wybierz następną liczbę  $k$  z  $A$ , która nie jest zawarta w tym samym zbiorze z  $i$ . Zamień  $j$  z  $k$ , dodaj krawędź  $(i, k)$  do drzewa.

(2.3) Usuń  $j$  (lub  $k$ ) z  $A$  i  $i$  z  $\bar{A}$ . Jeżeli  $j$  (lub  $k$ ) nie ukazuje się gdziekolwiek w pozostałej części  $A$ , wstaw to w  $\bar{A}$ .

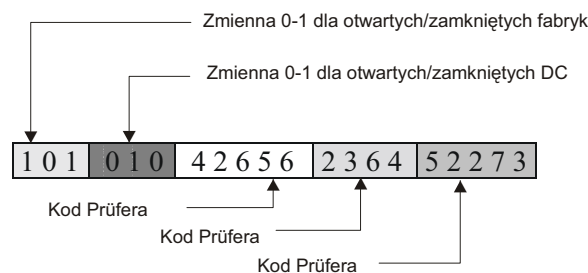
(2.4) Wyznacz dostępne ilości jednostki do  $x_{ij} = \min\{a_i, b_j\}$  lub  $x_{ij} = \min\{a_i, b_k\}$ .

(2.5) Aktualizuj dostępne  $a_i = a_i - x_{ij}$  i  $b_j = b_j - x_{ij}$ .

- Krok 3:* Jeżeli nie ma liczb pozostałych w  $A$ , wtedy są dokładnie dwa wierzchołki:  $r$  i  $s$ . Dodaj powiązanie z  $r$  do  $s$  w drzewie i ukształtuj drzewo z powiązaniem  $I + J - 1$ .
- Krok 4:* Jeżeli nie ma jednostek do przydzielenia, to koniec. W przeciwnym przypadku pozostaje dostawca wierzchołka  $r$  i zapotrzebowanie wierzchołka  $s$ . Dodaj krawędź  $(r, s)$  do drzewa i przydziel dostępne ilości jednostki  $x_{rs} = \min\{a_r, b_s\}$  do krawędzi. Jeżeli wystąpi cykl, wtedy usuń krawędź, która oznacza zero przepływu. Nowe drzewo częściowe jest ukształtowane z  $n + n - 1$  krawędziami.

W podobny sposób można generować dwa inne kody Prüfera reprezentujące  $J-K$  i  $K-L$  sub-drzew. Te dwa kody Prüfera zawierają odpowiednio  $J + K - 2$  i  $K + L - 2$  liczb. Procedury kodowania ( $J-K$  i  $K-L$ ) i dekodowania ( $J-K$  i  $K-L$ ) dla drugiego i trzeciego poziomu są również podobne.

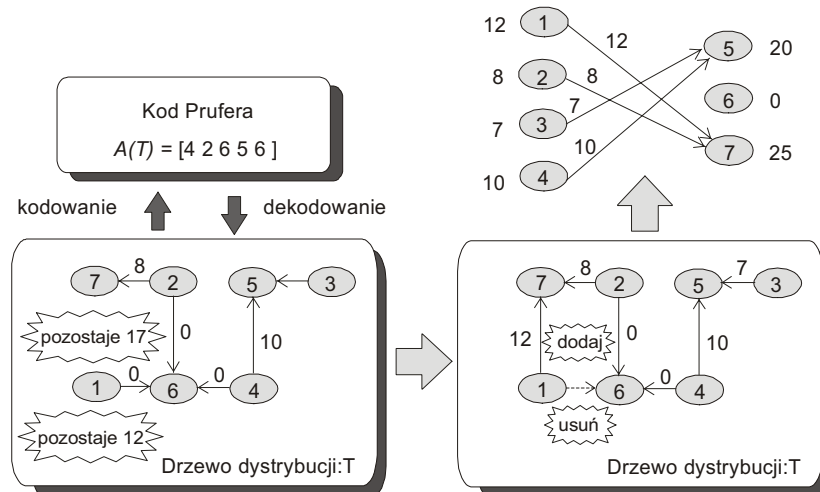
Rozważmy problem dla 4 dostawców, 3 dopuszczalnych fabryk, 3 dopuszczalnych CD i 4 klientów. Rozważamy 2 fabryki i 2 CD jako górną granicę otwartych fabryk i otwartych CD dla tego problemu. Na rysunku 8.7 zilustrowano reprezentację dopuszczalnego chromosomu.



Rys. 8.7. Ilustracja dopuszczalnego chromosomu [187]

Jak zdefiniowano wcześniej chromosom zawiera pięć subłańcuchów. Pierwszy i drugi subłańcuch są trzema binarnymi liczbami reprezentującymi odpowiednio otwarte/zamknięte fabryki i CD. Ostatnie trzy subłańcuchy są kodami Prüfera zawierającymi 5, 4 i 5 liczb do reprezentacji wzorca dystrybucji odpowiednio dla każdego poziomu. Powyższe reprezentacje pokazują, że dwie fabryki i jedno CD są otwarte. Kiedy dekodujemy chromosom, po pierwsze musimy zmienić możliwości zamkniętych urządzeń na zero. Następnie po wygenerowaniu kodów Prüfera i sprawdzeniu ich dopuszczalności możemy dekodować przez użycie procedur dekodowania dla znalezienia chromosomów wzorców dystrybucji i obliczyć wartość dopasowania. Poniżej podane jest krok po kroku dekodowanie pierwszego kodu Prüfera w powyższej reprezentacji. Po pierwsze, dla kodu Prüfera  $A = [4 2 6 5 6]$  określa się  $\bar{A} = [1 3 7]$ . Wierzchołek 1 jest najmniejszą liczbą-wierzchołkiem w  $\bar{A}$ , a wierzchołek 4 jest najbardziej na lewo położoną liczbą w  $A$ . Te dwa wierzchołki są w tym samym zbiorze. Tak więc wybiera się następnego wierzchołek w  $A$ , który nie jest w tym samym zbiorze, tj. wierzchołek 6.

Następnie dodaje się krawędź (1,6) do drzewa, a potem usuwa wierzchołek 1 z  $\bar{A}$ . Wierzchołek 6 z  $A$  pozostaje, zatem  $A = [4\ 2\ 5\ 6]$  i  $\bar{A} = [3\ 7]$ . Po pierwsze oznacza się  $x_{16} = \min \{a_1, b_6\} = 0$ , a następnie aktualizuje  $a_1 = a_1 - x_{16}$  i  $b_6 = b_6 - x_{16}$ . Po drugie, wierzchołek 3 jest najmniejszą liczbą w  $\bar{A}$  i wierzchołek 5 jest najbardziej na lewo położoną liczbą w  $A$ , która nie jest w tym samym zbiorze. Do drzewa dodaje się krawędź (3,5), a następnie usuwa wierzchołek 3 z  $\bar{A}$ . Odtąd wierzchołek 5 jest najmniejszą liczbą ukazaną w pozostałej części  $A$ , a więc dodaje się go do  $\bar{A}$ . Aktualnie  $A = [4\ 2\ 6]$  i  $\bar{A} = [5\ 7]$ . Oznacza się  $x_{35} = \min \{a_3, b_5\} = 7$ . Aktualizuje się  $a_3 = a_3 - x_{35} = 0$  i  $b_5 = b_5 - x_{35} = 13$ . Proces powtarza się tak długo, aż  $A$  jest puste i tylko wierzchołek 2 i 7 są w  $\bar{A}$ . Dalej następuje dodanie krawędzi (2,7) do drzewa. Odtąd jest wciąż dostępny dostawca w wierzchołku 7 i zapotrzebowanie w wierzchołku 1. Dodajemy krawędź (1,7) i usuwamy wierzchołek (1,6). Rysunek 8.8 ilustruje kod Prüfera i jego drzewo częściowe. Procedury dekodowania drugiego i trzeciego poziomu są podobne.



Rys. 8.8. Ilustracja procedury dekodowania [187]

## 8.2. Zastosowanie algorytmów ewolucyjnych do planowania i harmonogramowania produkcji

Zastosowanie algorytmów ewolucyjnych do zarządzania to przede wszystkim wspomaganie decyzji planistycznych. Od wielu lat trwają badania mające na celu stworzenie systemów zarządzania produkcją z rozbudowaną funkcją planowania symulacyjnego. Cechą wspólną tych podejść jest uwzględnienie – w czasie planowania produkcji – wielu ograniczeń. Algorytm ewolucyjny można zaimplementować do optymalizacji na wielu poziomach zarządzania produkcją, m.in. na poziomie strategicznym, taktycznym i operacyjnym. Na poziomie strategicznym można zastosować algorytm ewolucyjny do konfigurowania łańcucha dostaw. Planowanie taktyczne z algorytmem ewolucyjnym to przede

wszystkim tworzenie tzw. harmonogramu głównego MPS (*Master Production Scheduling*) w systemach MRPII/ERP. Na najniższym poziomie, operacyjnym, algorytmy ewolucyjne są powszechnie wykorzystywane do szeregowania zadań w systemach produkcyjnych [36, 43, 50, 52, 86, 91, 93, 113, 123, 124, 126, 132, 138, 139, 145, 192, 196, 208, 209].

W ostatnich latach w ramach badań dotyczących zastosowania algorytmów ewolucyjnych do zarządzania produkcją rozwijano dwa kierunki: pierwszy, dotyczący reprezentacji problemu oraz drugi, mający na celu zwiększanie wydajności przeszukiwania genetycznego przez włączenie tradycyjnych metod heurystycznych.

Bardzo ważnym pojęciem w algorytmach genetycznych jest *funkcja przystosowania* (*fitness function*), która stanowi miarę przystosowania (dopasowania) danego osobnika do populacji. Funkcja ta pozwala wybrać osobniki najlepiej przystosowane (czyli o największej wartości funkcji przystosowania), zgodnie z ewolucyjną zasadą przetrwania „najsilniejszych”. Funkcja przystosowania musi być odpowiednio zdefiniowana. Jeżeli osobnikiem jest harmonogram produkcji, a przyjętym kryterium jest termin zakończenia wszystkich zadań w systemie  $C_{max}$  (*makespan*), to osobniki o mniejszej wartości  $C_{max}$  mają większą wartość miary przystosowania. Ogólny problem harmonogramowania polega na tym, aby przy uwzględnieniu ograniczeń minimalizować okres wykonania (*makespan*), tj. wymagany czas od rozpoczęcia do całkowitego zakończenia wszystkich zadań. W przypadku systemów przepływowych problem ten może być sformułowany następująco [37]:

$$\min \max_{1 \leq k \leq m} \{ \max_{1 \leq i \leq n} \{ c_{ik} \} \} \quad (8.17)$$

$$c_{ik} - t_{ik} + M(1 - a_{ihk}) \geq c_{ih}, \quad i = 1, 2, \dots, n \quad i \quad h, k = 1, 2, \dots, m \quad (8.18)$$

$$c_{jk} - c_{ik} + M(1 - x_{ijk}) \geq t_{jk} \quad i, j = 1, 2, \dots, n \quad i \quad k = 1, 2, \dots, m \quad (8.19)$$

$$c_{ik} \geq 0, \quad i = 1, 2, \dots, n \quad i \quad k = 1, 2, \dots, m \quad (8.20)$$

$$x_{ijk} = 0 \text{ lub } 1, \quad i, j = 1, 2, \dots, n \quad i \quad k = 1, 2, \dots, m \quad (8.21)$$

gdzie:

$c_{jk}$  – termin zakończenia zadania  $j$  na maszynie  $k$ ,

$t_{jk}$  – czas wykonania zadania  $j$  na maszynie  $k$ ,

$M$  – całkowita liczba dodatnia,

$a_{ihk}$  – wskaźnik zmiennej definiowanej jako:

$a_{ihk} = \begin{cases} 1, & \text{jeżeli wykonanie na maszynie } h \text{ poprzedza wykonanie} \\ & \text{na maszynie } k \text{ dla zadania } i \\ 0, & \text{w przeciwnym przypadku} \end{cases}$

$x_{ijk}$  – wskaźnik zmiennej definiowanej jako:

$x_{ijk} = \begin{cases} 1, & \text{jeżeli zadanie } i \text{ poprzedza zadanie } j \text{ na maszynie } k \\ 0, & \text{w przeciwnym przypadku} \end{cases}$

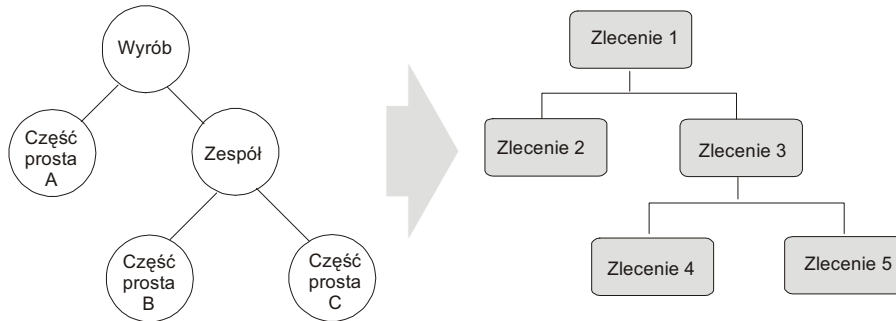
W ostatnim okresie, w problemach harmonogramowania badano m.in. następujące możliwości reprezentacji [37]:

- reprezentacja bazująca na operacjach (*operation*),
- reprezentacja bazująca na zadaniach (*job*),
- reprezentacja bazująca na listach preferencyjnych (*preference list*),
- reprezentacja bazująca na parach zadań (*job pair*),
- reprezentacja bazująca na regułach priorytetu (*prioryty rule*),
- reprezentacja bazująca na grafach dysjunkcyjnych (*disjunctive graph*),
- reprezentacja bazująca na terminach zakończenia (*completion time*),
- reprezentacja bazująca na maszynach (*machine*),
- reprezentacja losowa (*keys random*),
- inne.

Reprezentacje mogą być klasyfikowane jako: bezpośrednie i pośrednie. Reprezentacja bezpośrednie to takie, jak: reprezentacja bazująca na operacjach, reprezentacja bazująca na zadaniach, reprezentacja bazująca na parach zadań, reprezentacja bazująca na terminie zakończenia i reprezentacja losowa. W reprezentacjach pośrednich, takich jak np. bazujących na regułach priorytetu kodowanie w chromosom i algorytm genetyczny są wykorzystywane do określenia najlepszej kolejności reguł dyspozytorskich. Harmonogram jest konstruowany w oparciu o kolejność tych reguł. Do reprezentacji pośrednich należą również: reprezentacja bazująca na listach preferencyjnych, reprezentacja bazująca na grafie dysjunkcyjnym i reprezentacja bazująca na maszynach.

Planowanie taktyczne w przedsiębiorstwie produkcyjnym można realizować z zastosowaniem algorytmu ewolucyjnego, w którym stosuje się przede wszystkim reprezentacje bazujące na zadaniach. Na poziomie taktycznym harmonogram główny jest tworzony w oparciu o plan sprzedaży z uwzględnieniem rozwinięcia specyfikacji materiałowej. W tym podejściu harmonogram jest konstruowany odpowiednio do kolejności zleceń na liście  $n$ -zleceń. Chromosom jest więc łańcuchem znaków odpowiadających kolejnym zleceniom. Proces tworzenia harmonogramu polega na tym, że wszystkie operacje kolejnych zleceń (zadań) są umieszczane na harmonogramie w najwcześniejszych możliwych terminach ich rozpoczęcia.

Ze względu na strukturę zleczanych do produkcji wyrobów istnieją ograniczenia kolejnościowe między zleceniami. Zgodnie z logiką systemów MRP wyrób można przedstawić w postaci grafu. Ograniczenia kolejnościowe wynikają ze struktury wyrobu. Wyrób wykonywany jest z tzw. *pozycji materiałowych*: zespołów różnych szczebli i części prostych. Wyrób może być wyprodukowany po wytworzeniu zespołów wchodzących w jego skład, a zespoły można wykonać wtedy, kiedy zakończy się produkcja jego składników, tj. zespołów niższego stopnia złożoności lub części prostych. Zgodnie z logiką systemów MRP zlecenia tworzą strukturę o postaci grafu-drzewa i odpowiadają każdej pozycji materiałowej na poziomie struktury BOM (*Bill of Materials*). Odzworowanie to pokazano na rysunku 8.9.



Rys. 8.9. Odzworowanie struktury BOM w zlecenia [opracowanie własne]

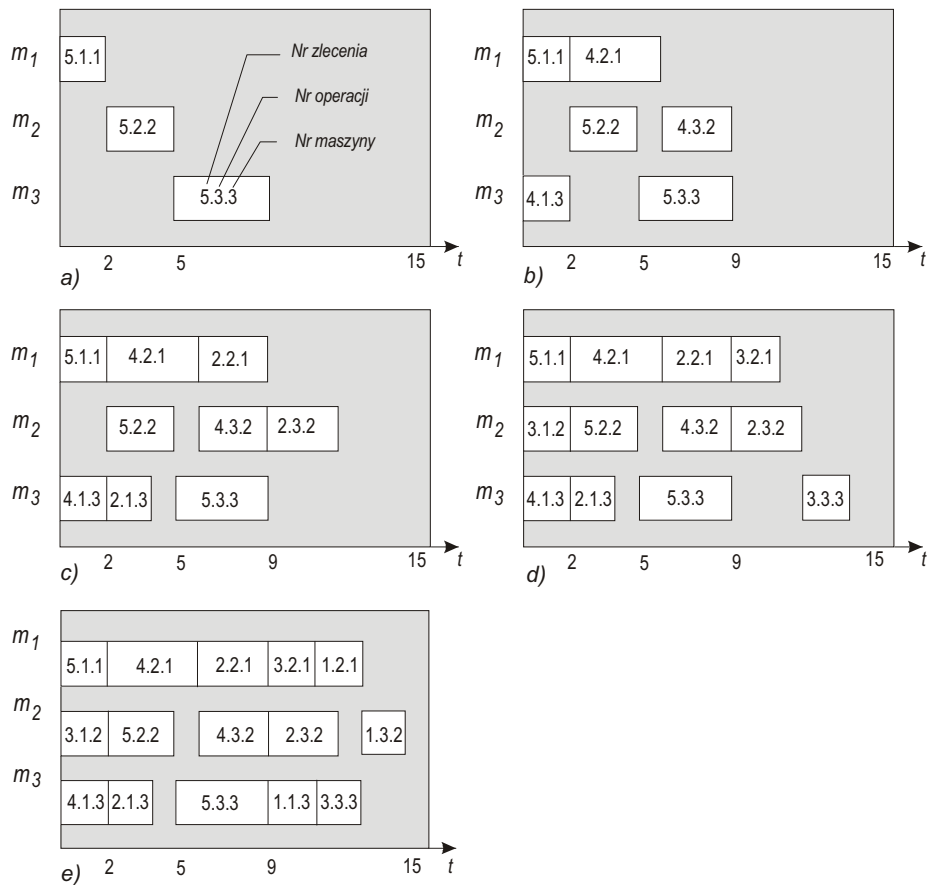
Każda permutacja zleceń odpowiada dopuszczalnemu harmonogramowi. Operacje pierwszego zlecenia na liście są umieszczane na harmonogramie jako pierwsze. Następnie są lokowane na harmonogramie operacje drugiego zlecenia itd. Pierwsza operacja zlecenia jest umieszczana na harmonogramie w możliwie najlepszym terminie na wymaganej dla niej maszynie. W dalszej kolejności, w podobny sposób umieszczana jest druga operacja. Postępowanie to jest kontynuowane tak długo, aż umieszczone zostaną wszystkie operacje. Tworzenie harmonogramu dla zleceń można przedstawić na przykładzie zleceń zaprezentowanych w tabeli 8.1.

Tabela 8.1. Przykład problemu dla 5 zleceń

Zlecenie	1			2			3			4			5		
Operacja	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
Czas wykonania	2	2	2	2	3	2	2	2	2	2	4	3	2	3	4
Maszyna	3	1	2	3	1	2	2	1	3	3	1	2	1	2	3

I tak, na przykład dla chromosomu [5 4 2 3 1] pierwszym wykonywanym zleceniem będzie zlecenie 5. Operacje dla zlecenia 5 wykonywane są na maszynach  $m_1, m_2, m_3$  w ustalonej kolejności i odpowiednich czasach wykonania (2 3 4). One jako pierwsze są umieszczane na harmonogramie (rys. 8.10a). Następnie na harmonogramie umieszczane są operacje zadania 4, dla kolejności maszyn  $m_3, m_1, m_2$  i odpowiednio czasów (2 4 3). Każda operacja jest umieszczana na harmonogramie w możliwie najlepszym miejscu (rys. 8.10b). Rysunek 8.10c ilustruje lokowanie na harmonogramie operacji dla zlecenia 2, a rysunek 8.10d – dla zlecenia 3. Na końcu lokowane są na harmonogramie operacje dla zlecenia 1 (rys. 8.10e).





Rys. 8.10. Dekodowanie harmonogramu z chromosomu [opracowanie własne]

Badania w zakresie zastosowania algorytmów ewolucyjnych zdominowane są ich implementacją do harmonogramowania wewnątrzkomórkowego. Poniżej przedstawiono różne możliwości reprezentacji w planowaniu operacyjnym.

### 8.2.1. Reprezentacja bazująca na operacjach

W tej reprezentacji koduje się harmonogram jako kolejność operacji i każdy gen staje się operacją. W ten sposób nazwa każdej operacji jest liczbą naturalną. Dla problemu przepływowego  $n$ -zadań i  $m$ -maszyn chromosom zawiera  $n \times m$  genów. Harmonogram jest dekodowany z chromosomu według następującej procedury: po pierwsze, translacja chromosomu na listę uporządkowanych operacji, po drugie, generowanie harmonogramu przez jednorzbiegową heurystykę bazującą na liście. Pierwsza operacja na liście jest jako pierwsza umieszczana na harmonogramie, następnie druga operacja itd. W wyniku takiego postępowania każda operacja jest lokowana w najlepszym, możliwym czasie

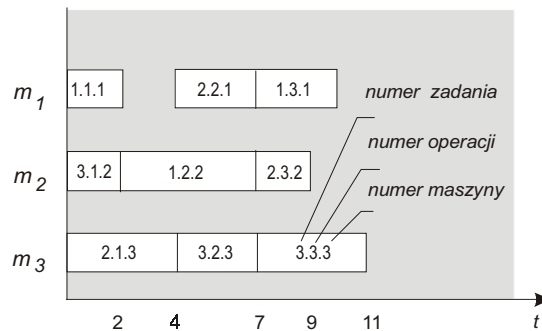
wykonania na odpowiedniej dla niej maszynie. Ten proces jest powtarzany tak długo, aż wszystkie operacje zostaną umieszczone na harmonogramie.

W tabeli 8.2 przedstawiono problem harmonogramowania dla 3-zadań i 3-maszynach.

Tabela 8.2. Przykład problemu dla 3-zadań i 3-maszyn

Zadanie	1			2			3		
Operacja	1	2	3	1	2	3	1	2	3
Czas wykonania	2	5	3	4	3	2	2	3	4
Maszyna	1	2	1	3	1	2	2	3	3

Przyjmijmy, że chromosom ma postać [1 2 2 2 1 3 1 3 3]. Każde zadanie ma trzy operacje. Każdy gen unikalnie wskazuje operację oraz określa ją odpowiednio do kolejności występującej w sekwencji. Niech  $o_{jim}$  oznacza  $i$ -tą operację zadania  $j$  na maszynie  $m$ . Chromosom może być przetworzony w unikalną listę uporządkowanych operacji [ $o_{111}$ ,  $o_{213}$ ,  $o_{221}$ ,  $o_{232}$ ,  $o_{122}$ ,  $o_{312}$ ,  $o_{131}$ ,  $o_{323}$ ,  $o_{333}$ ]. Operacja  $o_{111}$  ma najwyższy priorytet i jest umieszczana na harmonogramie jako pierwsza, następnie  $o_{213}$ , itd. Rezultaty harmonogramowania pokazano na rysunku 8.11.



Rys. 8.11. Dekodowanie harmonogramu [opracowanie własne]

### 8.2.2. Reprezentacja bazująca na zadaniach

Na poziomie planowania operacyjnego, podobnie jak w przypadku planowania taktycznego, można zastosować algorytmy z reprezentacją bazującą na zadaniach. W takich reprezentacjach zadanie to zlecenie wykonania określonej części prostej.

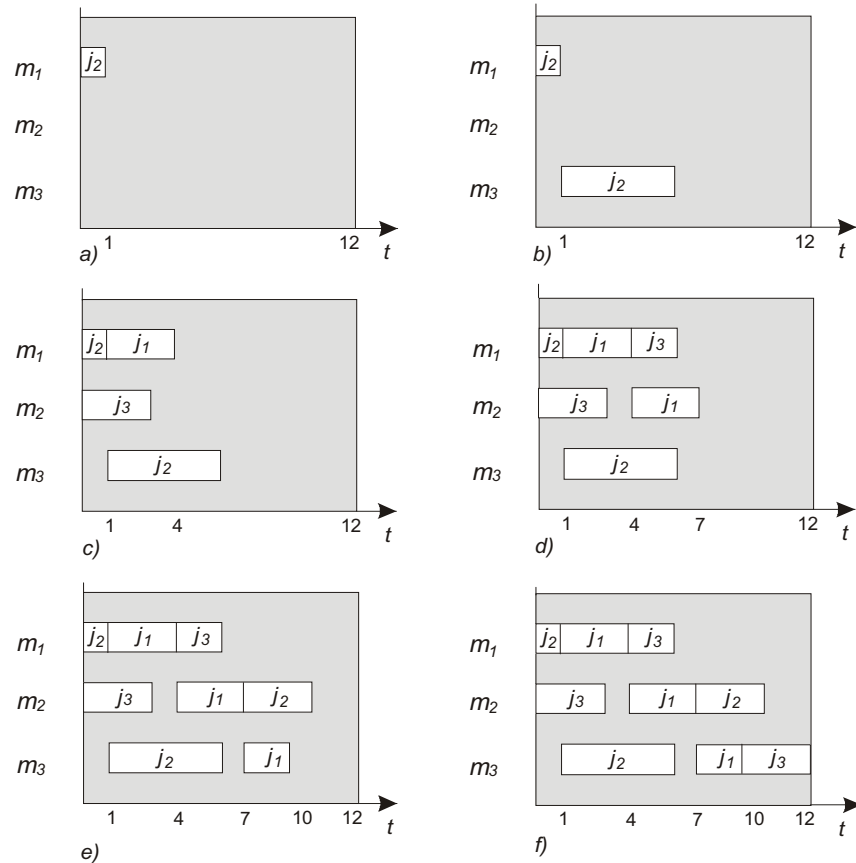
### 8.2.3. Reprezentacja bazująca na listach preferencyjnych

Ta reprezentacja została zastosowana do klasycznego problemu JSP (*job shop scheduling problem*) przez F. Croce, R. Tadei i G. Volta [49]. Dla problemu harmonogramowania  $n$ -zadań na  $m$ -maszynach chromosom jest tworzony z  $m$ -subchromosomów (każdy dla jednej maszyny). Subchromosom jest łańcuchem symboli o długości  $n$ , gdzie każdy symbol identyfikuje operację, która może być wykonana na przypisanej jej maszynie. Subchromosom nie opisuje kolejności operacji na maszynie. Każda maszyna ma własną listę preferencyjną. Aktualny harmonogram jest dekodowany z chromosomu. Analizuje się stany kolejek oczekujących przed maszyną i w rezultacie wybierane są pierwsze operacje z listy preferencyjnej. Dekodowanie z chromosomu w tej reprezentacji można przeanalizować dla systemu opisanego w tabeli 8.3.

Tabela 8.3. Przykład problemu dla 3-zadań i 3-maszyn

Zadanie	1			2			3		
Operacja	1	2	3	1	2	3	1	2	3
Czas wykonania	3	3	2	1	5	3	3	2	3
Maszyna	1	2	3	1	3	2	2	1	3

Przyjmijmy, że chromosom ma postać  $[(2\ 3\ 1)\ (1\ 3\ 2)\ (2\ 1\ 3)]$ . Pierwszy gen  $(2\ 3\ 1)$  jest listą preferencyjną dla maszyny  $m_1$ . Na podstawie tej listy preferencyjnej wnioskuje się, że pierwsza będzie operacja zadania  $j_2$  na maszynie  $m_1$ , następnie  $j_1$  na  $m_2$  i  $j_2$  na  $m_3$ . Odpowiednio do ograniczeń kolejności operacji, możliwe jest umieszczenie na harmonogramie tylko  $j_2$  na  $m_1$  (rys. 8.12a). Następną operacją możliwą do umieszczenia na harmonogramie jest  $j_2$  na  $m_3$ , jak pokazano na rysunku 8.12b. Teraz aktualną preferowaną operacją jest  $j_3$  na  $m_1$ ,  $j_1$  na  $m_2$  i  $m_3$ . W związku z tym, że nie jest możliwe umieszczenie ich na harmonogramie w aktualnym czasie, należy rozważyć drugie operacje na każdej liście. Na listach preferencyjnych są:  $j_1$  na  $m_1$  oraz  $j_3$  na  $m_2$  i  $m_3$ . Operacjami, które teraz można umieścić na harmonogramie są  $j_1$  na  $m_1$  i  $j_3$  na  $m_2$  (rys. 8.12c). Następnie jest możliwe umieszczenie na harmonogramie operacji  $j_3$  na  $m_1$  i  $j_1$  na  $m_2$  (rys. 8.12d), później  $j_2$  na  $m_2$  i  $j_1$  na  $m_3$  (rys. 8.12e). Ostatnie operacje, które zostaną umieszczone na harmonogramie, to  $j_3$  na  $m_3$  (rys. 8.12f). W ten sposób zostały umieszczone na harmonogramie wszystkie operacje i został określony czas wykonania wynoszący 12.



Rys. 8.12. Dekodowanie harmonogramu z chromosomu [37]

### 8.2.4. Reprezentacja bazująca na parach zadań

R. Nakano i T. Yamada [157] zastosowali macierz binarną do kodowania harmonogramu, którą tworzą relacje kolejnościowe par zadań dla odpowiednich maszyn. Taki problem reprezentacji można rozważyć na przykładzie 3 zadań i 3 maszyn opisanych tabelą 8.4.

Tabela 8.4. Przykład problemu dla reprezentacji bazującej na parach

Uporządkowanie operacji				Dopuszczalny harmonogram			
Zadanie	Porządek maszyn			Maszyna	Porządek zadań		
$j_1$	$m_1$	$m_2$	$m_3$	$m_1$	$j_2$	$j_1$	$j_3$
$j_2$	$m_1$	$m_3$	$m_2$	$m_2$	$j_3$	$j_1$	$j_2$
$j_3$	$m_2$	$m_1$	$m_3$	$m_3$	$j_2$	$j_1$	$j_3$

Zmienna binarna jest definiowana następująco:

$$x_{ijm} = \begin{cases} 1, & \text{jeżeli zadanie } i \text{ poprzedza zadanie } j \text{ na maszynie } m \\ 0, & \text{w przeciwnym przypadku} \end{cases}$$

Przeanalizujmy relację kolejnościową dla par  $(j_1, j_2)$  na maszynach  $(m_1, m_2, m_3)$ . Odpowiednio do danego harmonogramu mamy  $(x_{121}, x_{122}, x_{123}) = (0 \ 1 \ 0)$ . Dla par zadań  $(j_1, j_3)$  mamy  $(x_{131}, x_{132}, x_{133}) = (1 \ 0 \ 1)$  i dla par zadań  $(j_2, j_3)$  relacje kolejnościowe na maszynach  $(m_1, m_3, m_2)$  to  $(x_{231}, x_{233}, x_{232}) = (1 \ 1 \ 0)$ . Można zauważyć, że kolejność zmiennej  $x_{ijm}$  dla par zadań musi być zgodna z kolejnością operacji pierwszego zadania  $i$ . I tak, np. dla pary zadań  $(j_2, j_3)$ , kolejność operacji dla zadania  $j_2$  to  $(1 \ 3 \ 2)$ . Tak więc, zmienne względne są kolejka  $(x_{231}, x_{233}, x_{232})$ , raczej niż  $(x_{231}, x_{232}, x_{233})$ . Podsumowując te rezultaty otrzymujemy macierz binarną reprezentacji dla danego dopuszczalnego harmonogramu:

$$\begin{array}{l} (j_1, j_2) \text{ na } (m_1, m_2, m_3): \\ (j_1, j_3) \text{ na } (m_1, m_2, m_3): \\ (j_2, j_3) \text{ na } (m_1, m_3, m_2): \end{array} \begin{array}{ccc} x_{121} & x_{122} & x_{123} \\ x_{131} & x_{132} & x_{133} \\ x_{231} & x_{233} & x_{232} \end{array} \begin{array}{ccc} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{array}$$

### 8.2.5. Reprezentacja bazująca na regułach priorytetu

U. Dorndorf i E. Pesch [54] proponują algorytm genetyczny bazujący na regułach priorytetu, gdzie chromosom jest kodowany w kolejności reguł dyspozytorskich dla oznaczonych zadań. Algorytm genetyczny jest wykorzystywany do wyszukania najlepszej kolejności reguł dyspozytorskich. Dyspozytorskie reguły priorytetu są prawdopodobnie najlepszym zastosowaniem heurystyk dla rozwiązania problemu harmonogramowania w praktyce, ze względu na łatwość ich implementacji oraz najkrótszy całkowity czas zakończenia. Algorytm B. Gifflera i G. Thompsona [70] stanowić może wspólne podejście dla wszystkich heurystyk bazujących na regułach priorytetu. Dla problemu  $n$ -zadań i  $m$ -maszyn chromosom jest łańcuchem  $n \times m$  wejść  $(p_1, p_2, \dots, p_{nm})$ . Wejście  $p_i$  reprezentuje regułę zbioru wstępnie specyfikowanych dyspozytorskich reguł priorytetu. Wejście w  $i$ -tej pozycji oznacza, że konflikt w iteracji algorytmu B. Gifflera i G. Thompsona musi być rozwiązany z wykorzystaniem reguły priorytetu  $p_i$ , a ściślej, operacja z konfliktu może być umieszczona poprzez wybór z reguły  $p_i$ ; związany z niepewnością ze względu na wybór losowy. Niech:

- $PS_t$  = harmonogram częściowy zawiera  $t$  umieszczonych na nim operacji,
- $S_t$  = zbiór operacji, które mogą być umieszczone na harmonogramie w iteracji  $t$ , odpowiednio do danych  $PS_t$ ,
- $\sigma_i$  = najwcześniejszy czas, w którym operacje mogą się rozpocząć,
- $\phi_i$  = najwcześniejszy czas, w którym operacje  $i \in S_t$  mogą być zakończone,
- $C_t$  = zbiór operacji konfliktowych w iteracji  $t$ .

Procedura wyprowadzenia potomków z danego chromosomu  $(p_1, p_2, \dots, p_{nm})$  jest następująca:

Procedura: Wyprowadzanie potomków dla kodowania bazującego na regułach priorytetu [5].

*Krok 1:* Niech  $t = 1$  i na początku  $PS_t$  jest pustym harmonogramem częściowym i  $S_t$  zawiera wszystkie operacje bez poprzedników.

*Krok 2:* Określa się  $\phi_t^* = \min_{i \in S_t} \{\phi_i\}$  i maszynę  $m^*$ , na której  $\phi_i$  może być wykonane. Jeżeli jest więcej takich maszyn, wybór jest niepewny z uwagi na losowość.

*Krok 3:* Tworzenie zbioru konfliktów  $C_t$  zawierającego wszystkie operacje  $i \in S_t$ , z  $\sigma_i < \phi_t^*$ , które wymagają maszyn  $m^*$ . Wybór jednej operacji z  $C_t$ , w oparciu o regułę  $p_i$  i dodanie do operacji  $PS_t$  jako najwcześniejszej możliwej, odpowiednio do tworzonego nowego harmonogramu częściowego  $PS_{t+1}$ . Jeżeli jest więcej niż jedna operacja wybierana przez regułę priorytetu  $p_i$ , wybór jest niepewny ze względu na losowość.

*Krok 4:* Aktualizacja  $PS_{t+1}$  poprzez usuwanie wybranej operacji z  $S_t$  i dodawanie bezpośrednich potomków operacji do  $S_t$ . Powiększenie  $t$  o 1.

*Krok 5:* Powrót do kroku 2, aż zostanie wygenerowany zakończony harmonogram.

Procedurę tę można przeanalizować dla planu produkcji z tabeli 8.2 i reguł priorytetu zawartych w tabeli 8.5.

Tabela 8.5. Reguły priorytetu

Lp.	Reguła	Opis
1.	SPT ( <i>Shortest Processing Time</i> )	Wybór operacji z najkrótszym czasem wykonania
2.	LPT ( <i>Longest Processing Time</i> )	Wybór operacji z najdłuższym czasem wykonania
3.	MWR ( <i>Most Work Remaining</i> )	Wybór operacji, która należy do zadania o najdłuższym pozostałym jeszcze czasie do wykonania
4.	LWR ( <i>Least Work Remaining</i> )	Wybór operacji, która należy do zadania o najkrótszym pozostałym jeszcze czasie do wykonania

Rozważany chromosom jest następujący [1 2 2 1 4 4 2 1 3], gdzie 1 jest regułą SPT, 2 regułą LPT, 3 regułą MWR i 4 regułą LWR. Dla początkowego kroku:

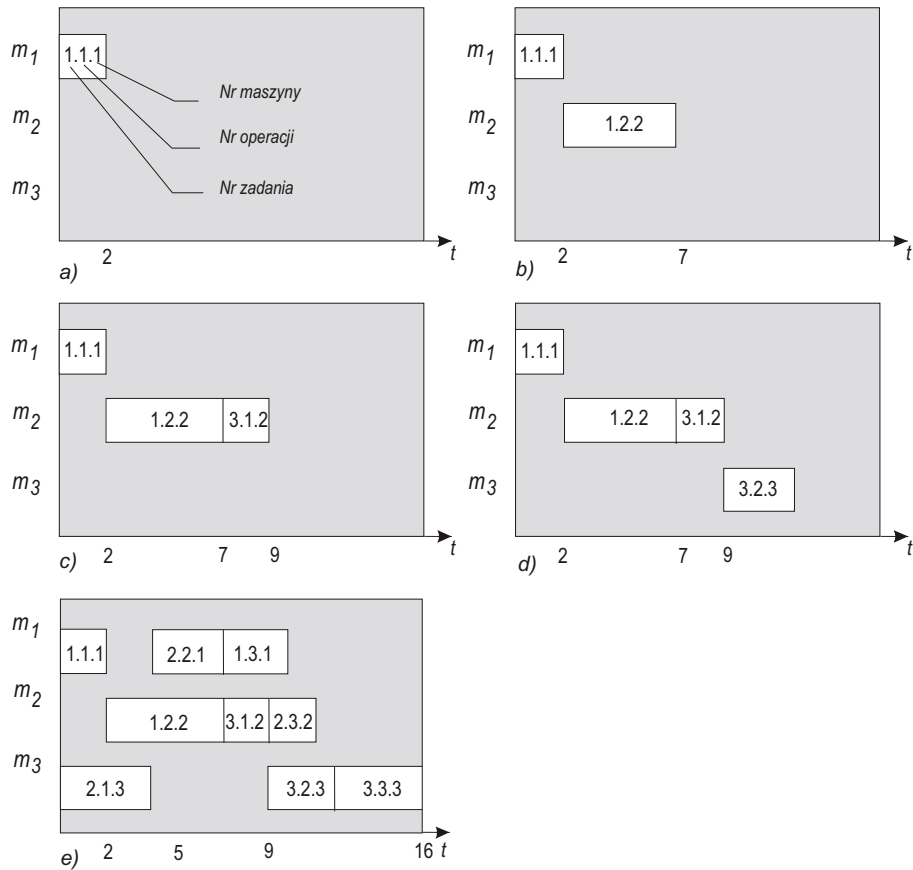
$$S_1 = \{o_{111}, o_{213}, o_{312}\}$$

$$\phi_1^* = \min \{2 \ 4 \ 2\} = 2$$

$$m^* = 1$$

$$C_1 = \{o_{111}\}$$

Operacja  $o_{111}$  jest umieszczana na harmonogramie na maszynie  $m_1$ , jak pokazano na rysunku 8.13a. Po aktualizacji danych otrzymuje się:



Rys. 8.13. Wyprowadzanie harmonogramu z chromosomu w reprezentacji bazującej na regułach priorytetu [opracowanie własne]

$$S_2 = \{o_{122}, o_{213}, o_{312}\}$$

$$\phi_2^* = \min \{7 \ 4 \ 2\} = 2$$

$$m^* = 2$$

$$C_2 = \{o_{122}, o_{312}\}$$

Operacje  $o_{122}$  i  $o_{312}$  współzawodniczą o maszynę  $m_2$ . W związku z tym, że drugi gen w chromosomie jest 2 (i określa regułę priorytetu LPT), operacja 1 2 2 jest umieszczana na harmonogramie na maszynie  $m_2$  (rys. 8.13b). Po aktualizacji otrzymuje się:

$$S_3 = \{o_{131}, o_{213}, o_{312}\}$$

$$\phi_3^* = \min \{10 \ 4 \ 2\} = 2$$

$$m^* = 2$$

$$C_3 = \{o_{312}\}$$

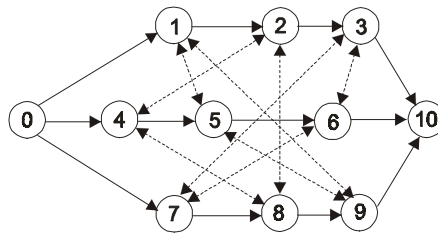
Operacja 3 1 2 jest umieszczana na maszynie  $m_2$  (rys. 8.13c). Po aktualizacji otrzymuje się:

$$\begin{aligned} S_4 &= \{o_{131}, o_{213}, o_{323}\} \\ \phi_4^* &= \min \{10 \ 4 \ 12\} = 4 \\ m^* &= 3 \\ C_4 &= \{o_{213}, o_{323}\} \end{aligned}$$

Obecnie operacje  $o_{213}$  i  $o_{323}$  współzawodniczą o maszynę  $m_3$ . W związku z tym, że czwarty gen chromosomu jest 1 (który określa regułę priorytetu SPT) operacja  $o_{323}$  jest umieszczana na harmonogramie na maszynie  $m_3$ , jak pokazano na rysunku 8.13d. Kroki procedury powtarzane są aż do całkowitego wyrowadzenie harmonogramu z chromosomu (rys. 8.13e).

