

**Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie**

Wydział Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej

KATEDRA AUTOMATYKI I ROBOTYKI



ROZPRAWA DOKTORSKA

KATARZYNA GROBLER-DEBSKA

**METODA PRZEŁĄCZANIA MODELI
ALGEBRAICZNO-LOGICZNYCH DLA PRODUKCJI
DYSKRETNEJ Z BRAKAMI JAKOŚCIOWYMI**

PROMOTOR:

Prof. dr hab. inż. Ewa Dudek-Dyduch

Kraków 24 czerwca 2019

AGH
University of Science and Technology in Krakow

Faculty of Electrical Engineering, Automatics, Computer Science
and Biomedical Engineering

DEPARTMENT OF AUTOMATICS AND ROBOTICS



PH.D OF SCIENCE THESIS

KATARZYNA GROBLER-DĘBSKA

**METHODOLOGY OF SWITCHING ALGEBRAIC-LOGICAL
MODELS FOR DISCRETE MANUFACTURING WITH DEFECTS**

SUPERVISOR:

Prof. Ewa Dudek-Dyduch Ph.D

Krakow 24 czerwca 2019

Podziękowania

Pragnę podziękować wszystkim osobom, które przyczyniły się do powstania tej pracy. Prezentowana praca doktorska została przygotowana pod opieką Pani prof. dr hab. Ewy Dudek-Dyduch, której dziękuję za cenne uwagi i sugestie, które wpłynęły na kształt tej pracy.

Dziękuję również wszystkim moim współpracownikom z AGH, którzy wspierali mnie podczas pisania tej pracy. Ogrom pomocy i życzliwości, który spotkał mnie w ostatnich latach jest nie do opisania. Szczególne podziękowania kieruję w stronę Edyty Kucharskiej, na której pomoc mogłam liczyć o każdej porze dnia, a czasem i nocy, za jej szczerą radę oraz uwagi (i jej Rodzinie Jackowi, Honoratce i Helence za ich cierpliwość). Praca ta nie powstała by również gdyby nie współpraca, dyskusje i dawanie przykładu, za który dziękuję w szczególności: Lidii, Krzyškowi, obu Jarkom, Markowi, Konradowi, Jurkowi i Waldkowi. Dziękuję również Kierownikowi Katedry Automatyki i Robotyki prof. dr hab. inż. Markowi Gorgoniowi za wsparcie i życzliwość. Nie mogę też zapomnieć o Paniach Barbarze Leśniak, Dorocie Kaganek, Danucie Korzeniowskiej i Barbarze Bysiewicz-Tokarz życzliwość i pilnowanie spraw formalnych. Dziękuję również niewymienionym koleżankom i kolegom z Katedr Automatyki i Robotyki oraz Informatyki Stosowanej.

Na koniec, ale nie na końcu chciałabym podziękować mojej Rodzinie: Rodzicom, którzy włożyli ogromny trud w moje wychowanie i kształcenie oraz miłość, Dzieciom Piotrusiowi i Halszce oraz Mężowi Krzyškowi za wyrozumiałość, miłość i pozytywną energię, Dziadkowi, Rodzeństwu i ich Rodzinom za motywację oraz Eli za ogromne wsparcie duchowe i opiekę nad dziećmi, w kryzysowych sytuacjach. Z nie mniejszą siłą dziękuję moim Przyjaciołom, Beacie, Julii, Paulinie, Kasi, Gosi, Agacie i Adamowi za mocne trzymanie kciuków. Osobne podziękowania kieruję do Mietka Jagodzińskiego za motywowanie i pomoc w moim rozwoju również biznesowym oraz Pani Bogusi za nieocenioną pomoc w opiece nad dziećmi i nie tylko.

Streszczenie

We współczesnych warunkach rynkowych przedsiębiorstwa produkcyjne zmuszone są do szybkiej dostawy produktów o jakości spełniającej postawione wymagania, w cenie i czasie, które są akceptowalne przez odbiorcę. Wymaga to odpowiednio szybkiej reakcji producentów na różne zachodzące zdarzenia związane z procesem produkcyjnym. Planowanie produkcji obecnie nie może się ograniczać do rozwiązywania klasycznych problemów szeregowania zadań, ale należy wziąć pod uwagę niepewność oraz niespodziewane zdarzenia, które wpływają na realizację założonych harmonogramów.

Tematyka szeregowania zadań z uwzględnieniem występujących zaburzeń jest obecnie intensywnie rozwijającą się dziedziną. W tym obszarze umiejscowiona jest klasa problemów rozważana w tej pracy. Dla wielu z tych problemów nie istnieją formalne modele pozwalające na tworzenie efektywnych algorytmów rozwiązujących problemy decyzyjne. W niniejszej pracy doktorskiej zaproponowano zastosowanie meta modelu algebraiczno-logicznego do tworzenia formalnych modeli dla klasy niedeterministycznych problemów dyskretnej produkcji przepływowej, w której występują defekty elementów produkowanych i konieczna jest ich naprawa. Zastosowane podejście pozwala właśnie na tworzenie formalnych modeli dyskretnych procesów produkcyjnych dla potrzeb sterowania na poziomie operacyjnym.

W ramach pracy przeanalizowano problemy szeregowania zadań w produkcji przepływowej z brakami jakościowymi, uwzględniono wykrywanie różnego typu braków jakościowych oraz sposób naprawy wybrakowanych elementów. W tym celu wyróżniono trzy rodzaje maszyn specjalnych: maszyny z kontrolą jakości \mathcal{M}_Q , maszyny naprawcze \mathcal{M}_D oraz maszyny powrotne \mathcal{M}_R . Ze względu na typy i sposób wykrywania braków jakościowych wyróżniono pięć klas problemów produkcji przepływowej z różną liczbą i rodzajem maszyn specjalnych. Zmodyfikowano i opracowano nowe formalne modele algebraiczno-logiczne wybranych problemów produkcji przepływowej z ograniczeniami czasowymi oraz opracowano metodę modelowania produkcji z brakami jakościowymi za pomocą metody przełączania modeli algebraiczno-logicznej dla tej klasy problemów, uwzględniające specyficzne cechy problemów wynikające z rodzajów i konfiguracji maszyn specjalnych. Ponadto podano następujące etapy konstruowania metody przełączania modeli dla tej klasy problemów: analiza zakłóceń, podział problemu na problemy pomocnicze, ustalenie reguł przełączania, definicja zbiorów stanów przełączania i konstruowanie funkcji przełączającej.

Abstract

Nowadays, manufacturing companies are forced to deliver products quickly and of adequate quality at the same time both in price and on time that are acceptable to the customer. This requires an adequately fast reaction to various occurring events related to the production process. Nowadays, production planning cannot be limited to solving classical scheduling problems, but both uncertainty and unexpected events that affect the implementation of the planned schedules should be under consideration.

Therefore, production scheduling under uncertainty has attracted much researchers attention in recent years. However, the flow-shop problems with defects are not equivalent to the flow-shop problem with stochastic uncertainties. The most important thing in the problems is the detection of unexpected events when the schedule is executed and the possibility of including in schedule data or parameters changes which come from the occurrence of the event. There are no formal models that allow creating effective algorithms for solving this kind of decision problems. This dissertation proposes the use of an algebraic-logical metamodel (ALMM) for creating formal models for the class of non-deterministic discrete flow-shop manufacturing problems, in which there are defects of manufactured elements and their repairing is necessary. The ALMM approach allows the creation of formal models of discrete production processes in which some events that would influence job processing occurs during the process execution.

In this dissertation, the flow-shop scheduling problems with defect were analyzed. In particular, the detection of various types of quality defects and the methods of repairing them were taken into account. For this purpose, three types of special machines were distinguished: machines with quality control \mathcal{M}_Q , repairing machines \mathcal{M}_D and returning machines \mathcal{M}_R . Due to the types and methods of detecting quality defects, five classes of flow-shop scheduling problems with different number and type of special machines were specified. Formal algebraic-logical models of selected flow-shop problems with time limits were developed and modified. For modeling flow-shop problems with quality defects the algebraic-logic switching method was developed. This method takes into account the specific features of problems i.e. the types and configurations of special machines. In particular, the following elements of the presented approach are proposed: disturbances type analyzing, problem division into the special subproblems, the definition of the set of switching states, determination of the switching rules and the switching function.

Spis treści

1. Wstęp	15
1.1. Wprowadzenie	15
1.2. Geneza i cele pracy	17
1.3. Zawartość pracy	19
2. Przegląd problemów szeregowania i metod ich rozwiązywania	21
2.1. Podstawowe pojęcia teorii szeregowania.....	22
2.1.1. Zadania \mathcal{J}	22
2.1.2. Maszyny \mathcal{M}	23
2.1.3. Zasoby \mathcal{R}	24
2.1.4. Funkcja celu φ	25
2.1.5. Notacja $\alpha \beta \gamma$	26
2.2. Klasyfikacja problemów	28
2.2.1. Podział ze względu na zmienność cech i wpływ czasu	28
2.2.2. Podział ze względu na charakter cech i sposób opisu.....	29
2.3. Metody modelowania i rozwiązywania rzeczywistych problemów produkcji przepływowej	29
2.3.1. Klasyfikacja metod modelowania	29
2.3.2. Zastosowanie metod modelowania w rozwiązywaniu problemów produkcji przepływowej	30
2.4. Metody rozwiązywania problemu przepływowego z ponownym przetwarzaniem zadań	34
3. Meta model algebraiczno-logiczny (ALMM)	37
3.1. Definicja meta modelu algebraiczno-logicznego	37
3.2. Metody optymalizacji dyskretnych dynamicznych problemów DDP bazujące na meta modelu algebraiczno-logicznym.....	40
3.2.1. Trój etapowa metoda projektowania algorytmów heurystycznych przeszukiwania z lokalnym kryterium	40
3.2.2. Metody uczenia maszynowego bazujące na ALMM	41
3.2.3. Metoda zadań zastępczych (MZZ).....	43
4. Model algebraiczno-logiczny dla problemu produkcji przepływowej FS	45

4.1.	Opis problemu produkcji przepływowej FS	45
4.2.	Model algebraiczno-logiczny problemu	45
4.2.1.	Stan systemu	46
4.2.2.	Decyzje.....	49
4.2.3.	Funkcja przejścia.....	50
4.3.	Kryterium jakości	52
5.	Problemy szeregowania zadań w produkcji przepływowej z brakami jakościowymi	53
5.1.	Opis procesu wykrywania braków jakościowych.....	53
5.2.	Opis wyróżnionych problemów produkcji przepływowej z brakami jakościowymi.....	54
5.2.1.	Problem z jedną maszyną z kontrolą jakości, brak maszyny naprawczej oraz jedną maszyną powrotną ($FS-M_qM_r$)	55
5.2.2.	Problem z jedną maszyną z kontrolą jakości, jedną maszyną naprawczą oraz jedną maszyną powrotną ($FS-M_qM_dM_r$)	56
5.2.3.	Problem z jedną maszyną z kontrolą jakości, więcej niż jedną maszyną naprawczą oraz jedną maszyną powrotną ($FS-M_qmM_dM_r$).....	57
5.2.4.	Problem z jedną maszyną z kontrolą jakości, więcej niż jedną maszyną naprawczą oraz więcej niż jedną maszyną powrotną($FS-M_qmM_dmM_r$).....	57
5.2.5.	Problem z więcej niż jedną maszyną z kontrolą jakości, więcej niż jedną maszyną naprawczą oraz więcej niż jedną maszyną powrotną ($FS-mM_qmM_dmM_r$).....	58
5.3.	Opis procesu technologicznego w malarni proszkowej jako przykład produkcji przepływowej z brakami jakościowymi	59
5.3.1.	Złożoność obliczeniowa problemu przepływowego z kryterium C_{max}	60
6.	Modelowanie problemu $FS-M_qM_r$ za pomocą podejścia ALMM z zastosowaniem metody przełączania	61
6.1.	Analiza problemu	61
6.2.	Podział problemu na problemy pomocnicze	62
6.3.	Model algebraiczno-logiczny problemu pomocniczego.....	62
6.4.	Funkcja przełączająca.....	66
6.4.1.	Reguły przełączania	66
6.4.2.	Zbiór stanów przełączania S_{switch}	67
6.4.3.	Algorytm przełączania	67
7.	Modelowanie problemu $FS-M_qM_dM_r$ za pomocą podejścia ALMM z zastosowaniem metody przełączania.	71
7.1.	Analiza problemu	71
7.2.	Podział problemu na problemy pomocnicze	72
7.3.	Modele problemów pomocniczych	73

7.3.1. Model algebraiczno-logiczny drugiego problemu pomocniczego z maszyną naprawczą	74
7.4. Funkcja przełączająca.....	76
7.4.1. Rodzaje i reguły przełączania \mathcal{R}_{switch}	77
7.4.2. Zbiór stanów przełączania	78
7.4.3. Algorytm przełączania	79
8. Metoda przełączania modeli algebraiczno-logicznych dla problemów z brakami jakościowymi $FS-mM_qmM_d m M_r$.....	85
8.1. Modelowanie problemu $FS-M_qmM_d m M_r$	85
8.1.1. Analiza problemu	86
8.1.2. Podział problemu na problemy pomocnicze	87
8.1.3. Modele problemów pomocniczych.....	87
8.2. Modelowanie problemu $FS-M_qmM_d m M_r$	92
8.2.1. Analiza problemu	92
8.2.2. Podział problemu na problemy pomocnicze	94
8.3. Modelowanie problemu $FS-mM_qmM_d m M_r$	96
8.3.1. Analiza problemu	97
8.3.2. Podział problemu na problemy pomocnicze	98
8.4. Właściwości modeli problemów pomocniczych	101
9. Metoda przełączania modeli algebraiczno-logicznych dla produkcji przepływowej z brakami jakościowymi.....	103
9.1. Analiza problemu	104
9.2. Rodzaje przełączania	105
9.3. Podział problemu na problemy pomocnicze	106
9.4. Ustalanie reguł przełączania R_{switch}	106
9.5. Zbiór stanów przełączania \mathcal{S}_{switch}	107
9.6. Funkcja przełączająca.....	108
Podsumowanie.....	111
A. Skorowidz ważniejszych oznaczeń	113
B. Skorowidz ważniejszych nazw	117
C. Dowód twierdzenia.....	118

1. Wstęp

Rozprawa doktorska dotyczy zastosowania meta modelu algebraiczno-logicznego do modelowania i optymalizacji niedeterministycznego problemu dyskretnej produkcji przepływowej, w której występują defekty elementów produkowanych i konieczna jest ich naprawa.