### 8.2.6. Reprezentacja bazująca na grafie dysjunkcyjnym

H. Tamaki i Y. Nishikawa [190] zaproponowali reprezentację bazującą na grafie dysjunkcyjnym. Jest ona pewnym rodzajem reprezentacji bazującej na parach zadań. Problem harmonogramowania może być reprezentowany z wykorzystaniem grafu dysjunkcyjnego  $G = (N, A, E)$ , który jest definiowany następująco;  $N$  zawiera węzły reprezentujące wszystkie operacje,  $A$  zawiera łuki powiązań kolejności operacji dla tych samych zadań i  $E$  zawiera łuki dysjunkcyjne powiązań operacji dla tych samych maszyn. Ograniczenia dysjunkcyjne są reprezentowane przez łuki w  $E$ . Dysjunkcyjne łuki mogą być ustalone przez jego obydwie możliwe orientacje. Konstrukcja harmonogramu będzie ustalona orientacją wszystkich łuków dysjunkcyjnych określających kolejność operacji na maszynach. Rysunek 8.14 ilustruje graf dysjunkcyjny dla 3-zadań i 3-maszyn (tab. 8.3).



Rys. 8.14. Graf dysjunkcyjny dla problemu 3-zadań i 3-maszyn

Chromosom składa się z łańcucha binarnego odpowiednio do uporządkowanej listy łuków dysjunkcyjnych w  $E$ , gdzie  $e_g$  dla łuku dysjunkcyjnego pomiędzy węzłami  $i$  i  $j$  jest definiowane następująco:

$$e_{ij} = \begin{cases} 1, & \text{dla orientacji łuków dysjunkcyjnych z węzła } j \text{ do węzła } i \\ 0, & \text{dla orientacji łuków dysjunkcyjnych z węzła } i \text{ do węzła } j \end{cases}$$

lista porządku łuków dysjunkcyjnych:  $e_{15} \ e_{19} \ e_{59} \ e_{24} \ e_{28} \ e_{48} \ e_{36} \ e_{37} \ e_{67}$   
 chromosom:  $[0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1]$ .



Problemem harmonogramowania jest określenie kolejności operacji na każdej maszynie, tj. ustalenie orientacji łuków dysjunkcyjnych w taki sposób, żeby wynikowy graf był acykliczny, co stanowi gwarancję, że nie ma konfliktu pierwszeństwa między operacjami. Jednakże może się zdarzyć, że chromosom dostarczy graf cykliczny i wtedy harmonogram jest niedopuszczalny [37]. Tak więc chromosom nie jest wykorzystywany do reprezentacji harmonogramu, tylko do preferencji decyzji. W tej reprezentacji są stosowane krytyczne procedury bazujące na ścieżce. Podczas procesu dedukcji, wtedy kiedy występuje konflikt dwóch węzłów (operacji) na maszynie, odpowiedni bit chromosomu jest wykorzystywany do ustalenia kolejności wykonania dwóch operacji. Inaczej mówiąc – do ustalenia orientacji dysjunkcyjnej łuków pomiędzy dwoma węzłami.

### 8.2.7. Reprezentacja bazująca na terminie zakończenia

T. Yamada i R. Nakano [205] zaproponowali reprezentację bazującą na terminie zakończenia, gdzie chromosom jest uporządkowaną listą czasów zakończenia operacji. Dla przykładu z tabeli 8.3 chromosom może być reprezentowany następująco:

$$[c_{111} \ c_{122} \ c_{133} \ c_{211} \ c_{223} \ c_{232} \ c_{312} \ c_{321} \ c_{333}]$$

gdzie  $c_{jir}$  oznacza czas zakończenia dla operacji  $i$  zadania  $j$  na maszynie  $r$ . Ta reprezentacja w ocenie R. Chenga, M. Gena i Y. Tsujimury [37] nie jest odpowiednia dla większości operatorów genetycznych, ponieważ będzie następowało generowanie nielegalnych harmonogramów. W związku z tym T. Yamada i R. Nakano [205] zaprojektowali specjalne operatory krzyżowania.

### 8.2.8. Reprezentacja bazująca na maszynach

U. Dorndorf i E. Pesch [54] zaproponowali algorytm genetyczny, w którym chromosom jest kodowany jako kolejność maszyn i harmonogram jest skonstruowany z heurystycznym przesuwaniem *wąskich gardeł* bazującym na kolejności. Heurystyczne przesuwanie „wąskich gardeł” (*shifting bottleneck heuristic*) zaproponowane przez J. Adamsa, E. Balasa i D. Zawacka [4] jest prawdopodobnie najmocniejszą procedurą spośród wszystkich problemów harmonogramowania wewnątrzkomórkowego.

Heurystyka przesuwania „wąskich gardeł” bazuje na klasycznej koncepcji nadawania im priorytetów. Różne miary jakości „wąskich gardeł” określają różne kolejki do „wąskich gardeł”. Jakość harmonogramów uzyskanych przez heurystyki przesuwania „wąskich gardeł” silnie zależy od kolejności „wąskich gardeł” maszyn. J. Adams proponuje wyliczanie heurystyk przesuwania „wąskich gardeł” dla różnej kolejności maszyn. Zamiast wyliczania, tj. przeszukiwania drzewa, U. Dorndorf i E. Pesch [54] proponują strategię genetyczną do określania najlepszej kolejności maszyn dla heurystyk przesuwania „wąskich gardeł”. Chromosom jest listą uporządkowanych maszyn. Algorytm genetyczny

jest wykorzystywany do rozwinięcia chromosomów i odkrycia najlepszej kolejności dla heurystyk przesuwania „wąskich gardeł”. Niech  $M_0$  będzie zbiorem maszyn już uporządkowanych i dany jest chromosom  $[m_1, m_2, \dots, m_m]$ . Procedura wyprowadzania harmonogramu z chromosomu przebiega następująco [54]:

Procedura. Wyprowadzanie harmonogramu z chromosomu bazujące na kodowaniu maszyn.

*Krok 1:* Niech  $M_0 \leftarrow \emptyset$ ,  $i \leftarrow 1$  i chromosom  $[m_1, m_2, \dots, m_m]$ .

*Krok 2:* Kolejność maszyny  $m_i$  optymalna. Aktualizacja zbioru  $M_0 \leftarrow M_0 \cup \{m_i\}$ .

*Krok 3:* Powtórna optymalizacja kolejności każdej maszyny krytycznej włączonej w ruch podczas trzymania innych ustalonych kolejności.

*Krok 4:* Niech  $i \leftarrow i + 1$ . Jeżeli  $i > m$  to koniec, w innym przypadku powrót do kroku 2.

Szczegółowy opis kroku 3 można znaleźć w pracy [6].

### 8.2.9. Reprezentacja losowa

Reprezentacja losowa została po raz pierwszy zastosowana przez J. Beana [17]. Reprezentacja losowa koduje rozwiązanie z liczbą losową. W problemie harmonogramowania dla  $n$ -zadań na  $m$ -maszynach każdy gen (klucz losowy) składa się z dwóch części: liczby całkowitej ze zbioru  $[1, 2, \dots, m]$  i ułamkowej wygenerowanej losowo z  $(0,1)$ . Część całkowita według klucza losowego jest interpretowana jako oznaczenie maszyny dla danego zadania. Sortowanie części ułamkowej wyprowadza kolejności zadań na każdej maszynie. Poniżej rozważono przykład z tabeli 8.3. Niech chromosom ma postać:

[1.34 1.09 1.88 2.66 2.91 2.01 3.23 3.21 3.44].

Sortowanie według klucza dla maszyny 1, w porządku rosnącym, daje rezultat w postaci kolejności zadania  $2 \rightarrow 1 \rightarrow 3$ , dla maszyny 2 kolejność zadania  $3 \rightarrow 1 \rightarrow 2$  i dla maszyny 3 kolejność zadania  $2 \rightarrow 1 \rightarrow 3$ . Niech  $o_{jm}$  oznacza zadanie  $j$  na maszynie  $m$ . Chromosom może być translacją z unikalnej listy uszeregowanych operacji, jak następuje:

$[o_{21} o_{11} o_{31} o_{32} o_{12} o_{22} o_{23} o_{13} o_{33}]$ .

Można zauważyć, że ustalona kolejność zadań może naruszyć ograniczenia kolejnościowe.

### 8.2.10. Symbiotyczny algorytm ewolucyjny

Y. Kim, K. Park i J. Ko [101] zaproponowali symbiotyczny algorytm ewolucyjny do integracji planowania procesu technologicznego i harmonogramowania w elastycznym systemie produkcyjnym. Podejście to ze względu na przyjęte założenia ma ograniczone zastosowanie praktyczne. Prezentowane jest tu jednak w celu pokazania sposobu reprezentacji problemu i wynikającego z tego

stopnia skomplikowania problemu. Autorzy przy budowie takiego modelu planowania i harmonogramowania w elastycznym systemie produkcyjnym przyjęli następujące założenia:

1. Zadania nie mogą być dzielone i każda maszyna może wykonywać tylko jedno zadanie w danym czasie.
2. Wszystkie zadania są równocześnie dostępne w momencie zero.
3. Zadanie, po wykonaniu go na danej maszynie, jest natychmiast transportowane na następną maszynę, a czas tego transportu nie jest brany pod uwagę.
4. Czas przygotowania operacji na maszynie jest zależny od sekwencji operacji i jest uwzględniany w czasie wykonania operacji.

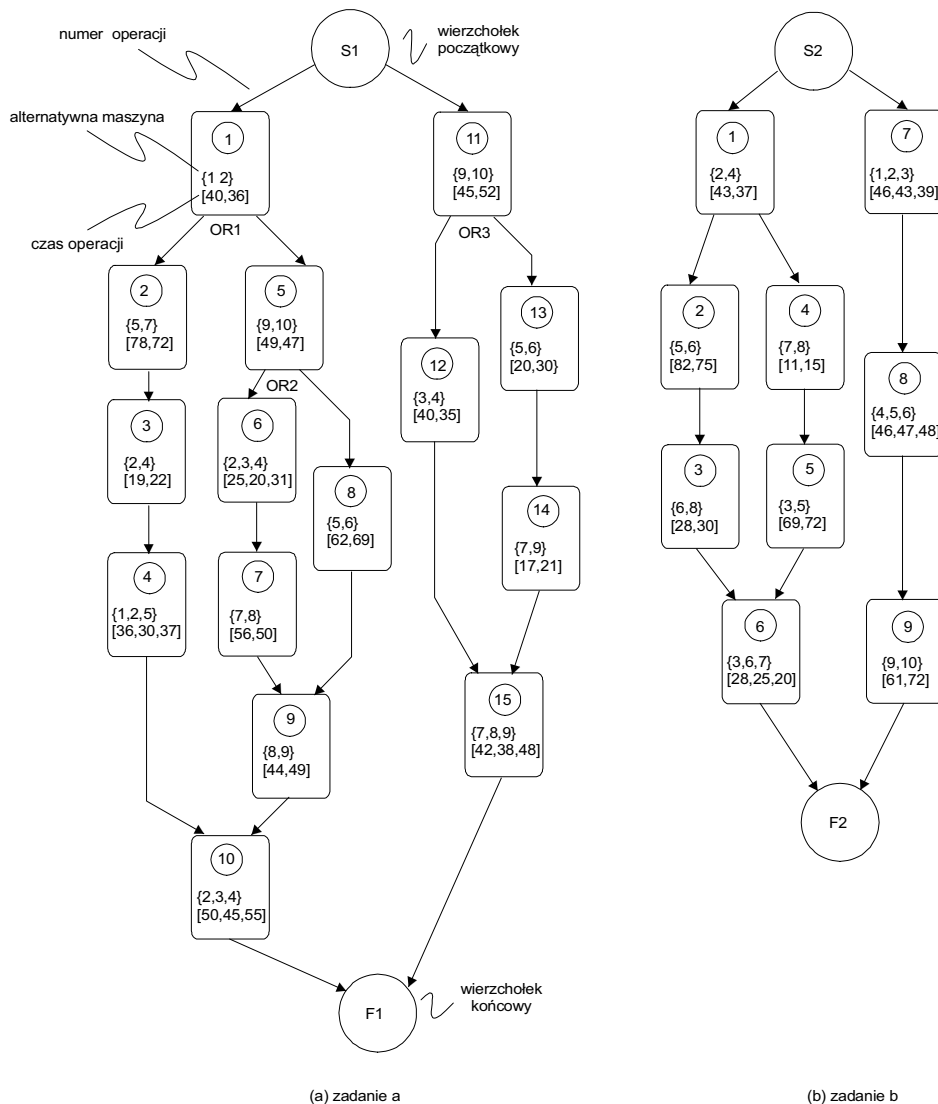
Na rysunku 8.15 zilustrowano analizowany proces technologiczny.

W prezentowanym tu podejściu autorzy rozważają trzy typy planów procesu: elastyczne operacje, elastyczną sekwencję i elastyczny proces. Elastyczna operacja odnosi się do możliwości jej wykonania na alternatywnej maszynie, z możliwością różnych czasów wykonania i kosztu. Ten typ jest często nazywany *elastycznością marszruty*. Elastyczność sekwencji odpowiada możliwości wymiany sekwencji, w której wymagane operacje są wykonywane. Pierwszy i drugi typ wymagają alternatywnych maszyn i alternatywnych sekwencji, ale operacje do wykonania są niezmiennie.

Elastyczny proces jest określony możliwościami produkcji o tych samych cechach wytwarzania z alternatywną operacją lub sekwencją operacji. W ten sposób można uzyskać korzystniejsze średnie czasy przepływu, lepszą przepustowość systemu i wykorzystanie maszyn.

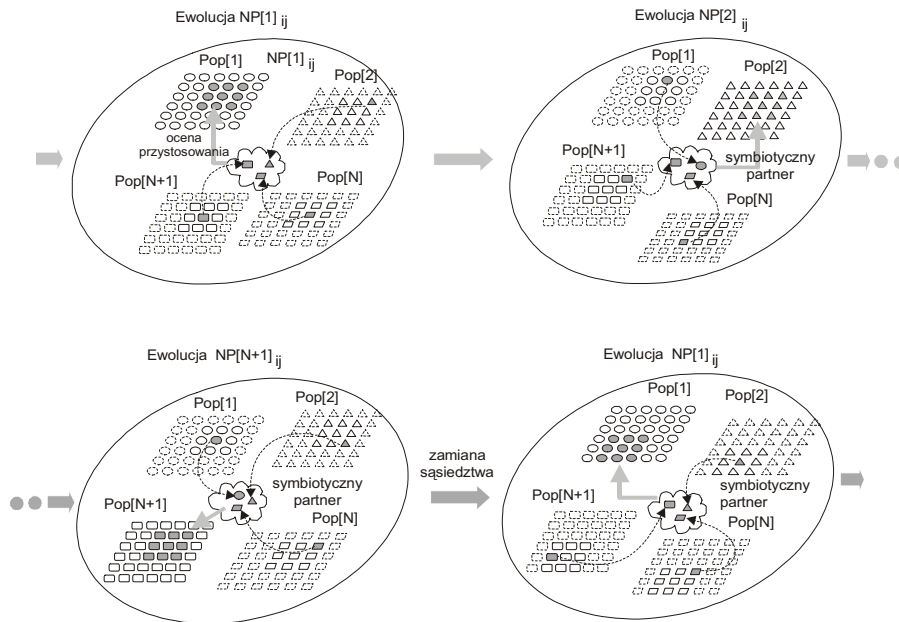
W prezentowanej tu koncepcji osobnik w populacji reprezentuje częściowe rozwiązanie całego problemu. Całkowite rozwiązanie jest konstruowane w wyniku kombinacji wszystkich częściowych rozwiązań. W tym podejściu ocena dopasowania osobnika wymaga selekcji jego symbiotycznych partnerów. Symbiotyczni partnerzy są osobnikami (częściowymi rozwiązaniami) połączonymi w ocenie dopasowania.

W rozważanym tu algorytmie ewolucyjnym wyróżnia się dwa typy różnych populacji: populację planu procesu ( $Pop[k]$ ),  $k = 1, 2, \dots, N$ ) i populację harmonogramu ( $Pop[N + 1]$ ). ( $Pop[k]$ ) dla  $k = 1, 2, \dots, N$  zawiera osobniki reprezentujące proces planowania dla zadania  $k$  i ( $Pop[N + 1]$ ) zawiera osobniki reprezentujące harmonogram. Każda populacja formułuje dwuwymiarową strukturę. W koewolucji jest używana struktura sąsiedztwa  $3 \times 3$ .  $NP[k]_{ij}$ , gdzie  $k = 1, 2, \dots, N + 1$  oznacza, że sąsiedztwo włącza osobnika  $(i, j)$  i jego ośmiu sąsiadów w ( $Pop[k]$ ). Każda komórka  $Pop[k]$ ,  $k = 1, 2, \dots, N + 1$  jest inicjalizowana z losowego tworzenia osobnika reprezentującego częściowe rozwiązanie. Przez pewien czas jedna populacja ewoluuje w interakcji z innymi populacjami w następujący sposób. Pierwszym krokiem algorytmu jest selekcjonowanie centrum ewolucji  $(i, j)$ . Osobniki w  $NP[q]_{i,j}$  kooperują z osobnikami z  $NP[k]_{ij}$ ,  $k = 1, 2, \dots, N + 1$  i  $k \neq q$ . Po wykonaniu lokalizacji ewolucji, dla populacji  $q$  rozwijane jest sąsiedztwo następnej populacji  $NP[q + 1]_{i,j}$ .



Rys. 8.15. Proces produkcyjny [101]

Lokalizacja ewolucji dla centrum  $(i, j)$  jest realizowana od  $NP[1]_{ij}$  do  $NP[N+1]_{ij}$ . Ten proces jest powtarzany ze zmienianym centrum ewolucji, np. ze zmianą sąsiedztwa, aż do satysfakcjonującego kryterium zatrzymania. Rysunek 8.16 przedstawia ogólną ideę proponowanego algorytmu.



Rys. 8.16. Ogólna idea symbiotycznego algorytmu [101]

Poniżej prezentowany jest symbiotyczny algorytm ewolucyjny.

- Krok 1:** Inicjalizacja. Generowanie zbioru  $N + 1$  populacji początkowej  $Pop[k]$ ,  $k = 1, 2, \dots, N + 1$ .
- Krok 2:** Początkowa ocena dopasowania. Ocena każdego osobnika, gdzie osobnik jest kombinacją wszystkich symbiotycznych partnerów, którzy są lokowani na tej samej pozycji we wszystkich innych populacjach. Zbiór  $f_{best}$  jest dopasowaniem najlepszych kombinacji.
- Krok 3:** Konstruowanie sąsiedztwa. Selekcja arbitralnych lokalizacji  $(i, j)$  i sąsiadów  $NP[k]_{ij}$  dla  $k = 1, 2, 3, \dots, N + 1$ . Zbiór  $q = 1$ .
- Krok 4:** Koewolucja.
- Krok 4.1:** Wybieranie dwóch rodziców z  $NP[q]_{i,j}$  w oparciu o dopasowanie i tworzenie dwóch potomków przez zastosowanie operatorów krzyżowania.
  - Krok 4.2:** Zastępowanie potomstwem nowo utworzonym w kroku 4.1 dwóch osobników, mających najgorsze dopasowanie w  $NP[q]_{i,j}$ .
  - Krok 4.3:** Zastosowanie operatora mutacji (jeżeli jest uaktywniony) do osobników w  $NP[q]_{i,j}$ .
  - Krok 4.4:** Ocenianie dopasowania dla osobników nowo wyprodukowanych, gdzie partnerzy symbiotyczni są losowo wybierani z każdego  $NP[k]_{i,j}$  dla  $k = 1, 2, 3, \dots, N + 1$  i  $k \neq q$ . Jeżeli dopasowanie najlepszego osobnika jest wyższe niż  $f_{best}$ , wtedy  $f_{best}$  jest aktualizowane z nowym najlepszym dopasowaniem.

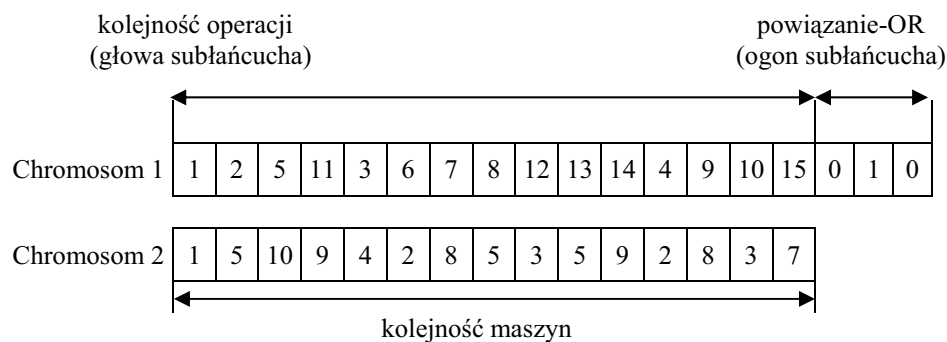
*Krok 4.5:* Zbiór  $q \leftarrow q + 1$ . Jeżeli  $q \leq N + 1$ , to następuje przejście do kroku 4.1.

W przeciwnym przypadku – przejście do kroku 5.

*Krok 5:* Warunek zatrzymania. Jeżeli kryteria zatrzymania są zadowalające, wtedy koniec. W przeciwnym przypadku – przejście do kroku 3.

Krok 4 jest odpowiedzialny za lokalizację ewolucji (bazowanie na typie *steady-state* GA).

Każdy osobnik w populacji planowania procesu  $Pop[k]$ ,  $k = 1, 2, \dots, N + 1$  zawiera dwa chromosomy z różną długością, jak pokazano na rysunku 8.17.



Rys. 8.17. Kodowanie procesu planowania [101]

Chromosom 1 dotyczy kolejności operacji dla zakończonych zadań, a chromosom 2 odpowiada przydzielonym maszynom. Chromosom 1 jest łańcuchem znaków, który jest podzielony na dwa subłańcuchy: głowę i ogon (jak pokazano na rys. 8.17). Głowa reprezentuje wszystkie operacje dla zadań, takich jak lista sekwencyjna. Lista ta zawiera wszystkie możliwe operacje, mimo że mogą one nie zostać wykonane ze względu na alternatywne procedury operacji. Kwestia ta może być rozwiązana przez użycie ogona. Ogon koduje powiązania-OR jako liczby binarne 0 lub 1. Liczba 0 oznacza wybór prawego powiązania-OR ścieżki, a liczba 1 lewego powiązania-OR ścieżki. Chromosom 2 jest łańcuchem, gdzie  $i$ -ty element reprezentujący maszynę odpowiada  $i$ -temu elementowi chromosomu 1. Jeżeli  $i$ -ta wartość elementu w chromosomie 1 i 2 jest odpowiednio  $j$  i  $m$ , to reprezentacja ta oznacza, że operacja  $j$  jest wykonywana na maszynie  $m$ . Schemat kodowania pokazano na przykładzie sieci przedstawionej na rysunku 8.15a, której odpowiada chromosom 1 i chromosom 2 na rysunku 8.17.

Kodowanie jest bezpośrednie. Operacje razem z odpowiednimi maszynami, które należą do nie wybranej ścieżki powiązań-OR, interpretowane jako ogon chromosomu 1, są pierwsze usuwane z osobnika. Następnie uporządkowania ukazujące się w wynikowych chromosomach 1 i 2 są interpretowane jako kolejności operacji i kolejności maszyn dla zadania. W powyższym przykładzie operacje 2-4, 8 i 12 są usuwane.

Kiedy generuje się osobników populacji początkowej, brana jest pod uwagę dopuszczalna kolejność operacji w planie procesu. Dopuszczalna kolejność operacji oznacza, że uporządkowanie elementów w kodowaniu nie narusza ograniczeń kolejnościowych. Jak wspomniano wyżej, sekwencja operacji ukazuje się w głowie łańcucha w chromosomie 1. Dopuszczalna kolejność operacji jest dostarczana w wyniku sortowania topologicznego, które transformuje częściowe uporządkowanie w liniowy porządek zgodnie z następującą procedurą:

- Krok 1:* Sformułuj początkowy dostępny zbiór operacji nie mających poprzednika i utwórz pewien pusty łańcuch.
- Krok 2:* Zakończ, jeżeli dostępny zbiór jest pusty. W przeciwnym przypadku idź do kroku 3.
- Krok 3:* Wybierz operację z dostępnego zbioru losowo i dołącz ją do łańcucha.
- Krok 4:* Aktualizuj dostępny zbiór przez usunięcie wybranej operacji i przez dodanie każdej najbliższej następnej operacji, jeżeli wszystkie najbliższe poprzedniki następników są już w łańcuchu. Idź do kroku 2.

Dopasowanie jest wartością z zakresu (0,1).

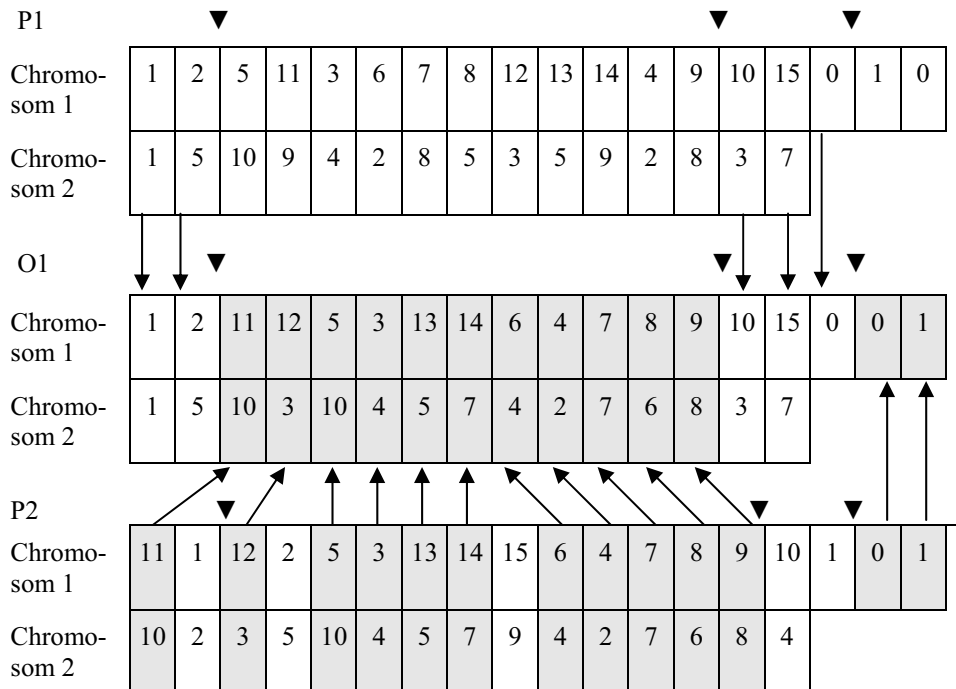
Funkcja dopasowania jest następująca:

$$f_q(s,t) = \frac{g_q(s,t) \square \{\max_{(u,v) \square NP[q]_j} g_q(u,v) \square 1\}}{\min_{(u,v) \square NP[q]_j} g_q(u,v) \square \{\max_{(u,v) \square NP[q]_j} g_q(u,v) \square 1\}}$$

gdzie  $g_q(u,v)$  jest długością uszeregowania (*makespan*).

W tym podejściu są używane dwa schematy krzyżowania dla rekombinacji osobników: pewna odmiana *two-point crossover* i tradycyjny *one-point crossover*. Odmiana *two-point crossover* jest zastosowana do głowy chromosomu 1 i 2. Dwa punkty przecięcia są wybierane losowo w głowie łańcucha. *Two-point crossover* jest zmodyfikowany dla generowania dopuszczalnych rozwiązań, które satysfakcjonują ograniczenia kolejnościowe i unikają powielania lub pominięcia operacji. Schemat tego krzyżowania jest następujący. Dwa punkty przecięcia wybierane są losowo w głowie subłańcucha i elementy przed pierwszym punktem przecięcia i po drugim punkcie przecięcia u rodzica (P1) przenoszone są na takie same pozycje jak w potomku (O1). Te elementy są usuwane z drugiego rodzica (P2), a pozostałe elementy są kopiowane w niezdeternowane pozycje (tj. pozycje pomiędzy dwa punkty przecięcia) w potomku, w tym samym uporządkowaniu jak w P2. Tradycyjnie *one-point crossover* jest zastosowany do subłańcucha ogona chromosomu 1. Operacja krzyżowania jest stosowana tylko wtedy, kiedy liczba elementów w ogonie jest większa niż 1. Inny potomek (O2) może być produkowany przez przełączenie roli z P1 na P2.

Przykład krzyżowania zaprezentowano na rysunku 8.18.

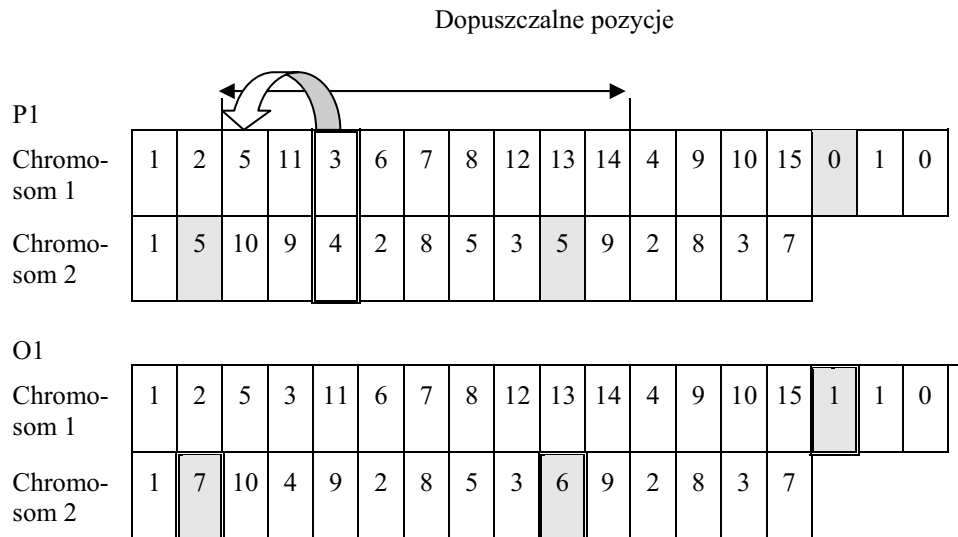


Rys. 8.18. Krzyżowanie dla planowania procesu [101]

W mutacji niektóre osobniki są losowo wybierane z częstością mutacji osobnika. Każdy wybrany osobnik jest mutowany jak następuje. Pierwszy schemat mutacji jest zastosowany do zmian kolejności operacji reprezentowanych w głowie chromosomu 1. Pewna operacja jest wybierana losowo z wyselekcjonowanych osobników. Następnie jest identyfikowana w osobniku potencjalna pozycja, na której wyselekcjonowana operacja może być umiejscowiona, bez naruszenia ograniczeń uporządkowania. W dopuszczalnej sekwencji, selekcja operacji musi następować we wszystkich jego najbliższych poprzednikach i poprzedzać wszystkie jego najbliższe następniki. Spośród potencjalnych wprowadzonych pozycji jedna zostaje wyselekcjonowana. Wybrana operacja razem z elementem ulokowanym na tej samej pozycji w chromosomie 2 jest umieszczana na wyselekcjonowanej pozycji.

Drugi schemat mutacji to przeprowadzanie zmiany przydzielonych maszyn i ścieżki połączeń-OR. Jest on związany z chromosomem 2 i ogonem łańcucha chromosomu 1. Element chromosomu 2 jest mutowany przez zmianę liczby maszyny na jedną alternatywną, jedną losową, a ogon przez zmianę z 1 na 0 lub odwrotnie. W przykładzie przedstawionym na rysunku 8.19 piąty element jest pierwszy wyselekcjonowany dla mutacji i operacja ta jest wstawiana w losową pozycję spośród dopuszczalnych. Po tej mutacji, poceniowane elementy są mutowane – losowo wybierane odpowiednio do każdego współczynnika mutowanego genu dla chromosomu 2 i ogona chromosomu 1.





Rys. 8.19. Mutacja dla procesu planowania [101]

Osobniki w  $Pop[N + 1]$  są reprezentowane przez permutację operacji dla każdego zadania. Jest to naturalna reprezentacja uporządkowania. Wszystkie operacje skojarzone z powiązaniem – OR są włączone w permutację. Przypomnijmy, że operacja w każdym zadaniu ma ograniczenia kolejnościowe. Dlatego nie wszystkie permutacje reprezentują bezpośrednio dopuszczalne harmonogramy. Operacja jest możliwa do umieszczenia na harmonogramie tylko wtedy, jeśli zadania poprzedzające ją są już umieszczone na harmonogramie. W niniejszym podejściu to ograniczenie kolejnościowe jest zaniechane na poziomie kodowania i poziomie operatorów genetycznych. Dopuszczalność harmonogramu jest zapewniona przez procedurę dekodowania. Osobniki populacji początkowej są generowane w wyniku permutacji losowej operacji (wszystkich rozważanych zadań). C. Bierwirth i D.C. Mattfeld [19] prezentują procedurę dekodowania permutacji w harmonogramach: pseudoaktywnych, aktywnych, bez opóźnień, hybrydowych. Te harmonogramy oprócz harmonogramu hybrydowego są definiowane i objaśnione przez K.R. Bakera [9]. Wymienione powyżej cztery typy harmonogramów mogą być generowane w oparciu o dobrze znany algorytm B. Gifflera i G. Thompsona [70]. C. Bierwirth i D.C. Mattfeld dowodzą, że procedura generowania harmonogramu hybrydowego może poprawiać jednocześnie jakość rozwiązań i redukować czas obliczeń. Dlatego zastosowano ją w niniejszym podejściu. Jak wspomniano powyżej, poziom dekodowania i partnerzy symbiotyczni osobnika populacji harmonogramowania są określane w ten sposób, że jest dany stały, przypisany plan procesu dla każdego zadania.

W prezentowanej poniżej procedurze dekodowania zastosowano następującą notację:

- $o_{ij}$  –  $j$ -ta operacja zadania  $i$ ,
- $\sigma_{ij}$  – najwcześniejszy termin rozpoczęcia operacji  $o_{ij}$ ,
- $p_{ij}$  – czas wykonania operacji  $o_{ij}$ ,
- $\tau_{ij}$  – najwcześniejszy termin zakończenia operacji  $o_{ij}$ , tj.  $\tau_{ij} = \sigma_{ij} + p_{ij}$ .

Procedura:

- Krok 1:* Konstrukuj zbiór wszystkich początkowych operacji;  $A := \{o_{i1} \mid 1 \leq i \leq N\}$ .
- Krok 2:* Określ  $\tau^* = \min \{\tau_{ij} \mid o_{ij} \in A\}$  i maszynę  $m^*$ , na której  $\tau^*$  mogło, by być realizowane.
- Krok 3:* Buduj zbiór  $B$  ze wszystkich operacji w  $A$ , które są wykonywane na  $m^*$  i wtedy określ  $\sigma^* = \min \{\sigma_{ij} \mid o_{ij} \in B\}$ .
- Krok 4:* Buduj zbiór  $C$  w zgodzie z parametrem  $\theta$  takim, że  $C := \{o_{ij} \in B \mid \sigma_{ij} \leq \theta\tau^* + (1 - \theta)\sigma^*, 0 \leq \theta \leq 1\}$ .
- Krok 5:* Wybierz operację  $o_{ij}^*$  z  $C$ , która znajduje się najbardziej na lewo w permutacji i usuń z  $A$ ,  $A := A \setminus \{o_{ij}^*\}$ .
- Krok 6:* Dołącz operację  $\{o_{ij}^*\}$  do harmonogramu i oblicz jego termin rozpoczęcia i zakończenia.
- Krok 7:* Jeżeli następnik  $o_{ij+1}^*$  operacji  $o_{ij}^*$  istnieje, to wstaw to w  $A$ ;  $A := A \cup \{o_{ij+1}^*\}$ .
- Krok 8:* Jeżeli  $A \neq \emptyset$  – idź do kroku 2. W przeciwnym przypadku – zakończenie.

Do rekombinacji osobnika w  $Pop\{N + 1\}$  zastosowano operator *precedence preservative crossover* (PPX) rozwijany dla problemów harmonogramowania przez C. Bierwirtha [20]. Operator PPX działa następująco. Osobnikowi losowo utworzonemu z elementów zbioru  $\{1,2\}$  odpowiada identyczna długość łańcucha. Łańcuch ten definiuje uporządkowanie, w którym operacje są sukcesywnie wybierane z rodziców 1 i 2. Na początku następuje inicjacja pustego potomka. Operacja na lewo, w jednym z dwóch rodziców jest wybierana zgodnie z uporządkowaniem rodzica w łańcuchu. Wybrana operacja jest dołączana do potomka, a następnie jest usuwana w obu rodzicach. Proces jest powtarzany tak długo, aż oboje rodzice są „pustymi”. Rysunek 8.20 pokazuje przykład zastosowania operatora PPX. Rozważany jest problem harmonogramowania zawierający dwa zadania: zadanie  $a$  i zadanie  $b$  pokazane na rysunku 8.15, z całkowitą liczbą 24 operacji. Liczba operacji 16-24 na rysunku 8.15 odpowiada 1-9 w zadaniu  $b$  w uporządkowaniu liczbowym. PPX utrzymuje uporządkowanie danych operacji w dwóch rodzicielskich permutacjach z tym samym współczynnikiem. Jest to istotne dla problemu harmonogramowania, ponieważ uporządkowanie operacji przypisanych do każdej maszyny ma szczególne znaczenie.

Operator mutacji działa następująco. Początkiem jest losowy wybór osobników z częstością mutacji osobnika i zdejmowanie elementów w wybranym osobniku, po czym następuje wstawienie tych elementów w pozycję losowo wybranego osobnika.

P1	1	2	3	6	8	4	14	13	9	21	16	18	24	5	10	22	7	11	17	12	15	23	19	20
P2	7	15	2	3	6	9	14	21	1	4	5	16	17	12	18	19	20	24	10	8	11	13	22	23
S	1	2	2	1	2	1	2	1	1	2	1	1	1	2	1	2	2	1	2	2	2	1	2	1
O1	1	7	15	2	3	6	9	8	4	14	13	21	16	5	18	17	12	24	19	20	10	22	11	23

Rys. 8.19. Krzyżowanie utrzymujące uporządkowanie [101]

Przyjęte przez autorów [101] założenia do prezentowanego powyżej modelu, a głównie pominięcie czasu transportu ograniczają jego zastosowanie, ponieważ w przypadkach zastosowania maszyn outsourcingowych (np. maszyn znajdujących się w odległych geograficznie zakładach podwykonawców) czasy transportu mogą być bardzo długie. Ponadto jak wspomniano wcześniej w rozprawie duże skomplikowanie algorytmów szeregowania zadań znacznie wydłuża czasy obliczeń. Z tego względu model ma niewielką przydatność praktyczną. Autorzy nie przetestowali systemu na danych rzeczywistych, więc nie są znane rezultaty zastosowań praktycznych.

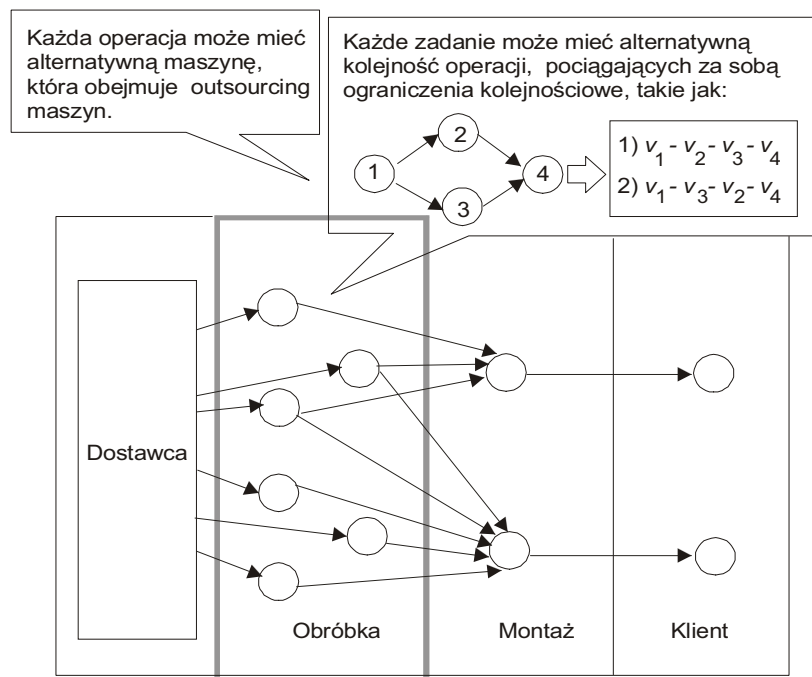
### 8.2.11. Zaawansowane planowanie i harmonogramowanie z outsourcingiem

W elastycznych systemach produkcyjnych bardzo ważne jest takie harmonogramowanie produkcji, które zapewni dotrzymanie dyrektywnych terminów jej zakończenia. Specyfiką elastycznej produkcji są małe serie i operacje, które dają długie cykle produkcyjne. W przypadkach, w których termin realizacji zlecenia klienta nie może być dotrzymany (istnieje zagrożenie przekroczenia terminu zakończenia) pewnym rozwiązaniem problemu może być użycie alternatywnych maszyn, tj. włączony zostaje outsourcing maszyn dla operacji zadania objętego zleceniem klienta. Koncepcję planowania i harmonogramowania produkcji w elastycznym systemie produkcyjnym z uwzględnieniem outsourcingu zaproponowali Y.H. Lee, Ch.S. Jeong i Ch. Moon [116]. Poniżej przedstawiono rozważania autorów dotyczące modelu działań planistycznych, których celem jest zminimalizowanie całkowitego czasu zakończenia zadania (*makespan*) oraz dotrzymanie terminu realizacji zlecenia klienta.