1.1. Wprowadzenie

We współczesnych warunkach rynkowych przedsiębiorstwa produkcyjne zmuszone są do szybkiej dostawy produktów o jakości spełniającej postawione wymagania, w cenie i czasie, które są akceptowalne przez odbiorcę. Wymaga to odpowiednio szybkiej reakcji producentów na różne zachodzące zmiany. Planowanie produkcji obecnie musi uwzględniać zarówno realia rynkowe, jak i uwarunkowania prawne, w szczególności te związane z ochroną środowiska. Aktualnie zarządzanie przedsiębiorstwem bez wspomaganie przez narzędzia i systemy informatyczne jest praktycznie niemożliwe. Niezwykle ważnym obszarem, w którym można zastosować efektywnie metody optymalizacji jest planowanie produkcji. W tym obszarze rozpatrywane były klasyczne problemy szeregowania zadań i opracowane dla nich metody. I do pewnego momentu takie rozwiązania były wystarczające mimo, że nie brały pod uwagę wszystkich aspektów związanych z rzeczywistą realizacją procesu produkcyjnego. Jednak obecnie ze względu na coraz większe i różnorodne żądania wykorzystanie tych metod w rzeczywistych problemach szeregowania jest trudne lub wręcz niemożliwe z różnych powodów, m.in. trudno jest określić dokładny czas przetwarzania zadania lub występują nieprzewidziane zakłócenia wpływające na system. W pracy Pinedo (2008) przedstawiono główne różnice między modelami teoretycznymi, a rzeczywistymi problemami planowania zadań i są to:

- Liczba zadań nie jest stała, a nowe zadania stale pojawiają się w systemie.
- Harmonogram nie jest stały i konieczne są zmiany harmonogramu.
- Linie produkcyjne (systemy maszynowe) są bardziej skomplikowane niż podział na maszyny równoległe i klasyczne systemy maszyn dedykowanych.
- Wagi (priorytety) zadań zmieniają się w czasie.
- Ważne są dodatkowe preferencje w doborze maszyn.
- Maszyny nie są dostępne stale, ale istnieją ograniczenia ich dostępności według schematów zmian pracy lub innych rozkładów.

- Funkcja kary jest nieliniowa.
- Często rozważane jest więcej niż jedno kryterium jakości.
- Należy brać pod uwagę dostępną zdolność produkcyjną poszczególnych maszyn.
- Czasy przetwarzania na poszczególnych maszynach nie są znane z góry, ani przewidywalne zgodnie z rozkładami prawdopodobieństwa.
- Czasy przetwarzania nie są stałe, ale mogą się zmieniać w zależności od doświadczenia załogi.

Na podstawie powyższych różnic można stwierdzić, że przy planowaniu produkcji należy wziąć pod uwagę niepewność oraz niespodziewane zdarzenia, które wpływają na realizację założonych harmonogramów.

Ponieważ w rzeczywistych systemach produkcyjnych, ze względu na dostępność materiałów, popyt na rynku i nieoczekiwane zakłócenia, nie można dokładnie przewidzieć wszystkich parametrów związanych z podejmowaniem decyzji podczas planowania kolejności wykonywania zadań proponowane są nowe metody rozwiązywania problemów szeregowania. Między innymi w ramach podejścia polegającego na ponownym harmonogramowaniu (rescheduling) proponowane są algorytmy do rozwiązywania problemów szeregowania zadań z awariami maszyn, nowymi przychodzącymi zadaniami w trakcie symulacji i stochastycznymi czasami przetwarzania zadań (Liu (2019); Shen and Zhu (2017)). Natomiast rzadziej można znaleźć podejścia łączące kilka z tych zakłóceń oraz inne zakłócenia takie jak anulowanie zamówienia, zmiany priorytetu kolejności wykonywania zadań, niemożliwość wykonania zadania z powodu niedostępności materiałów, czy też nieudany proces technologiczny i konieczność naprawienia źle wykonanego zadania.

W niniejszej pracy, do rozwiązania postawionego problemu badawczego wykorzystano meta model algebraiczno-logiczny ALMM wieloetapowych, dynamicznych procesów decyzyjnych. Paradygmat ALMM został zdefiniowany przez Dudek-Dyduch (1989, 1990). Od tego momentu był kolejno rozwijany w pracach Dudek-Dyduch (1989, 1991, 1993, 2000); Dudek-Dyduch and Dyduch (1988); Dudek-Dyduch (2015c). ALMM pozwala na tworzenie formalnych modeli dyskretnych procesów produkcyjnych dla potrzeb sterowania na poziomie operacyjnym. Meta-model algebraiczno-logiczny ALMM jest podejściem, które łączy modelowanie z symulacją połączoną z optymalizacją i stanowi on formalną reprezentację dynamicznego dyskretnego problemu. Prace związane z ALMM rozwijane są w trzech wzajemnie powiązanych ze sobą kierunkach:

1. Tworzenie metodologii modelowania ze szczególnym uwzględnieniem dyskretnych procesów produkcyjnych - Na podstawie ALMM opracowano wiele modeli algebraiczno-logicznych dyskretnych dynamicznych problemów produkcyjnych i logistycznych m.in. szeregowanie prac w kopalni Dudek-Dyduch (2000), szeregowanie zadań z przebrojeniami zależnymi od stanu systemu Kucharska (2006), szeregowanie zadań z zasobami zależnymi od stanu Dutkiewicz (2005), problem firm wielooddziałowych Dutkiewicz et al. (2016), dynamiczny problem marszrutyzacji Kucharska et al. (2019). Ponadto ALMM jest to ujednoczone podejście do tworzenia dyskretnych modeli problemów optymalizacji, reprezentujących wiedzę o tych problemach Dudek-Dyduch et al. (2005).

2. Opracowywanie nowych metod optymalizacji bazujących na meta-modelu algebraiczno-logicznym ALMM - Opracowane metody są ogólne i służą do konstruowania wyspecjalizowanych algorytmów heurystycznych na bazie schematu meta modelu algebraiczno-logicznego (ALMM). Do metod tych należą: trój etapowa metoda projektowania algorytmów heurystycznych przeszukiwania z lokalnym kryterium Dudek-Dyduch (1990), metoda uczenia bazująca na meta-modelu algebraiczno-logicznym Dudek-Dyduch (2000); Dudek-Dyduch and Dyduch (2006a), metoda z gromadzeniem informacji Kucharska and Dudek-Dyduch (2014), Dudek-Dyduch and Kucharska (2011a), metoda zadań zastępczych Dutkiewicz (2005).
3. Rozwijanie inteligentnego systemu do optymalizacji procesów dyskretnych - celem Inteligentnego Systemu ALMM jest nie tylko rozwiązywanie problemu optymalizacji dyskretniej, ale także pomoc użytkownikom w wyborze odpowiedniej metody rozwiązywania, skonfigurowaniu algorytmów i w tworzeniu modelu problemu. Podejście to wykorzystuje bazę wiedzy połączoną z inteligentnym interfejsem użytkownika oraz dodatkowym modułem biblioteki właściwości problemów Dudek-Dyduch (2017); Dudek-Dyduch et al. (2018, 2019).

1.2. Geneza i cele pracy

Geneza niniejszej pracy związana jest z koniecznością opracowania metody rozwiązywania niedeterministycznego problemu produkcji dyskretniej przepływowej, w której występują defekty elementów produkowanych i konieczna jest ich naprawa. Polega ona na obróbce na dodatkowej maszynie naprawczej spoza marszrut technologicznej, następnie naprawione elementy powracają do ponownej obróbki na maszynę w marszrucie technologicznej. Celem optymalizacji rozważanego problemu jest jak najszybsze wykonanie wszystkich zadań. Problem taki występuje między innymi w przedsiębiorstwie usługowym, które maluje prozkowo detale i pod koniec procesu malowania sprawdzana jest jakość powłoki lakierniczej. Konieczność poprawiania wybrakowanych elementów podczas procesu produkcyjnego w rzeczywistości występuje bardzo często. Dodatkowo większość z tych problemów należy do klasy problemów NP-trudnych. Vieira et al. (2003) rozróżnia dwie kategorie nieprzewidzianych zakłóceń, występujących w normalnie działających systemach rzeczywistych:

- Zakłócenia wydajności, czyli zakłócenia związane z zasobami produkcyjnymi, takimi jak maszyna (przykładowo awarie, niedostępność narzędzi, brak operatorów);
- Zakłócenia w zamówieniach, czyli zakłócenia związane z pracą, takie jak pośpiech, anulowanie pracy, niedobór surowca zmiana priorytetu, konieczność przeróbki itp.

Zagadnienia rozpatrywane w pracy dotyczą więc drugiej kategorii nieprzewidzianych zakłóceń.

Projektowanie metod umożliwiających odpowiednie harmonogramowanie zadań (w szczególności w produkcji przepływowej) z uwzględnieniem możliwości ponownej obróbki niektórych zadań stało się w ostatnich latach przedmiotem badań naukowych. Dla wielu z tych problemów nie istnieją formalne modele pozwalające na tworzenie efektywnych algorytmów rozwiązujących problemy decyzyjne. Meta model algebraiczno-logiczny ALMM wieloetapowych, dynamicznych procesów decyzyjnych pozwala

na tworzenie formalnych modeli dyskretnych procesów produkcyjnych dla potrzeb sterowania na poziomie operacyjnym. Bazując na podejściu ALMM zaproponowano koncepcje dwóch metod do rozwiązywania problemów z występującymi zaburzeniami w trakcie realizacji.

Koncepcja modelowania zakłóceń za pomocą metody przełączania modeli algebraiczno-logicznych została zaproponowana przez E. Dudek-Dyduch i przedstawiona po raz pierwszy w pracy Grobler-Dębska et al. (2013). E. Dudek-Dyduch zaproponowała w pracach Dudek-Dyduch (2015a,b) dwuetapową metodę transformacji modeli algebraiczno-logicznych (2SALMT) (ang. TwoStage AL Model Transformation Method). Metoda ta należy do metod sztucznej inteligencji i jej idea polega na tym, że w momencie wystąpienia zaburzenia w automatyczny sposób dokonywana jest transformacja modeli. Jest to ogólna metoda, która służy do modelowania problemów produkcji dyskretniej z różnego rodzaju niedeterministycznymi zakłóceniami.

Celem rozprawy doktorskiej jest opracowanie nowych formalnych modeli algebraiczno-logicznych wybranych klas procesów produkcji przepływowej oraz zdefiniowanie ogólnej metody modelowania produkcji przepływowej z brakami jakościowymi za pomocą tzw. metody przełączania modeli algebraiczno-logicznych.

Proponuje się następującą tezę:

Istnieje możliwość zamodelowania szerokiej klasy dyskretnych procesów produkcyjnych z brakami jakościowymi metodą przełączania odpowiednich modeli algebraiczno-logicznych i wykorzystanie tej metody do efektywnego tworzenia algorytmów szeregowania.