Strukturę łańcucha dostaw rozważanego tu elastycznego procesu przemysłowego pokazuje rysunek 8.21.

Zaprezentowany łańcuch ma czteroelementową strukturę (dostawca, obróbka, montaż, klient). Na poziomie klienta generowane są zlecenia z określonymi terminami zakończenia. Każde zlecenie określa produkt i jego montaż, po wykonaniu obróbki partii części. Każdy produkt jest wykonywany w trakcie

określonego procesu montażu i każdy produkt może potrzebować różnych części, wykonywanych w powiązanych zadaniach poziomu obróbki. Montaż może rozpocząć się tylko po zakończeniu wszystkich zadań obróbki i każdy proces montażu jest w relacji z innymi procesami montażu. Każde zadanie na poziomie obróbki specyfikuje część produktu ze zlecenia, która może wymagać różnych operacji. Ponadto operacje sformułowanego zadania obróbki mogą zawierać ograniczenia kolejnościowe. W rzeczywistości bowiem, pewne operacje zadania są związane ze sobą ograniczeniem uporządkowania.

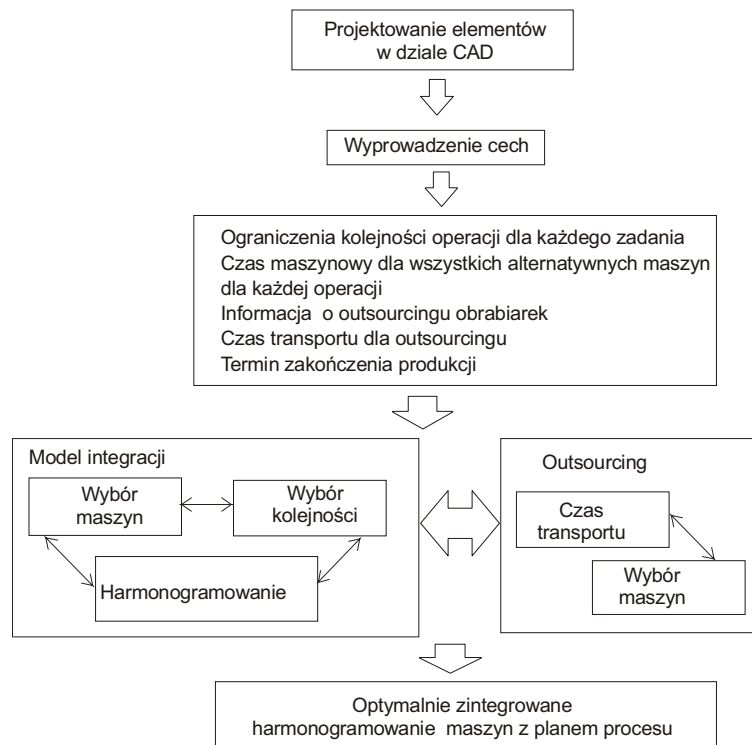


Rys. 8.21. Łańcuch dostaw wytwarzania w FMS [116]

Analizowany tu problem wytwarzania obejmuje  $k$  maszyn (1,2, ..., $k$ ) i  $n$  różnych zadań (1, 2, ...,  $n$ ). Wszystkie zadania są rozpoczynane i realizowane nieprzerwanie jako partie, odpowiednio do wcześniej określonych technologicznych kolejności zawartych w planie procesu. Każde zadanie wymaga pewnej liczby operacji i każda z tych operacji może być wykonana na pewnej liczbie alternatywnych, nieidentycznych maszyn, również w ramach outsourcingu maszyn. Maszyna musi być wybrana spośród alternatyw dla kolejności operacji dla każdego zadania. Dla danego zlecenia produkcji, które dotyczy mieszanki zadań i maszyn, kolejność operacji musi być wybrana w taki sposób, żeby zapewniona została maksymalna efektywność produkcji dla wszystkich zadań.

W tym opracowaniu analiza koncentruje się na poziomie obróbki, ponieważ te procesy stanowią „wąskie gardła” w dotrzymaniu terminu zakończenia.

Rysunek 8.22 ilustruje schematyczny diagram modelu APS (*Advanced Planning System*) z outsourcingiem. Elementy modelu są zintegrowane i działają jednocześnie (równolegle). Outsourcing jest najmocniej związany z terminem zakończenia.



Rys. 8.22. Schemat modelu APS [116]

W związku z tym, że pewne operacje produkcji w toku mogą zostać wybrane do outsourcingu model musi obejmować także czas transportu części z/do głównej fabryki do/z miejsca outsourcingu w trakcie procesu. Model obejmuje więc:

- określanie optymalnego harmonogramu z minimalnym terminem zakończenia każdego zlecenia (*makespan*),
- wybieranie najlepszej maszyny dla każdej operacji,
- określanie optymalnej kolejności operacji każdego zadania,
- wybieranie operacji do outsourcingu, kiedy zdolności produkcyjne maszyny nie pozwalają dotrzymać terminu zakończenia.

W tym podejściu jest wykorzystywany algorytm sortowania topologicznego (*topological sort*) – TS [87] do rozwiązania problemu określania sekwencji kolejności operacji dla zadań. Zastosowanie algorytmu TS do testu dopuszczalności czynności w wierzchołkach (AOV) sieci będzie generować liniowy

porządek  $v_1, v_2, \dots, v_j$  wierzchołków. Ten liniowy porządek jest priorytetem takim, że jeżeli  $v_1$  jest poprzednikiem  $v_2$  w sieci, wtedy  $v_1$  poprzedza  $v_2$  w liniowym porządku. Liniowy porządek z priorytetem jest ustalonym porządkiem topologicznym – TO (*topological order*).

Procedura: Sortowanie topologiczne.

**Input:** Sieć AOV. Niech  $J$  będzie liczbą wierzchołków;

**For**  $j=1$  do  $J$  do

**Begin**

**If** każdy wierzchołek ma poprzednik

Wtedy sieć jest niedopuszczalna stop;

**Else** wybierz wierzchołek  $v_j$ , który nie ma poprzednika;

**Output**  $v_j$

**Delete**  $v_j$  i wszystkie krawędzie wychodzące  $v_i$  z sieci;

**End;**

**End;**

W prezentowanym tu modelu APS kolejność operacji i wybór maszyny muszą być zintegrowane i *makespan* musi być minimalizowany w związku z koniecznością uwzględnienia terminu zakończenia zlecenia klienta. W sytuacji, kiedy wewnątrzzakładowe maszyny nie są dostępne i nie jest możliwe dotrzymanie terminu zakończenia zlecenia może być rozważany outsourcing. Do sformułowania zintegrowanego modelu przyjęto następującą notację:

- $r$  – indeks zlecenia klienta,  $r = 1, 2, 3, \dots, R$ ,
- $i$  – indeks części dla zlecenia  $r$ ,  $i = 1, 2, 3, \dots, I$ ,
- $o$  – indeks alternatywnego planu procesu TO dla części  $i$ ,  $o = 1, 2, 3, \dots, O_i$ ,
- $j$  – indeks operacji o TO,  $j = 1, 2, 3, \dots, J_o$ ,
- $k$  – indeks maszyny obejmujący outsourcingowe maszyny  $k = 1, 2, 3, \dots, K$ ,
- $R$  – całkowita liczba zleceń,
- $I_r$  – całkowita liczba części w zleceniu  $r$ ,
- $O_i$  – całkowita liczba TO dla części  $i$ ,
- $J_o$  – całkowita liczba operacji w TO  $o$ ,
- $K$  – całkowita liczba maszyn z włączeniem outsourcingowych maszyn,
- $mat_{riojk}$  – czas wykonania operacji  $j$  w TO  $o$  dla części  $i$  w zleceniu  $r$  na maszynie  $k$ ,
- $TM_{riojk}$  – termin zakończenia operacji  $j$  w TO  $o$  dla części  $i$  w zleceniu  $r$  na maszynie  $k$ ,
- $MS_r$  – makespan obróbki dla zlecenia klienta  $r$ ,
- $DT_r$  – czas dostawy dla zlecenia klienta  $r$ ,
- $ATr$  – czas montażu dla zlecenia klienta  $r$ ,
- $MP_{rk}$  – termin zakończenia wykonania dla maszyny  $k$  na zlecenie  $r$ ,
- $TR_k$  – czas transportu do maszyny  $k$ , ulokowanej w outsourcingowym przedsiębiorstwie,
- $D_r$  – termin zakończenia zlecenia klienta  $r$ ,

$S_{rijk}$  – termin rozpoczęcia operacji  $j$  dla części  $i$  w zleceniu  $r$  na maszynie  $k$ ,  
 $E_{rijk}$  – termin zakończenia operacji  $j$  dla części  $i$  w zleceniu  $r$  na maszynie  $k$ ,  
 $L$  – duża liczba dodatnia.

$X_{riojk} \begin{cases} \square 1, & \text{jeżeli operacja } j \text{ w TO } o \text{ dla części } i \text{ w zleceniu } r \text{ jest wykonywana} \\ \square & \text{na maszynie } k \\ \square & \text{w przeciwnym przypadku} \end{cases}$

$R_{riojefgk} \begin{cases} \square 1, & \text{jeżeli operacja } j \text{ w TO } o \text{ dla części } i \text{ poprzedza operację } g \text{ w TO } f \\ \square & \text{dla części } e \text{ w zleceniu } r, \text{ gdzie operacja } j \text{ i } g \text{ są wykonywane} \\ \square & \text{na maszynie } k \\ \square & \text{w przeciwnym przypadku} \end{cases}$

$Y_{rio} \begin{cases} \square 1, & \text{jeżeli plan procesu } o \text{ dla części } i \text{ w zleceniu } r \text{ jest wybrany} \\ \square & \text{w przeciwnym przypadku} \end{cases}$

$Z_k \begin{cases} \square 1, & \text{jeżeli } k \text{ jest na maszynie outsourcingowej} \\ \square & \text{w przeciwnym przypadku} \end{cases}$

Wprowadza się dwie zmienne decyzyjne. Pierwszą dla wyboru maszyny, tj. binarną zmienną decyzyjną  $X_{riojk}$ , która równa się 1, jeżeli operacja  $j$  w TO  $o$  dla części  $i$  w zleceniu  $r$  jest wykonywana na maszynie  $k$ . Drugą, dla wyboru kolejności operacji, tj. binarną zmienną decyzyjną  $Y_{rio}$ , która równa się 1, jeżeli TO  $o$  dla części  $i$  na zlecenie  $r$  zostaje wybrane.

W przypadkach, kiedy TS generuje alternatywną kolejność operacji, każda część ma zbiór kolejności operacji, które są ustalone porządkiem topologicznym TO. Termin zakończenia dla operacji  $j$  w TO  $o$  dla części  $i$  w zleceniu  $r$  na maszynie  $k$  ( $TM_{riojk}$ ) jest dany przez (1). Wartość  $TM_{riojk}$  otrzymuje się wtedy, kiedy tylko jeden TO jest wybrany dla każdej części, i uwzględnia się ograniczenia, że tylko jedna maszyna może być wybrana dla każdej operacji:

$$TM_{riojk} \begin{matrix} \square & \square & \square & \square & \square \\ r & i & o & j & \end{matrix} \text{mat}_{riojk} X_{riojk} Y_{rio}, \square k \quad (8.22)$$

gdzie:

$$\begin{matrix} K \\ \square & X_{riojk} \square 1, & \square r, i, o, j; \\ k \square 1 \end{matrix} \quad \begin{matrix} O_i \\ \square & Y_{rio} \square 1, & \square r, i \\ o \square 1 \end{matrix}$$

$MP_{rk}$  ocenia termin zakończenia procesu na maszynie  $k$  dla zlecenia  $r$  i definiuje jako:

$$MP_{rk} \square TM_{rI_r, O_i, J_o, k}, \square r, k \quad (8.23)$$

Równanie (8.23) jest stosowane do otrzymywania terminu zakończenia procesu dla każdej maszyny dla ostatniej operacji w zleceniu  $r$ .  $MS_r$  jest określone przez sumaryczną liczbę części do wytworzenia dla zlecenia  $r$  i czasu

wykonania związanego z każdą operacją. Maksymalna wartość  $MP_{rk}$  jest wartością funkcji celu i definiuje się ją jako:

$$\text{Min } MS_r \square \max_{k \square 1, \dots, K} (MP_{rk}) \quad (8.24)$$

Funkcja celu wyrażona w równaniu (8.24) jest stosowana do minimalizacji terminu zakończenia dla ostatniej operacji  $j = J_o$  w TO  $o$  dla części  $i = I_r$  w zleceniu  $r$ , jeżeli TO  $o$  dla części  $i$  i maszyny  $k$  dla operacji  $j$  są wybierane przez ograniczenia (8.22) i (8.23).

Równanie (8.25) satysfakcjonuje termin zakończenia zlecenia klienta  $r$ :

$$MS_r \leq D_r - AT_r - DT_r, \quad \forall r. \quad (8.25)$$

Dla operacji na tej samej części i wyborze operacji do outsourcingu:

$$TM_{riojk} + L(1 - Y_{rio}) \geq \text{mat}_{riojk}, \quad \forall r, i, o, j, k \quad (8.26)$$

gdzie  $j = 1$ ;

$$TM_{riojk} - TM_{rio(j-1)m} + TR_k Z_k + TR_m Z_m + L(1 - Y_{rio}) \geq \text{mat}_{riojk}, \quad \forall r, i, o, j, k, m \quad (8.27)$$

gdzie  $1 < j \leq J_o$  i  $X_{riojk} = X_{rio(j-1)} = 1$ .

Równania (8.26) i (8.27) zapewniają, że dla danej części  $i$ , operacji  $j-1$  wykonanie na maszynie  $m$  poprzedza następną operację ( $j$ ) wykonywaną na maszynie  $k$ . Jeżeli  $k$  jest maszyną outsourcingową to  $Z_k = 1$  i operacja  $j$  będzie może być wybrana do outsourcingu. Równanie 8.26 zadowala warunek, że termin zakończenia dwóch operacji (zarówno  $j - 1$ , jak i  $j$ ) dla części powinien mieć większe różnice, aniżeli czas maszynowy dla operacji  $j$ .

Dla każdej pary operacji, które używają maszyny  $k$  w TO  $j$  dla części  $i$  i w TO  $f$  dla części  $e$ :

$$TM_{riojk} - TM_{refgk} + TR_k Z_k + LR_{riojefgk} \geq \text{mat}_{riojk} \quad (8.28)$$

gdzie:  $X_{riojk} = X_{refgk} = 1$  i  $Y_{rio} = Y_{ref} = 1$

$$TM_{refgk} - TM_{riojk} + TR_k Z_k + L(1 - R_{riojefgk}) \geq \text{mat}_{refgk} \quad (8.29)$$

Równania (8.28) i (8.29) to spełnienie warunku, iż maszyna nie może być przydzielona do procesu więcej niż do jednej części w tym samym czasie i że dwie różne operacje nie mogą być wykonywane równolegle na tej samej maszynie:

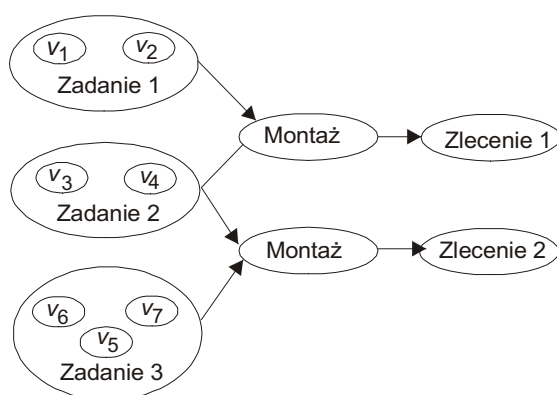
$$TM_{riojk} \geq \text{mat}_{riojk} \cdot \forall j \quad (8.30)$$

gdzie:  $X_{riojk} = 1$  i  $Y_{rio} = 1$ .

Równanie (8.30) zadowala warunek, że termin zakończenia operacji musi przynajmniej przekraczać jego czas procesu.



Tabela 8.6 prezentuje przykład alternatywnych maszyn dla operacji. Odpowiednio do informacji zawartej na rysunku 8.23 i w tabeli 8.6 mamy dwa zlecenia klientów, które mają różne terminy zakończenia. Maszyna 5 jest outsourcingowa i z drogą transportu o czasie 3 ( $TR_5 = 3$ ). Jeżeli terminy zakończenia są  $D_1 \leq 20$ ,  $D_2 \leq 23$  i  $AT_1 = AT_2 = DT_1 = DT_2 = 5$ , to całkowite czasy zakończenia (*makespans*) dla zleceń wynoszą  $MS_1 \leq 10$  i  $MS_2 \leq 13$ . Rozwiązanie optymalne tego prostego przykładu jest pokazane na rysunku 8.24. Zlecenie 2 satysfakcjonuje warunek  $D_2 \leq 23$  przez outsourcing drugiej operacji ( $v_4$ ) dla części 3 (P3). Jeżeli część 3 w zleceniu 2 jest nie do przesunięcia na outsourcingową maszynę, to termin zakończenia dla zlecenia 2 nie będzie dotrzymany.



Rys. 8.23. Prosty przykład elastycznego procesu produkcyjnego [116]

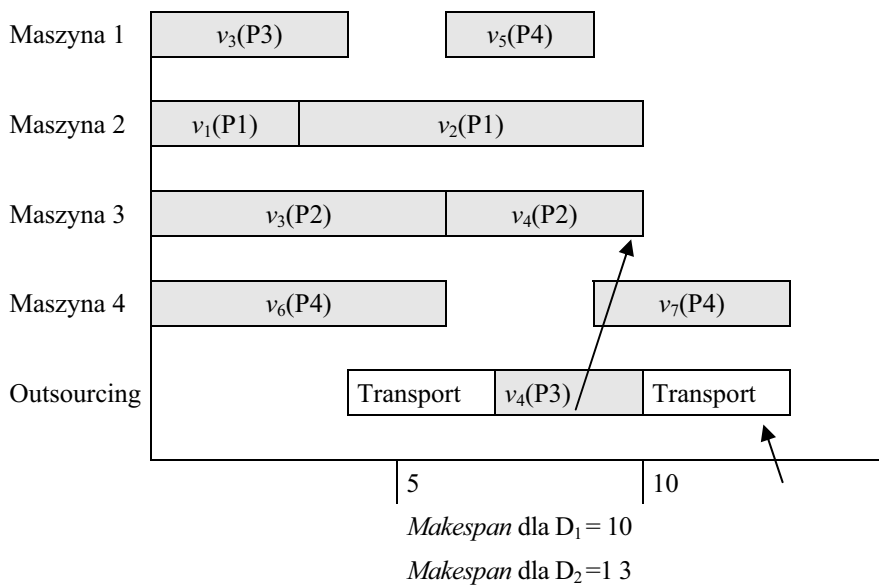
Tabela 8.6. Cykl produkcyjny i alternatywne maszyny dla operacji

Nr części	Typ zadania dla części (nr operacji)	Operacja dla alternatywnej maszyny (nr operacji, nr maszyny, czas wykonania operacji na maszynie)	Alternatywne sekwencje operacji dla zadania przez TS
1	Zadanie 1 ( $v_1, v_2$ )	$(v_1, M_1, 5)$ $(v_1, M_2, 3)$ $(v_2, M_2, 7)$	$v_1 - v_2$
2-3	Zadanie 2 ( $v_3, v_4$ )	$(v_3, M_3, 6)$ $(v_3, M_1, 4)$ $(v_4, M_3, 4)$ $(v_4, M_5, 3)$	$v_3 - v_4$
4	Zadanie 3 ( $v_5, v_6, v_7$ )	$(v_5, M_1, 3)$ $(v_6, M_2, 4)$ $(v_6, M_4, 6)$ $(v_7, M_3, 7)$ $(v_7, M_4, 4)$	$v_5 - v_6 - v_7$ $v_6 - v_5 - v_7$ $v_6 - v_7 - v_5$

(a) określony wybór maszyny i najlepsza sekwencja operacji

	Sekwencja operacji i wybrany nr maszyny	
Część 1	1	2
Nr maszyny	$M_2$	$M_2$
Część 2	3	4
Nr maszyny	$M_3$	$M_3$
Część 3	3	4
Nr maszyny	$M_1$	$M_5$
Część 4	6	5
Nr maszyny	$M_4$	$M_1$

(b) harmonogram Gantta (nr operacji, nr części)



Rys. 8.24. Rezultaty harmonogramowania [116]

W tym opracowaniu proponuje się efektywne podejście bazujące na algorytmie genetycznym, które zawiera algorytm topologicznego sortowania dla rozwiązania modelu APS z ograniczeniami kolejnościowymi. Niech  $P(t)$  i  $C(t)$  będą rodzicem i potomkiem w aktualnej generacji  $t$ . Ogólna procedura proponowana w podejściu bazującym na algorytmie genetycznym jest opisana poniżej.

Procedura: podejście bazujące na GA.

**Begin:**

$T \leftarrow 0$ ;

początkowa populacja rodziców  $P(t)$ ;

ocena  $P(t)$ ;

**while** (warunek zakończenia = nie) **do**

rekombinacja  $P(t)$  do  $C(t)$  przez krzyżowanie i mutację;

ocena  $C(t)$ ;

wybór  $P(t+1)$  z  $P(t)$  i  $C(t)$ ;

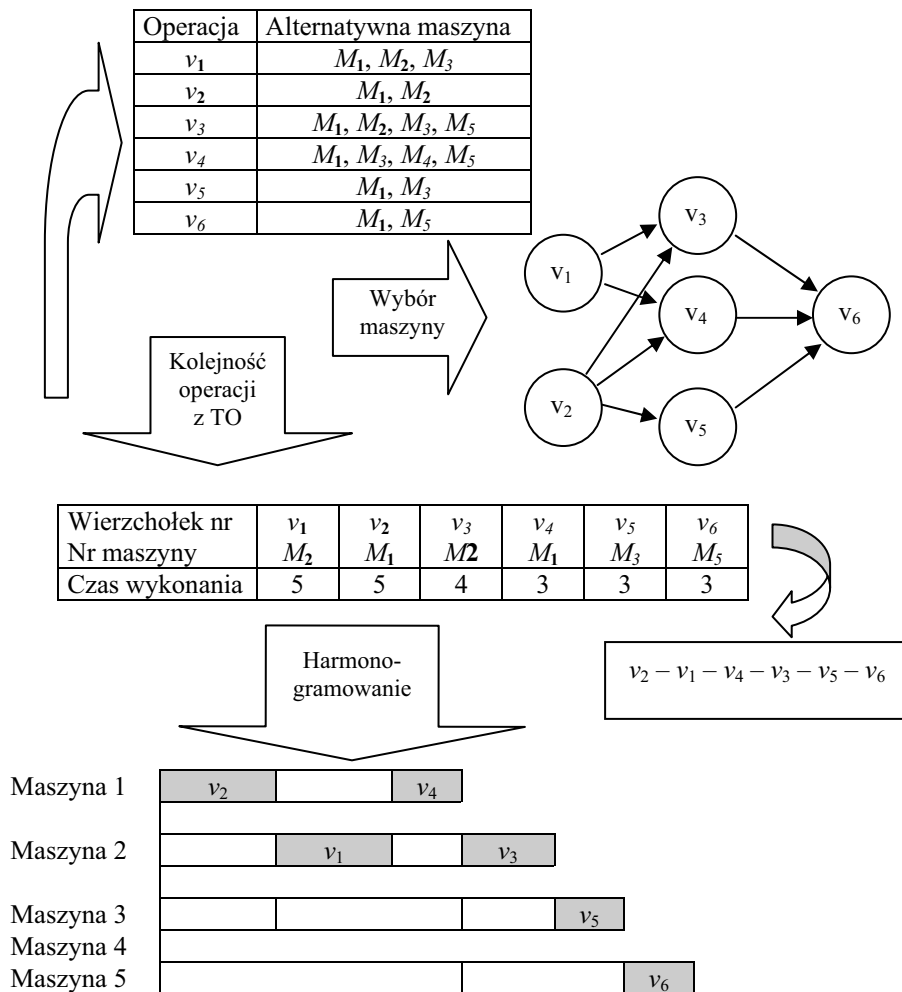
$t \leftarrow t + 1$ ;

**End;**

**End;**

Głównymi czynnikami rozwoju GA są: reprezentacja chromosomu, początkowa populacja, miary ewolucji, krzyżowanie, mutacja i strategia selekcji. W dodatku parametry genetyczne, takie jak wielkość populacji (`pop_size`), liczba generacji (`max_gen`), prawdopodobieństwo krzyżowania ( $P_c$ ) i prawdopodobieństwo mutacji ( $P_m$ ) są określane przed wykonaniem GA.

W problemie szeregowania wykorzystującym algorytm genetyczny, krytyczną kwestią jest rozwój schematu reprezentacji do reprezentacji dopuszczalnych rozwiązań. Bardzo trudno jest reprezentować ścieżkę z ograniczeniami kolejnościowymi w grafie. W generowaniu porządku TO, schemat reprezentacji musi być zdolny do generowania wszystkich możliwości TO dla danych sieci AOV. W dodatku, trasa rozwiązania zawsze odpowiada TO. Przypuśćmy, że jest jedno zadanie, które zawiera sześć operacji,  $v_1$  do  $v_6$ . Struktura chromosomu może być reprezentowana jak pokazano na rysunku 8.25. Na tym rysunku pierwszy rząd chromosomu wskazuje operacje, które dopasowują prawdopodobny wybór numeru maszyny do każdej operacji. Każda operacja losowo wybiera numer maszyny z możliwych alternatywnych maszyn. Drugi rząd wskazuje priorytety dla kandydatów wyboru, kiedy operacje nie napływają z istniejących krawędzi. Wartość genu jest generowana wyłącznie z prawdopodobieństwem  $[1, J]$ , gdzie  $J$  jest całkowitą liczbą operacji. Na przykład wtedy, kiedy  $v_1$  i  $v_2$  zostały wybrane z kolejności operacji, ponieważ ich liczbowy priorytet jest równy 2. Po wyborze  $v_2$  można usunąć  $v_2$  i krawędź wyprowadzoną z  $v_2$ . Teraz  $v_1$  i  $v_5$  nie mają poprzednika i  $v_1$  jest wybrana dla następnej operacji, ponieważ  $v_1$  ma wyższy priorytet niż  $v_5$ . W ten sam sposób kolejność operacji  $v_2 - v_1 - v_4 - v_3 - v_5 - v_6$  jest określona dla zadania i maszyn  $M_1 - M_2 - M_1 - M_2 - M_3 - M_5$  używanych dla każdej operacji sekwencyjnie. Przypuśćmy, że alternatywna maszyna i czas wykonania dla każdej operacji są pokazane w tabeli 8.7.



Rys. 8.25. Kodowanie problemu [116]

Tabela 8.7. Alternatywne maszyny i czas procesu dla operacji  $v_j$

Nr maszyny	Operacje					
	$v_1$	$v_2$	$v_3$	$v_4$	$v_5$	$v_6$
	Czas wykonania					
1	5	5	6	3	2	2
2	5	5	4	$\infty$	$\infty$	$\infty$
3	6	$\infty$	6	5	3	$\infty$
4	$\infty$	$\infty$	$\infty$	3	$\infty$	$\infty$
5	$\infty$	$\infty$	6	4	$\infty$	3

Tabela 8.8 może być prezentowana jako wykres Gantta, który pokazano na rysunku 8.25 z **makespan** wynoszącym 23. Kiedy chromosomy są modyfikowane

przez operacje genetyczne, takie jak krzyżowanie lub mutacja, **makespan** może być skrócony. W przypadku, gdy **makespan** jest dostatecznie zminimalizowany, chromosom pokazuje najlepszą wybraną maszynę i sekwencję operacji.

Tabela 8.8. Początkowy i końcowy termin każdej operacji (początkowy, końcowy)

Nr maszyny	Operacje					
	$v_1$	$v_2$	$v_3$	$v_4$	$v_5$	$v_6$
1		0,5		10,13		
2	5,10		13,17			
3					17,30	
4						
5						20,23

Procedura generowania rozwiązań dopuszczalnych jest pokazana poniżej.

**Procedure:** Reprezentacja dopuszczalnego rozwiązania.

**Input data:**

Liczba części i alternatywnych maszyn;

Liczba operacji dla każdej części;

Niech  $v_j(j = 1, \dots, J)$  będzie  $J$  operacją dla wszystkich typów części uporządkowanych w pewien sposób;

Zbiór skierowanych grafów z ograniczeniami kolejnościowymi;

Zbiór alternatywnych maszyn i czas dla każdej operacji włączając outsourcing;

Czas transportu dla outsourcingu;

**Begin:**

**Generowanie** chromosomu z wyborem maszyny:

Alokacja losowo wybranej maszyny ze zbioru alternatywnych maszyn dla każdej operacji do  $v_j$ ;

Generacja losowej liczby priorytetu (1 do  $J$ ) dla każdej operacji  $v_j$ ;

**Generowanie** sekwencji operacji:

**While** pozostałe wierzchołki **do**

**If** każdy wierzchołek ma poprzednika **then**

Sieć jest niedopuszczalna: **stop**;

**Else** wybierz wierzchołek  $v_j$ , który nie ma poprzednika;

**Output**  $v_j$ ;

**Delete**  $v_j$  i wszystkie krawędzie wyprowadzone z  $v_j$  ze skierowanego grafu;

**End**;

**Generowanie** dopuszczalnego harmonogramu:

Utwórz harmonogram z wybranych maszyn zawierających sekwencje operacji dla każdej części;

**End**;

W populacji początkowej chromosomu każdy chromosom zawiera liczbę priorytet i prawdopodobieństwo wyboru maszyny ze zbioru alternatywnych maszyn dla każdej operacji. Proces inicjalizacji jest wykonywany według procedury pokazanej poniżej i generuje populację chromosomu.

Chromosom zawiera informacje, które przydzielają operacje do każdej maszyny w sposób ograniczony przez określoną sekwencję operacji. Ocena dopasowania pokazana jest poniżej.

**Procedure:** Ocena dopasowania.

**Begin:**

$MP_{rk} = 0$  dla wszystkich  $k$ ;

$S_{ijk} = 0$  dla wszystkich  $i, j, k$ ;

Określenie sumy czasu procesów z użyciem maszyny  $k$ ;

**For**  $r = 1$  do  $R$

**If**  $r > 1$  then  $MP_{rk} = MP_{(r-1)k}$

**Else;**

**For**  $i = 1$  do  $I_r$  do

**For**  $j = 1$  do  $J$  do

$k$  wybór numeru maszyny dla operacji  $j$ ;

**If**  $k$  jest maszyną outsourcingową then;

**If** operacja  $j$  jest pierwszą operacją then;

$S_{rijk} = MP_{rk}$

**Else if** operacja  $j$  nie użyła maszyny  $k$  then;

$S_{rijk} = TR_k + MP_{rk}$

**Else**  $S_{rijk} = MP_{rk}$

$E_{rijk} = S_{rijk} + \mathbf{mat}_{rijk}$

**If** operacja  $j + 1$  nie użyje maszyny  $k$  then;

$MP_{rk} = E_{rijk} + TR_k$ ;

**Else**  $MP_{rk} = E_{rijk}$

**Else;**

$S_{rijk} = MP_{rks}$

$E_{rijk} = S_{rijk} + \mathbf{mat}_{rijk}$

$MP_{rk} = E_{rijk}$

**End;**

**End;**

**End;**

**For**  $r = 1$  do  $R$  do

$MS_r \square \max_{k=1 \dots K} (MP_{rk})$ ;

**End;**

**Return**  $MS_r$ ;

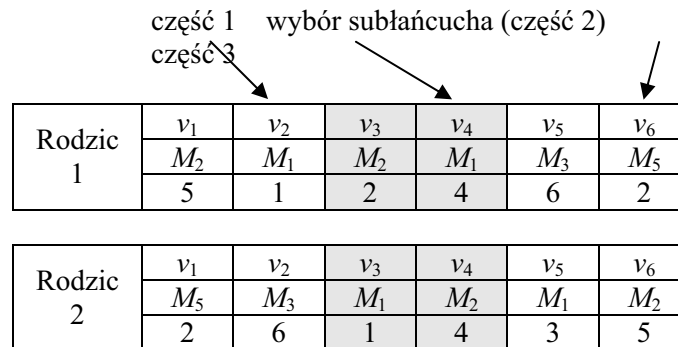
**End;**

Funkcja dopasowania dla modelu APS bazuje tylko na jednej wartości funkcji celu minimalizującej **makespan** ( $MS_r$ ) dla każdego zlecenia  $r$ .  $MS_r$  musi satysfakcjonować  $MS_r \leq D_r - AT_r - DT_r$  dla wszystkich zleceń  $r$ .

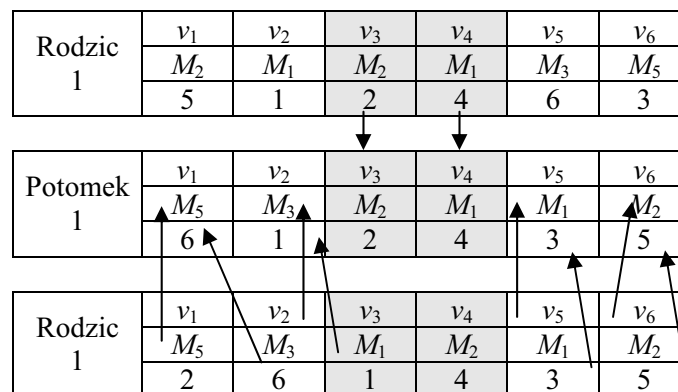
Po ocenie dopasowania wszystkich wygenerowanych chromosomów, strategia selekcji wybiera chromosomy z populacji. W ten sposób tworzy się nową populację dla następnych generacji bazując na rodzicach i potomstwie. W tym przypadku do selekcji jest wykorzystywana opisana wcześniej metoda koła ruletki.

Do tworzenia następnej generacji są używane także takie operatory genetyczne, jak: krzyżowanie i mutacja. W szczególności krzyżowanie ma największe znaczenie w produkcji nowego zbioru chromosomów potomnych. W niniejszym podejściu wykorzystuje się krzyżowanie **order-based**, które jest przedstawione na rysunku 8.26.

(1) Losowy wybór subłańcucha z rodzica 1



(2) Produkcja modyfikowanego rodzica 1



Rys. 8.26. Ilustracja porządku bazującego na krzyżowaniu [116]

Przypuśćmy, że dwa chromosomy są rodzicem 1 = [5 1 2 4 6 3] z numerami maszyn [2 1 2 1 3 5] i rodzicem 2 = [2 6 1 4 3 5] z numerami maszyn [5 3 1

2 1 2], w których parzyste kolumny z lewej reprezentują odpowiednio część 1, część 2 i część 3. Używając tych samych procedur możemy produkować potomstwo [6 1 2 4 3 5] z numerami maszyn [5 3 2 1 1 2]. Mutacja polega na losowym wyborze dwóch genów (tylko liczb priorytetu) z chromosomu i zamianie ich miejscami.

W niniejszym podejściu zaproponowano model APS integrujący planowanie, harmonogramowanie i outsourcing referowany przez autorów do zastosowania w łańcuchach dostaw wytwarzania. Model ten określa najlepszy harmonogram z użyciem alternatywnych sekwencji operacji i maszyn, obejmujących outsourcing i satysfakcjonujący terminy zakończenia zleceń klientów.

Proponowany powyżej model uwzględniający outsourcing bazuje na rozwiniętym podejściu z zastosowaniem GA, w którym proponuje się elastyczną metodę. Rezultaty eksperymentów pokazują, że w oparciu o model APS można generować umiarkowanie dobre rozwiązania. To podejście daje możliwość otrzymywania najbardziej efektywnych rozwiązań z pewnej przestrzeni rozwiązań dopuszczalnych, chociaż rozwiązania mogą nie być optymalne [116].



## **9. Proponowane podejście do zarządzania produkcją w ogniwie sieci dostaw**

### **9.1. Przesłanki poszukiwania nowych metod zarządzania produkcją**

Jak wykazano we wcześniejszych rozdziałach niniejszej rozprawy (tj. 2 i 4), zarządzanie produkcją w ogniwie sieci dostaw jest bardzo ważnym elementem zarządzania siecią dostaw i jednocześnie bardzo trudnym problemem optymalizacyjnym. Do niedawna przedsiębiorstwa budujące, instalujące i eksploatujące systemy informatyczne koncentrowały uwagę na procesach zachodzących głównie w ich ramach, tj. procesach wewnętrznych. Jednakże postępujące otwieranie się przedsiębiorstw na współpracę i kontakty w sieci dostaw zmusza je do stosowania technologii informacyjnych narzucanych przez współczesne tendencje w rozwoju sieci dostaw. Optymalizacja łańcucha dostaw polega więc, jak przedstawiono w rozdziale drugim na optymalizacji zarówno funkcjonowania poszczególnych jego ogniw, jak i współdziałaniu pomiędzy nimi.

Próby praktycznego wykorzystania globalnych modeli optymalizacyjnych, w wielu przypadkach opierają się na odejściu od realistycznego odwzorowania niektórych relacji na rzecz zastosowań w systemach przepływowych. Z tego powodu w modelowaniu sieci dostaw przyjmuje się wiele założeń upraszczających, ograniczając przy tym rozważania do systemów o ustabilizowanej produkcji, o czym wspomniano w podrozdziale 8.2. Jednakże rosnące wymagania klientów – oczekujących stale lepszych produktów – powodują, że przedsiębiorstwa muszą się do nich dostosowywać i w rezultacie szybko zmieniać asortyment produkcji. Powoduje to zmianę struktur przedsiębiorstw na bardziej elastyczne, których cechą jest na ogół występowanie nieustabilizowanej produkcji.

Kolejne uproszczenia dotyczą przyjmowania założenia, że w systemach produkcyjnych realizowane są zadania nie podzielone. Nie odpowiada to aktualnym tendencjom rozwoju sieci dostaw, w których występuje wydzielanie procesu produkcyjnego realizowanego w jednym z ogniw sieci dostaw i wykonanie tej części procesu w innym ogniwie sieci dostaw na jego wewnętrznych zasobach (tj. outsourcing).

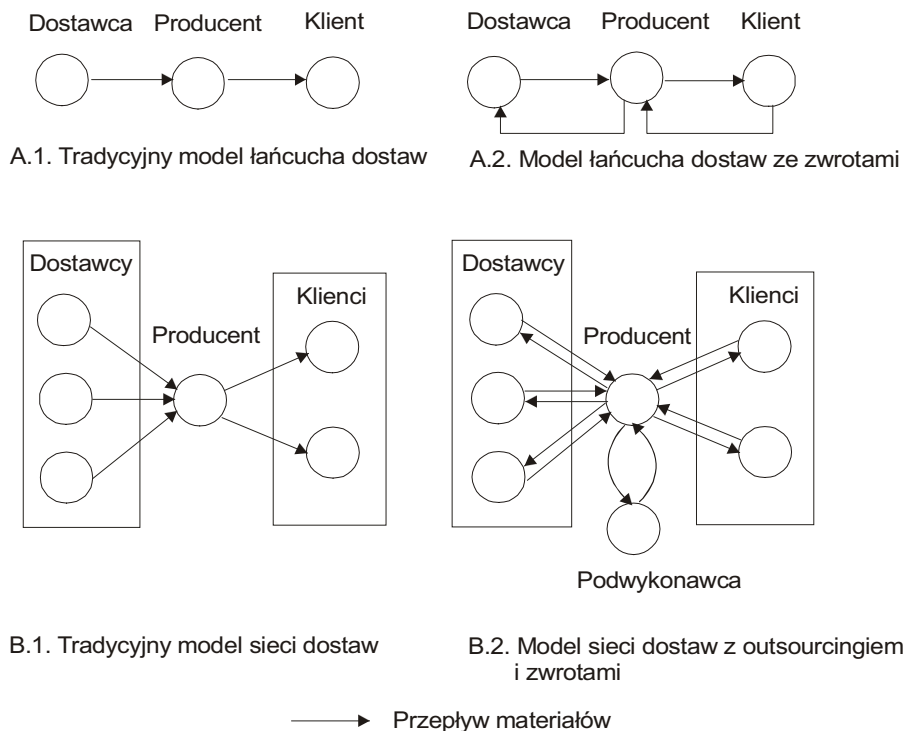
Ze względu na wagę problemu, na rysunku 9.1 zilustrowano nowe podejście do zarządzania łańcuchem dostaw. Przedstawione na tym rysunku modele A.1 i B.1 ilustrują tradycyjne podejście do łańcucha dostaw. Model A.2 (rys. 9.1) przedstawia model łańcucha dostaw, w którym pokazano, że w rzeczywistości przepływ materiałów może także dotyczyć zwrotów, jak np.: zwrot wadliwego materiału w ramach reklamacji, zwrot opakowań, czy też uzgodniony zwrot nadwyżek towaru. Model B.2 (rys. 9.1) prezentuje nowe zaproponowane przez autorkę podejście do problemu. W modelu tym pokazano możliwości wykonywania wydzielonej części procesu produkcyjnego producenta przez

jego podwykonawcę (na jego zasobach wewnętrznych) – outsourcing. Można przy tym wyróżnić następujące rodzaje wydziałania procesów:

- technologiczne,
- wyrównujące obciążenia zdolności produkcyjnych w sieci dostaw.

Outsourcing technologiczny może wystąpić z powodu specyfiki samego procesu technologicznego i wymaganych w związku z tym specjalistycznych maszynach i urządzeniach lub na skutek braku wiedzy o samym procesie. Natomiast wyrównywanie obciążeń zdolności produkcyjnych w sieci jest nową tendencją rozwojową w optymalizacji sieci dostaw, wymagającą stosowania nowych metod.