Wstępna idea zastosowania metody przełączania modeli algebraiczno-logicznych dla problemu przepływowego z jedną maszyną naprawczą została przedstawiona we współautorstwie z E. Dudek-Dyduch oraz E. Kucharską i przedstawiona w pracy Grobler-Dębska et al. (2013), a następnie kontynuowana w Grobler-Dębska et al. (2014); Kucharska et al. (2017a,b). Jest to metoda, która została zaproponowana w celu umożliwienia odpowiedniego zamodelowania niedeterministycznego zdarzenia, jakim jest wykrycie braku jakościowego i konieczność naprawy tego braku na jednej maszynie. W pracy metoda została doprecyzowana i uogólniona dla dowolnej liczby maszyn specjalnych.

Dla wykazania tezy pracy konieczne było zrealizowanie następujących zadań:

- analiza problemów z ponownym przetwarzaniem zadań (problemy re-entrant) wynikającym ze zdefiniowanego procesu produkcyjnego lub konieczności ponownego przetwarzania zadania z powodu niespełnienia kryterium jakości;
- analiza problemów szeregowania zadań w produkcji przepływowej z brakami jakościowymi, uwzględniającą cechy procesów wykrywania braków jakościowych oraz naprawy wybrakowanych elementów;
- modyfikacja i opracowanie nowych modeli algebraiczno-logicznych wybranych problemów produkcji przepływowej z deadlineami;
- opracowanie ogólnej metody modelowania produkcji z brakami jakościowymi za pomocą metody przełączania modeli algebraiczno-logicznej dla klasy problemów produkcji przepływowej z brakami jakościowymi

1.3. Zawartość pracy

Praca składa się ze wstępu, dziewięciu rozdziałów, podsumowania i trzech dodatków.

W rozdziale drugim przedstawiono przegląd problemów szeregowania zadań oraz metody ich rozwiązywania. Podano podstawowe pojęcia stosowane w teorii szeregowania wraz notacją $\alpha|\beta|\gamma$. Dokonano klasyfikacji problemów szeregowania ze względu na zmienność cech i wpływ czasu oraz charakter zmian i sposób opisu. Dodatkowo bardziej szczegółowo opisano metody modelowania i rozwiązywania rzeczywistych problemów produkcji przepływowej. W szczególności skupiono się nad metodami rozwiązywania problemu przepływowego z ponownym przetwarzaniem zadań (ang. re-entrant).

Rozdział trzeci zawiera definicję i właściwości meta modelu algebraiczno-logicznego. Podejście to zostało wykorzystane w dalszej części pracy do modelowania i rozwiązania klasy problemów produkcji przepływowej z brakami jakościowymi. Ponadto opisano ogólne metody rozwiązywania dyskretnych dynamicznych problemów produkcyjnych bazujące na meta modelu algebraiczno-logicznym: trójeta-pową metodę projektowania algorytmów przeszukiwania heurystycznego z lokalnym kryterium, metody uczenia maszynowego bazujące na tym podejściu oraz metodę zadań zastępczych.

W rozdziale czwartym sformułowano problem produkcji przepływowej FS , w którym dla zadań dodatkowo wymagany jest czas ukończenia (deadline). Dla tego problemu przedstawiono model algebraiczno-logiczny, w szczególności opisano stan systemu wraz ze zbiorami stanów niedopuszczalnych oraz docelowych, strukturę decyzji, definicję zbioru decyzji możliwych w danym stanie oraz funkcję przejścia pomiędzy kolejnymi stanami.

Rozdział piąty zawiera analizę problemów szeregowania zadań w produkcji przepływowej z brakami jakościowymi, uwzględniającą proces wykrywania braków jakościowych oraz konieczność naprawy wybrakowanych elementów. W rozważanych problemach wyróżniono maszynę na której kontrolowana jest jakość produkowanych elementów, maszynę na której dokonywana jest naprawa wybrakowanych detali oraz tak zwaną maszynę powrotną, od której realizowane są operacje z marszruty technologicznej produktu po naprawie. Uwzględniając potrzebę lub jej brak zastosowania maszyn naprawczej i powrotnej oraz różną liczbę tych maszyn wskazano pięć klas problemów. Ponadto opisano przykład rzeczywistego problemu produkcji przepływowej, malarni proszkowej, w którym występują braki jakościowe i konieczna jest ich naprawa.

Model algebraiczno-logiczny oraz rozwiązanie bazujące na przełączaniu modeli algebraiczno-logicznych dla problemu z jedną maszyną kontroli jakości, brakiem maszyny naprawczej oraz jedną maszyną powrotną przedstawiono w rozdziale szóstym. Z kolei w rozdziale siódmym zaproponowano modelowanie i rozwiązanie bazujące na meta modelu algebraiczno-logicznym dla problem z jedną maszyną kontroli jakości, jedną maszyną powrotną oraz dodatkową jedną maszyną naprawczą, na której naprawiane są detale z wykrytymi brakami jakościowymi.

Rozdział ósmy dotyczy uogólnionego problemu przepływowego z naprawą wybrakowanych elementów z więcej niż jedną maszyną z kontrolą jakości, więcej niż jedną maszyną naprawczą oraz więcej niż jedną maszyną powrotną. W szczególności podano modyfikacje w modelu algebraiczno-logicznym, podział problemu na problemy pomocnicze, nowe elementy modeli algebraiczno-logicznych problemów pomocniczych oraz elementy funkcji przełączającej.

W rozdziale dziewiątym zaproponowano zastosowanie koncepcji przełączania modeli algebraiczno-logicznych do modelowania i rozwiązywania problemów produkcji przepływowej z brakami jakościowymi. Szczegółowo opracowano elementy metody, czyli analizę problemu niedeterministycznego, podział podział problemu niedeterministycznego na deterministyczne problemy pomocnicze, budowę modeli algebraiczno-logicznych dla wszystkich wyróżnionych problemów, określenie rodzajów przełączania między modelami, ustalenie reguł przełączania, wyróżnienie stanów przełączania oraz opracowanie algorytmu przełączającego.

Dodatek A zawiera spis ważniejszych oznaczeń używanych w pracy. W dodatku B spisano ważniejsze nazwy wraz z ich angielskimi odpowiednikami. W dodatku C przedstawiono dowód twierdzenia o minimalnym czasie wykonywania wszystkich zadań w problemie $Fm|prmu, p_{ij} = p_j|C_{max}$.

2. Przegląd problemów szeregowania i metod ich rozwiązywania

Teoria szeregowania zadań jest rozwijana od połowy lat 50-tych XX wieku. Związane jest to z publikacją artykułu Johnson (1954), w którym przedstawiono po raz pierwszy algorytm rozwiązania rzeczywistego problemu przepływowego (*ang. flow-shop*) dla dwóch maszyn. Algorytm Johnson'a (*ang. Johnson rule or algorithm*) jest znaną do dzisiaj prostą techniką wyznaczenia optymalnego uszeregowania zadań dla dwumaszynowego problemu przepływowego Gupta and Stafford (2006). Od tego momentu problemy związane z planowaniem prac w systemie przepływowym wzbudziły znaczne zainteresowanie naukowe, które trwa aż do dzisiaj. W rezultacie, powstało wiele znaczących publikacji naukowych w obszarze rozwiązania problemu flow-shop oraz innych problemów szeregowania w różnych systemach i ich wariantach, między innymi Baker (1974); Blazewicz et al. (2001); Brucker (2007); Gawiejnowicz (2008); Janiak (1999); Pinedo (2008); Sawik (1998); Smutnicki (2002); Graham et al. (1979); Kacprzyk and Węglarz (2002). W głównej mierze są to tzw. klasyczne podejścia, w których algorytmy oparte są na werbalnych opisach problemów. Niemniej jednak rzeczywiste problemy produkcyjne są na tyle złożone i cechują się dynamicznymi zmianami oraz występowaniem nieprzewidzianych zaburzeń, że w większości przypadków nie da się ich sprowadzić do klasycznej postaci szeregowania zadań. Rozwiązywanie takich problemów stanowi przedmiot najnowszych badań w tej dziedzinie (González-Neira et al., 2017; Rossit et al., 2018; Ritzinger et al., 2016; Ouelhadj and Petrovic, 2008; Bukkur et al., 2018). W rzeczywistych systemach produkcyjnych mogą występować następujące zakłócenia: awaria maszyn, anulowanie zamówienia, zmiany terminu dostawy, niepewny termin płatności, niepewny czas przetwarzania, ponowne przetwarzanie związane z kontrolą jakości, dodawanie lub usuwanie operacji.

W rozdziale przedstawiono przegląd literatury związany z problematyką szeregowania zadań, w szczególności w systemie przepływowym i metod ich rozwijania. W pierwszej części rozdziału pokrótce przedstawiono teoretyczne podstawy. W drugiej części przedstawiono klasyfikację problemów szeregowania zadań, obejmującą dwa problemy statyczne i dynamiczne oraz problemy deterministyczne, niedeterministyczne i rozmyte. W trzeciej części przedstawiono przegląd metod modelowania i optymalizacji problemów szeregowania zadań. Omówiono rozwiązania, w których algorytmy bazują na formalnym modelu wraz z przykładami zastosowania.

2.1. Podstawowe pojęcia teorii szeregowania

Problemy szeregowania zadań formułowane są następująco: dany jest zbiór zadań oraz zbiór maszyn służących do wykonania tych zadań oraz wykonanie wszystkich zadań może wymagać zaangażowania dodatkowych określonych zasobów. Problem optymalnego szeregowania zadań polega na wyznaczeniu rozwiązania, dla którego przyjęta funkcja celu osiąga wartość najmniejszą. Funkcja celu reprezentuje pewną praktyczną miarę jakości funkcjonowania systemu.

Każdy problem szeregowania można zapisać jako czwórkę parametrów (Gawiejnowicz (2008)):

$$(\mathcal{J}, \mathcal{M}, \mathcal{R}, \varphi) \quad (2.1)$$

gdzie:

- \mathcal{J} – zbiór zadań do wykonania,
- \mathcal{M} – zbiór maszyn, które będą wykonywać zadania,
- \mathcal{R} – zbiór dodatkowych zasobów potrzebnych do wykonywania tych zadań,
- φ – funkcja celu.

Poniżej przedstawiono najważniejsze cechy zadań, maszyn, zasobów i funkcji celu:

2.1.1. Zadania \mathcal{J}

Każde zadanie J składa się z operacji, zazwyczaj z zadany porządkiem ich wykonywania. Operacja i dla j -tego zadania oznaczana jest przez O_{ij} . W szczególności, zadanie może składać się tylko z jednej operacji. Zbiór zadań oznaczany jest przez $\mathcal{J} = \{J_1, J_2, \dots, J_n\}$, a zbiór operacji j -tego zadania przez $J_j = \{O_{1j}, O_{2j}, \dots, O_{n_j}\}$.

Każde zadanie J_j należące do \mathcal{J} może być scharakteryzowane poprzez następujące parametry Grabowski et al. (2003):

n_j – liczba operacji,

p_{ij} – czas wykonywania operacji O_{ij} (i -tej operacji zadania J_j). Czasy p_{ij} mogą być:

- ustalone z góry i niezmiennie,
- zależne od momentu rozpoczęcia ich realizacji (np. im późniejszy termin rozpoczęcia tym dłuższy czas realizacji),
- zależne od ilości przydzielonego im zasobu (np. przydzielenie większej ilości zasobu może skrócić czas wykonywania operacji)
- zmienną losową o zadanym rozkładzie (np. rozkład wykładniczy)

d_j — pożądaný czas zakończenia wykonywania zadania (due date).

\bar{d}_j – termin krytyczny zakończenia wykonania zadania (deadline). Zadanie J_j musi zostać zakończone przed upływem tego terminu.

w_j – waga (priorytet), którą można interpretować jako koszt oczekiwania na wykonanie zadania J_j w ciągu jednostki czasu.

s_{kj} — czas przebrojenia (przygotowania maszyny) pomiędzy zadaniem J_k a zadaniem J_j . Czasy takich przebrojeń mogą być Chudzik and Janiak (1998):

- sekwencyjnie niezależne (czas przebrojenia maszyny pomiędzy kolejnymi zadaniami jest niezależny od pierwszego z zadań,
- sekwencyjnie zależne (czas przebrojenia maszyny pomiędzy kolejnymi zadaniami zależy od zadania poprzedzającego),
- zależne od ilości przydzielonych zasobów.

c_{kj} – koszt przebrojenia pomiędzy zadaniem J_k a zadaniem J_j . Koszt ten, analogicznie do czasu przebrojeń, również może być sekwencyjnie zależny lub sekwencyjnie niezależny, albo zależeć od ilości przydzielonych zasobów.

Ponadto zadania ze względu na możliwość podziału w trakcie ich realizacji dzieli się na:

- zadania podzielne, czyli takie, które można podzielić na mniejsze zadania (np. partia składająca się z kilku elementów),
- zadanie niepodzielne, czyli takie, które wykonuje się w całości (np. operacja technologiczna jednego elementu)

oraz ze względu na możliwość przerywania dzieli się na:

- zadana podzielne w czasie (przerywalne), jeżeli zadanie można przerwać w dowolnym momencie bez żadnych kosztów i wznowić wykonanie w późniejszym czasie (np. operacja obliczeniowa w systemie komputerowym),
- zadania niepodzielne w czasie (nieprzerywalne), czyli takie, których wykonywanie nie może być przerywane w żadnym momencie bez kosztów (np. operacja technologiczna w dyskretnym procesie produkcyjnym).

Dodatkowo, w zbiorze wszystkich zadań \mathcal{J} mogą być określone dodatkowe ograniczenia związane z kolejnością wykonywania zadań ze zbioru. Jeśli nie istnieją takie ograniczenia, czyli nie występują relacje częściowego porządku pomiędzy zadaniami, to zadania są zadaniami niezależnymi. W przeciwnym przypadku, gdy zadana jest przynajmniej jedna taka relacja w zbiorze zadań, to są to zadania zależne.

Opisane powyżej parametry nie wyczerpują wszystkich możliwości związanych z charakterystyką zadań w teorii szeregowania. Więcej własności zadań można znaleźć w Janiak (1999).

2.1.2. Maszyny \mathcal{M}

Zadania wykonywane są przez maszyny należące do zbioru $\mathcal{M} = \{M_1, M_2, \dots, M_m\}$. W klasycznej teorii szeregowania w danej chwili czasu każde zadanie może być wykonywane przez co najwyżej jedną maszynę i każda maszyna w danej chwili może wykonywać co najwyżej jedno zadanie. W zbiorze maszyn wyróżnia się zarówno różne funkcje maszyn, jak i różne systemy produkcyjne Sawik (1998):

1. Ze względu na spełniane funkcje maszyny dzieli się na:
 - równoległe – spełniające te same funkcje,

- dedykowane – różniące się spełnianymi funkcjami.
2. W przypadku maszyn równoległych każde zadanie J_j wymaga wykonania tylko jednej operacji na dowolnej maszynie M_i . Rozróżniane są trzy rodzaje maszyn równoległych:
- (a) identyczne – każda z maszyn pracuje z taką samą prędkością,
 - (b) jednorodne – maszyny różnią się prędkością, ale ich prędkość jest stała i nie zależy od wykonywanej operacji (są „proporcjonalne”),
 - (c) dowolne – prędkości maszyn zależą od wykonywanej operacji.
3. W przypadku maszyn dedykowanych rozróżniane są następujące systemy:
- (a) przepływowy (flow-shop) – każde zadanie wymaga wykonania przez wszystkie maszyny w ściśle określonym porządku,
 - (b) ogólny, gniazdowy (job-shop) – zarówno podzbiory maszyn mających wykonać operacje, jak i kolejność ich wykonywania są dowolne, choć określone dla każdego zadania,
 - (c) otwarty (open-shop) – wykonanie każdego zadania wymaga operacji na wszystkich maszynach, ale kolejność ich wykonywania jest dowolna i nieustalona.

Ponadto, w przypadku maszyn dedykowanych rozróżnia się również systemy *re-entrant*, w których zadania muszą przejść przez wszystkie maszyny w ściśle określonym porządku (według marszruty technologicznej) z wielokrotnymi powtórzeniami części lub całej marszruty technologicznej, aż wymagana jakość wykonania zostanie osiągnięta (Li et al. (2013)). Takie systemy często są zastosowane przy produkcji części elektrycznych (Danping and Lee (2011)). W nowszej literaturze znajdują się również systemy przepływowe typu hybrydowego (hybrid flow-shop) ((Dugardin et al., 2010; Hyun-Seon, 2009)) lub elastyczne (flexible flow-shop)(Pinedo (2008)). Określenie hybrydowe lub elastyczne używane jest wtedy, kiedy zadania muszą przejść określony ciąg technologiczny, ale maszyn danego typu jest wiele i pomiędzy maszynami z ciągu technologicznego występują magazyny międzyoperacyjne. Dla elementów znajdujących się w tych magazynach podejmowana jest decyzja przez którą konkretną maszynę danego typu zadanie ma być przetwarzane. Analogicznie, określenia systemu hybrydowego lub elastycznego stosowane są dla systemu gniazdowego (hybrid job-shop lub flexible job-shop).

2.1.3. Zasoby \mathcal{R}

Ze względu na różnorodność zasobów oraz ich własności są one klasyfikowane pod wieloma względami. Przede wszystkim dzieli się je ze względu na podzielność Gawiejnowicz (2008):

- dyskretne (np. narzędzia, roboty),
- podzielne w sposób ciągły (np. paliwo, prąd, gaz).

Ponadto wyróżnia się trzy podstawowe kategorie zasobów Smutnicki (2002):

- odnawialne – zasoby niezużywające się, dla których ograniczona jest jego dostępność tylko w pewnych chwilach czasu (np. procesor, maszyna, robot, siła robocza, pamięć maszyn cyfrowych),

- nieodnawialne – zasoby zużywające się, dla których ograniczona jest ich globalna ilość (np. surowce, materiały podlegające zużyciu, nakłady finansowe, energia),
- podwójnie ograniczone – zasoby, dla których ograniczona jest zarówno dostępność w danej chwili, jak i zużycie łączne (np. kapitał).

Dodatkowo spośród wielu istotnych cech zasobów wyróżnia się następujące parametry:

- dostępność w danym czasie,
- ilość,
- koszt (może być liczony w zależności od czasu wykorzystania lub od ilości zużytych jednostek),
- dopuszczalne obciążenie jednostki zasobu - dotyczy liczby operacji (zadań), które mogą być jednocześnie wykonywane przy użyciu tej jednostki; najczęściej przyjmuje się, że liczba ta jest równa jedności.

Opisane powyżej parametry nie wyczerpują wszystkich możliwości Smutnicki (2002); Janiak et al. (2002). Natomiast ze względu na to, że w tej pracy w rozważanych problemach produkcji przepływowej, nie uwzględnia się żadnych dodatkowych zasobów, pominięto bardziej szczegółowy opis parametrów zasobów.

2.1.4. Funkcja celu φ

Rozwiązaniem problemu szeregowania zadań jest uszeregowanie. Uszeregowanie to przypisanie maszyn (i ewentualnie zasobów) do zadań w czasie w taki sposób, że spełnione zostały następujące warunki Gawiejnowicz (2008):

- w każdym momencie, każda maszyna jest przypisana do co najwyżej jednego zadania i każde zadanie jest przetwarzane co najwyżej przez jedną maszynę,
- zadanie J_j , dla $1 \leq j \leq n$, jest przetwarzane w przedziale czasu $[r_j, +\infty)$,
- wszystkie zadania zostały ukończone,
- jeśli istnieją ograniczenia kolejnościowe zadań, to zadania są wykonywane w kolejności zgodnej z tymi ograniczeniami,
- jeśli istnieją ograniczenia zasobów, to są one spełnione,
- jeśli zadania są nieprzerywalne, to żadna operacja nie została przerwana.

Krócej mówiąc uszeregowanie jest to ustalenie kolejności wykonywania operacji na poszczególnych maszynach oraz wyznaczanie momentów, w których rozpoczyna się realizacja tych operacji oraz w przypadku rozpatrywania dodatkowych zasobów, należy również określić ich przydział. Rozwiązanie spełniające wszystkie ograniczenia w problemie nazywane jest rozwiązaniem dopuszczalnym. Zadanie optymalizacyjne polega na wyznaczeniu rozwiązania dopuszczalnego, dla którego przyjęte kryterium jakości (funkcja celu) osiąga wartość optymalną.

Najczęściej spotykane kryteria są następujące Grabowski et al. (2003):

$C_{max} = \max\{C_i : i = 1..n\}$ – maksymalny czas zakończenia wykonania wszystkich zadań (długość uszeregowania), gdzie C_i oznacza czas zakończenia zadania J_i .

$L_{max} = \max\{L_i : i = 1..n\}$ - maksymalna nieterminowość $L_i := C_i - d_i$, gdzie d_i oznacza pożądaną czas zakończenia wykonywania zadania J_i ,

$T_{max} = \max\{T_i : i = 1..n\}$ - maksymalne opóźnienie $T_i := \max\{0, C_i - d_i\}$,

$c_{max} = \max\{c_i : i = 1..n\}$ - maksymalny koszt wykonania zadania,

$\sum_{i=1}^n w_j C_j$ – suma ważonych czasów zakończenia wykonania zadań,

$\sum_{i=1}^n w_j T_j$ - suma ważonych opóźnień,

$\sum_{i=1}^n c_i$ – całkowity koszt wykonania zadań,

$\sum_{i=1}^n w_j U_j$ – suma ważonych opóźnień zadań $U_i := \begin{cases} 0 & \text{gdy } C_i \leq d_i \\ l & \text{w przeciwnym wypadku} \end{cases}$

Oprócz powyższych, rozpatrywanych jest wiele innych kryteriów, na przykład związanych z zasobami (całkowita ilość zużytego zasobu, całkowita ważona ilość zużytego zasobu, maksymalny koszt zużycia zasobu). W niektórych z powyżej rozpatrywanych kryteriów może być też dodatkowo uwzględniany koszt przebrojeń, przez dodanie do kryterium odpowiedniego składnika.

2.1.5. Notacja $\alpha|\beta|\gamma$

Notacja $\alpha|\beta|\gamma$ jest pewną uniwersalną klasyfikacją, która została początkowo przedstawiona i opracowana w Graham et al. (1979), a następnie dopracowana i rozszerzona Kan (2012). Ułatwia ona nie tylko zdefiniowanie problemu, ale także określenie jego szczególnych własności. Wprowadzona klasyfikacja pozwoliła też na uporządkowanie rezultatów badań. Posługując się wprowadzoną notacją dokonano zestawienia (zgrupowania) uzyskanych wyników, takich jak złożoność obliczeniowa problemu oraz znane metody rozwiązania. Są one prezentowane przykładowo w pracach Gawiejnowicz (2008); Janiak (1999); Grabowski et al. (2003); Smutnicki (2002). Klasyfikacja wyrażona jest w postaci trójpolowej notacji $\alpha|\beta|\gamma$:

α - opisuje zbiór maszyn \mathcal{M} czyli określa typ zagadnienia,

β - opisuje zbiór zadań oraz dodatkowe specyficzne ograniczenia zagadnienia

γ - określa kryterium optymalizacji (funkcję celu).

Symbol α jest złożeniem dwóch symboli α_1, α_2 . Symbol α_1 charakteryzuje rodzaj maszyn i może przyjmować następujące wartości Grabowski et al. (2003)

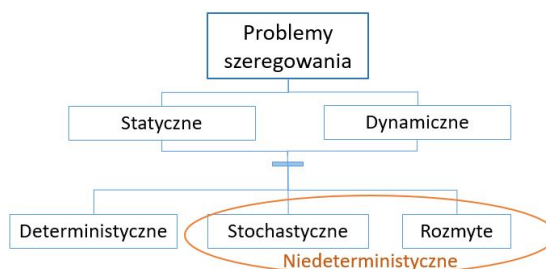
- F – system przepływowy (flow-shop)
- FP lub F^* – system przepływowy permutacyjny (permutation flow-shop), w którym dodatkowo kolejność obsługi zadań na wszystkich maszynach jest jednakowa,
- O – system otwarty (open-shop),

- J – system gniazdowy (job-shop) ,
- G - system ogólny (general-shop), w którym każde zadanie jest pojedynczą operacją, zaś zależności technologiczne dane są dowolnym grafem,
- P – system identycznych maszyn równoległych,
- Q – system jednorodnych maszyn równoległych,
- R – system niejednorodnych (niezależnych) maszyn równoległych ,
- \emptyset (symbol pusty) – system zawiera dokładnie jedną maszynę i w związku z tym nie jest określany jej rodzaj.

Drugi symbol α_2 określa liczbę maszyn w zbiorze \mathcal{M} . Jeśli liczba ta nie jest określona z góry to używa się symbolu pustego (\emptyset) mającego sens dowolnej liczby maszyn w systemie. Symbol pusty jest najczęściej pomijany w zapisie. Klasyfikacja ta nie obejmuje nowych problemów szeregowania przykładowo związanych z elastycznymi systemami wytwarzania.

Symbol β opisuje zbiór zadań i określa występowanie dodatkowych założeń i ograniczeń. β może zawierać dowolny podzbiór symboli, których znaczenie jest dość często modyfikowane. Spowodowane jest to głównie pojawiającymi się specyficznymi ograniczeniami wynikającymi z praktyki szeregowania. W związku z tym poniżej przedstawione zostaną te najczęściej spotykane i opisywane w literaturze.