Przy modelowaniu sieci dostaw należy zwrócić uwagę na to, że z matematycznego punktu widzenia model B.2 (rys. 9.1) przedstawia *graf skierowany* (inaczej *digraf*), który zawiera pętle. Używany powszechnie, a więc i w tej pracy termin *sieć dostaw* nie jest więc tożsamy z pojęciem *sieci* rozumianym w matematyce jako skierowany, skończony digraf bez pętli i łuków równoległych (por. [23]). W związku z tym aktualnie stosowane metody zarządzania sieciami nie nadają się do zastosowania w przypadkach zilustrowanych modelem B.2 (rys. 9.1).



Rys. 9.1. Modele sieci dostaw [opracowanie własne]

Innym ważnym powodem poszukiwania nowych rozwiązań problemu optymalizacji zarządzania ogniwem dostaw jest specyfika zmian zachodzących w ogniwie. Znaczna część opracowanych metod harmonogramowania prac w ogniwie dotyczy problemów teoretycznych i nie uwzględnia odchyłeń spowodowanych zakłóceniami występującymi w rzeczywistych warunkach przemysłowych. Występowanie nieprzewidywalnych zakłóceń w systemach wytwarzania wymaga podejmowania decyzji w krótkim czasie, co jak wykazano w rozdziale 3 i 4 wyklucza możliwość stosowania dokładnych metod analitycznych.

Analiza literatury przedmiotu (rozdział 4) wykazała, że problem zarządzania produkcją jest problemem NP-trudnym i nawet dość proste przypadki systemów produkcyjnych z małą liczbą zadań i maszyn są trudne do rozwiązania z zastosowaniem metod dokładnych. Jak pokazują dotychczasowe doświadczenia, o czym wspomniano w rozdziale 4, czas i koszt optymalnych obliczeń są przy tym tak duże, że metody dokładne nie znalazły praktycznych zastosowań. Z tego powodu lepszym podejściem do rozwiązania problemów zarządzania produkcją okazało się poszukiwanie suboptymalnych rozwiązań z zastosowaniem technik sztucznej inteligencji, co uzasadniono w rozdziale 8.

Rozważania przeprowadzone w niniejszej rozprawie dotyczące systemów eksperckich (rozdział 6) skłaniają do zastosowania tych systemów na wyższych poziomach zarządzania produkcją, tj. tworzenia planów produkcji, ponieważ na tym poziomie wiele decyzji uzależnionych jest od wiedzy eksperta. System ekspercki odtwarza sposób postępowania dyspozytora systemu, z uwzględnieniem dużej ilości (branych pod uwagę) różnorodnych warunków.

Na poziomie harmonogramowania, jak wykazano w rozdziale 8 rozprawy, badania w zakresie zastosowania algorytmów genetycznych do zarządzania produkcją są zdominowane problematyką szeregowania zadań (innymi słowy harmonogramowania wewnątrzkomórkowego). Na ogół w problemach szeregowania zakłada się istnienie zbioru zasobów i ustalonego systemu zadań, które mają być wykonane z wykorzystaniem tych zasobów. Korzystając z przyjętych właściwości systemu oraz nałożonych ograniczeń należy skonstruować efektywny algorytm znajdowania takiego uszeregowania zadań, które pozwala zoptymalizować lub zmierza do optymalizacji przyjętego kryterium. W proponowanej tu metodyce, przy implementowaniu algorytmu genetycznego wykorzystano podstawowe kryterium oceny bazujące na czasie wykonania. Jak wspomniano w podrozdziale 4.2 kryterium to jest przyjmowane najczęściej, ponieważ skracanie cykli produkcyjnych przynosi największe korzyści przedsiębiorstwom w postaci obniżenia kosztów i satysfakcji klienta z powodu skrócenia czasu realizacji zamówienia. Ogólny problem harmonogramowania polegał więc na tym, aby przy uwzględnieniu ograniczeń minimalizować długość uszeregowania, tj. czas od rozpoczęcia do całkowitego zakończenia wszystkich zadań realizowanych w komórce produkcyjnej.

## 9.2. Cele i zakres zastosowań proponowanej metody zarządzania produkcją w ogniwie sieci dostaw

Celem nadrzędnym tworzenia nowych metod i procedur zarządzania produkcją w ogniwie sieci dostaw było stworzenie spójnych metodycznych podstaw dla zintegrowanego podejścia do planowania i sterowania produkcją w dyskretnym systemie produkcyjnym przedsiębiorstwa przemysłowego.

Wynikającym z takiego podejścia celem cząstkowym było stworzenie kombinacji systemu eksperckiego i algorytmu genetycznego dla wyrównywania wykorzystania zdolności produkcyjnych i minimalizacji całkowitego cyklu produkcyjnego w systemach obróbki skrawaniem. Przyjęto przy tym założenia, że w nowym podejściu należy uwzględnić możliwości dzielenia zadań i wykonywania wybranych operacji tych zadań na zasobach w innych ogniwach sieci dostaw. Ponadto przyjęto założenie, że nowy sposób planowania produkcji powinien umożliwiać stosowanie alternatywnych procesów technologicznych, szczególnie w przypadkach „wąskich gardeł”, czy też awarii.

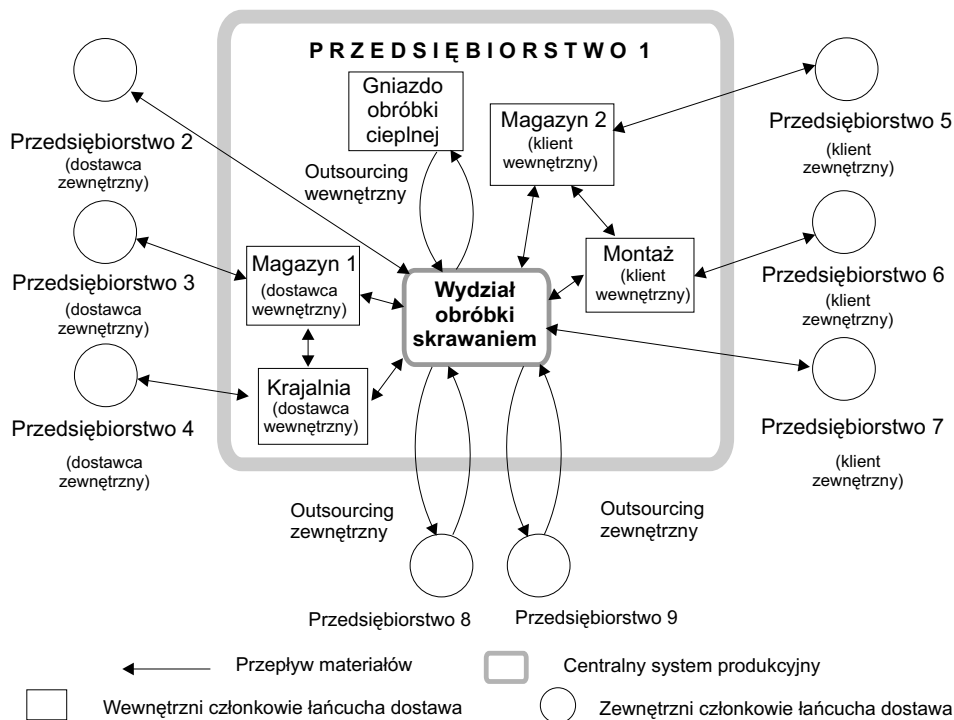
Podstawowe kryteria tworzenia planu produkcji dla systemu produkcyjnego to z jednej strony dotrzymanie terminu realizacji zamówienia określonego przez klienta, a z drugiej strony wyrównywanie obciążeń zasobów wewnętrznych. Jako kryterium optymalizacji harmonogramu przyjęto długość uszeregowania. Ogólny problem harmonogramowania polegał więc na tym, aby przy uwzględnieniu ograniczeń minimalizować okres wykonania (*makespan*), tj. czas od rozpoczęcia do całkowitego zakończenia wszystkich realizowanych w systemie produkcyjnym zadań.

Modelowane w tej pracy systemy produkcyjne różnią się od często analizowanych w literaturze problemów testowych, w których przyjmuje się pewne założenia upraszczające, np. takie, że: jest jedna maszyna każdego typu, liczba operacji w każdym zadaniu jest taka sama, liczba operacji równa się liczbie maszyn [8, 37]. Prezentowany tu system ekspercki i algorytm genetyczny mogą być zastosowane w systemach o szybkozmiennym asortymencie produkcji, także w przypadkach, w których występują alternatywne marszruty technologiczne i wykonywanie wydzielonych operacji z procesów (dla danej części wyrobu) w innych ogniwach sieci dostaw. Nowe metody powinny więc uwzględniać rezerwy zdolności produkcyjnych w sieci dostaw i umożliwiać szybkie reagowanie na zakłócenia występujące w tej sieci.

Przy opracowywaniu nowych metod ważnym kryterium ich tworzenia był koszt. Jak pokazuje praktyka, przedsiębiorstwa nadal nie posiadają takich środków finansowych, które pozwoliłyby na zakup drogich komputerów dużej mocy i drogiego komercyjnego oprogramowania. Mimo znacznej obniżki cen komputerów koszty nadal stanowią poważną barierę rozwoju systemów informatycznych. Szczególnie małe i średnie przedsiębiorstwa oczekują na tańsze oprogramowanie możliwe do użytkowania na popularnych komputerach.

### 9.3. Prezentacja nowej metody zarządzania produkcją w ogniwie sieci dostaw

W proponowanym tu podejściu do planowania produkcji zaimplementowano system ekspercki, natomiast szeregowanie zadań zrealizowano z zastosowaniem algorytmu genetycznego. W celu oceny efektywności algorytmu genetycznego do szeregowania zadań zastosowano także reguły priorytetu. Prezentowana w niniejszej pracy kombinacja systemu eksperckiego i algorytmu genetycznego była testowana na danych z rzeczywistych przedsiębiorstw opisanych poniżej (podrozdział 11.1). W nowej, prezentowanej tu metodzie na poziomie tworzenia planu produkcji, w sytuacjach kiedy jest to konieczne, może być uwzględnione dzielenie zadań. Oznacza to, że wybrane z procesów produkcyjnych operacje, dla których występują „wąskie gardła” mogą być zaplanowane do wykonania na maszynach spoza rozpatrywanego systemu wytwórczego. Planowanie operacji do wykonania poza rozpatrywanym systemem wytwórczym może również występować ze względu na specyfikę samego procesu technologicznego na przykład wtedy, kiedy w procesie technologicznym obok operacji obróbki skrawaniem występują operacje obróbki cieplnej realizowane w innym systemie. Rozważany w niniejszym podejściu przepływ materiałów przez obróbkowy system wytwórczy zilustrowano na rysunku 9.2.



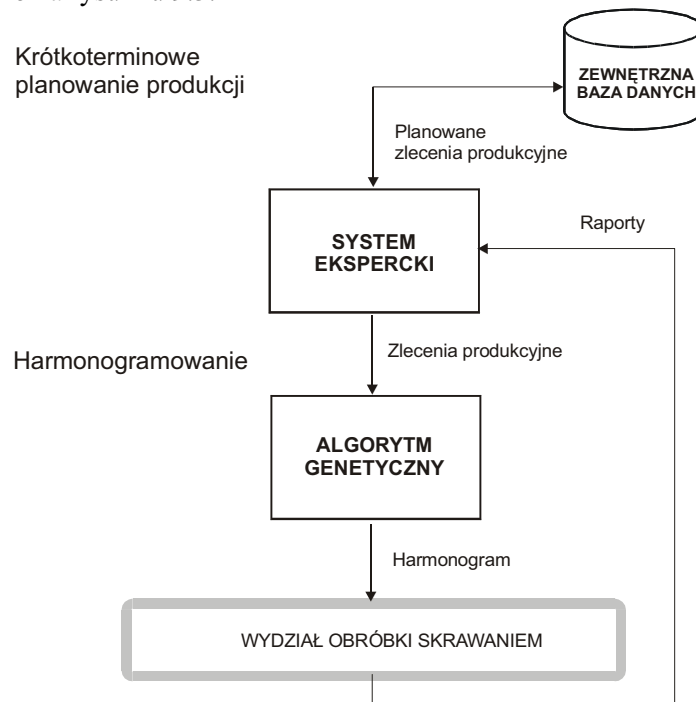
Rys. 9.2. Przepływ materiałów przez obróbkowy system wytwórczy z outsourcingiem maszyn [opracowanie własne]

Jak pokazano na rysunku dostawcami wewnętrznymi systemu obróbkowego są: magazyn 1 i krawalnia. Dostawcy zewnętrzni to przedsiębiorstwa 2, 3 i 4. Klientami wewnętrznymi, od których mogą napływać zamówienia są: magazyn 2 i montaż. Klienci zewnętrzni to przedsiębiorstwa 5, 6 i 7. W zaprezentowanym na rysunku 9.2 przykładzie przepływu materiałów występuje outsourcing zarówno wewnętrzny (gniazdo obróbki cieplnej), jak i zewnętrzny (przedsiębiorstwo 8 i 9).

Należy uściślić niektóre stosowane tu terminy. Przez *zadanie* należy rozumieć proces technologiczny wyrobu (inaczej części lub detalu), który jest wykonywany w gnieździe. Zazwyczaj harmonogramuje się pracę gniazda w celu wykonania kilku zadań. Aby wykonać określone zadanie, należy na przedmiocie zrealizować pewną liczbę *operacji* procesu technologicznego. Operacje te, dla danego przedmiotu, mają ściśle określoną kolejność wykonywania. Dana operacja zajmuje w określonym czasie tylko jedną maszynę (przyjęto, że żadna z operacji nie musi być wykonywana przez kilka maszyn jednocześnie).

W modelach systemów produkcyjnych bardzo często przyjmuje się ograniczenie niepodzielności zadań [186]. W tej rozprawie będą rozważone problemy dyskretne z zadaniami dzielonymi w fazie planowania produkcji i niepodzielnymi w fazie harmonogramowania.

Ideę implementacji kombinacji z zastosowaniem systemu eksperckiego i algorytmu genetycznego do planowania i harmonogramowania produkcji zaprezentowano na rysunku 9.3.



Rys. 9.3. Idea kombinacji systemu ekspertowego i algorytmu genetycznego [opracowanie własne]

Prezentowany tu system ekspercki był koncepcyjnie rozwijany w oparciu o przegląd literatury (np. [5, 10, 28, 29, 34, 46, 51, 78, 117, 118, 120, 150, 151, 167, 181, 183]) oraz wywiady z planistami, które przeprowadzono w przedsiębiorstwach przemysłowych.

Dzisiejsze przedsiębiorstwa funkcjonują w rozproszonej strukturze organizacyjnej, mimo że ich produkcja jest wynikiem ścisłej współpracy z wieloma kooperantami. Dlatego przy tworzeniu koncepcji systemu eksperckiego uwzględniono możliwości outsourcingu maszyn.

Hierarchiczne systemy zarządzania produkcją mają na ogół strukturę trójpoziomową, w której można wyróżnić poziomy: strategiczny, taktyczny i operacyjny [56]. Prezentowany na rysunku 9.3 schemat zarządzania dotyczy zarządzania na poziomie operacyjnym.

W dużych przedsiębiorstwach przemysłowych plany zadań dla obróbkowych systemów wytwórczych powstają na ogół na wyższych szczeblach planowania i to na ogół dla okresu miesięcznego. W prezentowanym tu podejściu, podobnie jak w rzeczywistych przedsiębiorstwach, tworzenie zleceń produkcyjnych może być realizowane zarówno w oparciu o plany zadań z wyższych szczebli zarządzania, jak i bezpośrednio – w oparciu o zlecenia klienta.

## 9.4. System ekspercki planowania i sterowania produkcją

### 9.4.1. Struktura systemu eksperckiego

Prezentowany tu system ekspercki umożliwia tworzenie planu produkcji dla komórek produkcyjnych obróbki skrawaniem w oparciu o badanie tzw. „wąskich gardeł”. Głównym celem tworzenia systemu eksperckiego było wyrównywanie wykorzystania zdolności produkcyjnych w obróbkowych systemach produkcyjnych o szybkozmiennym asortymencie produkcji.

Zbudowany system ekspercki planowania produkcji może być wykorzystywany do budowania planu produkcji dla różnych okresów, tj. tygodnia, dnia lub zmiany roboczej, w zależności od długości cykli produkcyjnych.

Plan produkcji jest zbiorem wybranych zleceń produkcyjnych, które powinny zostać wykonane w określonym horyzoncie planistycznym. Zlecenia produkcyjne są tworzone w zewnętrznej bazie danych.

Zlecenie produkcyjne zawiera zbiór operacji do wykonania określonego wyrobu (proces technologiczny lub jego człon). Przyjmuje się, że procesy technologiczne (inaczej zwane *planami procesu*), które mają być umieszczone w planie produkcji komórki produkcyjnej zostały wcześniej opracowane. Procesy technologiczne przechowywane są w zewnętrznej bazie danych.

Zleceniu produkcyjnemu dla celów planistycznych nadaje się unikalny wskaźnik priorytetu. Standardowo wszystkie operacje dla danego wyrobu posiadają ten sam wskaźnik priorytetu unikalny dla zlecenia produkcyjnego tego wyrobu. Wskaźnik priorytetu jest liczbą naturalną. Najniższy wskaźnik priorytetu nadawany jest najpilniejszemu zadaniu (wyrobowi). Wskaźnik priorytetu

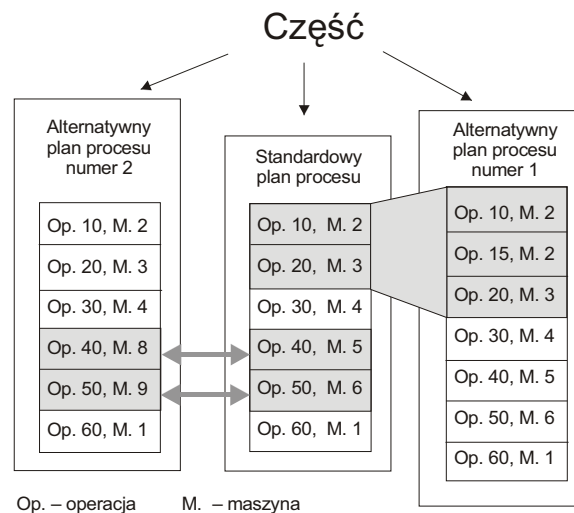
dla wyrobu określa planista odpowiednio do zamówień klienta. W przypadkach outsourcingu zlecenie jest wystawiane tylko na operacje należące do pojedynczego członu procesu technologicznego wyrobu i każdy człon ma inny wskaźnik priorytetu.

Przy tworzeniu planu produkcji analizuje się raporty. Są one tworzone zarówno dla wykonanych operacji, jak i opóźnionych zadań, tj. zadań, które były już umieszczone na harmonogramie i nie zostały wykonane z powodu zakłóceń w procesie wytwarzania. Przy tworzeniu planu produkcji na następny okres operacje z raportów traktowane są jako najpilniejsze.

#### 9.4.2. Zewnętrzna baza danych systemu eksperckiego

Proponowany system ekspercki planowania produkcji wykorzystuje plany procesów technologicznych przechowywane w jego zewnętrznej bazie danych. Założono, że możliwości zastosowania planów procesów alternatywnych i outsourcingu maszyn uwzględnia się na etapie opracowywania planów procesu technologicznego. Na rysunku 9.4 zaprezentowano przyjętą tu koncepcję stosowania procesów alternatywnych.

Przy obciążaniu maszyn – w pierwszym kroku – planuje się zastosowanie standardowych planów procesów technologicznych. Alternatywne plany procesu technologicznego mogą być zastosowane wtedy, kiedy stanowisko ma niewystarczającą zdolność produkcyjną, np. z powodu awarii lub po prostu stanowisko to stanowi „wąskie gardło” systemu.

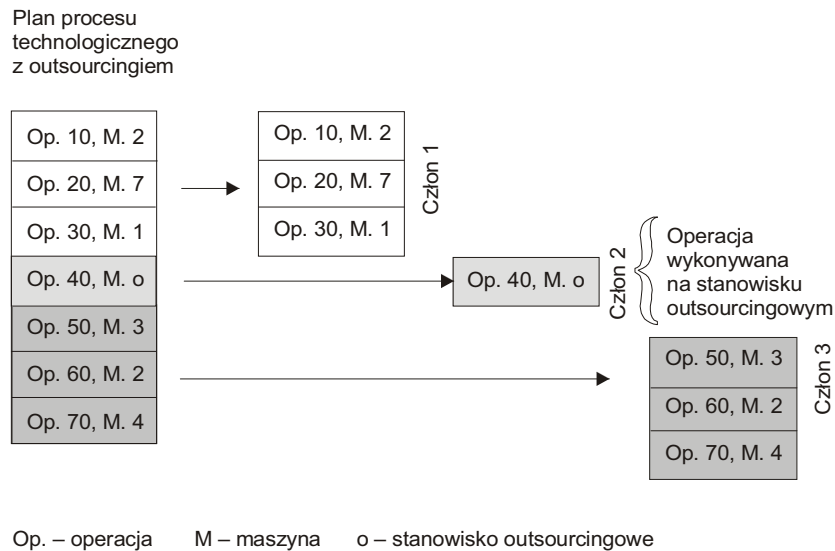


Rys. 9.4. Koncepcja stosowania procesów alternatywnych [opracowanie własne]

Z uwagi na to, że czasy transportu z/do miejsca ulokowania maszyny outsourcingowej mogą być bardzo długie zaproponowano dzielenie zadań. Polega ono na podziale zadania (tj. planu procesu technologicznego zawierającego operacje



outsourcingowe) na człony obejmujące operacje przed operacją outsourcingową i po operacji outsourcingowej. Dzielenie zadań (tj. planów procesów technologicznych lub inaczej marszrut) dla części obrabianej zilustrowano na rysunku 9.5.



Rys. 9.5. Dzielenie marszrut technologicznych [opracowanie własne]

W zewnętrznej bazie danych mogą więc być przechowywane cztery rodzaje marszrut (planów procesów technologicznych) dla jednego wyrobu:

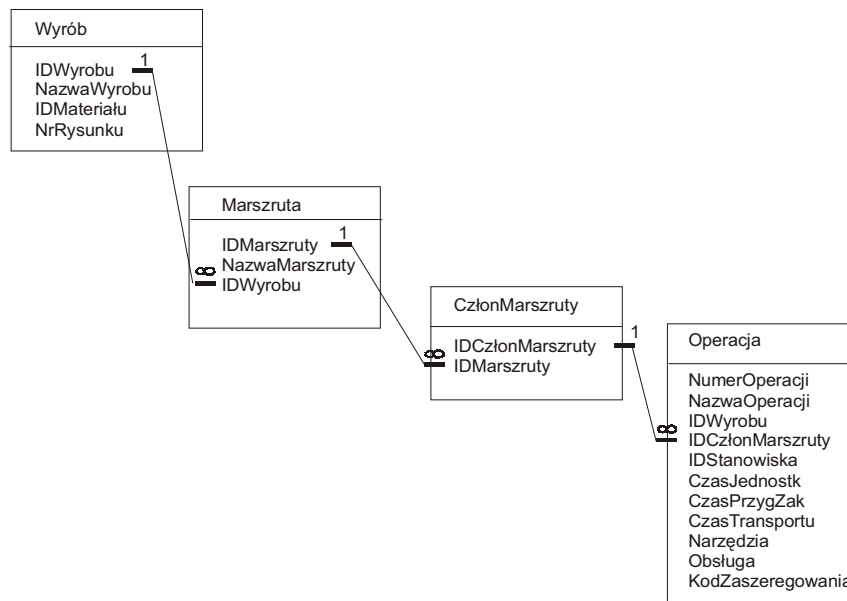
- marszruta standardowa,
- marszruta alternatywna 1,
- marszruta alternatywna 2,
- marszruta z outsourcingiem.

Na rysunku 9.6 pokazano powiązanie tabel opisujących marszruty w zaproponowanej tu zewnętrznej bazie danych zaimplementowanej w MS Access.

Pole Dostępność z wartością logiczną tak/nie w tabeli operacji planowanego zlecenia produkcyjnego określa dostępność materiałów i narzędzi do produkcji, a także dostępność środka transportu. W przypadku dzielenia zadań w związku z outsourcingiem istotne są czasy transportu. Mogą one mieć duży wpływ na termin rozpoczęcia kolejnych członów operacji. Pole Dostępność jest więc także zabezpieczeniem przed tworzeniem zleceń produkcyjnych obejmujących operacje na częściach, które nie powróciły jeszcze z outsourcingu.

Ważnym elementem planistycznym w proponowanym tu podejściu jest Zlecenie produkcyjne. Podstawą tworzenia zlecenia produkcyjnego jest Zamówienie, które jest dokumentem planistycznym wykorzystywanym na etapie planowania zgrubnego oraz średniookresowego do tworzenia planowanych zleceń produkcyjnych. Natomiast zlecenie produkcyjne jest stosowane w szczegółowym planowaniu i harmonogramowaniu produkcji.

Zapotrzebowania na zasoby określone zleceniami produkcyjnymi są podstawą szczegółowego harmonogramowania produkcji. Sprawdzenie dostępności zasobów może być zatem przeprowadzone podczas planowania, gdy są obliczane zapotrzebowania na zasoby do planu produkcji.



Rys. 9.6. Relacje pomiędzy tabelami opisującymi marszrutę technologiczną [opracowanie własne]

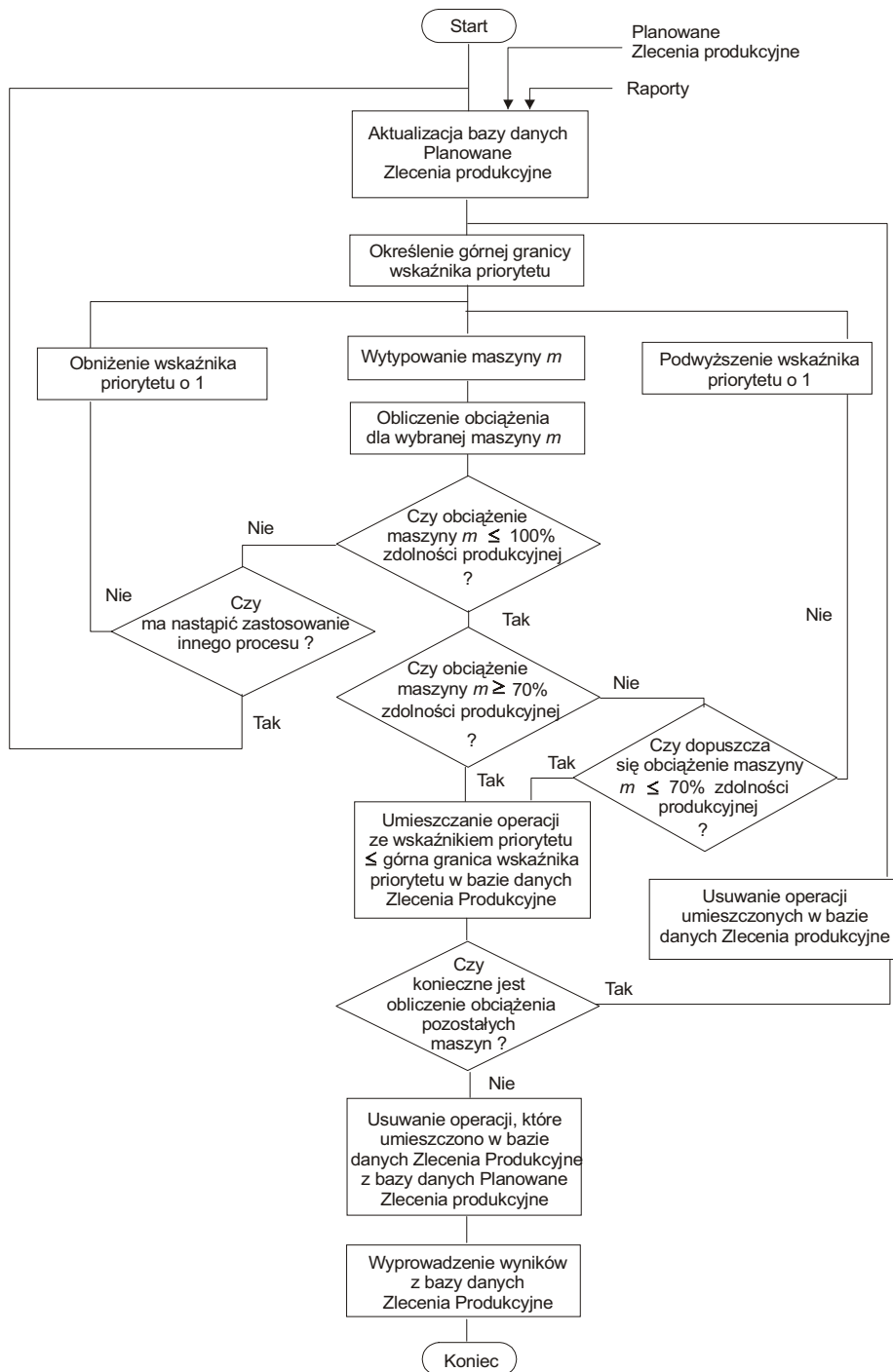
Ponadto na etapie tworzenia planu produkcji może być wykonane szczegółowe badanie obciążenia krytycznych miejsc produkcji.

Zamówienia mogą być składane zarówno przez klientów zewnętrznych, jak i przez klientów wewnętrznych rozpatrywanego tu systemu obróbki skrawaniem (np. wydziału). Na rysunku 9.2 pokazano przykład sieci dostaw, gdzie zamówienie może być złożone zarówno bezpośrednio przez klientów zewnętrznych (przedsiębiorstwa 5, 6 i 7), jak i klientów wewnętrznych (montaż, magazyn 2). W przypadku klientów wewnętrznych zamówienia mogą być generowane w związku z produkcją na magazyn, jak również mogą wynikać z potrzeb montażu własnego.

W oparciu o zamówienia tworzone są planowane zlecenia produkcji wykorzystywane następnie do krótkookresowego planowania produkcji i harmonogramowania w wydziale (lub gnieździe) obróbki skrawaniem.

### 9.4.3. Działanie systemu eksperckiego

System ekspercki krótkoterminowego planowania produkcji jest systemem konwersacyjnym. Działanie tego systemu eksperckiego prezentuje rysunek 9.7.



Rys. 9.7. Działanie systemu eksperckiego planowania produkcji [opracowanie własne]

Budowanie krótkoterminowego planu produkcji rozpoczyna się od wprowadzenia zleceń do wewnętrznej bazy danych Planowane Zlecenia produkcyjne. Możliwe jest ręczne i automatyczne wprowadzanie danych.

Baza danych Planowane Zlecenia produkcyjne jest tworzona i aktualizowana na bieżąco w trakcie dialogu z użytkownikiem (planistą) w oparciu o zewnętrzną bazę danych. W następnych krokach planista podczas dialogu z systemem określa górną granicę wskaźnika priorytetu oraz stanowisko, które typuje jako „wąskie gardło”.

Po wprowadzeniu niezbędnych danych jest obliczane obciążenie dla wybranego stanowiska. Obciążenie to jest sumą czasów wykonania wszystkich operacji przydzielonych do tego stanowiska.

W przypadku, gdy obciążenie stanowiska mieści się w granicach od 70% do 100% jego zdolności produkcyjnej wszystkie operacje ze wskaźnikiem priorytetu nie przekraczającym jego górnej granicy są umieszczane w krótkoterminowym planie produkcji obejmującym zlecenia produkcyjne. Następnie w trakcie dialogu z systemem planista podejmuje decyzję o sprawdzeniu obciążenia pozostałych stanowisk. Jeżeli nie jest konieczne sprawdzenie obciążenia pozostałych stanowisk to nastąpi wyprowadzenie wyników i aktualizacja baz danych. Jeżeli jednak planista podejmie decyzję o dalszym badaniu obciążenia komórki produkcyjnej, wtedy system ekspercki anuluje ustalony krótkoterminowy plan produkcji i praca z systemem rozpocznie się od wytypowania maszyny, dla której badane będzie obciążenie.

W sytuacji, gdy wyliczone obciążenie stanowiska jest mniejsze niż 70% jego zdolności produkcyjnej, w trakcie dialogu (z systemem eksperckim) planista rozstrzyga, czy stanowisko może być w tak niskim stopniu wykorzystane. W przypadku zbyt niskiego obciążenia stanowiska wskaźnik priorytetu zostaje podwyższony i ponownie jest obliczone obciążenie dla wytypowanej maszyny. Może jednak zdarzyć się, że planista dopuszcza niskie wykorzystanie stanowiska i wtedy wszystkie operacje ze wskaźnikiem priorytetu nie przekraczającym jego górnej granicy są umieszczane w krótkoterminowym planie produkcji. W kolejnym kroku planista podejmuje decyzję o sprawdzeniu obciążenia pozostałych stanowisk.

W przypadku, gdy obliczone obciążenie maszyny jest większe od jego zdolności produkcyjnej możliwa jest zmiana wielkości partii lub zastosowanie alternatywnego planu produkcji lub outsourcingu. Wtedy, kiedy nie jest to możliwe nastąpi obniżenie wskaźnika priorytetu i praca z systemem rozpocznie się ponownie od wytypowania maszyny.

### **9.5. Algorytm genetyczny harmonogramowania produkcji**

W prezentowanej tu metodyce – przed przystąpieniem do opracowania algorytmu genetycznego – przyjęto następujące założenia:

- w chwili początkowej znane są wszystkie zadania i opisujące je informacje,
- liczba operacji dla części jest różna lub równa liczbie maszyn w systemie,
- dana część może być realizowana wiele razy na tej samej maszynie,
- każda część ma inną marszrutę technologiczną (inny porządek operacji).

Implementacja algorytmu genetycznego wymagała określenia:

- genetycznej reprezentacji rozwiązania problemu,
- sposobu generowania populacji początkowej,
- postaci funkcji przystosowania,
- operatorów genetycznych,
- wartości parametrów:
  - rozmiaru populacji początkowej,
  - prawdopodobieństwa zastosowania operatora krzyżowania,
  - prawdopodobieństwa zastosowania operatora mutacji,
  - warunku zatrzymania.

### 9.5.1. Struktura algorytmu genetycznego

W proponowanym przez autorkę podejściu zaadaptowano klasyczny algorytm genetyczny zaproponowany przez D. E. Goldberga [74, 75], który przedstawiono w rozdziale 7. Proces ewolucji realizowany jest w następujących krokach:

Procedura: Algorytm genetyczny.

```

BEGIN
  t := 0;
  ustalenie początkowego P(t);
  ocena dopasowania P(t);
  WHILE (nie spełniony warunek zatrzymania) DO
  BEGIN
    t := t + 1;
    selekcja P(t) z pokolenia P(t-1);
    rekombinacja (przez krzyżowanie i mutację) P(t);
    ocena dopasowania P(t);
  END;
END.
```

### 9.5.2. Reprezentacja i tworzenie harmonogramu

Ze względu na prostotę reprezentacji i w oparciu o analizę prezentowanych w literaturze wyników badań [37], dla rozpatrywanych tu przedsiębiorstw zastosowano reprezentację bazującą na operacjach opisaną w podrozdziale 8.2.1. W tej reprezentacji każdy gen w chromosomie jest operacją, reprezentowaną za pośrednictwem łańcucha zadań.

### 9.5.3. Tworzenie populacji początkowej

Algorytm genetyczny startuje od początkowej populacji ciągów kodowych, po czym generuje kolejne populacje ciągów. Następne populacje są tworzone już przy użyciu algorytmu genetycznego. W większości zastosowań metod ewolucyjnych przetwarza się populację osobników, przy czym każdy osobnik reprezentuje potencjalne rozwiązanie rozpatrywanego zadania (w tym przypadku

harmonogram), a proces selekcji jest następująco ukierunkowany: lepsze osobniki mają większe szanse na przeżycie i reprodukcję. W prezentowanych tu badaniach selekcję realizowano z wykorzystaniem dobrze znanej metody koła ruletki [73], o której wcześniej wspomniano w podrozdziale 7.4 niniejszej rozprawy.

#### 9.5.4. Operatory krzyżowania i mutacji

Do krzyżowania wykorzystano operator PMX opisany w podrozdziale 7.5, ponieważ operator ten gwarantuje uzyskanie rozwiązań dopuszczalnych. Zastosowano także opisaną w niniejszej rozprawie mutację (podrozdział 7.5), polegającą na losowym wyborze dwóch genów w chromosomie i zamianie ich miejscami.

#### 9.5.5. Funkcja przystosowania

Można przyjmować różne kryteria oceny harmonogramów w zależności od uwarunkowań wewnętrznych i zewnętrznych przedsiębiorstwa. W przypadku analizowanych w niniejszej rozprawie przedsiębiorstw decydującym kryterium oceny harmonogramu jest termin zakończenia. W implementowanym systemie jako kryterium oceny harmonogramu przyjęto termin realizacji wszystkich zleceń  $C_{max}$ , inaczej długość uszeregowania (*makespan*). W rozpatrywanym przypadku mamy więc do czynienia z problemem minimalizacji. Zastosowana w niniejszym podejściu procedura działa dla liczb całkowitych. Dlatego wszystkie czasy muszą zostać przeliczone na najmniejszą jednostkę, w tym przypadku na minuty. W badaniach przyjęto, że: *czas zakończenia j (finish time<sub>j</sub>)* oznacza czas zakończenia ostatnio ustawionej operacji dla zadania *j*. Ponadto zakłada się, że w sytuacji, gdy nie została ustawiona jeszcze żadna operacja zadania *j* (na harmonogramie) *czas zakończenia j = 0*. Liczba zadań  $j = 1, 2, \dots, n$ . Poniżej opisano kolejne kroki procedury obliczania wartości długości uszeregowania.

*Krok 1:* Niech *finish time<sub>j</sub>* oznacza termin zakończenia ostatniej operacji zadania *j* umieszczonej na harmonogramie.

*Krok 2:* Odkoduj operację z chromosomu położoną najbardziej na lewo. Jeżeli jest to pierwsza operacja zadania *j*, wtedy przejdź do następnego kroku, w przeciwnym przypadku przejdź do kroku 4.

*Krok 3:*  $finish\ time_j = 0$ .

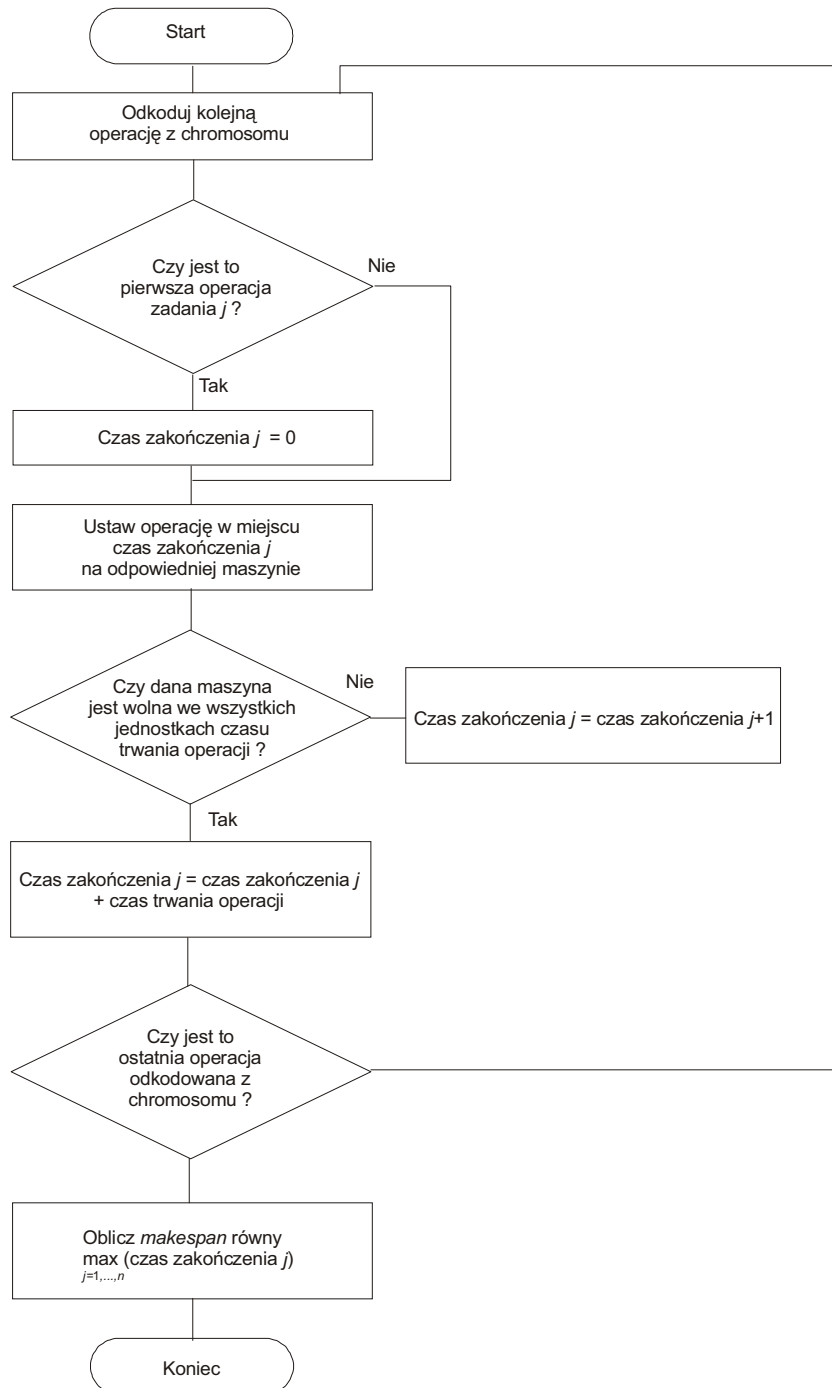
*Krok 4:* Umieść na harmonogramie operację w punkcie *finish time<sub>j</sub>* na przypisanej do niej maszynie. Jeżeli dana maszyna jest wolna we wszystkich kolejnych jednostkach czasu trwania operacji, wtedy przejdź do kroku 6, w przeciwnym przypadku przejdź do następnego kroku.