- setup (przebrojenia) - występują czasy przebrojenia maszyn pomiędzy wykonywaniem zadań (mogą dodatkowo określone być rodzaje tych przebrojeń: sekwencyjnie niezależne, sekwencyjnie zależne, zależne od zasobów)
- batch (porcjowanie) - występuje żądanie grupowania zadań, tzn. wszystkie zadania z pewnej rodziny muszą być realizowane po kolei, nie można dzielić zadań z danej rodziny na partie (grupy), przy czym można zmieniać kolejność zadań w obrębie rodziny
- no wait (bez czekania) - czas rozpoczęcia operacji następnych (w sensie porządku technologicznego) jest równy czasowi zakończenia operacji poprzednich, czyli zadanie jest wykonywane bez przerw
- pmtn (przerywać) - dopuszcza się możliwość przerywania wykonywania operacji
- prec - istnieje narzucony częściowy porządek technologiczny wykonywania zadań
- tree, outtree, intree, sp-graph - graf relacji w postaci drzewa zakorzenionego, nie zakorzenionego lub grafu szeregowo - równoległego
- r_j - zadania mają różne terminy zgłoszeń
- $p_{ij} = 1$ - czasy wykonania wszystkich operacji są jednakowe i równe jedności lub na przykład $p_{ij} < P$ co oznacza, że czas wykonania operacji jest ograniczony stałą P)
- no store (bez magazynu) - brak możliwości lub zakaz składowania w magazynie międzyoperacyjnym; w tym przypadku może odbywać się składowanie jedynie na maszynach poprzez sztuczne wydłużenie czasu obróbki.



Rysunek 2.1: Systematyka problemów szeregowania. Opracowanie własne na podstawie Karkula (2013).

Symbol γ przyjmuje wartość jednej z symbolicznych postaci funkcji celu (podanych punkcie 2.1.4). Dlatego jest albo maksymalnym kosztem φ_{max} albo sumarycznym kosztem ważonym $\sum w_j \varphi_j$ Sysło et al. (1995). W związku z tym można przyjąć, że

$$\begin{aligned} \varphi_{max} &\in \{C_{max}, L_{max}, T_{max}, c_{max}\}, \\ \sum w_j \varphi_j &\in \{\sum w_j C_j, \sum w_j T_j, \sum w_j U_j, \sum c_j\} \end{aligned} \quad (2.2)$$

Przykładowo używając podanej notacji, przepływowy problem szeregowania zadań na dwóch maszynach, bez opóźnień między operacjami z kryterium maksymalnego czasu zakończenia wykonania wszystkich zadań zapisywany jest następująco:

$$F2|no - wait|C_{max}.$$

2.2. Klasyfikacja problemów

Istnieje wiele typów problemów oraz kategorii ich klasyfikacji. Jedną z klasyfikacji umożliwiającą usystematyzowanie problemów szeregowania jest klasyfikacja modeli przedstawiona przez Karkula (2013). Systematyka ta dzieli modele ze względu na zmienność cech i wpływ czasu lub na charakter cech i sposób opisu. Zatem wyróżniamy następujące rodzaje problemów:

- statyczne i dynamiczne,
- deterministyczne, probabilistyczne i rozmyte.

2.2.1. Podział ze względu na zmienność cech i wpływ czasu

Podział na systemy statyczne i dynamiczne jest związany z możliwością zmian systemów w czasie.

W problemach statycznych upływający czas nie odgrywa roli. Zmienne stanu systemu nie są zależne od czasu, a także parametry systemu nie zmieniają się. W szeregowaniu zadań dla problemu statycznego plan przydziału zadań do zasobów sporządzany jest z góry (Smutnicki (2002)). Jest to możliwe tylko wtedy, gdy zadania wraz z ich charakterystykami czasowymi znane są z góry.

W szeregowaniu dynamicznym system jest reprezentowany w perspektywie upływającego czasu. Stan systemu zmienia się w czasie oraz właściwości i atrybuty są zależne od czasu.

2.2.2. Podział ze względu na charakter cech i sposób opisu

Podział na problemy deterministyczne i stochastyczne jest związany ze sposobem opisu problemu i charakterem cech.

Problemy deterministyczne szeregowania zadań są to problemy, w których cechy systemu zdefiniowane są wcześniej lub obliczane na bieżąco według znanych zależności czasowych. W problemach tych nie występują zmienne losowe, nie występują losowe zdarzenia.

Stochastyczne problemy szeregowania zadań to problemy, w których pewne zdarzenia występują losowo, czyli nie istnieje określony schemat ich występowania. Zatem w takich problemach charakterystyki zadań, zasobów lub maszyn są realizacjami zmiennych losowych o znanych lub nieznanach (estymowanych) rozkładach prawdopodobieństwa, bądź dane napływają w trakcie funkcjonowania systemu (*ang. stochastic uncertainties*).

Problemy rozmyte są to problemy, w których wartości danych liczbowych są wielkościami rozmytymi, ale znane są odpowiednie funkcje przynależności lub ich parametry Smutnicki (2002). Problemy szeregowania zadań w takich systemach należą do klasy problemów z niepewnością Behnamian (2016).

Rzeczywiste problemy produkcyjne zazwyczaj należą do klasy dynamicznych, niedeterministycznych problemów. Często w nich występują niezaplanowane zdarzenia takie jak pojawienie się nagłego zamówienia lub awaria maszyny, czy też niedostępność wystarczającej ilości zasobów (np. nieprzewidziana nieobecność pracownika, spadki natężenia prądu, niższa kaloryczność gazu). Ponadto możliwe jest również pojawienie się defektu elementu produkowanego i związku z tym naprawa tego defektu lub ponowne wykonanie zamówienia. Sytuacje te powodują występowanie niezaplanowanych zmian parametrów systemu produkcyjnego.

2.3. Metody modelowania i rozwiązywania rzeczywistych problemów produkcji przepływowej

W celu odpowiedniej reprezentacji wiedzy problemu tworzy się model formalny. W reprezentacji wiedzy podstawową rolę odgrywa informacja numeryczna Flasiński (2011). Poniżej przedstawiono klasyfikację wybranych modeli reprezentacji wiedzy, które stosowane są w literaturze do problemów szeregowania zadań:

- modele inspirowane matematyką,
- modele inspirowane naturą.

2.3.1. Klasyfikacja metod modelowania

Do rozwiązywania rzeczywistych problemów produkcyjnych używanych jest wiele modeli formalnych. Do modeli matematycznych można zaliczyć:

- Formalizm DEVS (*ang. Discrete Event System Specification*),

- Programowanie z ograniczeniami (*ang. Constraint Programming*),
- Automaty komórkowe (*ang. Cellular Automata*),
- Meta-model algebraiczno-logiczny.

Do modeli inspirowanych naturą można zaliczyć:

- Modele oparte na paradygmacie agentowym,
- Modele ewolucyjne,
- Modele stadne.

2.3.2. Zastosowanie metod modelowania w rozwiązywaniu problemów produkcji przepływowej

Formalizm DEVS stanowi formalne podejście do opisu systemów zdarzeń dyskretnych i wykorzystuje zalety modelu obiektowego Karkula (2013). Został on zaproponowany w pracy Zeigler (1984). Formalizm ten umożliwi opis działania systemów na dwóch poziomach:

- zatomizowany model DEVS (*ang. Atomic DEVS*),
- sprzężony model DEVS (*ang. Coupled DEVS*).

Na pierwszym poziomie zapisuje się zachowanie dyskretnych zdarzeń systemu za pomocą sekwencji deterministycznych przejść pomiędzy stanami, w jaki sposób system reaguje na zdarzenia zewnętrzne oraz jak generuje zdarzenia wyjściowe.

Definicja 1 Zatomizowany model DEVS zapisany jest jako następująca struktura:

$$DEVS = \langle X, S, Y, \delta_{int}, \delta_{ext}, \lambda, ta \rangle \quad (2.3)$$

gdzie:

X - zbiór zdarzeń wejściowych,

S - zbiór kolejnych stanów,

Y - zbiór zdarzeń wyjściowych,

$\delta_{int} : S \rightarrow S$ - wewnętrzna funkcja przejścia,

$\delta_{ext} : Q \rightarrow S$ - zewnętrzna funkcja przejścia, gdzie Q jest zbiorem wszystkich stanów,

$\lambda S \rightarrow Y$ - funkcja wyjścia,

$ta : S \rightarrow \mathbb{R}^+ \cup \{0\}$ - funkcja odmierzenia czasu.

Na poziomie sprzężonym system opisywany jest jako sieć połączeń (relacji) między modelami. Dzięki temu zdarzenia wyjściowe jednego modelu mogą być zdarzeniami wejściowymi sprzężonego z nim modelu. Dokładny opis działania sprzęgania modeli można znaleźć m.in. w Zeigler (1984). Natomiast definicja jest następująca:

Definicja 2 Sprzężony model DEVS zapisany jest jako następująca struktura:

$$N = \langle X, Y, D, M_i, C_{xx}, C_{yx}, C_{yy}, Select \rangle \quad (2.4)$$

gdzie:

X - zbiór zdarzeń wejściowych,

Y - zbiór zdarzeń wyjściowych,

D - zbiór nazw modeli,

$\{M_i\}$ - zbiór modeli, gdzie $\forall i \in D$, M_i jest modelem zatomizowanym DEVS lub modelem sprzężonym DEVS,

$C_{xx} \subseteq X \times \bigcup_{i \in D} X_i$ - zbiór sprzężonych zewnętrznych zdarzeń wejściowych,

$C_{yx} \subseteq \bigcup_{i \in D} Y_i \times \bigcup_{i \in D} X_i$ - zbiór sprzężeń wewnętrznych,

$C_{yy} : \bigcup_{i \in D} Y_i \rightarrow Y^\phi$ - funkcja sprzężona zewnętrznych zdarzeń wyjściowych,

$Select : 2^D \rightarrow D$ - funkcja, która określa, jak wybrać zdarzenie ze zbioru jednoczesnych zdarzeń (ang. tie-breaking function);

Formalizm DEVS często jest stosowany do opisu złożonych rzeczywistych procesów produkcyjnych ze względu na, że jest narzędziem umożliwiającym analizę systemów produkcyjnych uwzględniając ich dynamikę. Ze względu na złożoność tworzenia modelu, metodę tę używa się tylko do bardzo złożonych problemów. Przykłady zastosowania tej metody dla problemów szeregowania można znaleźć m.in. w Lin and Chen (2015); Pujo et al. (2006).

Do charakterystycznych cech układów zdarzeń dyskretnych można zaliczyć fakt, że zmienne opisujące działanie systemu, ulegają zmianie tylko w dyskretnych, zazwyczaj losowych, momentach czasu.

Programowanie z ograniczeniami Kolejną techniką modelowania jest programowanie z ograniczeniami. Istotnym elementem tej metody, wpływającym na efektywność jej działania, jest sposób modelowania ograniczeń problemu. Odpowiedni dobór ograniczeń skutkuje tym, czy zostanie znalezione rozwiązanie lub jak szybko zostanie ono znalezione. Znaczenie modelowania w programowaniu z ograniczeniami jest przedmiotem wielu prac, m.in. Freuder (1999), Puget (2004).

Definicja 3 Problem spełniający ograniczenia (ang. Constraint Satisfaction Problem (CSP)) jest to trójka:

$$CSP = \langle X, D, C \rangle \quad (2.5)$$

gdzie:

$X = \{X_1, \dots, X_n\}$ - zbiór zmiennych,

$D = \{D_1, \dots, D_n\}$ - zbiór dziedzin odpowiednich zmiennych,

$C = \{C_1, \dots, C_m\}$ - zbiór ograniczeń.

Każda zmienna X_i może przyjmować wartości z niepustej dziedziny D_i . Każde ograniczenie $C_j \in C$ jest parą $\langle t_j, R_j \rangle$, gdzie $t_j \subset X$ jest podzbiorem k zmiennych i R_j jest relacją k -argumentową na odpowiednim podzbiórze dziedziny D_j .

Programowanie z ograniczeniami ma wiele zastosowań w modelowaniu szeregowania zadań, zarówno dla procesów dynamicznych, deterministycznych jak i stochastycznych. Ta technika ma szczególne zastosowanie dla problemów z wieloma ograniczeniami. Rozwiązania oparte na technice programowania z ograniczeniami dla rzeczywistych problemów produkcji przepływowej można znaleźć m.in. w Novas and Henning (2014), Fang et al. (2013).

Automaty komórkowe Automaty komórkowe, to kolejny formalizm, który umożliwia formalne modelowanie dyskretnego procesu dynamicznego i może stanowić reprezentację wiedzy o problemie. W automatach komórkowych struktura systemu opisana jest przez siatkę komórek oraz ich stany, przejścia i reguły tych przejść, które są modelami matematycznymi. Automaty komórkowe dzielimy Dudek-Dyduch and Wąs (2006):

- ze względu na jednorodność stanów i reguł przejścia pomiędzy stanami
- ze względu na sposób aktualizacji stanów (synchroniczne i niesynchroniczne)
- ze względu na typ siatki
- ze względu na definicję sąsiedztwa

Automat homogeniczny posiada następujące cechy: jednakowy zbiór stanów dla całej siatki, jednakowy zbiór reguł dla całej siatki, stała siatka automatu, jednakowy schemat sąsiedztwa, jednakowa metoda aktualizacji stanów komórek na całej siatce. Zdecydowana większość stosowanych dziś automatów stanowią asynchroniczne, niehomogeniczne automaty (czyli automaty, które nie spełniają choć jednej z wymienionych powyżej cech). W szeregowaniu zadań automaty komórkowe bardziej sprawdzają się do szeregowania problemów na maszynach równoległych Kucharska et al. (2016); Swiecicka and Seredynski (2000); Swiecicka et al. (2006); Seredynski and Zomaya (2002), nie mniej jednak podjęto próby zastosowania tego formalizmu do problemów przepływowych Abdolzadeh and Rashidi (2010), Antczak et al. (2009).

Meta-model algebraiczno-logiczny ALMM Meta-model algebraiczno-logiczny ALMM jest podejściem, które łączy modelowanie z symulacją połączoną z optymalizacją. Stanowi on formalną reprezentację dynamicznego dyskretnego problemu, definiując zbiór stanów S , zbiór decyzji U , funkcję przejścia f , wyróżniony stan początkowy s_0 oraz dwa wyróżnione podzbiory stanów: zbiór stanów docelowych S_G oraz zbiór stanów niedopuszczalnych S_N . Cechą charakterystyczną podejścia ALMM jest to, że

funkcja przejścia służy do wyznaczania kolejnych stanów i odpowiadających im momentów czasu i nie występuje stałe taktowanie czasowe. Podejście *ALMM* zostało wykorzystane do zamodelowania rozważanych w pracy problemów produkcji przepływowej z brakami jakościowymi oraz opracowania algorytmów optymalizacyjnych dla tego problemu, w związku z tym szczegółowy opis tej metody znajduje się w rozdziale 3.

Modele oparte na paradygmacie agentowym Bożejko and Pempera (2012) Metody wieloagentowe charakteryzują się wykorzystaniem do rozwiązania problemu pewnych specjalnych autonomicznych jednostek programowych zwanych agentami. Zwykle stanowią one system rozproszony i mają zdolność monitorowania swego otoczenia, reagowania na zmiany środowiska oraz podejmowania samodzielnych decyzji. Agenci działają w pewnej przestrzeni, zwanej środowiskiem wieloagentowym. Współdziałanie agentów, dla osiągnięcia wspólnego celu bazuje na odpowiednim mechanizmie koordynacji. Każdy agent jest kompletnym systemem rozwiązywania bazującym na wiedzy dotyczącej pewnego podproblemu lub problemu szczególnego. Może on reprezentować zarówno dedykowany algorytm, zestaw algorytmów jak i moduł ekspertowy. Metoda wieloagentowa jest właściwie kompozycją metod i algorytmów zaprojektowanych w innych technikach. Stosowana jest również w szeregowaniu zadań z niepewnościami Chu et al. (2015).

Modele ewolucyjne Bożejko and Pempera (2012) Metody ewolucyjne odwołują się do natury zakładając, że niejawnym celem ewolucji jest optymalizacja dopasowania osobników do środowiska. Cechą charakterystyczną tych metod jest tworzenie podzbioru rozproszonych rozwiązań nazywanego populacją. Każde rozwiązanie zwane osobnikiem jest kodowane przez zbiór jego atrybutów, tak zwane chromosomy. Populacja jest kontrolowana przez cykliczne następujące po sobie procesy reprodukcji, krzyżowania i mutacji oraz przeżycia lub selekcji. Poniżej przedstawiono ogólny schemat algorytmów ewolucyjnych Smutnicki (2002). Pierwszym krokiem jest inicjalizacja populacji początkowej. Zazwyczaj jest ona tworzona w sposób losowy. Po zbudowaniu populacji określany jest stopień przystosowania każdego jej osobnika. Miara przystosowania zależy od oceny, jaką dany osobnik uzyska z punktu widzenia przyjętego kryterium.

Aby zdefiniować model ewolucyjny, należy określić w związku z tym następujące elementy: sposobu kodowania atrybutów w chromosomach, postać funkcji przystosowania, schemat wyboru puli rodzicielskiej i kojarzenia rodziców, operatorów krzyżowania, schemat mutacji, schemat selekcji.

Metody ewolucyjne są bardzo szeroko stosowane w rzeczywistych problemach szeregowania zadań i są implementowane specyficznie do rozważanego problemu. Stosuje się je zarówno w złożonych deterministycznych Danping and Lee (2011); Lin et al. (2011) jak i stochastycznych Almeder and Hartl (2013); Aytug et al. (2005); Horng et al. (2012) problemach produkcji przepływowej.

Modelowanie stadne Algorytmy stadne należą do rodziny algorytmów populacyjnych które odwierciedlają procesy zachodzące w naturalnych systemach roju lub stada i należą one do klasy algorytmów probabilistycznych. Podobnie jak algorytmy genetyczne wykorzystują mechanizmy istniejące w populacji takie jak proces selekcji i mutacji osobników do eksploracji przestrzeni rozwiązań. Istnieje wiele

struktur (modeli) tych algorytmów, które zostały zaprojektowane na podstawie zbiorowych zachowań owadów tworzących społeczności, praw łączących zachowanie ryb, stadami ptaków oraz stadami zwierząt. Przykładowymi tych struktur algorytmów należących do dziedziny inteligencji stadnej są: mrówkowa, pszczoła, świetlika, kukułki, rój cząstek, kolonia karaluchów. Ogólny schemat modelowania algorytmów stadnych można przedstawić następująco Kwiecień (2015):

- reprezentacja i inicjacja początkowej populacji,
- zdefiniowanie sąsiedztwa, odległości i ruchu
- określenie mechanizmów wspomagających modyfikację rozwiązań, dobór kluczowych parametrów charakterystycznych dla metody,
- określenie kryterium akceptacji lub odrzucenia rozwiązań kandydujących,
- specyfikacja funkcji dopasowania.

Dla problemów szeregowania zadań dodatkowo należy dokonać procedury dyskretyzacji przestrzeni ciągłej przez przekształcenie wartości w ograniczony zbiór możliwych do osiągnięcia rozwiązań.

Modelowanie stadne są szeroko stosowane w problemach szeregowania zadań dla problemów zarówno deterministycznych jak i stochastycznych np. Liao et al. (2007); Blum and Sampels (2004); Lian et al. (2006); Pan et al. (2011) . W celu zastosowania tej techniki do rzeczywistych problemów wymaga odpowiednich modyfikacji lub kombinacji z innymi metodami Tang et al. (2016); Nouri et al. (2017).

2.4. Metody rozwiązywania problemu przepływowego z ponownym przetwarzaniem zadań

Problem produkcji przepływowej często występuje w przemyśle i również jest częstym tematem rozważanym w literaturze naukowej. Rzeczywiste problemy szeregowania zadań są w większości przypadków dynamiczne i są zależne od zmiennych stochastycznych, takich jak stochastyczny czas przetwarzania, czy nieprzewidziane nagłe nowe zamówienia. Dlatego harmonogramowanie produkcji w warunkach niepewności jest w ostatnich latach przedmiotem wielu badań i przedstawione między innymi w pracach: Almeder and Hartl (2013); Aytug et al. (2005); Choi and Wang (2012); Shabtay (2012); Katragjini et al. (2015). Nie mniej jednak większość artykułów naukowych zakłada, że wykonywane zadania nie powtarzają procesu produkcyjnego. Innymi słowy zakłada się, że zadanie nie może być przetwarzane przez tą samą maszynę więcej niż jeden raz. Takie założenie upraszcza wiele rzeczywistych środowisk produkcyjnych, w których przykładowo występuje konieczność naprawienia niektórych niepoprawnie wykonanych zadań, wykrytych w trakcie kontroli jakości Lee et al. (2011). W konsekwencji nie jest rozważany ani czas, ani koszt koniecznej naprawy, co z punktu widzenia przedsiębiorcy może generować ogromne straty z powodu niedotrzymania czasu wykonania zlecenia, poniesienia dodatkowych (nieoszacowanych) kosztów realizacji produkcji, przeciążenia linii produkcyjnej dodatkowymi, niezaplanowanymi i nieoszacowanymi operacjami naprawy oraz przede wszystkim braku procesu zarządczego związanego z

realizacją procesu naprawy. Problemy, w których występuje konieczność powtórzenia wszystkich lub części operacji z marszuty technologicznej należą do problemów re-entrant flowshop. Do tej klasy zaliczane są zarówno problemy, w których powtórzenie operacji z marszuty technologicznej wynika z wymagań technologicznych (np. produkcja wyświetlaczy ciekłokrystalicznych LCD), jak i z konieczności wykonania procesu naprawy w wyniku negatywnej kontroli jakości wyrobu. Z powodu złożoności tych problemów większość rozwiązań bazuje na algorytmach heurystycznych lub meta-heurystykach.

W pracy Lee et al. (2011) przedstawiony jest rzeczywisty przepływowy problem re-entrant, w którym minimalizowane jest zarówno maksymalne opóźnienie jak i maksymalny czas wykonania wszystkich zadań, zaś naprawa wybrakowanych zadań polega na całkowitym powtórzeniu marszuty technologicznej. Autorzy zaproponowali algorytm genetyczny, którego efektywność została zweryfikowana na danych rzeczywistych firmy.

Hinze w kilku pracach Hinze et al. (2013); Hinze and Sackmann (2014); Hinze (2016) rozważa różne problemy typu re-entrant, w których ponowne przetwarzanie zadań wymagane jest z powodów braków jakościowych produktów. W pracy Hinze and Sackmann (2014), rozważa permutacyjny problem produkcji przepływowej typu re-entrant z ponownym przetwarzaniem zadań na wybranej maszynie w ciągu technologicznym. W wyniku kontroli jakości podejmowana jest decyzja o ponownym przetwarzaniu zadania na wybranej maszynie w ciągu technologicznym, tyle razy, aż przetwarzane zadanie spełni wymogi technologiczne, by kontynuować obróbkę na kolejnych maszynach w ciągu technologicznym.

W pracy Dugardin et al. (2010) przedstawiono zastosowanie algorytmu genetycznego Lorenza (L-NSGA) oraz algorytmu ewolucyjnego Pareto (SPEA2), aby zminimalizować maksymalny czas ukończenia dwumaszynowego problemu przepływowego re-entrant RHFS, gdzie ponowne przetwarzanie zadań jest wynikiem niespełnienia wymogów technologicznych produktów. W pracy Dugardin et al. (2010) autorzy przedstawili rozwiązanie wielokryterialnego problemu hybrydowej produkcji przepływowej typu re-entrant bazujące na algorytmie genetycznym.

W pracy Yalaoui et al. (2009) zaproponowano rozwiązanie problemu szeregowania zamówień klientów w systemie hybrydowej produkcji przepływowej, w którym każde zamówienie składała się z kilku partii, a niektóre partie są poddawane ponownej obróbce. Do rozwiązania zastosowano metodę optymalizacji algorytmem rojowym.

W pracy Hekmatfar et al. (2011) przedstawiono nowy hybrydowy algorytm genetyczny do rozwiązania dwuetapowego hybrydowego problemu przepływowego re-entrant RHFS. Pierwszy etap problemu to nakładanie kolejnych warstw na elementy z powtórzeniami, aż wszystkie warstwy zostaną nałożone. Drugi etap to sprawdzenie jakości wykonania wszystkich operacji zadań zakończonych w pierwszym etapie i ewentualna naprawa. Ten drugi etap może być używany tylko jako końcowy etap testowy i jest realizowany tylko po zakończeniu pierwszego etapu, a nie może kontrolować defektów generowanych w trakcie nakładania warstw.

W pracy Desprez et al. (2009) rozważany jest rzeczywisty problem przemysłowy typu hybrydowego problemu przepływowego re-entrant RHFS. Celem jest zminimalizowanie całkowitej ważonej liczby opóźnionych prac. Opracowano algorytm genetyczny, aby poradzić sobie z problemami o dużych rozmiarach.

W pracy Chakhlevitch and Glass (2009) zaproponowano specjalny dwustopniowy hybrydowy harmonogram planowania w systemie przepływowym. Celem jest zminimalizowanie czasu realizacji. Udowodniono, że problem jest NP-trudny, następnie opracowano algorytm heurystyczny, na podstawie analizy cech charakterystycznych problemu.

W pracy Zhou et al. (2018) rozważono problem planowania przepływowego re-entrant z operacjami sprawdzającymi wykonanie i operacjami naprawczymi. Zaproponowany został model matematyczny i hybrydowy algorytm ewolucyjny, aby zminimalizować całkowity ważony czas zakończenia.