*Krok 5:*  $finish\ time_j = finish\ time_j + 1$  i powróć do kroku 4.

*Krok 6:*  $finish\ time_j = finish\ time_j + \text{czas trwania operacji}$ . Jeżeli jest to ostatnia operacja, która została odkodowana z chromosomu, wówczas przejdź do następnego kroku, w przeciwnym przypadku powracaj do kroku 2 tyle razy, aż wszystkie operacje zostaną umieszczone na harmonogramie.

*Krok 7:*  $Makespan = \min_{j=1, \dots, n} (finish\ time_j)$ .

Na rysunku 9.8 zilustrowano działanie procedury.



Rys. 9.8. Procedura obliczania długości uszeregowania [opracowanie własne]

Zastosowanie systemu eksperckiego w połączeniu z algorytmem genetycznym umożliwia efektywne zarządzanie procesami produkcyjnymi w obecności zakłóceń. System ekspercki odtwarza sposób postępowania dyspozytora systemu, z uwzględnieniem różnych warunków. Zaproponowane tu algorytmy nie wymagają dużej pojemności pamięci oraz długiego czasu obliczeń, dlatego celowe jest ich stosowanie, gdy wynik obliczeń trzeba otrzymać szybko.

## 9.6. Implementacja nowej metody zarządzania produkcją

Opracowane w ramach nowego podejścia do zarządzania produkcją w ogniwie sieci dostaw programy komputerowe zaimplementowano na komputerze z płytą główną Gigabyte 81E, procesorem Celeron 1,7 GHz/256 MB i HDD 40GB.

### 9.6.1. Implementacja systemu eksperckiego

Opisany powyżej system ekspercki planowania produkcji jest systemem dedykowanym, zaimplementowanym w języku Turbo Prolog 2.0. System ten jest systemem konwersacyjnym. Kolejne opcje głównego „menu” pozwalają użytkownikowi na: tworzenie planu produkcji oraz zakładanie, aktualizowanie i wyprowadzanie zawartości wewnętrznych baz danych. Aktywowanie wybranej z głównego „menu” opcji zakładania, aktualizowania i wyprowadzania zawartości wewnętrznych baz danych powoduje rozwinięcie podrzędnego „menu” w celu dokonania wyboru wewnętrznej bazy danych, której zawartość chcemy poznać lub modyfikować.

Opracowany w ramach niniejszych badań program składa się z kilku sekcji poprzedzonych słowami kluczowymi: *DOMAINS*, *DATABASE*, *PREDICATES*, *CLAUSES*, *GOAL*. Poniżej przedstawiono podstawowe sekcje programu.

#### DOMAINS

```
plan,raportWy,raportOp=symbol
idZamow,idWyrobu,idStanow,iloscWyr,idOper,priorytet=integer
tpz,tj,czasW,czasTr,wskazWyk,zdolProd=real
lista=czasW*
```

#### DATABASE - db\_zlecProd

```
db_zlecProd(idZamow,idWyrobu,idOper,tpz,tj,czasTr,iloscWyr,idStanow,
priorytet,czasW)
```

#### DATABASE - db\_planProd

```
db_planProd(idZamow,idWyrobu,idOper,tpz,tj,czasTr,iloscWyr,idStanow,
priorytet,czasW,plan)
```



```

DATABASE - db_raportW
  db_raportW(idZamow,idWyrobu,idOper,tpz,tj,czasTr,iloscWyr,idStanow,
  priorytet,czasW,raportWy)
DATABASE - db_raportO
  db_raportO(idZamow,idWyrobu,idOper,tpz,tj,czasTr,iloscWyr,idStanow,
  priorytet,czasW,raportOp)
DATABASE - db_planOper
  db_planOper(idZamow,idWyrobu,idOper,tpz,tj,czasTr,iloscWyr,idStanow,
  priorytet,czasW)
PREDICATES
  wykonanieProgramu
  opcja(integer)
  wyswietl
  sumaczasW(lista,czasW,integer)
  obliczenia
  wariant(integer)
  zapis
  usuwanie
  zapisywanie(integer)
  kasowanie(integer)
  pliki
  baza(integer)
  czytanie
  czytaj(integer)
GOAL
  wykonanieProgramu.
CLAUSES
  wykonanieProgramu :-
  write(„MENU”),nl,
  write(„1-wyswietlanie danych”),nl,
  write(„2-wprowadzanie danych”),nl,
  write(„3-usuwanie danych”),nl,
  write(„4-tworzenie planu produkcji”),nl,
  write(„5-wczytanie danych z pliku”),nl,
  write(„6-zapisanie danych w pliku”),nl,
  write(„Esc-koniec pracy”),nl,
  readint(I),
  opcja(I),
  wykonanieProgramu.
  wykonanieProgramu.

```

Sekcja *DOMAINS* zawiera deklaracje dziedzin użytkownika. W programie zadeklarowano następujące dziedziny:

plan	– plan	=symbol
raportwy	– raport wykonanych zadań	=symbol
raportop	– raport opóźnionych zadań	=symbol
idzmow	– kod zamówienia	=symbol
idwyrobu	– kod wyrobu	=integer
idstanow	– kod stanowiska	=integer
iloscwyr	– wielkość partii	=integer
idoper	– numer operacji	=integer
priorytet	– wskaźnik priorytetu	=integer
tpz	– czas przygotowawczo–zakończeniowy	=real
tj	– czas jednostkowy	=real
czasw*	– czas wykonania operacji	=lista
czastr	– czas transportu	=real
Wskazwyk	– wskaźnik wykorzystania stanowiska	=real
Zdolprod	– zdolność produkcyjna stanowiska	=real

Baza wiedzy tworzona jest przez wyodrębnienie części pamięci operacyjnej, w której zapisywane są fakty tworzące wiedzę.

Zbiór faktów w Turbo Prologu można interpretować jako relacyjną bazę danych. Dla umożliwienia odwoływania się do bazy danych wprowadzono sekcję *DATABASE* i predykaty standardowe umożliwiające operacje na bazach danych. Standardowe predykaty *consult* i *save* pozwalają na wczytywanie oraz zapisywanie zawartości baz danych przechowywanych w pamięci operacyjnej do pamięci zewnętrznej.

Fakt jest zapisywany poprzez użycie specjalnie w tym celu zdefiniowanego predykatu. Predykatem służącym do wprowadzania faktów do bazy wiedzy jest w tym przypadku *assert*.

W programie zadeklarowano następujące wewnętrzne bazy danych:

db_zlecProd	– wewnętrzna baza danych zawierająca zbiór operacji ze zleceń produkcyjnych,
db_planProd	– wewnętrzna baza danych zawierająca zbiór operacji z planu produkcji,
db_raportW	– wewnętrzna baza danych zawierająca zbiór operacji z raportu o wykonanych operacjach,
db_raportO	– wewnętrzna baza danych zawierająca zbiór operacji z raportu o opóźnionych operacjach,
db_planOper	– baza danych zawierająca zbiór operacji wytypowanych do planu produkcji tworzona w pamięci operacyjnej.



```
write(„Czy chcesz wprowadzić operacje do planu (T/N)?”),readln(O),O=t,
db_planOper(IdZamow,IdWyrobu,IdOper,TPz,Tj,CzasTr,IloscWyr,IdStanow,
Priorytet,CzasW),
retract(db_zlecProd(IdZamow,IdWyrobu,IdOper,TPz,Tj,CzasTr,IloscWyr,
IdStanow,Priorytet,CzasW)),fail;
```

```
retractall(db_planOper(_____,_____)).
```

obliczenia :-nl.

```
SumaCzasW([ ],0,0).
sumaCzasW([H|T],Suma,N) :-
sumaCzasW(T,S1,N1),
Suma=H+S1,
N=N1+1.
```

Predykat standardowy *assert* powoduje umieszczenie wybranych operacji w wewnętrznej bazie danych Plan produkcji.

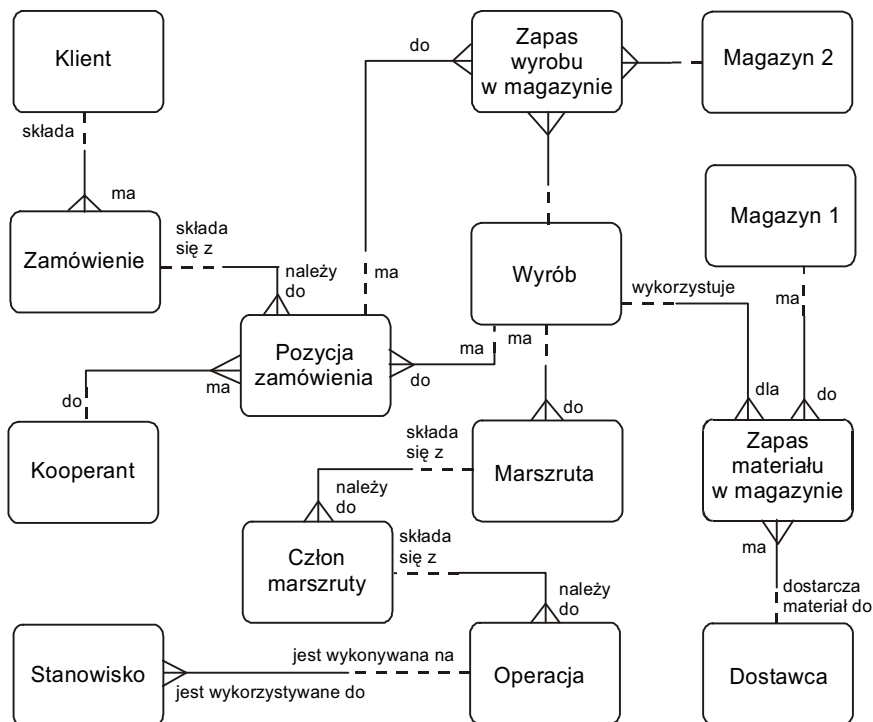
Predykat standardowy *fail* jest predykatem, którego wartością jest fałsz. Oznacza to, że klauzula, która w swej definicji zawiera *fail* nigdy nie może być uzgodniona. Predykat ten jest umieszczony jako ostatni predykat definiujący daną klauzulę w celu wymuszenia przeszukiwania całego obszaru wartości zmiennych pozostałych predykatów.

W programie zdefiniowano predykaty arytmetyczne umożliwiające porównanie wartości liczbowych oraz nadawanie zmiennym wartości określonych w programie wyrażeń arytmetycznych.

Sekcja *GOAL* może wystąpić tylko raz w programie. W tym przypadku sekcja *GOAL* to wykonanie programu.

Do zarządzania zamówieniami i zleceniami w wydziale (lub gnieździe) obróbki skrawaniem zaprojektowano zewnętrzną bazę danych, którą następnie zaimplementowano w MS ACCESS. Na rysunku 9.9 pokazano diagram związków encji (*entity diagram*) tej bazy, opracowany według metodologii Oracle.

Planowane Zlecenie produkcyjne pochodzi z kwerendy utworzonej w oparciu o Pozycje zamówienia oraz Operacje i zawiera następujące pola: IdPozycjiZamówienia, IDWyrobu, NazwaWyrobu, IlośćWyrobu, IdOperacji, NazwaOperacji, CzasJednostk, CzasPrzygZak, CzasTransportu, IDStanowiska, NazwaStanowiska, NazwaMarszruty, Priorytet.



Rys. 9.9. Diagram związków encji zewnętrznej bazy danych systemu eksperckiego wg metodologii Oracle [opracowanie własne]

### 9.6.2. Implementacja algorytmu genetycznego

W programie wykorzystano bibliotekę narzędzi dla algorytmów genetycznych Galib do problemów optymalizacji, stworzoną przez badaczy z Mechanical Engineering Department, Massachusetts Institute of Technology. Biblioteka ta zawiera m.in. klasyczny algorytm genetyczny zaproponowany przez D.E. Goldberga bez procedury obliczania funkcji celu. Ponadto w bibliotece tej znajdują się narzędzia, które umożliwiają użytkownikowi dokonanie wyboru sposobu selekcji, mutacji i krzyżowania.

Opracowany przez autorkę rozprawy i wykorzystany następnie do testów program obejmujący procedurę obliczania funkcji celu oraz prezentację wyników w postaci harmonogramu Gantta zaimplementowano w języku Borland C++ Builder 5 w systemie Windows 2000 Professional.

Poniżej zaprezentowano fragment programu obliczania funkcji celu, który opracowano w ramach tworzenia nowego prezentowanego tu systemu harmonogramowania i zaimplementowano w systemie Galib.

```

float Objective(GAGenome &c)
{
    int i, j, t, m;
    float score = 0;
    //czas zakonczenia ostatniej operacji zadania
    int finished;
    //ostatnia operacja zadania
    int op;
    //zadanie
    int task;
    int tim;

    bool busy;
    char s[1];

    ofstream outfile;

    int test;
    int ind=0;

    //  GAStringGenome &g = (GAStringGenome &c);
    GA1DArrayAlleleGenome<int> &g = (GA1DArrayAlleleGenome<int>
    &c);

    for(i=0; i<numTasks; i++)
    {
        j=0;
        while (machines[i][j]!=-1)
        {
            opind[i][j]=ind;
            j++;
            ind++;
        }
        numop[i]=j;
    }

    i=0;
    t=0;

    //clear variables
    for (i=0; i<50; i++)
    {
        currop[i]=0;
        lastfinishtime[i]=0;
    }
}

```

```

for (i=0; i<50; i++)
{
  for (j=0; j<MAXTIME; j++)
  {
    gantt[i][j]=-1;
  }
}

i=0;
j=0;

while (i<g.size())
{
  task = g.gene(i)-1;

  op = currop[task];
  finished = lastfinishtime[task];

  busy=false;
  do
  {
    busy=false;
    tim = times[task][op];

    for (j=0; j<tim; j++)
    {
      t = finished+j;
      m = machines[task][op];

      if (gantt[m][t] != -1)
      {
        busy=true;
      }
    }
    if (busy==true)
    {
      finished++;
    }
  }
  while (busy==true);

  tim = times[task][op];
  for (j=0; j<tim; j++)

```

```

    {
        gantt[m][finished+j]=opind[task][op];
    }

    currop[task]= currop[task] + 1;
    lastfinishtime[task]=finished+j;
    i++;
}

for (j=MAXTIME-1; j>=0; j--)
{
    for (i=0; i<numMachines; i++)
    {
        if (gantt[i][j]!=-1)
        {
            if (j>score)
            {
                score = j;
            }
        }
    }
}
score++;

return score;
}

```

Zaimplementowany algorytm genetyczny wyprowadza wyniki w postaci chromosomu i graficznego harmonogramu. Taka prezentacja wyników wspomaga planistę w podejmowaniu decyzji dotyczących wyrównywania obciążeń maszyn. Poniżej przytoczono fragment programu, który umożliwi generowanie wyników.

```

void GenerateScoreModeResults(float score)
{
    int i, j;
    int k, f;
    bool done;
    AnsiString s;
    char ss[120];

    Form1->ScoreEdit->Text = AnsiString(score);

    Form1->ResultsStringGrid->RowCount = numMachines+1;

```



```

Form1->ResultsStringGrid->ColCount = score+2;

for (j=0; j<numMachines; j++)
{
    k = j+1;
    Form1->ResultsStringGrid->Cells[0][j]= (AnsiString) k;
}

for (j=1; j<score+1; j++)
{
    Form1->ResultsStringGrid->Cells[j][numMachines]= AnsiString(j);
}

done=false;
for (i=0; i<numMachines; i++)
{
    for (j=0; j<score; j++)
    {
        done=false;
        for (int ii=0; ii<numTasks; ii++)
        {
            for (int jj=0; jj<numop[ii]; jj++)
            {
                if (gantt[i][j]==opind[ii][jj])
                {
                    k = ii+1;
                    f = jj+1;
                    s = (AnsiString) k;
                    s+= „.”;
                    s+=(AnsiString) f ;
                    Form1->ResultsStringGrid->Cells[j+1][i]= s;
                    done = true;
                }
            }
        }
    }
    if (!done)
    {
        //      outfile << „000” << „.”;
    }
}
}
}
}

```

## 10. Uwagi dotyczące cyklu życia systemu zarządzania produkcją z zastosowaniem systemu eksperckiego i algorytmu genetycznego

Metodykę tworzenia systemów informatycznych można zdefiniować jako: spójny, logicznie uporządkowany zestaw metod i procedur o charakterze technicznym i organizatorskim, co pozwala zespołowi wykonawczemu realizować cykl życia systemu [68, 211].

Na podstawie analizy wiedzy oraz doświadczenia w tworzeniu systemów informatycznych cykl życia systemu zarządzania produkcją z systemem eksperckim i algorytmem genetycznym można podzielić na sześć powiązanych sprzężeniami zwrotnymi następujących faz (por. [12, 13, 125, 129, 131]):

- 1) planowanie,
- 2) analiza,
- 3) projektowanie,
- 4) budowanie i dokumentowanie,
- 5) wdrażanie,
- 6) eksploatacja.

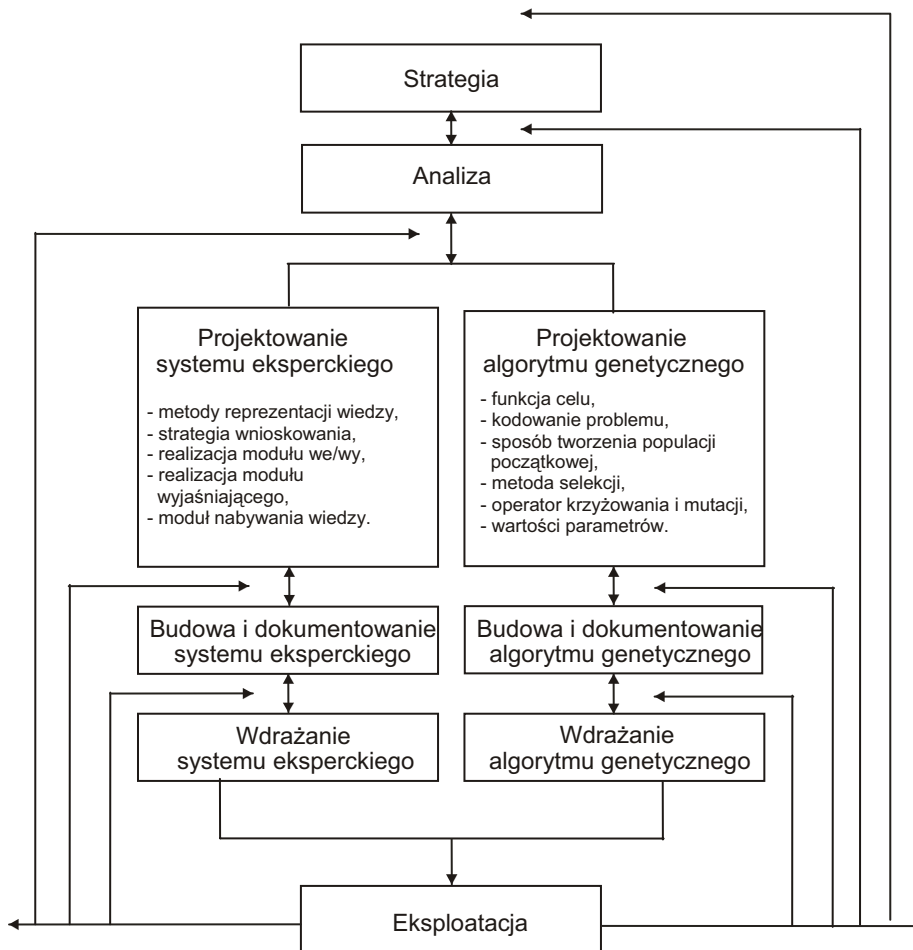
Na rysunku 10.1 zilustrowano graficznie następstwo faz cyklu życia systemu zarządzania produkcją z systemem eksperckim i algorytmem genetycznym.

Pierwszą fazą cyklu życia systemu jest planowanie, zwane również w literaturze przedmiotu strategicznym planowaniem systemu [12, 134]. Przedmiotem zainteresowania w tej fazie są: misja firmy i jej cele, biznesplan, a także otoczenie. Uzgadniane są cele przedsiębiorstwa i ich priorytety, jak również określa się zasięg przedsięwzięcia informatycznego. Podejmowane są kluczowe decyzje dotyczące zastosowania metod sztucznej inteligencji. Rozważa się wszystkie możliwe rozwiązania alternatywne.

Wypracowana wraz z kierownictwem przedsiębiorstwa strategia informatyzacji wymaga uwzględnienia różnego rodzaju zagrożeń w osiąganiu celów oraz ograniczenia w funkcjonowaniu przedsiębiorstwa. Dokonywana jest ponadto analiza rynku sprzętu, oprogramowania i sieci komputerowych, jak również technologii informatycznych dostępnych na rynku. W tej fazie występuje projektowanie zstępujące (*top-down*). W trakcie cyklicznie prowadzonych wywiadów zbierane są podstawowe informacje o przedsiębiorstwie i wymaganiach użytkowników dotyczących tworzonego systemu informatycznego. Modelowane są specyficzne funkcje i dane, które są wielokrotnie opiniowane i weryfikowane.

W małych firmach analiza strategiczna powinna objąć całą firmę. W bardzo dużych przedsiębiorstwach, podzielonych na niezależne działy studium obejmuje funkcje działu. Efektem fazy planowania jest ramowy plan informatyzacji oraz modele struktury przedsiębiorstwa – funkcjonalnej i informacyjnej. Plan informatyzacji określa, m.in.: przedmiot, zakres i podstawowe funkcje systemu, czas realizacji przedsięwzięcia, szacunkowe koszty realizacji poszczegól-

gólnych etapów, zmiany w systemie informacyjnym i w strukturze zatrudnienia, wymagania w zakresie szkolenia przyszłych użytkowników systemu, sposób kontroli przebiegu prac, zasady współpracy z użytkownikami ze wskazaniem kompetencji w podejmowaniu decyzji, wstępny projekt architektury systemu, harmonogram realizacji przedsięwzięcia. Harmonogram realizacji przedsięwzięcia obejmuje przyporządkowanie zadań do skali czasu (z uwzględnieniem priorytetów realizacyjnych) oraz określa sposoby i zasady zatwierdzania wyników analizy oraz osoby odpowiedzialne za jej przeprowadzenie. W miarę postępu prac harmonogram realizacji przedsięwzięcia jest weryfikowany. Należy podkreślić, że planowanie jest procesem ciągłym, powtarzanym stosownie do oceny kierownictwa naczelnego firmy oraz czynników zewnętrznych.



Rys. 10.1. Cykl życia systemu zarządzania produkcją z systemem eksperckim i algorytmem genetycznym [opracowanie własne]

W fazie analizy pogłębia się analizę systemu rzeczywistego i modeli opracowanych na etapie analizy strategicznej. Analiza szczegółowa dotyczy na ogół jednej aplikacji. Podstawowymi czynnościami w fazie analizy są [204]:

- identyfikacja i charakterystyka problemów i celów,
- studium dziedziny przedmiotowej – opis istniejącego systemu,
- studium wykonalności systemu informatycznego,
- definicja i ustalenie priorytetów zidentyfikowanych przez użytkownika,
- analiza kosztów i zysków.

Stosowane na tym etapie podejście wstępujące (*bottom-up*) ma chronić projektantów przed pominięciem istotnych szczegółów. Rozważaniom poddawane są problemy pewnej dziedziny przedmiotowej organizacji z udziałem kluczowych wykonawców. W ramach analizy prowadzone są wywiady z szefami pionów i kierownikami działów lub ich zastępcami. Produktami tego etapu są szczegółowe modele przedsiębiorstwa oraz zaktualizowany plan informatyzacji (por. [160]). Szczegółowe modele odwzorowują przedsiębiorstwo w ujęciu statycznym i dynamicznym. Ujęcie statyczne odzwierciedlają modele podstawowych struktur organizacji, tj. struktury organizacyjnej, funkcjonalnej, informacyjnej i przestrzennej, jak również powiązania struktur: funkcjonalnej z organizacyjną, organizacyjnej z informacyjną oraz organizacyjnej z przestrzenną. Analizę szczegółową na ogół rozpoczyna się od zbudowania tzw. *diagramu hierarchii funkcji* i *diagramu przepływu danych* [131], które następnie weryfikowane są z pracownikami systemu rzeczywistego pod kątem poprawności.

Studium wykonalności systemu składa się z czterech części: technicznej, organizacyjnej, prawnej i ekonomicznej. Najpóźniej w tej fazie powinny zostać podjęte decyzje o zastosowaniu technik sztucznej inteligencji oraz integracji z istniejącym systemem informatycznym.

Podczas analizy szczegółowej wymagana jest ścisła współpraca analityków systemu z jego przyszłymi użytkownikami. Rezultatem tej współpracy powinny być precyzyjnie zdefiniowane wymagania użytkowników, które umożliwią tworzenie różnych wariantów rozwiązań na etapie projektowania. W związku z tym ta faza niekiedy nazywana jest fazą *analizy potrzeb użytkownika* (*user requirements analysis*).

W przypadku podjęcia decyzji o budowie systemu eksperckiego do prac powinien zostać włączony inżynier wiedzy. Podczas tego etapu ekspert i inżynier wiedzy określają problem do rozwiązania i jego zakres. Ponadto powinny zostać określone źródła wiedzy. Źródłem wiedzy może zostać ekspert dziedzinowy (*domain expert*) – specjalista z dziedziny lub zespół ekspertów. Do podstawowych źródeł wiedzy należą również wszelkiego rodzaju publikacje z rozważanej dziedziny. W toku analizy inżynier wiedzy i ekspert przedstawiają kluczowe koncepcje, relacje i charakterystykę przepływu informacji, które są niezbędne w procesie rozwiązywania problemu w danej dziedzinie. Wynik analizy powinien dać podstawę do uzyskania odpowiedzi na pytania [15]: jakie dane są do dyspozycji, co jest znane w odniesieniu do faktów, a co

musi zostać jeszcze wypracowane w toku np. wnioskowania, czy powiązania, przyczynkowość i inne uzyskane informacje mogą być uwzględnione i przedstawione w diagramach, wykresach itp.? Rozstrzygnięcia wymaga także to, czy tworzony system ekspercki będzie systemem dedykowanym, tj. tworzonym od podstaw przez inżyniera wiedzy współpracującego z informatykiem, czy też może zostać wykorzystany system szkieletowy z pustą bazą wiedzy.

W przypadku podjęcia decyzji o zastosowaniu algorytmu genetycznego należy podjąć decyzję o wyborze jednego z trzech podejść:

- zastosowanie algorytmu genetycznego bazującego na ciągach binarnych i modyfikacja oryginalnego problemu do postaci odpowiedniej dla algorytmu genetycznego,
- pozostawienie rzeczywistego problemu w niezmienionej postaci i zastosowanie reprezentacji chromosomów, odpowiadającej potencjalnym rozwiązaniom z użyciem naturalnych struktur danych,
- trzecie podejście to adaptacja zarówno problemu do algorytmu genetycznego, jak i algorytmu genetycznego do problemu.

W fazie tej powstaje ogólna koncepcja algorytmu genetycznego oraz określana jest funkcja celu.

Projektowanie systemu informatycznego obejmuje przede wszystkim oprogramowanie i architekturę systemu. Ponadto sporządzane są plany kontroli danych, wdrożenia systemu i testowania. Elementy te zawarte są w finalnym projekcie systemu, który powinien być zatwierdzony przez kierownika systemu, jak również osobę odpowiedzialną za współpracę ze strony użytkownika.

W trakcie projektowania definiowane jest środowisko aplikacji i jej bezpieczeństwo oraz określone są funkcje administracyjne. Podczas projektowania uwzględnia się szczegółowe wymagania sformułowane w trakcie analizy. Na podstawie wcześniejszych modeli oraz istniejącego systemu rzeczywistego ustala się granice każdego programu, z uwzględnieniem ograniczeń finansowych i czasowych.

Tak jak i poprzednie fazy z prezentowanej tu metodyki, projektowanie jest realizowane w sposób iteracyjny. Przygotowywane warianty projektów są poddawane weryfikacji tak długo, dopóki nie zostanie osiągnięty kompromis w sprawie finalnego rozwiązania. Podczas projektowania może okazać się, że zrealizowanie wymagań określonych wcześniej przez użytkownika jest niemożliwe, np. ze względu na brak środków finansowych, czas realizacji, ograniczenia sprzętowe czy brak fachowej wiedzy. W przypadku bardzo dużych zmian wymagań postawionych przez użytkownika niekiedy może być konieczny powrót do etapów analizy.

W przypadku tworzenia systemów eksperckich formalizacja wiedzy wiąże się z potrzebą doboru postaci, w jakiej zostanie ujęta i przedstawiona wcześniej odkryta lub wypracowana koncepcja. W tej fazie inżynier wiedzy dokonuje wyboru reprezentacji wiedzy. W przypadku systemu eksperckiego, w tej fazie następuje przeniesienie strukturalnie sformalizowanej wiedzy, np. z postaci

papierowej na formę odpowiadającą specyficie języka programu komputerowego.

W przypadku algorytmów genetycznych formalizacja dotyczy opracowania: sposobu tworzenia populacji początkowej, metod selekcji, operatorów krzyżowania i mutacji, inwersji oraz wartości parametrów genetycznych.

Etap budowy to kodowanie programów. Równoległe z budową sporządza na jest dokumentacja dla operatorów systemu i dokumentacja użytkowników. Zbudowanie prototypu umożliwia ocenę przyjętych założeń i koncepcji systemu, pozwala weryfikować przydatność przyjętych strategii i ułatwia dalszy proces projektowania. Pozytywne oceny prototypu uzasadniają przystąpienie do budowy właściwego systemu, przy czym stosowane są dwie strategie: ewolucyjny rozwój prototypu do postaci pełnego systemu lub odrzucenie prototypu i tworzenie systemu od początku.

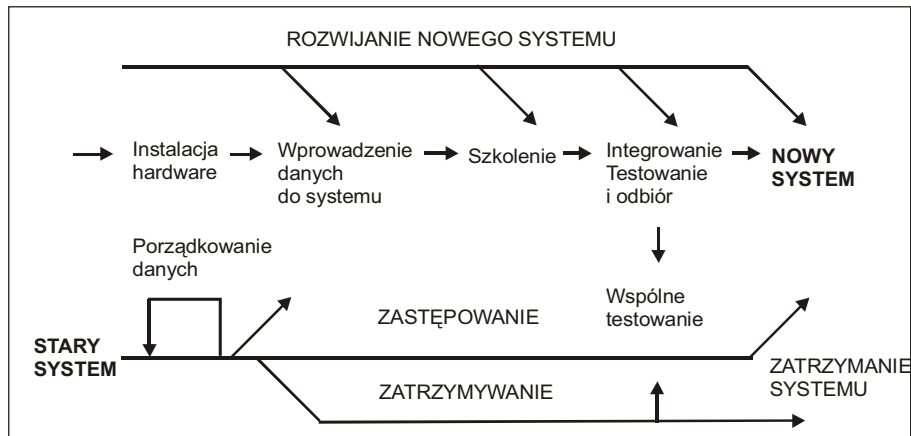
Oprogramowanie realizowane jest przeważnie bez bliższych kontaktów z użytkownikami. Jednak wszelkie wprowadzane zmiany nie uzgodnione na wcześniejszych etapach powinny być akceptowane przez analityka i użytkownika. Ponadto programiści powinni dokumentować wszelkie decyzje dotyczące zmian.

Testowanie systemu będące elementem fazy budowy i dokumentowania odbywa się zarówno dla części systemu, jak i całości systemu zgodnie z opracowanym w poprzedniej fazie planem testowania. Testowanie służy sprawdzeniu systemu pod względem jego zdolności wykonawczych. Za pomocą testowania jest badane zachowanie systemu podczas działania z punktu widzenia: jego zdolności i wydajności w rozwiązywaniu problemów, wykrywalności pojawiających się błędów.

Istotnym etapem budowania algorytmu genetycznego jest strojenie, to znaczy optymalne określanie takich wielkości, jak: rodzaj zastosowanej mutacji, rodzaj zastosowanego krzyżowania, sposób selekcji, prawdopodobieństwo mutacji, prawdopodobieństwo krzyżowania, wielkość populacji i parametry genetyczne.

Faza wdrożenia obejmuje: demonstrację systemu, zainstalowanie sprzętu komputerowego i oprogramowania, szkolenie przyszłych użytkowników systemu i personelu operacyjnego oraz przekazanie pełnej dokumentacji i odbiór testów. Wdrożenie nowego systemu informatycznego wymaga: ustalenia sposobu jego użytkowania, wyboru ochrony systemu, przeprowadzania badań w warunkach rzeczywistych i integracji z istniejącym systemem informatycznym.

W fazie wdrożenia oba systemy, stary i nowy mogą pracować tak długo, aż osiągnięte zostaną cele określone w planie informatyzacji. Na rysunku 10.2 pokazano jak przebiega wdrażanie nowego systemu.



Rys. 10.2. Wdrożenie [13]

W trakcie wdrożenia – dla zapewnienia ciągłości pracy przedsiębiorstwa – konieczna jest współpraca zespołu realizatorów z użytkownikami. Szkolenie użytkowników przebiega na ogół dwustopniowo. Na pierwszym etapie realizowane są szkolenia dla wybranej grupy użytkowników zwanych kluczowymi użytkownikami, które rozpoczynają się już w fazie budowy systemu. Kluczowi użytkownicy uczą się funkcjonowania systemu operując na rzeczywistych danych. Poznają metody szkolenia oraz zawartość dokumentacji systemu. Na drugim etapie kluczowi użytkownicy szkolą pozostałych użytkowników wdrażanego systemu. Dopiero po zapoznaniu wszystkich użytkowników z nowym systemem może nastąpić jego odbiór.

W trakcie eksploatacji systemu kontrolowany jest jego sposób funkcjonowania i wydajność. Utrzymanie dobrej jakości systemu wymaga jego stałej pielęgnacji. Wprowadzane zmiany nie powinny powodować przerw w pracy systemu.

## 11. Badania eksperymentalne

### 11.1. Opis badanych systemów produkcyjnych

Poniżej zaprezentowano krótką charakterystykę badanych systemów wytwórczych.

*Studium przypadku 1* – W ramach badań jako pierwsze analizowano przedsiębiorstwo zajmujące się produkcją części do montażu i remontów maszyn budowlanych. Są to części wykonywane na wydziałach mechanicznych, o bardzo dużych gabarytach, jak na przykład: wał, tuleja, elementy łożyska, wpusty przyrmatyczne, koła zamachowe, elementy sprzęgła. Produkcja realizowana jest pojedynczo (tj. bez tworzenia partii produkcyjnych), ze względu na transport i długie czasy obróbki. Podstawą włączenia danej części do planu produkcji jest zlecenie klienta na wykonanie lub remont maszyny budowlanej.

W pozostałych analizowanych przypadkach produkcja była realizowana w małych partiach.

*Studium przypadku 2* – Analizowane przedsiębiorstwo produkuje drobne części do montażu urządzeń elektrycznych oraz części zamienne dla przemysłu elektrycznego. Produkcja na wydziale mechanicznym realizowana jest – na zlecenie klienta – w małych seriach. Produkowane są przede wszystkim części do akumulatorów. Przedsiębiorstwo produkuje w systemie trzech zmian roboczych. Ze względu na konieczność wykonania różnych prac na innych wydziałach, np. w galwanizerni stosowany jest outsourcing wewnętrzny.

*Studium przypadku 3* – Przedsiębiorstwo produkuje – na zlecenie klienta – części do montażu własnego i remontów maszyn rolniczych. Badaniami objęto wydział mechaniczny, realizujący produkcję seryjną w systemie dwóch zmian roboczych.

*Studium przypadku 4* – W ramach badań podjęto także analizę planowania produkcji dla wydziału mechanicznego wytwarzającego seryjnie części armatury hydraulicznej. Produkcja realizowana jest – na zamówienie klienta – w systemie trzech zmian roboczych.

*Studium przypadku 5* – Badaniami objęto przedsiębiorstwo produkujące na zlecenie części do armatury wodociągowej. Przedsiębiorstwo to specjalizuje się w obróbce skrawaniem i wytwarza części w systemie dwóch zmian roboczych. Ze względu na długości cykli produkowanych w partiach części plan produkcji budowany jest dla dwóch dni roboczych.

*Studium przypadku 6* – Analizą objęto przedsiębiorstwo pracujące w systemie trzech zmian roboczych i produkujące seryjnie części do maszyn dla leśnictwa.

*Studium przypadku 7* – Badaniami objęto produkcję części do produkowanych na zlecenie klienta form wtryskowych. Produkcja jest realizowana w bardzo małych seriach, a czasami występuje produkcja jednostkowa.

*Studium przypadku 8* – Eksperyment przeprowadzono dla części do maszyn spożywczych, produkowanych w bardzo małych seriach na zlecenie klienta.



*Studium przypadku 9* – W ramach tych badań analizowano fazę obróbką produkcji narzędzi. W tym systemie produkcja realizowana jest w stosunkowo dużych seriach w porównaniu z innymi rozważanymi tu systemami wytwórczymi.

*Studium przypadku 10* – Badaniami objęto obróbkę mechaniczną w przedsiębiorstwie produkującym części do pojazdów i maszyn rolniczych. Na wydziale obróbki mechanicznej przyjęto system dwóch zmian roboczych.

Tabela 11.1 prezentuje charakterystyki wymienionych wyżej systemów produkcyjnych.

Tabela 11.1. Charakterystyki badanych systemów produkcyjnych

Numer eksperymentu	Asortyment produkcji	Liczba maszyn w systemie	Zdolności produkcyjne		
			Horyzont planowania (liczba dni)	Liczba zmian roboczych	Fundusz kalendarzowy (w minutach)
1	Części do maszyn budowlanych	11	1	2	960
2	Części do urządzeń elektrycznych	9	1	3	1440
3	Części do maszyn rolniczych	16	1	2	960
4	Części do armatury hydraulicznej	13	1	3	1440
5	Części do armatury wodociągowej	14	2	2	1920
6	Części dla urządzeń leśnictwa	9	1	3	1440
7	Części do form wtryskowych	11	1	3	1440
8	Części do maszyn spożywczych	9	1	2	960
9	Narzędzia	14	1	3	1440
10	Części do pojazdów i maszyn rolniczych	21	2	2	1920

## 11.2. Strojenie algorytmu genetycznego

Zaproponowany w rozdziale dziewiątym algorytm genetyczny strojono w następujących etapach:

- określanie wielkości początkowej populacji,
- określenie prawdopodobieństwa mutacji,
- ustalenie minimalnej liczby generacji, dla której uzyskanie najlepszego chromosomu jest najbardziej prawdopodobne,
- weryfikacja metody selekcji,
- weryfikacja sposobu krzyżowania.

Strojenie algorytmu genetycznego zrealizowano z wykorzystaniem danych testowych dla systemu produkcyjnego opisanego jako przypadek 3, który jest typowym przedstawicielem przedsiębiorstwa realizującego produkcję o szybkozmennym asortymencie, w trakcie dwóch zmian roboczych. Eksperymenty przeprowadzono z wykorzystaniem dziesięciu planów produkcji dla tego systemu, które zaprezentowano w tabelach od 1 do 10, umieszczonych w aneksie niniejszej rozprawy.

W celu ustalenia wpływu wielkości populacji początkowej na wartość długości uszeregowania wykonano testy dla 10 wymienionych wyżej planów produkcji, z zastosowaniem populacji początkowej – 20 oraz 40 i następujących pozostałych parametrach genetycznych:

- liczba generacji (warunek zatrzymania) – 260,
- prawdopodobieństwo krzyżowania – 1,
- prawdopodobieństwo mutacji – 0,05.

Wyniki testów zaprezentowano poniżej w tabeli 11.2.

Tabela 11.2. Porównanie wartości długości uszeregowania dla różnych wielkości populacji początkowych

Numer eksperymentu	Wartości najlepszej długości uszeregowania	
	Populacja początkowa = 20	Populacja początkowa = 40
1	912	<b>864</b>
2	918	<b>905</b>
3	<b>916</b>	917
4	924	<b>917</b>
5	<b>919</b>	926
6	<b>912</b>	928
7	<b>959</b>	968
8	924	<b>917</b>
9	<b>919</b>	<b>919</b>
10	960	<b>928</b>
Liczba najlepszych chromosomów	5	6
Suma długości uszeregowania	9263	9189

Opierając się na wynikach testów zawartych w powyższej tabeli można stwierdzić, że zwiększanie wielkości początkowej populacji powyżej 20 osobników nie gwarantuje uzyskania lepszych rozwiązań. W przeprowadzonych testach, w których populacja początkowa była równa 20 osobników najlepsze długości uszeregowania otrzymano dla 5 przypadków, natomiast w testach dla populacji początkowej wynoszącej 40 osobników najlepsze długości uszeregowania otrzymano 6-krotnie. W celu głębszej analizy dotyczącej wpływu wielkości populacji początkowej na długość uszeregowania obliczono średnie odchylenie długości uszeregowania  $O_{usr}$ , w oparciu o wyniki zawarte w tabeli 11.2. Średnie odchylenie długości uszeregowania obliczono według wzoru:

$$O_{usr} = \frac{(S_{40} - S_{20})}{S_{40}} \cdot 100\% \quad (11.1)$$

gdzie:

- $S_{20}$  – suma długości uszeregowania dla 10 harmonogramów uzyskanych z zastosowaniem algorytmu genetycznego, w którym populacja początkowa wynosiła 20 osobników,
- $S_{40}$  – suma długości uszeregowania dla 10 harmonogramów uzyskanych z zastosowaniem algorytmu genetycznego, w którym populacja początkowa wynosiła 40 osobników.

Tak obliczone średnie odchylenie długości uszeregowania wynoszące 0,81% wskazuje, że zwiększanie populacji początkowej powyżej 20 osobników nie przynosi istotnych korzyści w postaci skrócenia długości uszeregowania. Natomiast jak pokazały przeprowadzone testy, czas obliczeń dla przypadków z populacją początkową wynoszącą 40 osobników wydłuża się ponad dwukrotnie (z około 4 minut do 8 minut), w stosunku do czasu obliczeń z populacją początkową wynoszącą 20 osobników.