Powyżej przedstawiono wiele metod modelowania i rozwiązywania wybranych problemów szeregowania zadań. Wybrane problemy szeregowania zadań stanowią reprezentację dynamicznych, złożonych problemów planowania, o cechach podobnych do rozważanych w pracy problemów przepływowych. Przedstawiony w tym rozdziale, wśród metod modelowania, meta-model algebraiczno-logiczny *ALMM* został wykorzystany do rozwiązania rozważanych w pracy problemów. Szczegółowo podejście to zostało opisane w następnym rozdziale 3. Wybór tego podejścia podyktowany został następującymi względami:

- automaty komórkowe mają zastosowanie dla problemów przestrzennych i bardziej nadają się do reprezentacji problemów planowania zadań na maszynach równoległych niż w systemie przepływowym,
- w formalizmie DEVS stan zmienia się natychmiast, gdy nadejdzie zdarzenie wyjściowe, nowa wartość stanu zależy od wartości zdarzenia wejściowego, poprzedniej wartości stanu i czasu, nie zależy natomiast od podjętej decyzji,
- w formalizmach programowania z ograniczeniami, ewolucyjnych i stadnych brakuje możliwości zamodelowania wieloetapowego podejmowania decyzji.

3. Meta model algebraiczno-logiczny (ALMM)

W niniejszym rozdziale przedstawiony został meta model algebraiczno-logicznego (ALMM). Paradigmat ALMM został zdefiniowany w pracach Dudek-Dyduch (1989, 1990) i kolejno rozwijany w pracach Dudek-Dyduch (1989, 1991, 1993, 2000); Dudek-Dyduch and Dyduch (1988); Dudek-Dyduch (2015c). Meta model algebraiczno-logiczny należy do klasy modeli matematycznych i służy do formalnego opisu wieloetapowego procesu decyzyjnego.

Do modelowania problemu wykorzystywana jest przestrzeń stanów, jednakże ani współrzędne stanu ani sterowania (decyzji) nie muszą przyjmować tylko wartości liczbowych. W konsekwencji ograniczenia dotyczące podzbiorów stanów, decyzji oraz ich wzajemnych relacji zapisywane są zarówno za pomocą zależności algebraicznych jak i formuł logicznych. Podejście ALMM uwzględnia możliwość zmiennego zbioru decyzji nieznanych z góry. Cechą charakterystyczną jest nie indywidualnie, a wspólne podejmowanie decyzji dla wszystkich obiektów.

ALMM umożliwia symulację procesu, optymalizację problemu oraz ścisłą analizę problemów. Za pomocą meta modelu algebraiczno-logicznego (ALMM) można stworzyć formalne modele wielu różnych dyskretnych problemów optymalizacyjnych, a w szczególności dyskretnych dynamicznych problemów optymalizacyjnych takich jak sterowanie dyskretnymi procesami produkcyjnymi (wytwarzania) czy problemy logistyczne.

W poniższej definicji meta modelu algebraiczno-logicznego *stanem właściwym procesu x* nazywamy zbiór danych opisujących zachowanie systemu (stan fizyczny systemu), a *stanem uogólnionym procesu s* nazywamy parę zawierającą zarówno stan właściwy, jak i moment czasu.

3.1. Definicja meta modelu algebraiczno-logicznego

Meta-model algebraiczno-logiczny (ALMM) służy do formalnego opisu wieloetapowego procesu decyzyjnego. Został on zdefiniowany przez E. Dudek-Dyduch następująco:

Definicja 4 *Dudek-Dyduch (2015c)* Wieloetapowym dynamicznym decyzyjnym procesem nazywamy dyskretny proces P jednoznacznie określony przez szóstkę:

$$P = (U, S, s_0, f, S_N, S_G) \quad (3.1)$$

gdzie:

U - zbiór decyzji sterujących lub sygnałów sterowania,

$S = X \times T$ - zbiór stanów uogólnionych,

X - zbiór stanów właściwych,

$s_0 = (x, t)$ - uogólniony stan początkowy, $s_0 \in S$,

$T \subset \mathbb{R}^+ \cup \{0\}$ - podzbiór nieujemnych liczb rzeczywistych reprezentujących chwile czasowe,

$f : U \times S \rightarrow S$ - funkcją przejścia, zdefiniowana jest przy pomocy dwóch funkcji $f = (f_x, f_t)$:

- $f_x : U \times X \times T \rightarrow X$ określa następny stan właściwy,

- $f_t : U \times X \times T \rightarrow T$ określa następny moment czasu i spełnia następujący warunek:
 $\Delta t = f_t(u, x, t) - t > 0$ i ma wartość skończoną,
 przy czym zbiory U, X, T są niepuste,

$S_N \subset S$ - zbiór uogólnionych stanów niedopuszczalnych,

$S_G \subset S$ - niepusty zbiór uogólnionych stanów docelowych, a więc stanów, w których proces powinien się znaleźć w wyniku działania właściwych sterowań.

Funkcja przejścia została zdefiniowana jako funkcja częściowa (określona tylko dla pewnych par $(u, s) \in U \times S$), co umożliwia na uwzględnienie wszystkich ograniczeń dotyczących decyzji sterujących za pomocą tzw. zbiorów sterowań możliwych w stanie s , oznaczonych $U_p(s)$. Jeśli decyzja u jest możliwa w stanie s , to funkcja przejścia jest określona dla tej pary (u, s) . W przeciwnym wypadku nie jest określona. Zbiór decyzji możliwych w danym stanie $U_p(s)$ jest zdefiniowany następująco:

$$U_p(s) = \{u \in U : (u, s) \in \text{Dom} f\} \quad (3.2)$$

Definicja określająca zbiór stanów możliwych jest istotną częścią składową definicji funkcji przejścia. Ograniczenia dotyczące stanów uogólnionych definiujące S_N , można również uwzględnić za pomocą zbiorów sterowań dopuszczalnych w stanie s , oznaczonych $U_d(s)$. Zbiór decyzji dopuszczalnych w danym stanie $U_d(s)$ definiowany jest następująco:

$$U_d(s) = \{u \in U_p(s) : f(u, s) \notin S_N\} \quad (3.3)$$

Trajektorie procesu

Równoważnie definicję procesu P można wyrazić jako zbiór wszystkich trajektorii procesu Dudek-Dyduch (1990).

Definicja 5 Dudek-Dyduch (1990) Niech symbole $S, s_0, S_N, S_G, U_p(s)$ oznaczają tak jak powyżej odpowiednio zbiór stanów uogólnionych, uogólniony stan początkowy, zbiór uogólnionych stanów niedopuszczalnych, zbiór uogólnionych stanów docelowych, zbiór decyzji możliwych w stanie s . Procesem P nazywamy zbiór wszystkich ciągów skończonych $\tilde{s} = (s_0, s_1, \dots, s_n)$ oraz ciągów nieskończonych $\tilde{s} = (s_0, s_1, \dots)$ takich, że:

– w przypadku ciągów nieskończonych, dla każdego $i \in \mathbb{N} \cup 0$ zachodzi następujący warunek :

$$\text{istnieje } u_i \in U_p(s) \text{ takie, że } s_{i+1} = f(u_i, s_i) \quad (3.4)$$

– w przypadku ciągów skończonych, dla każdego $i = 0, 1, \dots, n - 1$ zachodzi warunek (3.4) oraz dodatkowo poniższy warunek:

$$s_n \in S_G \cup S_N \text{ lub } U_p(s_n) = \emptyset \quad (3.5)$$

Ciągi spełniające powyższe warunki nazywane są trajektoriami procesu P .

W każdym, nowo wyznaczonym stanie symulowanego procesu podejmowana jest decyzja dotycząca następnego kroku jego prowadzenia. Decyzja ta wybierana jest spośród możliwych (sensownych) dla danego stanu decyzji. Dla danego stanu i wybranej decyzji można wyznaczyć odpowiadający mu moment czasu oraz następny stan procesu. Wyznacza się je, korzystając z tzw. funkcji przejścia procesu. Jeśli wyznaczony stan należy do zbioru stanów docelowych procesu, przebieg symulacji jest zakończony pomyślnie i można dokonać jego oceny. Jeśli nowo wyznaczony stan i związany z nim czas nie spełniają warunków narzuconych ograniczeniami technologicznymi i czasowymi na proces, to stan taki znajduje się w zbiorze tzw. stanów niedopuszczalnych i nie ma sensu prowadzenia dalszej symulacji procesu.

Trajektorie oznaczane są symbolem \tilde{s} , a zbiór wszystkich trajektorii procesu P symbolem \tilde{S} .

Ponadto symbolem S_E oznaczany jest zbiór stanów uogólnionych, dla których funkcja przejścia f jest nieokreślona a równocześnie stany te nie należą do zbioru stanów docelowych:

$$S_E = \{s \in S \setminus S_G : U_p(s) = \emptyset\} \quad (3.6)$$

Zatem trajektoria startuje z wyróżnionego początkowego stanu s_0 i może być trajektorią nieskończoną lub skończoną i zakończyć się w zbiorze stanów docelowych S_G , w zbiorze stanów niedopuszczalnych S_N lub w zbiorze stanów, w którym zbiór „sensownych” decyzji jest pusty S_E . Trajektoria dopuszczalna, to trajektoria skończona, która kończy się w zbiorze stanów docelowych S_G , w pozostałych stanach trajektoria jest niedopuszczalna.

Symbolem \tilde{u} będzie oznaczany ciąg decyzji wyznaczający trajektorię. Jeśli ciąg decyzji wyznacza trajektorię dopuszczalną to nazywany jest dopuszczalnym ciągiem decyzji i odpowiednio, gdy wyznacza niedopuszczalną trajektorię, to jest niedopuszczalnym.

Zadanie poszukiwania rozwiązania dopuszczalnego polega na znalezieniu ciągu decyzji procesu P wyznaczającego trajektorię dopuszczalną. Zadanie optymalizacji polega na znalezieniu takiego ciągu sterowań dopuszczalnych \tilde{u} , który ekstremalizuje pewien funkcjonal Q (kryterium). Zadanie optymalizacji określone jest zatem przez model procesu P oraz funkcjonal Q i oznaczone będzie jako para (P, Q) .

Postać zbioru decyzji i zbioru stanów właściwych

W najogólniejszym przypadku zbiory U oraz X można przedstawić jako iloczyny kartezjańskie (górne wskaźniki oznaczają kolejne zbiory)

$$U = U^1 \times U^2 \times \dots \times U^m \quad X = X^1 \times X^2 \times \dots \times X^n \quad (3.7)$$

Najogólniej więc zarówno decyzja u jak i stan x są n -tkami.

$$u = (u^1, u^2, \dots, u^m) \quad x = (x^1, x^2, \dots, x^n) \quad (3.8)$$

Wartości poszczególnych elementów zarówno stanu jak i decyzji nie muszą przyjmować wartości liczbowych. Poszczególne współrzędne wektora u reprezentują odrębne decyzje, które mogą być podejmowane równocześnie w danym stanie.

W modelu nie nałożono żadnych ograniczeń dotyczących wartości współrzędnych stanu oraz decyzji. Wartościami tych zmiennych mogą być nie tylko liczby, ale też nazwy elementów jak i pewne obiekty, np. skończone zbiory, ciągi.

3.2. Metody optymalizacji dyskretnych dynamicznych problemów DDP bazujące na meta modelu algebraiczno-logicznym

W tej części pracy zostaną przedstawione metody rozwiązywania problemów szeregowania, które w odróżnieniu od tzw. klasycznych podejść bazują na formalnym modelu matematycznym reprezentującym problem szeregowania, a nie na werbalnym opisie problemu Dudek-Dyduch (2015c). Metody te są ogólne i służą do konstruowania wyspecjalizowanych algorytmów heurystycznych na bazie meta modelu algebraiczno-logicznego (ALMM).

3.2.1. Trój etapowa metoda projektowania algorytmów heurystycznych przeszukiwania z lokalnym kryterium

Metoda ta została zdefiniowana w pracach Dudek-Dyduch (1990); Dudek-Dyduch and Dyduch (1988) i zastosowana w Dudek-Dyduch and Dyduch (1996); Dudek-Dyduch and Fuchs-Seliger (1993). W metodzie tej wykorzystano pojęcie *semimetryki*. Semimetryka, oznaczana jako Ψ , w odróżnieniu od metryki nie musi spełniać warunku $\Psi(s_1, s_2) = 0 \Leftrightarrow s_1 = s_2$.