Najlepsze chromosomy uzyskane w przeprowadzonych testach zaprezentowano w tabeli 11.3.

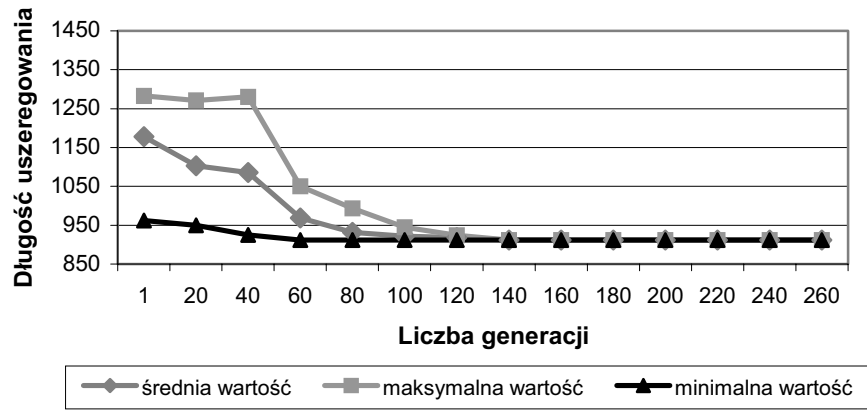
Tabela 11.3. Najlepsze chromosomy uzyskane w testach [opracowanie własne]

Numer eksperymentu	Najlepszy chromosom	Długość uszeregowania	Ilość osobników w populacji początkowej	Liczba generacji
1	6 9 2 11 8 15 14 1 5 10 5 7 11 2 13 3 15 12 3 12 3 8 8 12 8 2 10 4 14 2 5 1 12 14 12 4 6 13 13 10 3 11 14 5 14 9 15 11 15	864	40	80
2	4 8 2 15 5 10 12 4 3 11 8 12 1 5 14 3 13 10 7 11 14 12 9 14 2 3 10 2 11 14 3 8 16 9 12 8 12 6 5 13 1 15 14 5 13 15 11 2 15 6 16 16	905	40	40

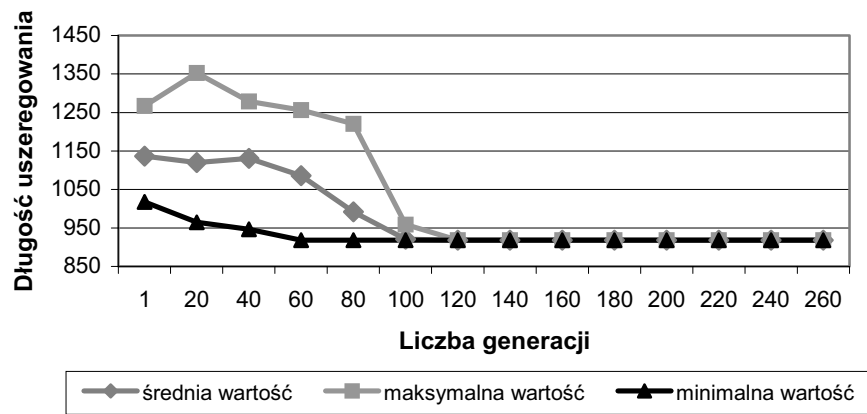
cd. tabeli 11.3

3	1 12 8 15 5 7 11 5 6 9 13 9 1 10 2 5 4 14 12 8 2 12 8 16 12 10 12 3 3 5 10 11 8 11 14 13 3 2 4 11 14 6 14 16 13 3 16 2 14	916	20	60
4	3 6 5 1 15 10 17 8 17 14 7 2 4 11 3 14 3 16 4 2 12 8 10 2 5 9 8 5 14 6 3 13 1 11 11 12 12 13 14 13 12 14 8 5 10 2 12 16 16 9 11 17 17	917	40	80
5	14 12 3 2 10 2 11 7 5 9 8 15 9 2 4 11 15 6 3 11 3 11 8 14 5 8 1 14 5 2 3 1 4 10 12 12 12 6 13 13 12 5 10 14 14 13 8	919	40	40
6	12 11 3 1 2 10 14 5 4 8 2 4 9 5 16 1 9 2 7 6 2 8 8 15 3 10 10 14 5 11 12 11 13 12 13 12 5 8 12 3 14 6 14 11 14 13 15 3	912	20	80
7	4 9 2 16 5 8 3 10 5 3 5 2 13 7 4 15 1 2 14 10 6 14 8 9 2 8 16 10 5 12 3 11 11 12 17 12 12 3 11 12 11 1 14 14 14 13 16 13 6 8 17	959	20	20
8	4 5 7 12 5 14 5 10 8 16 8 2 3 13 9 2 11 15 1 14 8 11 2 4 3 6 11 14 15 6 11 8 10 12 12 3 12 12 5 13 2 14 14 9 1 15 10 13 16 3	917	40	60
9	9 5 15 1 5 3 16 3 1 8 12 8 3 10 11 5 6 2 12 2 13 8 14 7 8 12 10 12 12 4 11 11 14 11 9 13 3 2 5 13 14 6 14 2 14 10 4 16 16	919	20	60
10	9 3 12 12 16 14 15 2 5 10 3 8 13 10 11 4 5 13 9 2 3 2 8 1 6 3 14 8 4 2 11 7 1 5 12 12 12 13 10 11 14 14 6 8 14 11 5 16 16	928	40	20

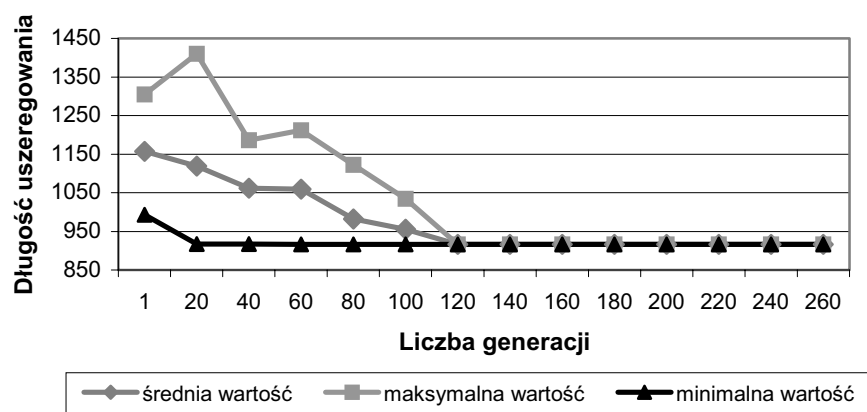
Rysunki od 11.1 do 11.10 prezentują graficznie wyniki uzyskane z wykorzystaniem algorytmu genetycznego dla przypadków planów produkcji z tablic od 1 do 10 umieszczonych w aneksie. Przedstawione na tych rysunkach wykresy pokazują wartości długości uszeregowania otrzymane dla różnej liczby generacji. Jak można zobaczyć na wykresach, rozwiązania stabilizują się na ogół po 100-140 generacjach.



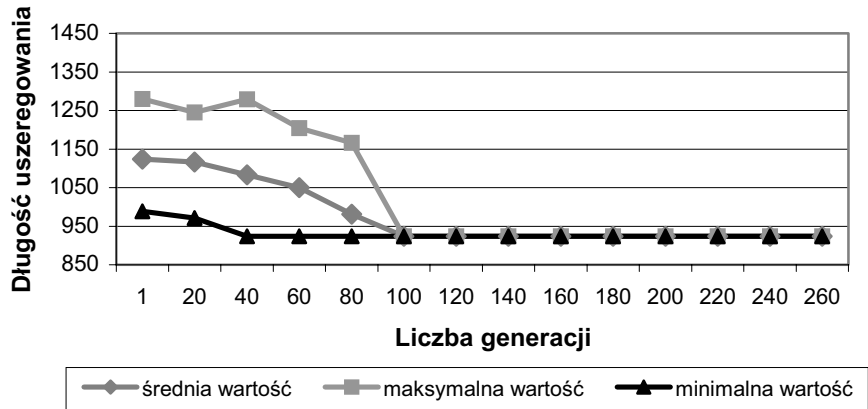
Rys. 11.1. Eksperyment dla planu produkcji 1 [opracowanie własne]



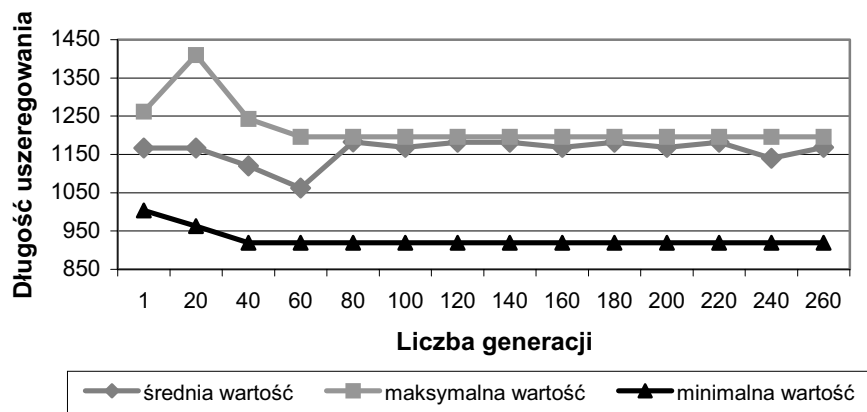
Rys. 11.2. Eksperyment dla planu produkcji 2 [opracowanie własne]



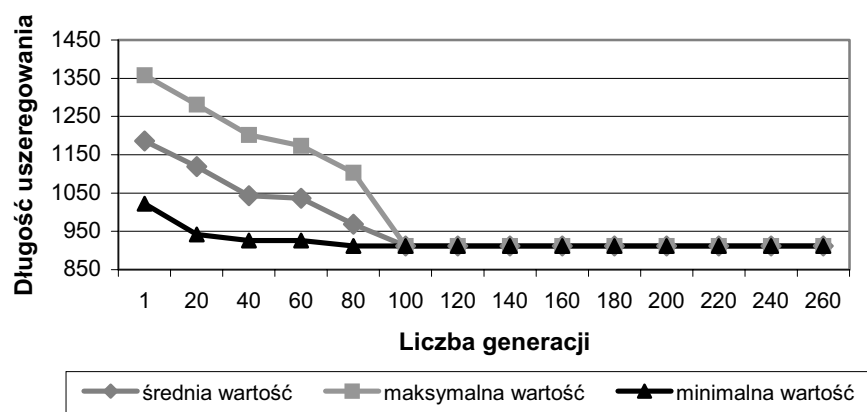
Rys. 11.3. Eksperyment dla planu produkcji 3 [opracowanie własne]



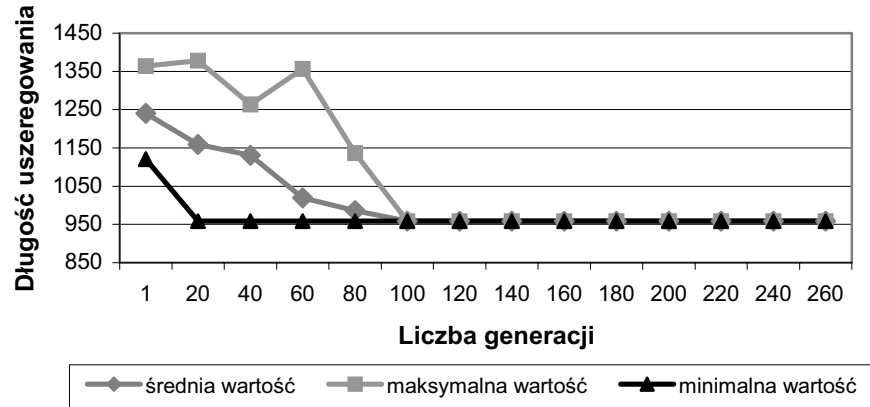
Rys. 11.4. Eksperyment dla planu produkcji 4 [opracowanie własne]



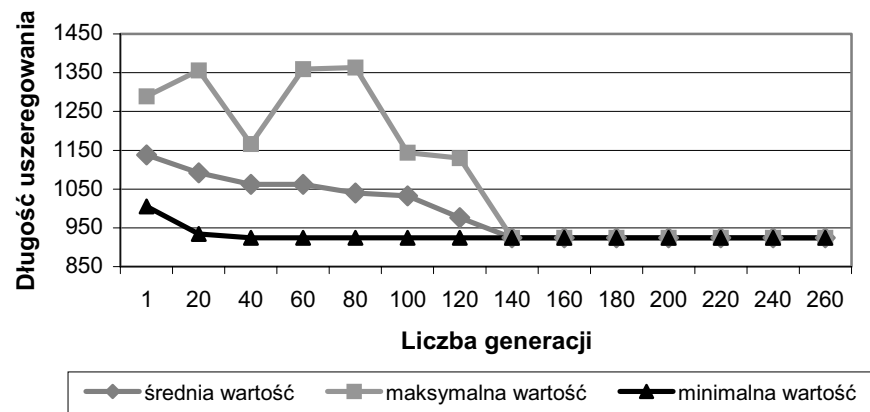
Rys. 11.5. Eksperyment dla planu produkcji 5 [opracowanie własne]



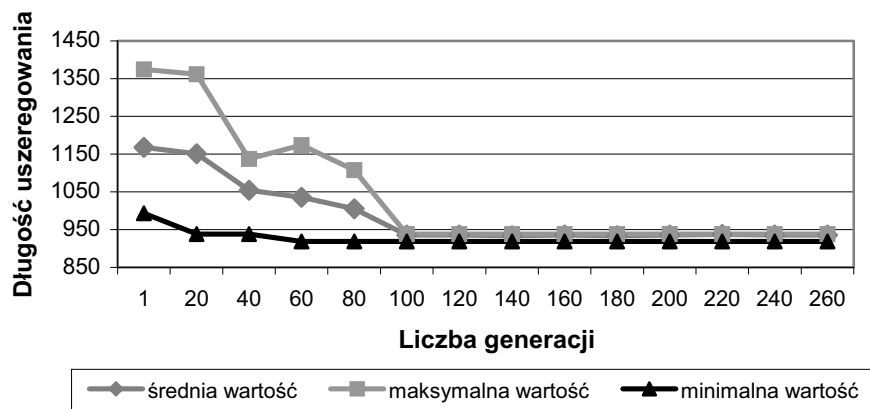
Rys. 11.6. Eksperyment dla planu produkcji 6 [opracowanie własne]



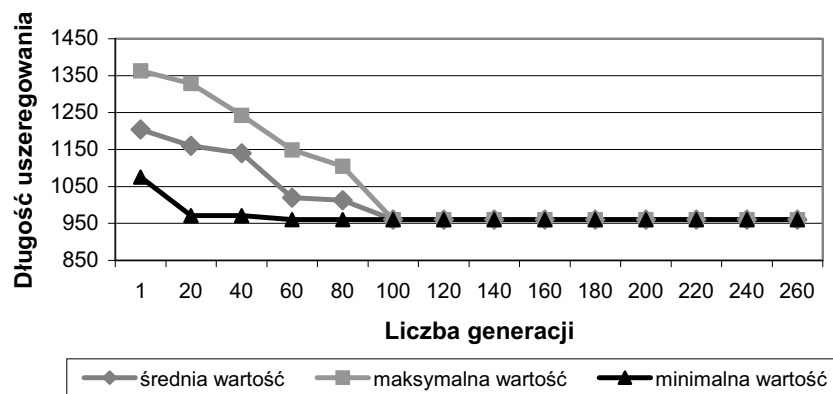
Rys. 11.7. Eksperyment dla planu produkcji 7 [opracowanie własne]



Rys. 11.8. Eksperyment dla planu produkcji 8 [opracowanie własne]



Rys. 11.9. Eksperyment dla planu produkcji 9 [opracowanie własne]



Rys. 11.10. Eksperyment dla planu produkcji 10 [opracowanie własne]

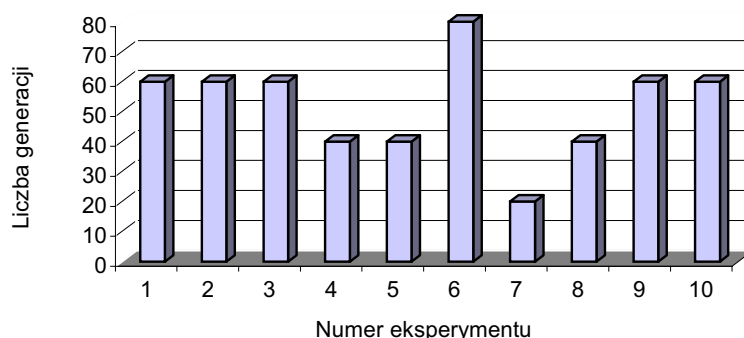
Analizując wyniki eksperymentów przedstawione na rysunkach od 11.1 do 11.10 można zauważyć, że minimalna wartość długości uszeregowania otrzymywana jest już po 20-80 generacjach. Zaprezentowana poniżej tabela zawiera wartości minimalne długości uszeregowania dla planów produkcji dla przypadku 3 (z tabel od 1 do 10 umieszczonych w aneksie), otrzymane dla różnej liczby generacji z populacją początkową równą 20.

Tabela 11.4. Wartości minimalne długości uszeregowania dla 10 testowych planów produkcji dla różnej liczby generacji z populacją początkową = 20

Liczba generacji	Numer eksperymentu									
	1	2	3	4	5	6	7	8	9	10
1	962	1018	993	989	1004	1022	1121	1005	993	1076
20	950	965	917	971	963	942	<b>959</b>	934	<b>938</b>	971
40	925	946	917	<b>924</b>	<b>919</b>	926	959	<b>924</b>	938	971
60	<b>912</b>	<b>918</b>	<b>916</b>	924	919	926	959	924	919	<b>960</b>
80	912	918	916	924	919	<b>912</b>	959	924	919	960
100	912	918	916	924	919	912	959	924	919	960
120	912	918	916	924	919	912	959	924	919	960
140	912	918	916	924	919	912	959	924	919	960
160	912	918	916	924	919	912	959	924	919	960
180	912	918	916	924	919	912	959	924	919	960
200	912	918	916	924	919	912	959	924	919	960
220	912	918	916	924	919	912	959	924	919	960
240	912	918	916	924	919	912	959	924	919	960
260	912	918	916	924	919	912	959	924	919	960

Dane zawarte w tabeli 11.4 zilustrowano na rysunku 11.11.





Rys. 11.11. Minimalna liczba generacji, dla której uzyskano najlepszy chromosom [opracowanie własne]

W oparciu o uzyskane powyżej wyniki można stwierdzić, że przyjmowanie – w rozważanych przypadkach – liczby generacji powyżej 80, dla populacji początkowej 20 nie prowadzi do poprawy harmonogramu.

Przeprowadzone testy dla prawdopodobieństwa mutacji 0,01 – 0,05 nie wykazały wpływu tego parametru na długość uszeregowania.

W celu weryfikacji wybranej do badań metody selekcji wykonano testy dla analizowanych powyżej 10 planów produkcji (dla przypadku 3) z zastosowaniem następujących metod selekcji:

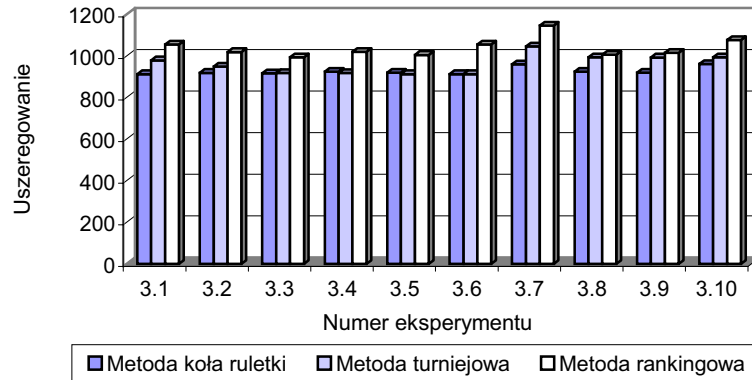
- koła ruletki,
- turniejowej,
- rankingowej.

Testy wykonano dla populacji początkowej równej 20 osobników, prawdopodobieństwa mutacji wynoszącego 0,05 i prawdopodobieństwa krzyżowania równego 1. Wyniki testów prezentuje tabela 11.5.

Tabela 11.5. Porównanie najlepszych rozwiązań otrzymanych dla różnych metod selekcji

Numer eksperymentu	Wartości najlepszych długości uszeregowania		
	metoda koła ruletki	metoda rankingowa	metoda turniejowa
3.1	<b>912</b>	1054	978
3.2	<b>918</b>	1018	948
3.3	<b>916</b>	993	917
3.4	924	1019	<b>917</b>
3.5	919	1004	<b>912</b>
3.6	<b>912</b>	1054	<b>912</b>
3.7	<b>959</b>	1145	1045
3.8	<b>924</b>	1005	993
3.9	<b>919</b>	1014	992
3.10	<b>960</b>	1076	993
Liczba najlepszych chromosomów	8	0	3
Suma długości uszeregowania	9263	10382	9607

Uzyskane powyższej wyniki zilustrowano graficznie na rysunku 11.12.



Rys. 11.12. Porównanie wartości najlepszych długości uszeregowania dla różnych metod selekcji

Wyniki przeprowadzonych testów dowodzą, że dokonano trafnego wyboru metody selekcji, ponieważ jak pokazano w tabeli 11.5 oraz na rysunku 11.12 najwięcej najlepszych chromosomów, z najkrótszą długością uszeregowania otrzymano z zastosowaniem metody koła ruletki.

Do rozpatrywanego algorytmu genetycznego zastosowano – poza operatorem PMX – także cztery inne rodzaje operatorów krzyżowania: *one-point crossover*, *two-point crossover*, *uniform crossover* oraz *order crossover*. Dla operatorów *one-point crossover*, *two-point crossover* oraz *uniform crossover* powstawały rozwiązania niedopuszczalne. Dla operatora *order crossover* otrzymano tylko jedno dopuszczalne rozwiązanie z długością uszeregowania 1179, dla populacji początkowej 4 i liczby generacji 1. Próby stosowania innych parametrów, jak np. przyjmowanie populacji początkowej równej 20 osobników i liczby generacji 10 nie doprowadziły w ciągu pięciu godzin do uzyskania jakichkolwiek wyników. Z uwagi na długi czas obliczeń zastosowanie tego operatora w algorytmie genetycznym jest niemożliwe. W przypadku awarii maszyn czy też innych zakłóceń w systemie produkcyjnym wielogodzinne oczekiwanie na wygenerowanie nowego harmonogramu jest zbyt kosztowne. Przeprowadzone testy dowodzą więc, że dokonano prawidłowego wyboru operatora krzyżowania.

Jak oceniono w trakcie badań najbardziej efektywne stosowanie proponowanego algorytmu genetycznego, w systemach wytwarzania o nieustabilizowanej produkcji, występuje dla populacji początkowej równej 20 osobników i liczbie generacji 80. Przyjmowanie w analizowanym systemie większej liczby osobników w populacji początkowej jest wyborem pomiędzy korzyścią z około 0,81% skróceniem długości uszeregowania a dwukrotnym wydłużeniem czasu obliczeń. Ostateczne decyzje dotyczące wielkości populacji początkowej mogą być podejmowane przez użytkownika, ponieważ są związane z rosnącymi kosztami użytkownika systemu.

### 11.3. Wyniki badań

Badaniami objęto 10 planów produkcji, zbudowanych z wykorzystaniem danych z dziesięciu systemów produkcyjnych opisanych w podrozdziale 11.1. Plany te zawarto w tabelach od 11 do 20, umieszczonych w aneksie niniejszej rozprawy. Eksperymenty wykonano z zastosowaniem zaproponowanego w podrozdziale 9.5 algorytmu genetycznego i parametrach ustalonych w podrozdziale 11.2. Eksperyment polegał na wygenerowaniu najbliższych optymalnym harmonogramów. Jak wspomniano wyżej eksperymenty przeprowadzono dla kryterium długości uszeregowania. W celu oceny efektywności algorytmów pod względem obliczeniowym mierzono czasy uzyskania wyniku.

Tabela 11.6 prezentuje charakterystyki planów produkcji (od 11 do 20 umieszczonych w aneksie), które stanowiły „wejście” do tworzenia harmonogramów z wykorzystaniem zaproponowanego w podrozdziale 9.5 algorytmu genetycznego.

Tabela 11.6. Charakterystyki badanych systemów produkcyjnych i przykładowych planów produkcji

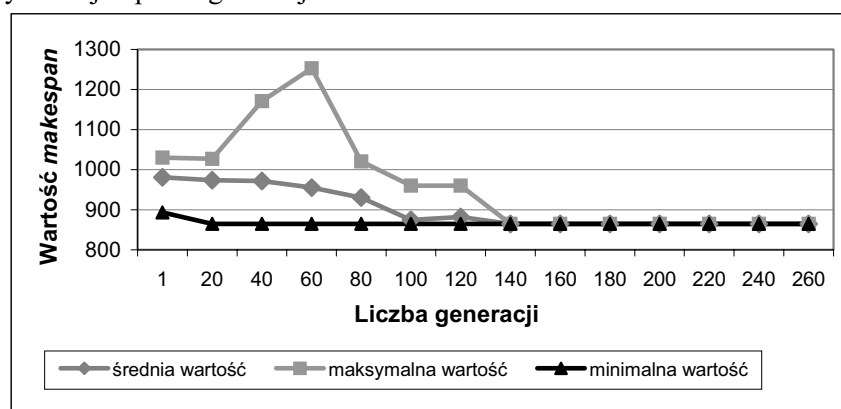
Numer eksperymentu	Asortyment produkcji	Charakterystyka planu produkcji	
		liczba zadań	liczba operacji
1	Części do maszyn budowlanych	11	70
2	Części do urządzeń elektrycznych	12	50
3	Części do maszyn rolniczych	14	45
4	Części do armatury hydraulicznej	7	42
5	Części do armatury wodociągowej	11	49
6	Części dla urządzeń leśnictwa	14	49
7	Części do form wtryskowych	11	43
8	Części do maszyn spożywczych	12	38
9	Narzędzia	17	83
10	Części do pojazdów i maszyn rolniczych	15	62

Wstępne badania wykazały, że system ekspercki może wspomagać planistę w generowaniu planów produkcji. Zbudowany w ramach prezentowanych tu badań system ekspercki wspomaga planistę w tworzeniu krótkookresowych planów produkcji i optymalizuje wykorzystanie zdolności produkcyjnych, a w szczególności zapobiega powstawaniu „wąskich gardeł”. W związku z tym plany produkcji, o charakterystykach jak w tabeli 11.6, wykorzystano jako dane wejściowe do algorytmu genetycznego.

Po zastosowaniu algorytmu genetycznego dla przypadku planu produkcji z tabeli 11 znajdującej się w aneksie, najlepszy harmonogram (tj. chromosom) był następujący:

2 6 8 3 1 2 6 2 4 1 2 4 9 6 7 3 1 3 2 10 3 3 1 1 7 8 4 8 5 5 5 7 6 5  
7 8 8 6 6 6 2 6 8 4 7 4 7 1 7 8 1 5 7 8 8 6 5 4 8 6 6 8 2 3 3 3 11  
10 10 11

Długość uszeregowania dla powyższego chromosomu wynosiła 865. Rysunek 11.13 ilustruje graficznie wyniki uzyskane z wykorzystaniem zaproponowanego algorytmu genetycznego dla eksperymentu 1 z tabeli 11.6. Prezentowany na tym rysunku wykres pokazuje wartości długości uszeregowania otrzymane dla różnej liczby generacji. Jak można zobaczyć na wykresie rozwiązania stabilizują się po 130 generacjach. Minimalną wartość długości uszeregowania otrzymano już po 20 generacjach.



Rys. 11.13. Wartości długości uszeregowania w zależności od liczby generacji dla eksperymentu 1 z tabeli 11.6 [opracowanie własne]

Do populacji początkowej w dalszych wykonywanych testach dla pozostałych planów produkcji wybierano 20 osobników, ponieważ jak wcześniej wykazano stosowanie większej liczby osobników w populacji początkowej nie gwarantowało uzyskania istotnie lepszych rezultatów. Dla wszystkich przypadków prawdopodobieństwo krzyżowania z operatorem PMX wynosiło 1, a prawdopodobieństwo mutacji 0,05.

Tabela 11.7 zawiera wyniki testów uzyskane z zastosowaniem zaproponowanego algorytmu genetycznego.

Tabela 11.7. Wyniki testów dla danych z różnych przedsiębiorstw

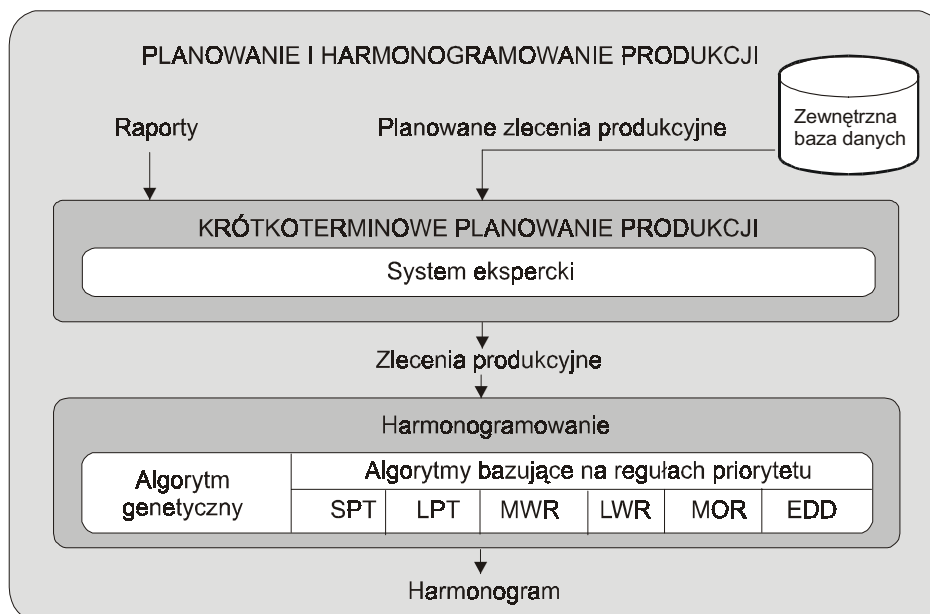
Numer eksperymentu	Długość uszeregowania	Minimalna liczba generacji, dla najlepszego uzyskanego chromosomu	Czas obliczeń (w sekundach)
1	865	20	11
2	1411	40	19
3	917	40	19
4	1403	20	10
5	1654	40	34
6	1379	40	30
7	1384	40	24
8	950	80	90
9	1416	20	16
10	1745	80	112

W oparciu o przeprowadzone badania testowe można stwierdzić, że zaproponowany algorytm genetyczny pozwala uzyskać rozwiązanie suboptymalne w bardzo krótkim czasie, nie przekraczającym 5 minut. Po 80 generacjach we wszystkich eksperymentach minimalne wartości długości uszeregowania pozostawały bez zmian. Rezultaty otrzymywane z zastosowaniem algorytmu genetycznego stabilizowały się w granicach od 80 do 160 generacji.

#### 11.4. Badania porównawcze algorytmu genetycznego z algorytmami bazującymi na kombinacji reguł priorytetu

W celu oceny skuteczności algorytmu genetycznego, do tworzenia harmonogramów zastosowano również algorytmy bazujące na różnych kombinacjach reguł priorytetu. Badania przeprowadzono z zastosowaniem jednej, tzw. reguły głównej: SPT lub LPT, MWR, LWR, MOR, LOR, EDD. W przypadku, kiedy zastosowano regułę główną i okazało się, że otrzymane wartości były jednakowe dla kilku operacji równocześnie, wówczas o wyborze jednej operacji spośród kilku rozstrzygano stosując kolejną regułę: wybierz operację z najniższym nadanym przez planistę wskaźnikiem priorytetu (tj. najwyższym priorytetem).

Badania z zastosowaniem reguł priorytetu przeprowadzono w opisanych wyżej 10 przedsiębiorstwach. Ogólną ideę badań porównawczych pokazano na rysunku 11.14.



Rys. 11.14. Schemat badania metod zarządzania produkcją [opracowanie własne]

Tworzenie harmonogramu z zastosowaniem algorytmów bazujących na kombinacji reguł priorytetu można zilustrować następującym przykładem.

Przyjmijmy, że plan produkcji dla komórki produkcyjnej jest jak w tabeli 11.8, a reguła główna to SPT. Wtedy kombinacja reguł zastosowana dla planu produkcji będzie następująca. Wybierz operację z najkrótszym czasem trwania operacji. Jeżeli jest więcej takich operacji, to wybierz spośród nich operację z najwyższym priorytetem (tj. najniższym wskaźnikiem priorytetu).

Tabela 11.8. Przykład planu produkcji

Zadanie	1			2			3		
Wskaźnik priorytetu	2			1			3		
Numer operacji	1	2	3	1	2	3	1	2	3
Czas operacji	2	5	3	4	3	2	2	3	4
Numer maszyny	1	2	1	3	1	2	2	3	3

Rozważmy teraz plan produkcji z tabeli 11.8. W pierwszym kroku dla powyższego planu produkcji podzbiór rozważanych operacji obejmuje: pierwszą operację zadania 1, pierwszą operację zadania 2 i pierwszą operację zadania 3. W tym podzbiorze najkrótszy czas trwania operacji mają dwie operacje: pierwsza operacja zadania 1 i pierwsza operacja zadania 3. Ponieważ pierwsza operacja zadania 1 ma najniższy wskaźnik priorytetu (a tym samym najwyższy priorytet), umieszczana jest na harmonogramie jako pierwsza. W drugim kroku podzbiór rozważanych operacji obejmuje: drugą operację zadania 1, pierwszą operację zadania 2 i pierwszą operację zadania 3. Teraz najkrótszy czas trwania ma tylko pierwsza operacja zadania 3 i dlatego zostaje ona umieszczona na harmonogramie. W trzecim kroku podzbiór rozważanych operacji obejmuje: drugą operację zadania 1, pierwszą operację zadania 2 i drugą operację zadania 3. Aktualnie najkrótszy czas trwania ma druga operacja zadania 3 i w związku z tym jest umieszczana na harmonogramie itd.

Porównanie jakości harmonogramów otrzymanych za pomocą rozpatrywanych algorytmów dla kryterium długości uszeregowania zawiera tabela 11.9, w której dla każdego algorytmu podano długość najlepszego uszeregowania, otrzymanego w wyniku testów.

Tabela 11.9. Porównanie długości uszeregowania otrzymanych z zastosowaniem różnych algorytmów [opracowanie własne]

Numer eksperymentu	Długość uszeregowania (w minutach)							
	GA	Algorytm bazujący na regułach priorytetu						
		SPT	LPT	MWR	LWR	MOR	LOR	EDD
1	865	1127	1004	916	1132	878	1066	944
2	1411	1987	1514	1558	1880	1571	1880	1803
3	917	1109	1087	1075	1106	1032	1350	1264
4	1403	2059	1696	1403	1996	1469	2059	1972
5	1654	2280	2185	1717	2413	1721	2189	2010
6	1379	1883	1717	1425	1917	1482	1888	1536
7	1384	1616	1604	1526	1470	1500	1890	1680
8	950	1057	1152	973	1281	966	1208	973
9	1416	1748	1883	1583	1730	1925	1707	1585
10	1745	1955	2070	1949	1955	2051	1990	1931

Na podstawie badań, których wyniki umieszczono w tabeli 11.9 można stwierdzić, że algorytm genetyczny zaproponowany przez autorkę jest narzędziem pozwalającym rozwiązywać praktyczne problemy optymalizacji. W przypadku rozważanych tu problemów harmonogramowania produkcji algorytm genetyczny pozwala uzyskać wyniki znacznie lepsze od uzyskanych metodami bazującymi na regułach priorytetu.

### 11.5. Analiza i ocena końcowa

Analizy zaproponowanej nowej metody zarządzania produkcją z zastosowaniem systemu eksperckiego i algorytmu ewolucyjnego można dokonać według kryteriów:

- ilościowych (mierzalnych) o charakterze technicznym i ekonomicznym,
- niewymiernych (niemierzalnych).

Źródłem efektów mierzalnych jest wyliczona wartość długości uszeregowania. Wpływa ona w sposób istotny na stan robót w toku, zgodnie ze wzorem:

$$Z_{tsr} = C \times P \quad (11.2)$$

gdzie:

- $Z_{tsr}$  – średni stan robót w toku,
- $C$  – długość uszeregowania (inaczej długość cyklu),
- $P$  – wielkość produkcji wykonywanej w jednostce czasu.

Efekty mierzalne wynikają więc przede wszystkim ze skrócenia cyklu produkcyjnego, co w rezultacie powoduje obniżenie wielkości produkcji w toku, a tym samym mniejsze zamrożenie środków obrotowych i obniżenie kosztów produkcji.

W tabeli 11.10 przedstawiono wyniki analizy porównawczej wartości długości uszeregowania otrzymanych w wyniku zastosowania algorytmu genetycznego i algorytmów bazujących na kombinacji reguł priorytetu. Do analizy zastosowano następujące formuły:

$$R_{ei} = \frac{(vm_{rule} - vm_{ga})}{vm_{ga}} \cdot 100\% \quad (11.3)$$

$$R_{es} = \frac{\sum_{i=1}^n R_{ei}}{n} \quad (11.4)$$

gdzie:

- $R_{ei}$  – odchylenie wartości długości uszeregowania, otrzymane w wyniku zastosowania kombinacji reguł (bazującej na regule głównej  $i$  oraz wskaźniku priorytetu), od wartości długości uszeregowania otrzymanej przez algorytm genetyczny (w procentach),
- $R_{es}$  – średnie odchylenie dla danego przypadku,
- $R_e$  – średnia arytmetyczna odchyleń dla wszystkich analizowanych przypadków,
- $i$  – reguła główna,
- $vm_{rule}$  – wartość długości uszeregowania otrzymana w wyniku zastosowania kombinacji reguł,
- $vm_{ga}$  – wartość długości uszeregowania otrzymana w wyniku zastosowania algorytmu genetycznego.

Wyniki obliczeń uzyskane w wyniku zastosowania powyższej formuły i danych z tabeli 11.9 przedstawiono w tabeli 11.10.

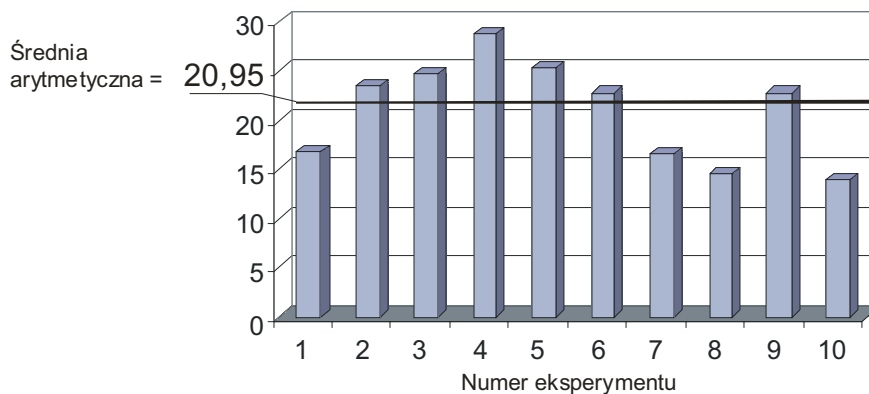
Tabela 11.10. Analiza odchyleń wartości długości uszeregowania [opracowanie własne]

Numer eksperymentu	Reguła główna w kombinacji, na której bazuje algorytm							$R_{es}$ [%]
	SPT	LPT	MWR	LWR	MOR	LOR	EDD	
	$R_{ei}$ [%]							
1	30,29	16,07	5,90	30,87	1,50	23,24	9,13	16,71
2	40,82	7,30	10,42	33,24	11,34	33,24	27,78	23,45
3	20,94	18,54	17,23	20,61	12,54	47,22	37,84	24,99
4	46,76	20,88	0	42,27	4,70	46,76	40,56	28,85
5	37,85	32,10	3,81	45,89	4,05	32,35	21,52	25,37
6	36,55	24,51	3,34	39,01	7,47	36,91	11,39	22,74
7	16,76	15,90	10,26	6,21	8,38	36,56	21,39	16,49
8	11,26	21,26	2,42	34,84	1,68	27,16	2,42	14,43
9	23,45	32,98	11,79	22,18	35,95	20,55	11,94	22,69
10	12,03	18,62	11,70	12,03	17,54	14,04	10,66	13,80
Średnia arytmetyczna [%]	27,67	20,82	7,69	28,72	10,52	31,80	19,46	20,95



Uzyskane w tabeli wyniki zilustrowano graficznie na rysunku 11.15.

Odchylenie długości  
uszeregowania  $R_{es}$  w [%]



Rys. 11.15. Porównanie średnich odchyleń długości uszeregowania  $R_{es}$  dla różnych przypadków [opracowanie własne]

W oparciu o wyniki z tabeli 11.10 i rysunku 11.15 można stwierdzić, że zaproponowany algorytm genetyczny pozwala uzyskać wyniki średnio o 20,95% lepsze niż zastosowane tu algorytmy bazujące na kombinacji reguł priorytetu.

Dodatkowo w ramach badań zweryfikowano efektywność różnych kombinacji reguł priorytetu. W badanych tu przypadkach najlepsze efekty uzyskano z zastosowaniem kombinacji reguł bazującej na regule głównej MWR. Dla tej reguły średnie odchylenie względne wartości długości uszeregowania od wartości długości uszeregowania dla algorytmu genetycznego wynosi 7,69%. Nieco gorsze, wynoszące 10,52% otrzymano z zastosowaniem kombinacji reguł bazującej na regule głównej MOR.

Inne efekty ekonomiczne trudniej mierzalne to:

- skrócenie czasu tworzenia harmonogramu (w rozpatrywanych przypadkach do około 5 minut) i w rezultacie szybkie reagowanie na zakłócenia,
- likwidacja „wąskich gardeł”,
- lepsze dociążenie maszyn.

Efekty niemierzalne to:

- skrócenie okresu realizacji zamówienia i satysfakcja klienta, co powoduje wzrost konkurencyjności na rynku,
- usprawnienie pracy planistów i dyspozytorów.

Zaproponowany w niniejszej rozprawie nowy algorytm nie wymaga, jak pokazały przeprowadzone eksperymenty, dużej pojemności pamięci oraz długiego czasu obliczeń, dlatego jest celowe jego stosowanie wtedy, gdy wynik

obliczeń trzeba otrzymać szybko przy stosunkowo niskich kosztach. Zaletą algorytmu genetycznego jest także jego uniwersalny charakter. Proponowany w rozprawie nowy algorytm genetyczny może być wykorzystany nie tylko do harmonogramowania prac na maszynach, ale także do harmonogramowania pracy ludzi.

Przedstawiona nowa metoda zarządzania produkcją w ogniwie sieci dostaw może znaleźć zastosowanie przede wszystkim w małych i średnich przedsiębiorstwach, gdzie najczęściej występuje szybkozmienny asortyment produkcji i outsourcing.

## 12. Wnioski

Na podstawie badań, których wyniki przedstawiono w niniejszej rozprawie, można stwierdzić, że jest możliwe zastosowanie sztucznej inteligencji do zarządzania produkcją w przedsiębiorstwie przemysłowym. Celem badań prowadzonych przez autorkę było zaimplementowanie systemu eksperckiego i algorytmu genetycznego do zarządzania produkcją w ogniwie sieci dostaw o niestabilizowanej produkcji z outsourcingiem. Oznaczało to konieczność prowadzenia, mających interdyscyplinarny charakter, wielostronnych badań i eksperymentów, uwzględniających ograniczenia wynikające ze struktury i sposobu funkcjonowania systemów produkcyjnych. Przegląd problematyki dotyczącej zarządzania produkcją wskazał luki metodologiczne związane z dotychczasowymi propozycjami metod i procedur oraz ich stosowaniem w przedsiębiorstwie przemysłowym, postrzeganym jako ogniwo sieci dostaw. W związku z tym w rozprawie przedstawiono propozycję spójnej metodyki zarządzania produkcją w obróbkowym systemie produkcyjnym o niestabilizowanej produkcji, w którym możliwe jest stosowanie alternatywnych marszrut produkcyjnych i outsourcingu. Poniżej przedstawione jest krótkie, syntetyczne podsumowanie ważniejszych wyników zrealizowanych prac badawczych, jak również wskazane są niektóre pożądane kierunki dalszych badań.

Najważniejsze wyniki szczegółowe rozprawy są następujące:

1. Wraz z nowymi tendencjami w zarządzaniu produkcją powstała potrzeba opracowania nowych podstaw metodologicznych zarządzania produkcją w ogniwie sieci dostaw. W szczególności, jak wykazano w niniejszej rozprawie (rozdział 9) zaistniała potrzeba uściślenia definiowania pojęcia sieci dostaw, ponieważ aktualne modele sieci dostaw proponowane w literaturze przedmiotu tylko wybiórczo odwzorowują rzeczywiste procesy w sieci dostaw.
2. Z metodologicznego i praktycznego punktu widzenia istotnym rozszerzeniem podejść do problemu zarządzania produkcją było uwzględnienie przez autorkę zadań podzielonych i outsourcingu. Optymalizacja dotyczyła więc nie tylko funkcjonowania poszczególnych ogniw, ale także współdziałania pomiędzy nimi.
3. Korzystając z rozwiązań przedstawionych w rozdziałach 6 i 8, w rozdziale 9 zaproponowano nową metodę zarządzania niestabilizowaną produkcją w ogniwie sieci dostaw z zastosowaniem sztucznej inteligencji. Tworząc nowe metody zarządzania produkcją zwrócono uwagę na to, że z matematycznego punktu widzenia używane powszechnie, a więc i w tej rozprawie pojęcie *sieci dostaw* nie jest tożsame z pojęciem *sieci* rozumianym w matematyce jako skierowany, skończony graf bez pętli i łuków równoległych (por. [23]). W związku z aktualnymi tendencjami rozwoju sieci dostaw, dotyczącymi wyrównywania obciążeń zasobów, do modelowania sieci zaproponowano graf.
4. W związku z powyższym, aktualnie znane metody zarządzania produkcją oparte na modelach sieciowych nie odpowiadają tendencjom rozwojowym

w zarządzaniu *siecią dostaw*. Powstała więc potrzeba opracowania nowych metod zarządzania produkcją w ogniwie sieci dostaw, ponieważ znane i dotychczas stosowane metody nie znajdują zastosowania w wielu produkcyjnych obiektach rzeczywistych, szczególnie w tych, które produkują różnorodne wyroby.

5. Opracowano oryginalną metodę krótkoterminowego planowania produkcji opartą na koncepcji „wąskich gardeł” z zastosowaniem stworzonego przez autorkę systemu eksperckiego. Metoda ta może być zastosowana w ogniwie sieci dostaw o niestabilizowanej produkcji. Zaspokojenie zmiennych oczekiwań klienta na coraz to nowsze, lepsze produkty, w możliwie najkrótszym czasie i przy możliwie najniższych kosztach było główną przesłanką tworzenia tego systemu.
6. W opracowanej metodzie zarządzania produkcją z zastosowaniem systemu eksperckiego nie występuje – przyjmowane zazwyczaj podczas optymalizacji planowania i harmonogramowania produkcji – ograniczenie niepodzielności zadań systemu produkcyjnego, co pozwala uwzględniać outsourcing wytwarzania i postrzegać przedsiębiorstwo jako ogniwo sieci dostaw.
7. Opracowana metoda planowania produkcji uwzględnia outsourcing wytwarzania i alternatywne procesy technologiczne w wyniku zastosowania nowej koncepcji systemu eksperckiego z zewnętrzną bazą danych – stworzoną przez autorkę rozprawy – do generowania zleceń produkcyjnych. Przy budowie tego systemu wzięto pod uwagę zarówno dotrzymywanie terminów zleceń klienta, jak i wyrównywanie obciążeń maszyn.
8. W zewnętrznej bazie danych zaproponowano nową koncepcję zarządzania planami procesów technologicznych.
9. Opracowano nowy, efektywny algorytm genetyczny dla problemu szeregowania zadań w obróbkowym systemie produkcyjnym o szybkozmiennym asortymencie produkcji z kryterium długości uszeregowania. Zebrano dane z dziesięciu rzeczywistych przedsiębiorstw przemysłowych do przeprowadzenia eksperymentów obliczeniowych. W pracy zawarto wyniki eksperymentów obliczeniowych uzyskane dla niezależnych zadań systemów produkcyjnych. Dla tych danych porównano efektywność zaimplementowanych algorytmów: genetycznego i bazującego na regułach priorytetu.
10. Wyniki teoretyczne oraz rezultaty eksperymentów obliczeniowych potwierdzają efektywność przyjętej przez autorkę metody do analizy i rozwiązywania problemów szeregowania zadań w obróbkowych systemach wytwórczych o szybkozmiennym asortymencie produkcji z outsourcingiem wytwarzania. W oparciu o wyniki badań można stwierdzić, że zaproponowany nowy algorytm genetyczny pozwala uzyskać wyniki średnio o 20,95% lepsze niż algorytmy bazujące na kombinacji reguł priorytetu. Tak duże skrócenie cykli produkcyjnych powoduje obniżenie wielkości produkcji w toku, a tym samym mniejsze zamrożenie środków obrotowych i znaczące obniżenie kosztów produkcji.
11. Uzupełniono wiedzę dotyczącą modeli szeregowania zadań opartych na regułach priorytetu o nowe wyniki. W ramach badań zweryfikowano efektywność

różnych kombinacji reguł priorytetu. W badanych przypadkach najlepsze efekty uzyskano dla kombinacji reguł priorytetu bazujących na regule głównej MWR. Średnie odchylenie względne długości uszeregowania w tym przypadku od długości uszeregowania dla algorytmu genetycznego wynosi 7,69%. Niewiele gorsze średnie odchylenie względne, tj. 10,52% otrzymano z zastosowaniem kombinacji reguł bazującej na regule głównej MOR.

12. Efekty ekonomiczne – zaproponowanych w rozprawie – metod zarządzania produkcją, które są trudniej mierzalne, to: skrócenie czasu tworzenia harmonogramu (w rozpatrywanych przypadkach najwyżej do 5 minut) i w rezultacie szybkie reagowanie na zakłócenia. Inne efekty to: likwidacja „wąskich gardeł”, lepsze obciążenie maszyn, skrócenie czasu realizacji zamówienia, usprawnienie pracy planistów i dyspozytorów.
13. Przedstawiono cykl życia systemu zarządzania produkcją w ogniwie sieci dostaw z zastosowaniem systemu eksperckiego i algorytmu genetycznego. Sformułowano zasady budowy i stosowania takiego systemu.
14. Uzyskane wyniki pozwalają przypuszczać, że zastosowanie opracowanej metodyki do badania innych problemów dyskretnych będzie równie efektywne jak w przypadku szeregowania zadań w obróbkowych systemach wytwarzania.

Trudności związane ze złożonością obliczeniową metod poszukujących dokładnych rozwiązań problemów harmonogramowania stanowią ważną przyczynę rezygnowania z tych metod. W związku z tym mają one raczej wartość teoretyczną niż praktyczną. Natomiast zaproponowane przez autorkę niniejszej rozprawy metody umożliwiają zarządzanie nieustabilizowaną produkcją w krótkich horyzontach planistycznych (1 do 3 dni). Proponowane metody umożliwiają wyrównywanie obciążeń i optymalizowanie długości uszeregowania systemu produkcyjnego typu *general job-shop*, w którym nie ma żadnych ograniczeń porządku technologicznego i w którym występują alternatywne marszruty technologiczne oraz outsourcing. W praktyce dyspozytor do planowania i harmonogramowania produkcji w takich systemach stosuje metody heurystyczne oparte na własnym doświadczeniu lub reguły priorytetu. W metodyce zaproponowanej w niniejszej rozprawie problem planowania jest rozwiązany z zastosowaniem systemu eksperckiego, a harmonogramowanie realizowane jest z zastosowaniem algorytmu genetycznego. Należy podkreślić, że klasa problemów szeregowania, dla której w tej pracy zaproponowano algorytm genetyczny obejmuje większość zastosowań praktycznych. Zaproponowany przez autorkę algorytm lokalnego przeszukiwania stanowi uniwersalne narzędzie rozwiązywania dużej klasy problemów dyskretnych. Proponowany w rozprawie nowy algorytm genetyczny może być wykorzystany nie tylko do harmonogramowania prac na maszynach, ale także do harmonogramowania pracy ludzi.

Na podstawie przeprowadzonych testów można stwierdzić, że zbudowany przez autorkę algorytm genetyczny jest bardzo efektywnym narzędziem pozwalającym rozwiązywać praktyczny problem optymalizacji harmonogramu. W celu oceny efektywności algorytmu genetycznego, do tworzenia harmonogramów zastosowano również algorytmy bazujące na różnych kombinacjach

reguł priorytetu. W oparciu o wyniki badań można stwierdzić, że zaproponowany algorytm genetyczny pozwala uzyskać wyniki znacznie lepsze, tj. średnio o 20,95% niż algorytmy bazujące na kombinacji najpopularniejszych reguł priorytetu. Efekty mierzalne wynikają więc przede wszystkim z dużego skrócenia cyklu produkcyjnego, co w rezultacie powoduje obniżenie wielkości produkcji w toku, a tym samym mniejsze zamrożenie środków obrotowych i obniżenie kosztów produkcji.

Dyskusja nad praktycznymi aspektami realizacji procesu tworzenia kombinacji systemu eksperckiego i algorytmu genetycznego do planowania i harmonogramowania produkcji (przeprowadzona w rozdziale 10) wskazuje, że sukces rozwiązania danego problemu zależy nie tylko od wyboru specyficznego algorytmu, ale także od właściwej realizacji innych etapów całego procesu.

Przedstawione nowe metody zarządzania produkcją w ogniwie sieci dostaw mogą znaleźć zastosowanie przede wszystkim w małych i średnich przedsiębiorstwach, gdzie najczęściej występuje szybkozmienny asortyment produkcji i outsourcing.

Przedstawione rezultaty badań, dotyczące określonych przypadków z rzeczywistych przedsiębiorstw wskazują na możliwości formułowania i rozwiązywania w ramach przedstawionego tu ogólnego podejścia problemów bardziej złożonych, które mogą być przedmiotem dalszych prac z tego zakresu. W szczególności powinny zostać podjęte badania dotyczące zastosowań systemów eksperckich i algorytmów genetycznych na wyższych szczeblach zarządzania produkcją, ponieważ korzystną cechą zaproponowanego tu systemu ekspertowego i algorytmu genetycznego jest uniwersalność. Rozwój hybrydowych systemów zarządzania z zastosowaniem sztucznej inteligencji można też ukierunkować na:

- wspomaganie decyzji dotyczących transportu,
- organizację i zarządzanie zautomatyzowanymi magazynami,
- zarządzanie relacjami z klientami.

Ponadto analiza dotychczasowych badań dotyczących optymalizacji sieci dostaw wskazuje na kierunki prowadzenia dalszych prac badawczych w zakresie zastosowań algorytmów genetycznych do optymalizacji, na przykład takie jak:

- rozmieszczenie obiektów w sieci dostaw,
- rozmieszczenie i grupowanie maszyn w ogniwie sieci dostaw,
- projektowanie montażu,
- klasyfikowanie rodzin części,
- harmonogramowanie pracy ludzi w realizacji czynności logistycznych.

Interesującym obszarem badań, który zarysował się podczas realizacji tej rozprawy, jest konstrukcja efektywnych algorytmów lokalnego przeszukiwania dla problemów dyskretno-ciągłych.

Jak pokazały zaprezentowane w niniejszej rozprawie badania, zarządzanie produkcją przemysłową ze sztuczną inteligencją może stać się dla polskiej gospodarki szansą na sprostanie warunkom dyktowanym przez partnerów na rynkach europejskich i światowych.

## Literatura

- [1] Aarts E.H.L., Laarhoven P.J.M., Lenstra J.K., Ulder N.L.J., 1994. *A computational study of local search shop scheduling*. ORSA Journal on Computing 6, 118-125.
- [2] Abt S., 1998. *Zarządzanie logistyczne w przedsiębiorstwie*. Polskie Wydawnictwo Ekonomiczne, Warszawa.
- [3] Adachi T., Talavage J., Moodie C., 1989. *A rule based control method for a multi-lop production system*. Artificial Intelligence in Engineering 4(3), 115-125.
- [4] Adams J., Balas E., Zawack D., 1988. *The shifting bottleneck procedure for job-shop scheduling*. International Journal Flexible Manufacturing Systems 34, 391-401.
- [5] Appels S., Ayel J., Laurent J.P., Lanusse A., Panet B.P., 1989. *Controller module for coordination of supervision activities on production management systems*, [in:] Proceedings of the 5th CIM Europe Conference "Computer Integrated Manufacturing", Springer, Berlin, 3-14.
- [6] Applegate D., Cook W., 1991. *A computational study of the job-shop scheduling problem*. ORSA Journal on Computing 3, 149-156.
- [7] Bak D., 1992, *Rozwój i rola logistyki w Wielkiej Brytanii*. Problemy Magazynowania i Transportu, Zeszyt Specjalny.
- [8] Baker K.R., 1975. *A comparative study of flow shop algorithms*. Operations Research 23, 62-73.
- [9] Baker K.R., 1974. *Introduction to sequencing and scheduling*. J. Wiley and Sons, New York.
- [10] Balas E., Lenstra J.K., Vazacopoulos A., 1995. *One machine scheduling with delayed precedence constraints*. Management Science 41, 94-109.
- [11] Banaszak Z., Jampolski L., 1991. *Komputerowo wspomagane modelowanie elastycznych systemów produkcyjnych*. Wydawnictwa Naukowo-Techniczne, Warszawa.
- [12] Barker R., Longman C., 1992. *CASE Method. Function and Process Modeling*. Addison-Wesley Publishing Company, Reading, Massachusetts.
- [13] Barker R., 1991. *CASE Method. Tasks and Deliverables*. Addison-Wesley Publishing Company. Reading, Massachusetts.
- [14] Barker J.R., McMahon G.B., 1985. *Scheduling the general job-shop*. Management Science 31, 594-598.
- [15] Bazewicz M., 1994. *Metody i techniki reprezentacji wiedzy w projektowaniu systemów*. Politechnika Wroclawska, Wroclaw.
- [16] Beamon B.M., 1998. *Supply chain design and analysis: Models and methods*. International Journal of Production Economics 55, 281-294.
- [17] Bean J., 1994. *Genetic Algorithms and random keys for sequencing and optimization*. ORSA Journal on Computing 5, 154-160.
- [18] Bierwirth C., 1995. *A generalized permutation approach to job shop scheduling with genetic algorithms*. OR Spectrum 17, 87-92.

- [19] Bierwirth C., Mattfeld D.C., 1999. *Production scheduling and rescheduling with genetic algorithms*. Evolutionary Computation 7, 1-17.
- [20] Bierwirth C., Mattfeld D.C., Kopfer H., 1996. *On permutation representation for scheduling problem*, [in:] Proceedings of Parallel Problem Solving from Nature, H.M. Voigt et al. editors. Springer, Berlin, 310-318.
- [21] Blackstone J.H., Phillips D.T., Hogg G.L., 1982. *A state of the art. Survey of dispatching rules for manufacturing job shop operations*. International Journal of Production Research 20, 27-45.
- [22] Blumenfeld D.E., Hali R.W., Jordan W.C., 1985. *Trade-off between freight expediting and safety stock inventory costs*. Journal of Business Logistics 6(1), 9-100.
- [23] Błażewicz J., Cellary W., Słowiński R., Węglarz J., 1981. *Algorytmy sterowania rozdziałem zadań i zasobów w kompleksie operacji*. Politechnika Poznańska, Poznań.
- [24] Błażewicz J., Cellary W., Słowiński R., Węglarz J., 1983. *Badania operacyjne dla informatyków*. Wydawnictwa Naukowo-Techniczne, Warszawa.
- [25] Błażewicz J., Domschke W., Pesch E., 1996. *The job shop scheduling problem: Conventional and new solution techniques*. European Journal of Operational Research 93, 1-33.
- [26] Bongaerts L., Monostori L., McFarlane D., Kádár B., 2000. *Hierarchy in distributed shop floor control*. Computers in Industry 43, 123-137.
- [27] Borowiecki R., Kwieciński M. (red. nauk.), 2004. *Informacja i wiedza w zintegrowanym systemie zarządzania*. Kantor Wydawniczy Zakamycze, Zakamycze.
- [28] Brucker P., 1994. *A polynomial algorithm for the two machine job-shop scheduling problem with a fixed number of jobs*. OR Spectrum 16, 5-7.
- [29] Brucker P., Jurisch B., 1993. *A new lower bound for the job-shop scheduling problem*. European Journal of Operational Research 64, 156-167.
- [30] Bruno G., Elia A., Laface P., 1988. *A rule-based systems to schedule production*, [in:] Proceedings of Conference Artificial Intelligence in Industry. Springer-Verlag, 452-465.
- [31] Brzeziński M. (red. nauk.), 2002. *Organizacja i sterowanie produkcją. Projektowanie systemów produkcyjnych i procesów sterowania produkcją*. Placet, Warszawa.
- [32] Bubnicki Z., 1990. *Wstęp do systemów ekspertowych*. Państwowe Wydawnictwo Naukowe, Warszawa.
- [33] Burbidge J.L., 1962. *The principles of production control*. Mc Donald and Evans Ltd., London.
- [34] Byrne M.D., Baker M.A., 1999. *Production planning using a hybrid simulation – analytical approach*. International Journal of Production Economics 59, 305-311.



- [35] Cellary W., Strykowski S., Remasz L., Toboła M., 2002. *Ewolucja łańcuchów dostaw w gospodarce elektronicznej*, [w:] *Logistyka on-line. Zarządzanie łańcuchem dostaw w globalnej gospodarce elektronicznej*, red. nauk. K. Rutkowski. Polskie Towarzystwo Ekonomiczne, Warszawa, 45-47.
- [36] Chen C.L., Vempati V.S., Aljaber N., 1995. *An application of genetic algorithms for flow shop problem*. European Journal of Operational Research 80, 389-396.
- [37] Cheng R., Gen M., Tsujimura Y., 1996. *A tutorial survey of job-shop scheduling problems using genetic algorithms. Part 1. Representation*. Computers and Industrial Engineering 4, 983-997.
- [38] Cheng R., Gen M., Tsujimura Y., 1999. *A tutorial survey of job-shop scheduling problems using genetic algorithms. Part II. Hybrid genetic search strategies*. Computers and Industrial Engineering 36, 343-364.
- [39] Childerhouse P., Towill D.R., 2003. *Simplified material flow holds the key to supply chain integration*. Omega. The International Journal of Management Science 31, 17-27.
- [40] Chiu C.C., Cook D.F., Kao J.L., Chou Y.Ch., 1997. *Combining a neural network and a rule based expert system for short-term load forecasting*. Computers and Industrial Engineering 4, 787-797.
- [41] Chlebus E., 2000. *Techniki komputerowe CAX w inżynierii produkcji*. Wydawnictwa Naukowo-Techniczne, Warszawa.
- [42] Ciesielski M., 2004. *Podstawy wiedzy logistycznej*. Wydawnictwa Akademii Ekonomicznej, Poznań.
- [43] Cleveland G.A., Smith S. F., 1989. *Using genetic algorithms to schedule flow shop releases*, [in:] Proceedings of the Third International Conference on Genetic Algorithms Applications, J.D. Schaffer (ed.), Morgan Kaufmann, San Mateo, CA, 160-169.
- [44] Coffman E.G.Jr. (red. nauk.), 1980. *Teoria szeregowania zadań*. Wydawnictwa Naukowo-Techniczne, Warszawa.
- [45] Cohen M.A., Lee H.L., 1989. *Resource deployment analysis of global manufacturing and distribution networks*. Journal of Manufacturing and Operations Management 2, 81-104.
- [46] Cohen M.A., Lee H.L., 1988. *Strategic analysis of integrated production – distribution systems: models and methods*. Operations Research 36(2), 216-228.
- [47] Conway R.W., Maxwell W. L., Miller L.W., 1967. *Theory of scheduling*. Addison-Wesley, Reading, Massachusetts.
- [48] Cooper M.C., Lambert D.M., Pagh J.D., 1997. *Supply chain management: More than a new name for logistics*. The International Journal of Logistics Management 8(1), 1-13.
- [49] Croce D.F., Tadei R., Volta G., 1995. *A genetic algorithm for the job shop problem*. Complex Systems 22, 15-24.
- [50] Daghli C.H., Sittisathanchai S., 1993. *Genetic neuro-scheduler for job-shop scheduling*. Computers and Industrial Engineering 25, 267-270.

- [51] Das C., 1975. *Supply and redistribution rules for two-location inventory systems: One-period analysis*. Management Science 21(7), 765-776.
- [52] Davis L., 1985. *Job shop scheduling with genetic algorithms*, [in:] Proceedings of the First International Conference on Genetic Algorithms and Their Applications, J. Grefenstette (ed.), Lawrence Erlbaum Associates, Hillsdale, NJ, 136-140.
- [53] Doumeingts G., Ducq Y., Vallespir B., Kleinhans S., 2000. *Production management and enterprise modelling*. Computers in Industry 42, 245-263.
- [54] Dorndorf U., Pesch E., 1995. *Evolution based learning in a job shop scheduling environment*. Complex Systems 22, 25-40.
- [55] Durlik I., 1996. *Inżynieria zarządzania. Strategia i projektowanie systemów produkcyjnych. Część I*. Agencja Wydawniczo-Poligraficzna Placet, Warszawa.
- [56] Durlik I., 1996. *Inżynieria zarządzania. Strategia i projektowanie systemów produkcyjnych. Część II*. Agencja Wydawniczo-Poligraficzna Placet, Warszawa.
- [57] Durlik I., 1998. *Restrukturyzacja procesów gospodarczych. Reengineering. Teoria i Praktyka. Business Process Reengineering w warunkach High-Technology*. Agencja Wydawniczo-Poligraficzna Placet, Warszawa.
- [58] Dwiliński L., 2002. *Zarządzanie produkcją*. Politechnika Warszawska, Warszawa.
- [59] Erengüc S.S., Simpson N.C., Vakharia A.J., 1999. *Integrated production/distribution planning in supply chains: An invited review*. European Journal of Operational Research 115, 219-236.
- [60] Evans G.N., Naim M.M., Towill D.R., 1998. *Application of a simulation methodology to the redesign of a logistical control system*. International Journal of Production Economics 56, 157-168.
- [61] Fertsch M., 1993. *Metodyka optymalizacji w adaptacyjnym projektowaniu systemów produkcyjnych*. Rozprawy Nr 288, Politechnika Poznańska, Poznań.
- [62] Fertsch M. (red. nauk.), 2003. *Logistyka produkcji*. Instytut Logistyki i Magazynowania, Poznań.
- [63] Fisher M., Hammond J., Obermeyer W., Raman A., 1997. *Configuring a supply chain to reduce the cost of demand uncertainty*. Production and Operations Management 6(3), 211-225.
- [64] Flakiewicz W., 2000. *Systemy informacyjne w zarządzaniu. Uwarunkowania, technologie, rodzaje*. Academia Oeconomica, Wydawnictwo C.H. Beck, Warszawa.
- [65] Fleming P.J., Purshouse R.C., 2002. *Evolutionary algorithms in control systems engineering: a survey*. Control Engineering Practice 10, 1223-1241.
- [66] Fox M.S., Smith S.F., 1984. *ISIS-A knowledge based system for factory scheduling*. Expert Systems 1, 25-49.

- [67] French S., 1982. *Sequencing and scheduling: an introduction to the mathematics of the job-shop*. J. Wiley and Sons, New York.
- [68] Fuglewicz P., Stapor K., Trojnar A., 1995. *CASE dla ludzi*. Lupus, Warszawa.
- [69] Garfinkel R.S., Nemhauser G.L., 1978. *Programowanie całkowitoliczbowe*. Państwowe Wydawnictwo Naukowe, Warszawa.
- [70] Giffler B., Thompson G., Ness V.V., 1960. *Algorithms for solving production scheduling problems*. Operations Research 8, 487-503.
- [71] Glaser H., Geiger W., Rohde V., 1992. *PPS Produktionsplanung und -steuerung. Grundlagen – Konzepte – Anwendungen*. Gabler, Wiesbaden.
- [72] Glover F., 1989. *Tabu search – part*. ORSA Journal on Computing 1, 190-206.
- [73] Goldberg D.E., 2003. *Algorytmy genetyczne i ich zastosowania*. Wydawnictwa Naukowo-Techniczne, Warszawa.
- [74] Goldberg D.E., 1989. *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley, Reading, MA.
- [75] Goldberg D., Lingle R., 1985. *Alleles, loci and the travelling salesman problem*, [in:] Proceedings of the First International Conference on Genetic Algorithms, J.J. Grefenstette (ed.), Lawrence Erlbaum Associates, Hillsdale, N.J., 154-159.
- [76] Gołomska E., 1994. *Logistyka jako zarządzanie łańcuchem dostaw*. Akademia Ekonomiczna w Poznaniu, Poznań.
- [77] Gołomska E. (red. nauk.), 1999. *Kompendium wiedzy o logistyce*. Wydawnictwo Naukowe PWN, Warszawa – Poznań.
- [78] Gordon V., Proth J.M., Chu C., 2002. *A survey of the state-of-the-art. Of common due date assignment and scheduling research*. European Journal of Operational Research 139, 1-25.
- [79] Green G.I., Appel L.B., 1981. *An empirical analysis of job shop dispatch rule selection*. Journal of Operations Management 1, 197-203.
- [80] Haupt R., 1989. *A survey of priority-rule based scheduling*. OR Spectrum 11, 3-16.
- [81] Hayes-Roth B., 1985. *A blackboard architecture for control*. Artificial Intelligence 26, 251-321.
- [82] Herron D.P., 1983. *Management science in industrial logistics*. Applications of Management Science 3, 49-85.
- [83] Ho J.C., Chang Y.L., 1990. *A new heuristic for the n-job, M-machine flow-shop problem*. European Journal of Operational Research 52, 194-202.
- [84] Ho J.C., Chang Y.L., 2001. *An integrated MRP and JIT framework*. Computers and Industrial Engineering 41, 173-185.
- [85] Holland J.H., 1975. *Adaptation in natural and artificial systems*. University of Michigan Press, Ann Arbor.
- [86] Holsapple C., Jacob V., Pakath R., Zaveri J., 1993. *A genetic-based hybrid scheduler for generating static schedules in flexible manufacturing contexts*. IEEE Transaction Systems, Man, and Cybernetics 23, 963-971.

- [87] Horowitz E., Sahni S., 1984. *Fundamentals of data structures in Pascal*. Computer Science Press, Rockville, MD.
- [88] Huang H.H., 2000. *Integrated production model in agile manufacturing systems*. The International Journal of Advanced Manufacturing Technology 20, 515-525.
- [89] Humpert B., Korvin De A., 1990. *A design and implementing blackboard architectures*. Expert Systems for Information Management 3, 165-194.
- [90] Hurink J., Jurisch B., Thole M., 1996. *Tabu search for job-shop scheduling problem with multi-purpose machines*. OR Spectrum 15, 205-215.
- [91] Husbands P., Mill F., Warrington S., 1991. *Genetic algorithms, production plan optimisation and scheduling*. Lecture Notes in Computer Science 496, 80-84.
- [92] Hvolby H.H., Trienekens J., 2002. *Supply chain planning opportunities for small and medium sized companies*. Computer in Industry 49, 3-8.
- [93] Ip W.H., Li Y., Man K.F., Tang K.S., 2000. *Multi-product planning and scheduling using genetic algorithm approach*. Computers and Industrial Engineering 38, 283-296.
- [94] Ito T., Salleh M.R., 2000. *A blackboard-based negotiation for collaborative supply chain system*. Journal of Materials Processing Technology 107, 398-400.
- [95] Ishibuchi H., Misaki S., Tanaka H., 1995. *Modified simulated annealing algorithms for the flow shop sequencing problem*. European Journal of Operational Research 81, 388-398.
- [96] Jasiński Z. (red. nauk.), 1999. *Zarządzanie pracą. Organizowanie, planowanie, motywowanie, kontrola*. Agencja Wydawnicza Placet, Warszawa.
- [97] Józefowska J., 1997. *Dyskretno-ciągłe problemy szeregowania zadań*. Rozprawy Nr 318, Wydawnictwo Politechniki Poznańskiej, Poznań.
- [98] Kanet J., Adelsberger H., 1987. *Expert systems in production scheduling*. European Journal of Operation Research 29, 51-57.
- [99] Kempf K., Uzsoy R., Smith S., Gary K., 2000. *Evaluation and comparison of production schedules*. Computers in Industry 42, 203-220.
- [100] Kim B., Kim S., 2001. *Extended model of a hybrid production planning approach*. International Journal of Production Economics 73(2), 165-173.
- [101] Kim Y.K., Park K., Ko J., 2003. *A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling*. Computers and Operations Research 30, 1151-1171.
- [102] Kisielnicki J., Sroka H., 1999. *Systemy informacyjne biznesu. Informatyka dla zarządzania. Metody projektowania i wdrażania systemów*. Agencja Wydawnicza Placet, Warszawa.
- [103] Knosala R., 1995. *Komputerowe systemy zarządzania produkcją. Wybrane zagadnienia*. Wydawnictwo Politechniki Śląskiej, Gliwice.
- [104] Knosala R. (red. nauk.), 2002. *Zastosowania metod sztucznej inteligencji w inżynierii produkcji*. Wydawnictwa Naukowo-Techniczne, Warszawa.

- [105] Knosala R., Wal T., 2001. *A production scheduling problem using genetic algorithm*. Journal of Materials Processing Technology 109, 90-95.
- [106] Korbicz J., Obuchowicz A., Uciński D., 1994. *Sztuczne sieci neuronowe*. Akademicka Oficyna Wydawnicza PLJ, Warszawa.
- [107] Kovács G.L., Paganelli P., 2003. *A planning and management infrastructure for large, complex, distributed projects – beyond ERP and SCM*. Computers in Industry 51, 165-183.
- [108] Krawczyk S., 2001. *Metody ilościowe w logistyce. Tom II*. Wydawnictwo C.H. Beck, Warszawa.
- [109] Lambert D.M., Cooper M.C., 2000. *Issues in supply chain management*. Industrial Marketing Management 29, 65-83.
- [110] Lambert D.M., Cooper M.C., Pagh J.D., 1998. *Supply chain management: Implementation issues and research opportunities*. The International Journal of Logistics Management 9(2), 1-19.
- [111] Lambert D.M., Emmelhainz M.A., Gardner J.T., 1996. *Developing and implementing supply chain partnerships*. The International Journal of Logistics Management 7(2), 1-17.
- [112] Latoś H., Ławrynowicz A., 1995. *Expert System Shell Esta*. [in:] Proceedings Science Conference Advanced Manufacturing Technology, Section CAD/CAM 4, Faculty of Mechanical and Manufacturing Engineering at the Technical University of Roussel, 19-21.
- [113] Lee H., Kim S.S., 2001. *Integration of process planning and scheduling using simulation based genetic algorithms*. The Journal of Advanced Manufacturing Technology 18, 586-590.
- [114] Lee Y.H., Kim S.H., 2002. *Production-distribution planning in supply chain considering capacity constraints*. Computers and Industrial Engineering 43, 169-190.
- [115] Lee K.S., 1996. *Customized purchase supporting expert systems: UNIK-SES*. Expert Systems with Applications 4, 431-441.
- [116] Lee Y.H., Jeong Ch.S., Moon Ch., 2002. *Advanced planning and scheduling with outsourcing in manufacturing supply chain*. Computers and Industrial Engineering 43, 351-374.
- [117] Lewis J.C., Naim M.M., Towill D.R., 1997. *An integrated approach to reengineering and logistic control*. International Journal of Physical Distribution and Logistics Management 23(3), 197-209.
- [118] Li H., Li Z., Li L.X., Hu B., 2000. *A production rescheduling expert simulation system*. European Journal of Operational Research 124, 283-293.
- [119] Liebowitz J., 1995. *Expert systems: A short introduction*. Engineering Fracture Mechanics 5/6, 601-607.
- [120] Liebowitz J., Potter W.E., 1995. *Scheduling objectives, requirements, resources, constrains, and processes: Implications for a generic expert scheduling system architecture and toolkit*. Expert Systems with Applications 3, 423-432.

- [121] Liou A.H.A., Wu M.T., 1996. *Mapping knowledge to rules for scheduling expert systems*. Expert Systems With Applications 3/4, 341-350.
- [122] Ławrynowicz A., 2005. *Advanced planning and scheduling in manufacturing system*, [in:] Proceedings of 6<sup>th</sup> Carpathian Control Conference. Vol. II. University of Miskolc, Miskolc, Hungary, 75-80.
- [123] Ławrynowicz A., 2004. *A genetic algorithm for job shop scheduling*, [in:] *Recent Developments in Artificial Intelligence Methods*, T. Burczyński, W. Cholewa, M. Moczulski (eds.), AI-METH Series, Silesian University of Technology, Gliwice 2004, 173-176.
- [124] Ławrynowicz A., 2003. *Algorytmy ewolucyjne w zarządzaniu produkcją*. Informatyka Teoretyczna i Stosowana 4, 149-168.
- [125] Ławrynowicz A., 1996. *Implementacja aplikacji narzędziami CASE firmy Oracle* Informatyka 5, 24-29.
- [126] Ławrynowicz A., 2004. *Combination of expert system and genetic algorithm for job shop planning and scheduling*, [in:] Proceedings of 5<sup>th</sup> International Carpathian Control Conference. Vol. I. Faculty of Mechanical Engineering and Robotics AGH-UST, Krakow, 943-948.
- [127] Ławrynowicz A., 1994. *Ekspercki system wspomagania sterowania produkcją*. Informatyka 6, 26-27.
- [128] Ławrynowicz A., 1997. *COMPLAN – ekspercki system doboru technik komputerowego wspomagania zarządzania produkcją*, [w:] red. nauk. Z. Bubnicki i A. Grzech. Inżynieria Wiedzy i Systemy Ekspertowe, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 51-58.
- [129] Ławrynowicz A., 1997. *Cykl życia systemu informatycznego*. Ekonomika i Organizacja Przedsiębiorstwa 5, 21-23.
- [130] Ławrynowicz A., 2005. *Integration of production planning and scheduling using expert system and genetic algorithm*, [in:] Proceedings of the Third European Conference on Intelligent Management Systems in Operations, The OR Society, Birmingham, UK, 33-42.
- [131] Ławrynowicz A., 1997. *Modelowanie działalności przedsiębiorstwa*. Ekonomika i Organizacja Przedsiębiorstwa 7, 23-25.
- [132] Ławrynowicz A., 2005. *Production planning and control with outsourcing using artificial intelligence*, [in:] Proceedings of the Third International Workshop on Supply Chain Management and Information Systems (SCMIS 2005), South East European Research Centre, Thessaloniki, Greece, 485-496.
- [133] Ławrynowicz A., 1993. *Reprezentacja wiedzy w sztucznej inteligencji*. Przegląd Mechaniczny 4, 16-20.
- [134] Ławrynowicz A., 1999. *Strategiczne planowanie przedsięwzięć informatycznych w przemyśle*, [w:] Materiały z Konferencji Strategia Systemów Informatycznych SIS'99, Rytko, AE Kraków, 151-160.

- [135] Ławrynowicz A., 2001. *Supply Chain Operations Reference model and its implementation*, [in:] Proceedings of the 3<sup>rd</sup> Conference IBIS'01. Implementation of Business Information Systems. Global Information Society. ARP, Malmö – Copenhagen, 95-102.
- [136] Ławrynowicz A., 2000. *System ekspercki o architekturze tablicowej w zarządzaniu produkcją*. Zeszyty Naukowe Politechniki Śląskiej, Zarządzanie 6, 53-62.
- [137] Ławrynowicz A., 1995. *Szkieletowy system ekspercki*, [w:] *Wspomaganie decyzji. Systemy eksperckie*, red. nauk. R. Kulikowski i L. Bogdan, PAN, Warszawa 303-307.
- [138] Ławrynowicz A., 2003. *Zastosowanie algorytmów genetycznych do planowania produkcji*. Organizacja i Kierowanie 4, PAN, Warszawa, 85-105.
- [139] Ławrynowicz A., 2004. *Zastosowanie kombinacji systemu eksperckiego i algorytmu ewolucyjnego do zarządzania produkcją*, Rozdział XXVIII, [w:] *Automatyzacja Procesów Dyskretnych*, red. nauk. M. Zaborowski, Wydawnictwa Naukowo-Techniczne, Warszawa, 239-246.
- [140] Maini M., 1989. *COSIMA Control Systems for Integrated Manufacturing*. [in:] Proceedings of the 5th CIM Europe Conference "Computer Integrated Manufacturing", Springer, Berlin, 423-434.
- [141] Manganelli R.L., Klein M.M., 1998. *Reengineering. Metoda usprawniania organizacji*. Polskie Wydawnictwo Ekonomiczne, Warszawa.
- [142] Márcus A., Váncza J., Kis T., Kovács A., 2003. *Project scheduling approach to production planning*. Annals of the CIRP 1, 359-362.
- [143] Maropoulos P.G., McKay K.R., Bramall D.G., 2002. *Resource-aware aggregate planning for the distributed manufacturing enterprise*. Annals of the CIRP 1, 363-366.
- [144] Martyniak Z. (red. nauk.), 1996. *Nowoczesne metody zarządzania produkcją*. Akademia Górniczo-Hutnicza w Krakowie, Kraków.
- [145] Mattfeld D.C., 1996. *Evolutionary search and job-shop*. Springer-Verlag, Heidelberg.
- [146] McKay K.N., Wiers V.C.S., 2003. *Integrated decision support for planning, scheduling, and dispatching tasks in focused factory*. Computers in Industry 50, 5-14.
- [147] McIvor R.T., Mulvenna M.D., Humphreys P.K., 1997. *A hybrid knowledge-based system for strategic purchasing*. Expert Systems with Applications 4, 497-512.
- [148] Michalewicz Z., 1999. *Algorytmy genetyczne + struktury danych = programy ewolucyjne*. Wydawnictwa Naukowo-Techniczne, Warszawa.
- [149] Michalski A., 1993. *Język opisu architektury tablicowej systemu eksperckiego*, [w:] Materiały konferencyjne Inżynieria Wiedzy i Systemy Ekspertowe, Tom II, Wydawnictwa Politechniki Wrocławskiej, Wrocław, 407-414.

- [150] Min H., Jayaraman V., Srisvastava R., 1998. *Combined location-routing problems: A synthesis and future research directions*. European Journal of Operational Research 108(1), 1-15.
- [151] Min H., Melachrinoudis E., 1999. *The relocation of a hybrid manufacturing / distribution facility from supply chain perspectives: A case study*. Omega 27(1), 75-85.
- [152] Min H., Zhou G., 2002. *Supply chain modelling: past, present and future*. Computers and Industrial Engineering 43, 231-249.
- [153] Mohanasundaram K.M., Natarajan K., Viswanathkumar G., Radhakrishnan P., Rajendran C., 2002. *Scheduling rules for dynamic shops that manufacture multi-level jobs*. Computers and Industrial Engineering 44, 119-131.
- [154] Moon C., Kim J., Hur S., 2002. *Integrated process planning and scheduling with minimizing total tardiness in multi-plants supply chain*. Computers and Industrial Engineering 43, 331-349.
- [155] Mulawka J.J., 1996. *Systemy ekspertowe*. Wydawnictwa Naukowo-Techniczne, Warszawa.
- [156] Nakagiri D., Kuriyama S., 1996. *A study production management system with MRP*. International Journal of Production Economics 44, 27-33.
- [157] Nakano R., Yamada T., 1991. *Conventional genetic algorithm for job-shop problems*, [in:] Proceeding of the Fourth International Conference on Genetic Algorithms, Belew and Booker (eds.), Morgan Kaufman, San Mateo, CA, 471-479.
- [158] Nazaretow W.M., Kim D.P., 1991. *Techniczna imitacja intelektu*. Wydawnictwa Naukowo-Techniczne, Warszawa.
- [159] Neumann K., Schwindt Ch., Trautmann N., 2002. *Advanced production scheduling for batch plants in process industries*. OR Spectrum 24, 251-279.
- [160] Nowicki A., red. nauk., 2005. *Wstęp do systemów informacyjnych zarządzania w przedsiębiorstwie*. Wydawnictwo Politechniki Częstochowskiej, Częstochowa.
- [161] Nowicki E., Smutnicki C., 1996. *A fast tabu search algorithm for the permutation flow shop problem*. European Journal of Operational Research 91, 160-175.
- [162] Ogbu F.A., Smith D.K., 1990. *The application of the simulated annealing algorithm to the solution of the  $n/m/C_{max}$  flowshop problem*. Computers and Operations Research 17, 243-253.
- [163] O'Grady P.J., Harrison C., 1985. *A general search sequencing rule for job shop sequencing*. International Journal of Production Research 23, 951-973.
- [164] Oh K.H., 1997. *Expert line balancing system (ELBS)*. Computers and Industrial Engineering 1-2, 303-306.



- [165] Osman I.H., Potts C.N., 1989. *Simulated annealing for permutation flow-shop scheduling*. OMEGA International Journal of Management Science 17, 551-557.
- [166] Owoc M.L. (red. nauk.), 1991. *Elementy systemów ekspertowych*. Akademia Ekonomiczna we Wrocławiu, Wrocław.
- [167] Palmer G.J., 1996. *A simulated annealing approach to integrated production scheduling*. Journal of Intelligent Manufacturing 7, 163-176.
- [168] Park S.J., Kim J. W., Kang H.W., 1996. *Heuristic knowledge representation of production scheduling: An integrated modeling approach*. Expert Systems with Applications 3/4, 325-339.
- [169] Pawlak M., 1999. *Algorytmy ewolucyjne jako narzędzie harmonogramowania produkcji*. Wydawnictwa Naukowe PWN, Warszawa.
- [170] Petrovic D., Roy R., 1998. *Modeling and simulation of a supply chain in an uncertain environment*. European Journal of Operational Research 109, 299-309.
- [171] Pierreval H., Caux C., Paris J.L., Viguier F., 2003. *Evolutionary approaches to the design and organization of manufacturing systems*. Computers and Industrial Engineering 44, 339-360.
- [172] Pirlot M., 1996. *General local search methods*. European Journal of Operational Research 92, 493-511.
- [173] Popończyk A., 1997. *Spirala wdrożeń MRP II*. Informatyka, Wyd. Specjalne MRP'97.
- [174] Popończyk A., 2000. *Zaawansowane systemy planowania i optymalizacji łańcucha dostaw typu APS*. Informatyka 6, 22-25.
- [175] Presutti W.D., 2003. *Supply management and e-procurement: creating value added in the supply chain*. Industrial Marketing Management 32, 219-226.
- [176] Radośniński E., 2001. *Systemy informatyczne w dynamicznej analizie decyzyjnej*. Wydawnictwa Naukowe PWN, Warszawa – Wrocław.
- [177] Rutkowska D., Piliński M., Rutkowski L., 1999. *Sieci neuronowe, algorytmy genetyczne i systemy rozmyte*. Wydawnictwo Naukowe PWN, Warszawa – Łódź.
- [178] Sabri E.H., Beamon B.M., 2000. *A multi-objective approach to simultaneous strategic and operations planning in supply chain design*. Omega 28, 581-598.
- [179] Santarek K., Strzelczak S., 1989. *Elastyczne systemy produkcyjne*. Wydawnictwo Naukowo-Techniczne, Warszawa.
- [180] Sawik T., 1995. *Scheduling flexible flow lines with no in-process buffers*. International Journal of Production Research 33, 1359-1370.
- [181] Shanthikumar J.G., Sargent R.G., 1983. *A unifying view of hybrid simulation /analytic models and modeling*. Operations Research 31(6), 1030-1052.

- [182] Sroka H., 1994. *Systemy ekspertowe. Komputerowe wspomaganie decyzji w zarządzaniu i finansach*. Akademia Ekonomiczna w Katowicach, Katowice.
- [183] Stefanowski J., 2001. *Algorytmy indukcji reguł decyzyjnych w odkrywaniu wiedzy*. Rozprawy Nr 361, Wydawnictwo Politechniki Poznańskiej, Poznań.
- [184] Stevens G.C., 1989. *Integration of the Supply Chain*. International Journal of Physical Distribution and Logistic Management 8.
- [185] Sun D., Batta R., Lin L., 1995. *Effective job shop scheduling through active chain manipulation*. Computers and Operations Research 22, 159-172.
- [186] Sundram R.M., Fu S.S., 1998, *Process planning and scheduling*. Computers and Industrial Engineering 15, 296-307.
- [187] Syaril A., Yun Y.S., Gen M., 2002. *Study on multi-stage logistic chain network: a spanning tree-based genetic algorithm approach*. Computers and Industrial Engineering 43, 299-314.
- [188] Taillard E., 1993. *Benchmarks for basic scheduling problems*. European Journal of Operational Research 64, 278-285.
- [189] Taillard E., 1990. *Some efficient heuristic methods for flow shop sequencing*. European Journal of Operational Research 47, 65-74.
- [190] Tamaki H., Nishikawa Y., 1992. *A paralleled genetic algorithm based on neighbourhood model and its application to the job-shop scheduling*, [in:] Proceedings of the Second International Conference on Parallel Problem Solving from Nature, Elsevier Science Publishers, North-Holland 573-582.
- [191] Thomas D. J., Griffin P.M., 1996. *Co-ordinated supply chain management*. European Journal of Operational Research 94, 1-15.
- [192] Todd D., Sen P., 1999. *Distributed task scheduling and allocation using genetic algorithms*. Computers and Industrial Engineering 37, 47-50.
- [193] Towill D.R., McCullen P., 1999. *The impact of an agile manufacturing programme on supply chain dynamics*. International Journal of Logistics Management 10(1), 83-96.
- [194] Tsai T., Sato R., 2004. *A UML model of agile production planning and control system*. Computers in Industry 53, 133-152.
- [195] Tzafestas S., 1989. *Petri-net and knowledge-based methodologies in manufacturing systems modeling simulation and control*, [in:] Proceedings of Conference Computer Integrated Manufacturing. Springer, Berlin, 39-50.
- [196] Uckun S., Bagchi S., Kawamura K., Miyabe Y., 1993. *Managing genetic search in job shop scheduling*. IEEE Expert-Intelligent Systems and Their Applications 8, 15-24.
- [197] Vidal C.J., Goetschalckx M., 1997. *Strategic production-distribution models: a critical review with emphasis on global supply chain models*. European Journal of Operational Research 9/8, 1-18.

- [198] Villa A., 2001. *Introducing some supply chain management problems*. International Journal of Production Economics 73, 1-4.
- [199] Warnecke H.J., Hüser M., 1995. *Lean production*. International Journal of Production Economics 41, 37-43.
- [200] Widmer M., Hertz A., 1989. *A new heuristic method for the flow shop sequencing problem*. European Journal of Operational Research 41, 186-193.
- [201] Wikner J., Towill D.R., Naim M., 1991. *Smoothing supply chain dynamics*. International Journal of Production Economics 22(3), 231-248.
- [202] Witkowski J., 2003. *Zarządzanie łańcuchem dostaw*. Polskie Wydawnictwo Ekonomiczne, Warszawa.
- [203] Wróblewski K.J., Krawczyński R., Kosieradzka A., Kasprzyk S., 1984. *Reguły priorytetu w sterowaniu przepływem produkcji*. Wydawnictwa Naukowo-Techniczne, Warszawa.
- [204] Wrycza S., 1999. *Analiza i projektowanie systemów informatycznych zarządzania. Metody, techniki, narzędzia*. Wydawnictwo Naukowe PWN, Warszawa.
- [205] Yamada T., Nakano R., 1992. *A genetic algorithm applicable to large-scale job-shop problems*, [in:] Proceedings of the Second International Conference on Parallel Problem Solving from Nature. Elsevier Science Publishers, North-Holland, 281-290.
- [206] Yamada T., Takahashi H., Kataoka S., 1996. *A heuristic for the mini-max spanning forest problem*. European Journal of Operational Research 1, 565-572.
- [207] Zegordi S.H., Itob K., Enklawa T., 1995. *Minimizing makespan for flow shop scheduling by combining simulated annealing with sequencing knowledge*. European Journal of Operational Research, 85, 515-531.
- [208] Zhou H., Feng Y., Han L., 2001. *The hybrid heuristic algorithm for job shop scheduling*. Computers and Industrial Engineering 40, 191-200.
- [209] Zhou G., Gen M., 1999. *Genetic algorithm approach on multi-criteria minimum spanning tree problem*. European Journal of Operation Research 114(1), 141-152.
- [210] Zhou G., Min H., Gen M., 2002. *The balanced allocation of customers to multiple distribution centres in the supply chain network: a genetic algorithm approach*. Computers and Industrial Engineering 42, 251-261.
- [211] Zieliński J.S. (red. nauk.), 2000. *Inteligentne systemy w zarządzaniu. Teoria i praktyka*. Wydawnictwa Naukowe PWN, Warszawa.
- [212] Zimmewicz K., 2003. *Współczesne koncepcje i metody zarządzania*. Polskie Wydawnictwo Ekonomiczne, Warszawa.

## ZARZĄDZANIE PRODUKCJĄ W OGNIWIE SIECI DOSTAW Z ZASTOSOWANIEM SYSTEMU EKSPERCKIEGO I ALGORYTMU GENETYCZNEGO

### Streszczenie

W niniejszej rozprawie zaproponowano nową metodologię zarządzania produkcją w ogniwie sieci dostaw z zastosowaniem sztucznej inteligencji. Studia koncentrują się na obróbkowym systemie produkcyjnym, ponieważ stanowi on „wąskie gardło” w dotrzymaniu terminów zakończenia realizowanych w nim procesów produkcyjnych.

Ogólnie podejście do zarządzania produkcją polegało na:

- zastosowaniu w pierwszej fazie systemu eksperckiego do tworzenia planu produkcji, odpowiedniego do zamówień klientów i raportów z wytwarzania,
- zastosowaniu w drugiej fazie algorytmu genetycznego do konstruowania harmonogramu odpowiedniego do planu produkcji.

Głównym celem badań było usprawnienie tradycyjnych metod planowania i sterowania produkcją opartych na bazach danych oraz regułach priorytetu oraz zaproponowanie nowego i efektywniejszego podejścia ze sztuczną inteligencją do problemu zarządzania w elastycznych systemach produkcyjnych z outsourcingiem. Proponowane nowe metody nie wymagają przyjmowania nierealnych założeń do funkcji celu, takich jak liniowość, wypukłość, różniczkowalność i mogą być zastosowane do bardzo skomplikowanych rzeczywistych problemów.

Autorka proponuje system ekspercki stworzony do wspomagania krótkoterminowego planowania i sterowania produkcją w ogniwie sieci dostaw. Planowanie produkcji inicjuje obciążenia produkcji i zdolności produkcyjne przez generowanie planu produkcji, który określa przepływ materiału i wykorzystanie zasobów w danym horyzoncie planistycznym. Zaproponowany system ekspercki uwzględnia alternatywne plany procesu dla zadań wtedy, kiedy występują „wąskie gardła”.

W prezentowanym podejściu są rozważani – jako źródła zleceń – wewnętrzni i zewnętrzni klienci, ponieważ w erze sieci dostaw decyzja o zastosowaniu zasobów musi być podjęta w oparciu zarówno o wewnętrzne, jak i zewnętrzne zdolności produkcyjne – przepływ materiału musi być zsynchronizowany z przyływami i wypływami. Dlatego system ekspercki generuje plany produkcji bazując na wewnętrznych zasobach i outsourcingowych maszynach. Wyrównywanie obciążeń jest możliwe także przez outsourcing, ponieważ zewnętrzna baza danych systemu eksperckiego obejmuje alternatywne plany procesu dla zadań.

W niniejszej rozprawie jest także rozwijane nowe podejście do harmonogramowania niestabilizowanej produkcji, z kryterium terminu zakończenia wszystkich zadań realizowanych w systemie produkcyjnym. W ramach badań efektywności nowej metody przeprowadzono testy porównawcze wartości długości

uszeregowania dla harmonogramu otrzymanego z zastosowaniem algorytmu genetycznego i harmonogramu otrzymanego z zastosowaniem reguł dyspozytorskich. W rozprawie rozważano siedem kombinacji najpopularniejszych reguł priorytetu, tj.: SPT, LPT, MWR, LWR, MOR, LOR, EDD.

Zaproponowana metoda bazująca na kombinacji systemu eksperckiego i algorytmu genetycznego była testowana na danych z rzeczywistych przedsiębiorstw.

Wstępne oceny wykazały, że system ekspercki może wspomagać planowanie przez generowanie krótkoterminowych planów produkcji. Zastosowanie systemu eksperckiego w przemyśle redukuje liczbę „wąskich gardeł”. Najważniejszą korzyścią z zastosowania zaproponowanych technik sztucznej inteligencji jest szybkie reagowanie na wymagania rynku, modyfikowanie zdolności produkcyjnych i wychodzenie naprzeciw potrzebom klientów. Rezultaty eksperymentów dowodzą, że zaproponowany algorytm genetyczny jest bardzo efektywny. Badania wykazały, że dla problemu harmonogramowania koncepcja z zastosowaniem algorytmu genetycznego daje lepsze rezultaty (średnio o 20,95%), niż koncepcja oparta na metodach bazujących na kombinacji reguł priorytetu.

W przyszłości badania powinny koncentrować się na zastosowaniu kombinacji systemu eksperckiego i algorytmu genetycznego na wyższych szczeblach zarządzania produkcją, ze względu na uniwersalność zaproponowanych metod.

**PRODUCTION MANAGEMENT IN A FOCUSED FACTORY  
OF A SUPPLY NETWORK USING EXPERT SYSTEM  
AND GENETIC ALGORITHM**

Abstract

This dissertation proposes a new methodology to production management in a focused factory of a supply net using artificial intelligence. In this study, the focus is on the fabrication level, because it is a bottleneck in the process and it is essential to satisfying the due date.

A general approach is: using in the first phase expert systems to create production plan according to the customer's orders and reports from manufacturing and using in the second phase genetic algorithms to construct a schedule according to the production plan.

The main purpose of the research was to improve the efficiency of the traditional planning and control methods based on a data base and a priority rule and propose a more effective and efficient approach to solving the production management problem with outsourcing using artificial intelligence. The proposed method does not require any unrealistic assumptions about the objective function, such as linearity, convexity and differentiability. It can be used to solve highly complicated and non-linear functions of a realistic production problem.

The author proposes an expert system designed to help companies in the short-term production planning and control production in a focused factory of a supply net. Production planning matches future production load and capacities by generating plans that determine the flow of materials and the use of resources over a given planning horizon. The proposed expert system considers alternative process plans for job, when a machine bottleneck exists. Thus, in this system capacity adjustment is also possible through outsourcing.

This approach considers orders of internal customer and/or orders of external customer as the source of production orders, because in the era of supply net, decision on the use of resources should concern both internal and external capacities; the internal flow of materials should be synchronized with the incoming and outgoing flows. Therefore, the expert system generates production plans based on available internal resources and/or outsourcing machines. It is possible, because the external data base of expert system includes alternative processing plans for job.

In this dissertation, a new approach with the genetic algorithm to the job shop scheduling problem with makespan as the criterion has been also developed. In this research, the effectiveness of the scheduling method was tested by comparing value of the makespan produced by the genetic algorithm with value of the makespan produced by methods based on dispatching rules. In this dissertation there have been considered seven combinations of popular dispatching

rules i.e. SPT, LPT, MWR, LWR, MOR, LOR, EDD. The comparison has been carried out by applying these methods to ten instances of production plans.

The proposed method based on the combination of expert system and genetic algorithm has been tested using data from real factories.

Preliminary evaluation indicates that the expert system can assist the planning by providing the production plans. By applying the expert system in industry, the bottleneck problems of job shop are reduced. The major benefit of using the artificial intelligence technique is to provide a quick response to changing market requirements, modify manufacturing capacities and production to meet the production need and fulfil the needs of customers. The results of experiments provide that the proposed genetic algorithm is a very effective algorithm. The research indicates, that for the scheduling problem, the concept using genetic algorithm yields better results (average 20,95%) than using the methods based on priority rule.

In the future, research can concentrate on using a combination of the expert system and the genetic algorithm at upper production management levels, for the sake of universal proposed methods.

## **ANEKS**

### **Plany produkcji**



Tabela 1. Plan produkcji 1 dla przypadku 3

Operacje	1	2	3	4	5	1	2	3	4	5
Zadania	Czas (w minutach)					Maszyny				
1	74	88	–	–	–	1	5	–	–	–
2	94	29	113	198	–	1	11	9	6	–
3	96	163	322	109	–	3	11	5	12	–
4	85	171	–	–	–	1	5	–	–	–
5	93	124	105	104	–	1	8	9	6	–
6	272	244	–	–	–	6	9	–	–	–
7	171	–	–	–	–	6	–	–	–	–
8	271	83	155	290	–	7	7	9	15	–
9	85	165	–	–	–	1	5	–	–	–
10	41	346	113	–	–	1	7	13	–	–
11	50	87	180	122	–	2	12	10	12	–
12	30	36	37	66	303	4	4	4	10	12
13	88	84	159	–	–	1	16	8	–	–
14	50	46	84	88	255	2	2	2	14	10
15	50	87	180	61	–	2	12	10	12	–

Tabela 2. Plan produkcji 2 dla przypadku 3

Operacje	1	2	3	4	5	1	2	3	4	5
Zadania	Czas (w minutach)					Maszyny				
1	74	88	–	–	–	1	5	–	–	–
2	94	29	113	198	–	1	11	9	6	–
3	96	163	322	109	–	3	11	5	12	–
4	85	171	–	–	–	1	5	–	–	–
5	93	124	105	104	–	1	8	9	6	–
6	272	244	–	–	–	6	9	–	–	–
7	171	–	–	–	–	6	–	–	–	–
8	271	83	155	290	–	7	7	9	15	–
9	85	165	–	–	–	1	5	–	–	–
10	41	346	113	–	–	1	7	13	–	–
11	50	87	180	122	–	2	12	10	12	–
12	30	36	37	66	303	4	4	4	10	12
13	88	84	159	–	–	1	16	8	–	–
14	50	46	84	88	255	2	2	2	14	10
15	50	87	180	90	–	2	12	10	12	–
16	88	84	159	–	–	1	16	8		

Tabela 3. Plan produkcji 3 dla przypadku 3

Operacje	1	2	3	4	5	1	2	3	4	5
Zadania	Czas (w minutach)					Maszyny				
1	74	88	–	–	–	1	5	–	–	–
2	94	29	113	198	–	1	11	9	6	–
3	96	163	322	109	–	3	11	5	12	–
4	85	171	–	–	–	1	5	–	–	–
5	93	124	105	104	–	1	8	9	6	–
6	272	244	–	–	–	6	9	–	–	–
7	171	–	–	–	–	6	–	–	–	–
8	271	83	155	290	–	7	7	9	15	–
9	85	165	–	–	–	1	5	–	–	–
10	41	346	113	–	–	1	7	13	–	–
11	50	87	180	122	–	2	12	10	12	–
12	30	36	37	66	303	4	4	4	10	12
13	88	84	159	–	–	1	16	8	–	–
14	50	46	84	88	255	2	2	2	14	10
15	171	–	–	–	–	6	–	–	–	–
16	88	84	159	–	–	1	16	8		

Tabela 4. Plan produkcji 4 dla przypadku 3

Operacje	1	2	3	4	5	1	2	3	4	5
Zadania	Czas (w minutach)					Maszyny				
1	74	88	–	–	–	1	5	–	–	–
2	94	29	113	198	–	1	11	9	6	–
3	96	163	322	109	–	3	11	5	12	–
4	85	171	–	–	–	1	5	–	–	–
5	93	124	105	104	–	1	8	9	6	–
6	272	244	–	–	–	6	9	–	–	–
7	171	–	–	–	–	6	–	–	–	–
8	271	83	155	290	–	7	7	9	15	–
9	85	165	–	–	–	1	5	–	–	–
10	41	346	113	–	–	1	7	13	–	–
11	50	87	180	122	–	2	12	10	12	–
12	30	36	37	66	303	4	4	4	10	12
13	88	84	159	–	–	1	16	8	–	–
14	50	46	84	88	255	2	2	2	14	10
15	171	–	–	–	–	6	–	–	–	–
16	88	84	159	–	–	1	16	8		
17	50	87	180	61	–	2	12	10	12	–

Tabela 5. Plan produkcji 5 dla przypadku 3

Operacje	1	2	3	4	5	1	2	3	4	5
Zadania	Czas (w minutach)					Maszyny				
1	74	88	–	–	–	1	5	–	–	–
2	94	29	113	198	–	1	11	9	6	–
3	94	29	55	198	–	1	11	9	5	–
4	85	171	–	–	–	1	5	–	–	–
5	93	124	105	104	–	1	8	9	6	–
6	272	244	–	–	–	6	9	–	–	–
7	171	–	–	–	–	6	–	–	–	–
8	271	83	155	290	–	7	7	9	15	–
9	85	165	–	–	–	1	5	–	–	–
10	41	346	113	–	–	1	7	13	–	–
11	50	87	180	122	–	2	12	10	12	–
12	30	36	37	66	303	4	4	4	10	12
13	88	84	100	–	–	1	16	8	–	–
14	50	46	84	88	255	2	2	2	14	10
15	74	88				1	5			–

Tabela 6. Plan produkcji 6 dla przypadku 3

Operacje	1	2	3	4	5	1	2	3	4	5
Zadania	Czas (w minutach)					Maszyny				
1	74	88	–	–	–	1	5	–	–	–
2	94	29	113	198	–	1	11	9	6	–
3	94	29	55	198	–	1	11	9	5	–
4	85	171	–	–	–	1	5	–	–	–
5	93	124	105	104	–	1	8	9	6	–
6	272	244	–	–	–	6	9	–	–	–
7	171	–	–	–	–	6	–	–	–	–
8	271	83	155	290	–	7	7	9	15	–
9	85	165	–	–	–	1	5	–	–	–
10	41	346	113	–	–	1	7	13	–	–
11	50	87	180	122	–	2	12	10	12	–
12	30	36	37	66	303	4	4	4	10	12
13	88	84	100	–	–	1	16	8	–	–
14	50	46	84	88	255	2	2	2	14	10
15	74	88	–	–	–	1	5	–	–	–
16	114	–	–	–	–	6	–	–	–	–

Tabela 7. Plan produkcji 7 dla przypadku 3

Operacje	1	2	3	4	5	1	2	3	4	5
Zadania	Czas (w minutach)					Maszyny				
1	74	88	–	–	–	1	5	–	–	–
2	94	29	113	198	–	1	11	9	6	–
3	96	163	322	109	–	3	11	5	12	–
4	85	171	–	–	–	1	5	–	–	–
5	93	124	105	104	–	1	8	9	6	–
6	272	244	–	–	–	6	9	–	–	–
7	171	–	–	–	–	6	–	–	–	–
8	271	83	155	290	–	7	7	9	15	–
9	85	165	–	–	–	1	5	–	–	–
10	41	346	113	–	–	1	7	13	–	–
11	50	87	180	122	–	2	12	10	12	–
12	30	36	37	66	303	4	4	4	10	12
13	88	84	159	–	–	1	16	8	–	–
14	50	46	84	88	255	2	2	2	14	10
15	171	–	–	–	–	6	–	–	–	–
16	88	84	159	–	–	1	16	8	–	–
17	104	128	–	–	–	1	5	–	–	–

Tabela 8. Plan produkcji 8 dla przypadku 3

Operacje	1	2	3	4	5	1	2	3	4	5
Zadania	Czas (w minutach)					Maszyny				
1	74	88	–	–	–	1	5	–	–	–
2	94	29	113	198	–	1	11	9	6	–
3	96	163	322	109	–	3	11	5	12	–
4	85	171	–	–	–	1	5	–	–	–
5	93	124	105	104	–	1	8	9	6	–
6	136	122	–	–	–	6	9	–	–	–
7	171	–	–	–	–	6	–	–	–	–
8	271	83	155	290	–	7	7	9	15	–
9	85	165	–	–	–	1	5	–	–	–
10	41	346	113	–	–	1	7	13	–	–
11	50	87	180	122	–	2	12	10	12	–
12	30	36	37	66	303	4	4	4	10	12
13	88	84	159	–	–	1	16	8	–	–
14	50	46	84	88	255	2	2	2	14	10
15	88	84	159	–	–	1	16	8	–	–
16	43	83	–	–	–	1	5	–	–	–

Tabela 9. Plan produkcji 9 dla przypadku 3

Operacje	1	2	3	4	5	1	2	3	4	5
Zadania	Czas (w minutach)					Maszyny				
1	104	132	–	–	–	1	5	–	–	–
2	94	29	113	198	–	1	11	9	6	–
3	96	163	322	109	–	3	11	5	12	–
4	85	171	–	–	–	1	5	–	–	–
5	93	124	105	104	–	1	8	9	6	–
6	272	244	–	–	–	6	9	–	–	–
7	171	–	–	–	–	6	–	–	–	–
8	271	83	155	290	–	7	7	9	15	–
9	85	165	–	–	–	1	5	–	–	–
10	41	346	113	–	–	1	7	13	–	–
11	50	87	180	122	–	2	12	10	12	–
12	30	36	37	66	303	4	4	4	10	12
13	88	84	159	–	–	1	16	8	–	–
14	50	46	84	88	255	2	2	2	14	10
15	171	–	–	–	–	6	–	–	–	–
16	88	84	159	–	–	1	16	8	–	–

Tabela 10. Plan produkcji 10 dla przypadku 3

Operacje	1	2	3	4	5	1	2	3	4	5
Zadania	Czas (w minutach)					Maszyny				
1	148	176	–	–	–	1	5	–	–	–
2	94	29	113	198	–	1	11	9	6	–
3	96	163	322	109	–	3	11	5	12	–
4	85	171	–	–	–	1	5	–	–	–
5	93	124	105	104	–	1	8	9	6	–
6	272	244	–	–	–	6	9	–	–	–
7	171	–	–	–	–	6	–	–	–	–
8	271	83	155	290	–	7	7	9	15	–
9	85	165	–	–	–	1	5	–	–	–
10	41	346	113	–	–	1	7	13	–	–
11	50	87	180	122	–	2	12	10	12	–
12	30	36	37	66	303	4	4	4	10	12
13	88	84	159	–	–	1	16	8	–	–
14	50	46	84	88	255	2	2	2	14	10
15	171	–	–	–	–	6	–	–	–	–
16	88	84	159	–	–	1	16	8	–	–



Tabela 12. Plan produkcji części do urządzeń elektrycznych

	Operacje (czas w minutach)										
	1	2	3	4	5	6	7	8	9	10	11
Zadanie 1											
Czas	65	170	75	–	–	–	–	–	–	–	–
Maszyna	4	2	2	–	–	–	–	–	–	–	–
Zadanie 2											
Czas	105	–	–	–	–	–	–	–	–	–	–
Maszyna	2	–	–	–	–	–	–	–	–	–	–
Zadanie 3											
Czas	105	–	–	–	–	–	–	–	–	–	–
Maszyna	2	–	–	–	–	–	–	–	–	–	–
Zadanie 4											
Czas	55	170	74	–	–	–	–	–	–	–	–
Maszyna	4	2	2	–	–	–	–	–	–	–	–
Zadanie 5											
Czas	76	100	250	172	77	64	73	–	–	–	–
Maszyna	1	8	9	3	5	4	4	–	–	–	–
Zadanie 6											
Czas	145	64	–	–	–	–	–	–	–	–	–
Maszyna	6	4	–	–	–	–	–	–	–	–	–
Zadanie 7											
Czas	410	80	–	–	–	–	–	–	–	–	–
Maszyna	6	4	–	–	–	–	–	–	–	–	–
Zadanie 8											
Czas	200	–	–	–	–	–	–	–	–	–	–
Maszyna	4	–	–	–	–	–	–	–	–	–	–
Zadanie 9											
Czas	74	76	72	89	156	125	83	80	78	84	–
Maszyna	1	5	4	5	7	7	3	3	9	4	–
Zadanie 10											
Czas	75	115	345	207	68	66	82	–	–	–	–
Maszyna	1	8	9	3	5	4	4	–	–	–	–
Zadanie 11											
Czas	75	285	95	–	–	–	–	–	–	–	–
Maszyna	4	2	2	–	–	–	–	–	–	–	–
Zadanie 12											
Czas	72	81	82	101	201	159	95	91	88	99	–
Maszyna	1	5	4	5	7	7	3	9	3	4	–

Tabela 13. Plan produkcji części do maszyn rolniczych

	Operacje (czas w minutach)										
	1	2	3	4	5	6	7	8	9	10	11
Zadanie 1											
Czas	74	88	–	–	–	–	–	–	–	–	–
Maszyna	1	5	–	–	–	–	–	–	–	–	–
Zadanie 2											
Czas	94	29	113	198	–	–	–	–	–	–	–
Maszyna	1	11	9	6	–	–	–	–	–	–	–
Zadanie 3											
Czas	96	163	322	109	–	–	–	–	–	–	–
Maszyna	3	11	5	12	–	–	–	–	–	–	–
Zadanie 4											
Czas	85	171	–	–	–	–	–	–	–	–	–
Maszyna	1	5	–	–	–	–	–	–	–	–	–
Zadanie 5											
Czas	93	124	105	104	–	–	–	–	–	–	–
Maszyna	1	8	9	6	–	–	–	–	–	–	–
Zadanie 6											
Czas	272	244	–	–	–	–	–	–	–	–	–
Maszyna	6	9	–	–	–	–	–	–	–	–	–
Zadanie 7											
Czas	171	–	–	–	–	–	–	–	–	–	–
Maszyna	6	–	–	–	–	–	–	–	–	–	–
Zadanie 8											
Czas	271	83	155	290	–	–	–	–	–	–	–
Maszyna	7	7	9	15	–	–	–	–	–	–	–
Zadanie 9											
Czas	85	165	–	–	–	–	–	–	–	–	–
Maszyna	1	5	–	–	–	–	–	–	–	–	–
Zadanie 10											
Czas	41	346	113	–	–	–	–	–	–	–	–
Maszyna	1	7	13	–	–	–	–	–	–	–	–
Zadanie 11											
Czas	50	87	180	122	–	–	–	–	–	–	–
Maszyna	2	12	10	12	–	–	–	–	–	–	–
Zadanie 12											
Czas	30	36	37	66	303	–	–	–	–	–	–
Maszyna	4	4	4	10	12	–	–	–	–	–	–
Zadanie 13											
Czas	88	84	159	–	–	–	–	–	–	–	–
Maszyna	1	16	8	–	–	–	–	–	–	–	–
Zadanie 14											
Cz	50	46	84	88	255	–	–	–	–	–	–
M	2	2	2	14	10	–	–	–	–	–	–



Tabela 14. Plan produkcji części do armatury hydraulicznej

	Operacje (czas w minutach)										
	1	2	3	4	5	6	7	8	9	10	11
Zadanie 1											
Czas	44	116	108	96	212	203	127	54	–	–	–
Maszyna	1	7	8	2	3	3	7	10	–	–	–
Zadanie 2											
Czas	28	485	67	240	72	114	–	–	–	–	–
Maszyna	1	5	11	12	9	9	–	–	–	–	–
Zadanie 3											
Czas	476	148	76	76	–	–	–	–	–	–	–
Maszyna	4	4	8	9	–	–	–	–	–	–	–
Zadanie 4											
Czas	29	532	53	480	62	69	–	–	–	–	–
Maszyna	1	6	11	2	11	10	–	–	–	–	–
Zadanie 5											
Czas	500	92	–	–	–	–	–	–	–	–	–
Maszyna	6	8	–	–	–	–	–	–	–	–	–
Zadanie 6											
Czas	583	47	76	73	185	73	180	110	–	–	–
Maszyna	4	11	9	10	7	10	13	9	–	–	–
Zadanie 7											
Czas	44	116	108	96	212	203	107	54			
Maszyna	1	7	8	2	5	3	7	10			

Tabela 15. Plan produkcji części do armatury wodociągowej

	Operacje (czas w minutach)										
	1	2	3	4	5	6	7	8	9	10	11
Zadanie 1											
Czas	362	248	353	–	–	–	–	–	–	–	–
Maszyna	12	7	13	–	–	–	–	–	–	–	–
Zadanie 2											
Czas	362	111	108	–	–	–	–	–	–	–	–
Maszyna	2	3	5	–	–	–	–	–	–	–	–
Zadanie 3											
Czas	430	111	131	–	–	–	–	–	–	–	–
Maszyna	2	3	6	–	–	–	–	–	–	–	–
Zadanie 4											
Czas	362	305	286	198	121	65	–	–	–	–	–
Maszyna	14	8	13	6	6	9	–	–	–	–	–
Zadanie 5											
Czas	387	261	60	–	–	–	–	–	–	–	–
Maszyna	4	13	9	–	–	–	–	–	–	–	–
Zadanie 6											
Czas	54	194	90	88	159	49	110	–	–	–	–
Maszyna	1	12	8	5	10	11	3	–	–	–	–
Zadanie 7											
Czas	477	–	–	–	–	–	–	–	–	–	–
Maszyna	4	–	–	–	–	–	–	–	–	–	–
Zadanie 8											
Czas	346	204	261	181	141	60	–	–	–	–	–
Maszyna	14	8	5	6	5	9	–	–	–	–	–
Zadanie 9											
Czas	265	181	60	–	–	–	–	–	–	–	–
Maszyna	4	6	9	–	–	–	–	–	–	–	–
Zadanie 10											
Czas	54	194	90	88	124	44	104	–	–	–	–
Maszyna	1	3	8	6	10	11	2	–	–	–	–
Zadanie 11											
Czas	56	209	96	94	171	59	116	–	–	–	–
Maszyna	1	12	8	12	5	11	3	–	–	–	–

Tabela 16. Plan produkcji części do maszyn dla leśnictwa

	Operacje (czas w minutach)										
	1	2	3	4	5	6	7	8	9	10	11
Zadanie 1											
Czas	50	270	125	44	110	–	–	–	–	–	–
Maszyna	1	3	6	8	7	–	–	–	–	–	–
Zadanie 2											
Czas	54	215	62	236	–	–	–	–	–	–	–
Maszyna	11	10	8	7	–	–	–	–	–	–	–
Zadanie 3											
Czas	53	309	140	46	122	–	–	–	–	–	–
Maszyna	1	2	10	8	7	–	–	–	–	–	–
Zadanie 4											
Czas	41	306	197	61	–	–	–	–	–	–	–
Maszyna	11	2	10	9	–	–	–	–	–	–	–
Zadanie 5											
Czas	41	43	–	–	–	–	–	–	–	–	–
Maszyna	1	9	–	–	–	–	–	–	–	–	–
Zadanie 6											
Czas	51	43	164	–	–	–	–	–	–	–	–
Maszyna	11	9	7	–	–	–	–	–	–	–	–
Zadanie 7											
Czas	260	35	–	–	–	–	–	–	–	–	–
Maszyna	2	9	–	–	–	–	–	–	–	–	–
Zadanie 8											
Czas	50	191	179	62	–	–	–	–	–	–	–
Maszyna	1	3	6	9	–	–	–	–	–	–	–
Zadanie 9											
Czas	86	252	109	–	–	–	–	–	–	–	–
Maszyna	11	4	5	–	–	–	–	–	–	–	–
Zadanie 10											
Czas	36	252	–	–	–	–	–	–	–	–	–
Maszyna	1	1	–	–	–	–	–	–	–	–	–
Zadanie 11											
Czas	50	155	626	67	218	–	–	–	–	–	–
Maszyna	11	6	4	9	7	–	–	–	–	–	–
Zadanie 12											
Czas	50	270	125	44	110	–	–	–	–	–	–
Maszyna	1	3	6	8	7	–	–	–	–	–	–



Tabela 17. Plan produkcji części do form wtryskowych

	Operacje (czas w minutach)										
	1	2	3	4	5	6	7	8	9	10	11
Zadanie 1											
Czas	38	260	80	50	–	–	–	–	–	–	–
Maszyna	1	2	8	9	–	–	–	–	–	–	–
Zadanie 2											
Czas	26	110	–	–	–	–	–	–	–	–	–
Maszyna	1	3	–	–	–	–	–	–	–	–	–
Zadanie 3											
Czas	260	350	140	–	–	–	–	–	–	–	–
Maszyna	5	7	8	–	–	–	–	–	–	–	–
Zadanie 4											
Czas	50	500	50	38	80	80	–	–	–	–	–
Maszyna	1	3	5	9	6	7	–	–	–	–	–
Zadanie 5											
Czas	56	44	272	80	44	–	–	–	–	–	–
Maszyna	11	9	2	5	8	–	–	–	–	–	–
Zadanie 6											
Czas	56	38	188	140	–	–	–	–	–	–	–
Maszyna	11	10	4	8	–	–	–	–	–	–	–
Zadanie 7											
Czas	38	80	44	–	–	–	–	–	–	–	–
Maszyna	1	2	7	–	–	–	–	–	–	–	–
Zadanie 8											
Czas	170	260	110	–	–	–	–	–	–	–	–
Maszyna	1	2	7	–	–	–	–	–	–	–	–
Zadanie 9											
Czas	32	110	44	–	–	–	–	–	–	–	–
Maszyna	1	2	4	–	–	–	–	–	–	–	–
Zadanie 10											
Czas	80	80	110	260	230	260	110	–	–	–	–
Maszyna	11	10	4	4	7	5	8	–	–	–	–

Tabela 18. Plan produkcji części do maszyn spożywczych

	Operacje (czas w minutach)										
	1	2	3	4	5	6	7	8	9	10	11
Zadanie 1											
Czas	63	440	62		–	–	–	–	–	–	–
Maszyna	1	2	6		–	–	–	–	–	–	–
Zadanie 2											
Czas	27	108	40	84	58	25	66	–	–	–	–
Maszyna	1	3	9	3	7	6	8	–	–	–	–
Zadanie 3											
Czas	29	211	76	–	–	–	–	–	–	–	–
Maszyna	1	2	6	–	–	–	–	–	–	–	–
Zadanie 4											
Czas	48	148					–	–	–	–	–
Maszyna	1	6					–	–	–	–	–
Zadanie 5											
Czas	98	225				–	–	–	–	–	–
Maszyna	1	3				–	–	–	–	–	–
Zadanie 6											
Czas	52	257	191	35	–	–	–	–	–	–	–
Maszyna	1	5	7	6	–	–	–	–	–	–	–
Zadanie 7											
Czas	94	277		–	–	–	–	–	–	–	–
Maszyna	1	3		–	–	–	–	–	–	–	–
Zadanie 8											
Czas	62	99		–	–	–	–	–	–	–	–
Maszyna	6	3		–	–	–	–	–	–	–	–
Zadanie 9											
Czas	108	115		–	–	–	–	–	–	–	–
Maszyna	6	2		–	–	–	–	–	–	–	–
Zadanie 10											
Czas	536	59						–	–	–	–
Maszyna	4	6						–	–	–	–
Zadanie 11											
Czas	79	36						–	–	–	–
Maszyna	1	6						–	–	–	–
Zadanie 12											
Czas	38	125	40	108	54	26	63	–	–	–	–
Maszyna	1	4	9	4	7	6	8	–	–	–	–

Tabela 19. Plan produkcji narzędzi

	Operacje (czas w minutach)										
	1	2	3	4	5	6	7	8	9	10	11
Zadanie 1											
Czas	90	27	209	23	27	–	–	–	–	–	–
Maszyna	1	5	8	4	5	–	–	–	–	–	–
Zadanie 2											
Czas	100	26	247	23	27	–	–	–	–	–	–
Maszyna	1	5	8	4	5	–	–	–	–	–	–
Zadanie 3											
Czas	127	30	335	23	33	–	–	–	–	–	–
Maszyna	1	5	12	4	5	–	–	–	–	–	–
Zadanie 4											
Czas	148	30	512	23	37		–	–	–	–	–
Maszyna	1	5	12	4	5		–	–	–	–	–
Zadanie 5											
Czas	364	20				–	–	–	–	–	–
Maszyna	7	5				–	–	–	–	–	–
Zadanie 6											
Czas	56	33	32	23	59	–	–	–	–	–	–
Maszyna	2	5	11	2	9	–	–	–	–	–	–
Zadanie 7											
Czas	56	23	33	33	23	62	–	–	–	–	–
Maszyna	2	2	5	11	2	9	–	–	–	–	–
Zadanie 8											
Czas	170	27	33	363	25	44	–	–	–	–	–
Maszyna	1	2	5	3	4	5	–	–	–	–	–
Zadanie 9											
Czas	65	67	18	25	24	–	–	–	–	–	–
Maszyna	8	8	4	5	5	–	–	–	–	–	–
Zadanie 10											
Czas	40	73	25	73	24	24		–	–	–	–
Maszyna	6	10	13	10	5	5		–	–	–	–
Zadanie 11											
Czas	31	25	20	40	41	18	25	24	–	–	–
Maszyna	10	14	13	8	8	4	5	5	–	–	–





Wszystkie publikacje  
Wydawnictw Uczelnianych  
Akademii Techniczno-Rolniczej w Bydgoszczy  
można nabywać w

Dziale Udostępniania  
Biblioteki Głównej ATR

ul. Prof. S. Kaliskiego 7, 85-796 Bydgoszcz

tel. (0-52) 340-80-78, fax (0-52) 340-80-63

<http://ksiegarnia.atr.bydgoszcz.pl/>