Trój etapowa metoda projektowania algorytmów heurystycznych przeszukiwania z lokalnym kryterium została szczegółowo opisana w Dudek-Dyduch (2015c) i składa się z następujących etapów:

- Etap 1 W pierwszym etapie formułuje się warunki konieczne i/lub wystarczające do otrzymania rozwiązania optymalnego (suboptymalnego) lub dopuszczalnego rozwiązania na podstawie analizy opisu werbalnego problemu. Warunki te mają się odnosić do podzbiorów decyzji lub określać zbiór stanów korzystnych S_{A_i} , $i = 1, 2, \dots$ lub zbiór stanów niekorzystnych S_{DA_i} dla spełnienia kryterium lub możliwości generowania trajektorii dopuszczalnej. Warunki mogą również określać pewne cele pośrednie przykładowo zbiór stanów, które należy osiągnąć w jak najkrótszym czasie lub w celu zminimalizowania wartości specjalnego kryterium pomocniczego.
- Etap 2 W drugim etapie formułowany jest problem optymalizacji lokalnej na podstawie wyróżnionych w poprzednim etapie: zbioru stanów korzystnych i zbioru stanów niekorzystnych. Ponadto wykorzystywane są informacje na temat zbioru stanów docelowych S_G , zbioru stanów dopuszczalnych S_N i zbioru decyzji możliwych. W celu wyznaczenia trajektorii w taki sposób by spełnić kryterium

lokalne (poprzez stany korzystne) zaproponowano użycie różnych semimetryk do obliczenia odległości między stanami w przestrzeni stanów. Kroki te prowadzą do zmiany globalnego kryterium Q na kryterium lokalne z wykorzystaniem strategii A^* (Perl (1984)) i zmaksymalizowaniu (zminimalizowaniu) odległości za pomocą semimetryki. Zastosowanie użycia semimetryk do obliczania odległości między stanami przedstawiono szczegółowo w Dudek-Dyduch (1990). W taki sposób zdefiniowany jest nowy zastępczy problem lokalny, który zazwyczaj jest wielokryterialny.

Etap 3 Trzeci etap polega na ustaleniu sposobu rozwiązania lokalnego problemu optymalizacji na podstawie wyboru decyzji. Wyróżniono cztery podstawowe sposoby postępowania. Niech $q = q_1, q_2, \dots, q_k$ oznacza wektorowe kryterium lokalne. Pierwszy sposób rozwiązania problemu lokalnego polega na wyborze (dowolnym) rozwiązania ze zbioru Pareto, gdy dla każdego stanu trajektorii zbiór Pareto jest odpowiednio mniejszy od zbioru decyzji możliwych $U_p(s)$. Drugi sposób polega na ustaleniu priorytetów dla kryteriów i dokonaniu wyboru leksykograficznego, czyli takiego u^* , które należy do zbioru:

$$U^*(s) = \{u^* \in U_p(s) : \forall u \in U_p(s) \exists i=1,2,\dots,k-1 : \\ q_1(u^*, s) = q_1(u, s), \dots, q_i(u^*, s) = q_i(u, s), q_{i+1}(u^*, s) < q_{i+1}(u, s)\}.$$

Trzeci sposób polega na wyborze większościowym. Wybierana jest decyzja u^* , dla której większość kryteriów przyjmuje korzystne wartości. Czwarty sposób polega na wyborze decyzji z wykorzystaniem skalaryzacji, polegającej na przypisaniu współczynników wag poszczególnym kryteriom. Takie kryterium jest postaci:

$$q = \sum_{i=1}^k w_i q_i(u, s)$$

gdzie q_i oznacza i -te kryterium, a w_i jest wagą przypisaną i -temu kryterium.

Metoda ta umożliwia zdefiniowanie różnych metod metaheurystycznych oraz algorytmów.

3.2.2. Metody uczenia maszynowego bazujące na ALMM

Metoda uczenia bazująca na meta-modelu algebraiczno-logicznym została zdefiniowana w pracach Dudek-Dyduch (2000); Dudek-Dyduch and Dyduch (2006a) i przedstawiona również w Dudek-Dyduch and Dyduch (2006a,b, 1996). Metoda ta została później rozszerzona w następujących pracach Kucharska (2006); Dudek-Dyduch and Kucharska (2011a,b); Dutkiewicz and Kucharska (2011).

Celem tej metody jest znalezienie rozwiązań dla niektórych klas dyskretnych problemy optymalizacyjne modelowanych za pomocą schematu meta-modelu algebraiczno-logicznego oraz spełniających następujące własności:

- wszystkie trajektorie muszą być skończone,
- kryterium Q musi być separowalne (definicja separowalności kryterium Q dla procesu Q oraz własności takiego kryterium podane zostały przez Dudek-Dyduch (1990)).

Pierwsza metoda uczenia oparta na ALMM

Metoda ta polega na generowaniu w sposób iteracyjny kolejnych trajektorii, w taki sposób by, jeśli to możliwe, z każdą iteracją poprawiać rozwiązanie. Trajektoria generowana jest na podstawie specjalnie skonstruowanego zadania lokalnej wielokryterialnej optymalizacji.

Postać kryterium lokalnego w tej metodzie jest następująca:

$$q(u, s) = \Delta Q(u, s) + \widehat{Q}(u, s) + a_1\varphi_1(u, s) + \dots + a_n\varphi_n(u, s) \quad (3.9)$$

gdzie:

$\Delta Q(u, s)$ - przyrost wartości wskaźnika jakości w wyniku podjętej decyzji u podjętej w stanie s ,

$\widehat{Q}(u, s)$ - oszacowania wartości wskaźnika jakości końcowego odcinka trajektorii po zrealizowaniu decyzji u ,

$\varphi_i(u, s)$ - składniki odzwierciedlające dodatkowe ograniczenia lub dodatkowe wymagania w przestrzeni stanów dla $i = 1, \dots, n$,

α_i - współczynniki, które określają wagi składników szacujących wpływ powyższych dodatkowych ograniczeń lub wymagań w kryterium $i = 1, \dots, n$.

W związku z postacią kryterium lokalnego, zadanie optymalizacji lokalnej polega na wybraniu takiej decyzji spośród wszystkich decyzji możliwych w danym stanie, aby następowała minimalizacja wartości kryterium jakości przy maksymalizacji odległości między następnym stanem a stanami zabronionymi i niekorzystnymi oraz minimalizacji odległości między następnym stanem a wyróżnionymi stanami korzystnymi, a także przy uwzględnieniu preferencji dla pewnych typów decyzji.

Znaczenie poszczególnych składników kryterium lokalnego może być różne. Im bardziej znaczący jest któryś składnik, tym większa powinna być wartość jego wagi. Trudno jest określić a priori optymalne wagi. Wagi te zależą zarówno od rozważanego problemu optymalizacji, jak również od danych konkretnej badanej instancji problemu. W trakcie generowania kolejnych trajektorii zbierana jest wiedza, w celu odpowiedniej modyfikacji współczynników.

Z drugiej strony wartości współczynników ustalone dla najlepszej trajektorii reprezentują zagregowaną wiedzę uzyskaną w trakcie eksperymentów. Każda wygenerowana trajektoria jest analizowana przez algorytm uczenia się. Jeżeli trajektoria jest niedopuszczalna, badana jest przyczyna niepowodzenia. Przykładowo badane jest, który podzbiór stanów niekorzystnych należy do stanów trajektorii. Następnie waga składnika kryterium związanego z tymi podzbiórami powinna zostać zmniejszona. W przypadku kiedy wygenerowana trajektoria jest dopuszczalna, można wzmocnić wagi tych składników dla następnej trajektorii. W oparciu o uzyskane informacje, zadanie lokalnej optymalizacji jest poprawiane podczas eksperymentów symulacyjnych. W związku z tym ten proces jest traktowany jako proces uczący lub inteligentny algorytm wyszukiwania.

Rozwinięcie metody uczenia opartej na ALMM

Pierwsze rozwinięcie metody uczenia opartej na meta-modelu algebraiczno-logicznym zostało przedstawione przez Dudek-Dyduch and Dyduch (2006a,b). Autorzy zaproponowali wykorzystanie wiedzy eksperckiej w celu zmniejszenia wielkości zbioru decyzji możliwych $U_p(s)$, a co za tym idzie zaproponowaniu lepszego algorytmu. Metoda ta ma następujące zastosowanie: jeśli wiedza ekspercka możliwa

wyróżnienie pewnych podzbiorów stanów S_{d_i} ($i = 1, \dots$) oraz dla tych podzbiorów reguł podejmowania decyzji R_i ($i = 1, \dots$), wtedy algorytm weryfikuje, do którego podzbioru należy zakwalifikować wygenerowany nowy stan i wykonuje odpowiednią regułę. Jeśli reguły podane przez eksperta wykluczają podjęcie niektórych decyzji, wtedy rozważany jest mniejszy zbiór decyzji możliwych $U_p(s)$. W podobny sposób można zastosować wyniki wykonanych przez algorytm eksperymentów.

Kolejnym rozwinięciem metody uczenia są zaproponowane przez Kucharska (2006) nowe elementy takie jak:

- zmiany związane z kryterium lokalnym, opisane w Kucharska (2006); Kucharska and Dudek-Dyduch (2014), polegające na:
 - dodaniu dodatkowych elementów odpowiadających za preferencje do kryterium lokalnego,
 - usunięciu niepotrzebnych elementów z lokalnej formy kryterium w przypadku, gdy pewne podzbiory stanów są nieosiągalne,
- hybrydowe połączenie pierwszej metody uczenia z wykorzystaniem takich elementów jak odcięcie nieperspektywicznych odcinków trajektorii.

3.2.3. Metoda zadań zastępczych (MZZ)

Metoda ta została zaproponowana w pracy doktorskiej Dutkiewicz (2005), a następnie rozwijana w pracach Dudek-Dyduch and Dutkiewicz (2013); Dutkiewicz and Dudek-Dyduch (2014) oraz przebadana dla różnych problemów NP-trudnych m.in. w Dutkiewicz and Kucharska (2011); Dutkiewicz et al. (2016).

W metodzie zadań zastępczych konstruowana jest pojedyncza trajektoria. W każdym rozważanym stanie s procesu decyzja wyznaczana jest na podstawie specjalnie skonstruowanego zadania optymalizacji, nazwanego zadaniem zastępczym $ZZ(s)$. Celem tworzenia takich zadań jest ułatwienie wyznaczenia decyzji w danym stanie poprzez zastąpienie optymalizacji zadania globalnego prostszym zadaniem lokalnym. Po wyznaczeniu decyzji generowany jest następny stan trajektorii s' , dla którego przeprowadzana jest „automatyczna analiza” procesu w celu wygenerowania najlepszego odcinka trajektorii ze stanu s' do zbioru stanów docelowych S_G . Następnie w wyniku tej analizy określone jest nowe, zmodyfikowane zadanie zastępcze. W związku z powyższym w każdej iteracji obliczenia prowadzone są na dwóch poziomach:

1. poziom automatycznej analizy procesu i skonstruowania zadania zastępczego
2. poziom wyznaczania możliwie dobrej decyzji dla zrealizowania zadania zastępczego i obliczania następnego stanu.

Trajektoria generowana jest aż do osiągnięty stanu docelowego lub gdy zostanie osiągnięty stan niedopuszczalny.

Zadanie zastępcze zdefiniowane jest jako para $ZZ(s) = (P_Z, Q_Z)$, gdzie P_Z oznacza proces zastępczy, zaś Q_Z oznacza kryterium zastępcze.

Proces zastępczy $P_Z(s)$ zdefiniowany jest następująco:

$$P_Z(s) = (s_{0z}, f_z, S_{Nz}, S_{Gz})$$

gdzie:

s_{0z} - oznacza stan początkowy dla procesu zastępczego P_Z , będący równocześnie aktualnym stanem s procesu podstawowego P ,

f_z - funkcja przejścia procesu zastępczego P_Z ,

S_{Nz} - zbiór uogólnionych stanów niedopuszczalnych dla danego procesu zastępczego P_Z ,

S_{Gz} - niepusty zbiór uogólnionych stanów docelowych dla danego procesu zastępczego P_Z .

Aby można było określić zadanie zastępcze, konieczna jest analiza procesu w danym stanie. Na jej podstawie w danym stanie określone są, w sposób heurystyczny, tzw. cele pośrednie. Cele pośrednie służą do zdefiniowania zbioru stanów docelowych zadania zastępczego S_{Gz} . Jako cel pośredni d określane jest jak najszybsze osiągnięcie przez proces stanu należącego do pewnego wyróżnionego zbioru stanów S_d . Zbiór celów pośrednich d oznaczany jest jako D .

Dla skonstruowania zadania zastępczego w danym stanie s realizowane są następujące kroki:

1. określenie zbioru celów D
2. określenie priorytetów w zbiorze celów D
3. określenie podzbioru zbioru D czyli zbioru celów $D_W(s)$, które będą realizowane
4. określenie zbioru stanów docelowych S_{Gz} procesu zastępczego P_Z .

Opisane powyżej podejście *ALMM* zostało wykorzystane do zamodelowania problemów produkcji przepływowej z brakami jakościowymi oraz opracowania algorytmów optymalizacyjnych dla tych problemów.

4. Model algebraiczno-logiczny dla problemu produkcji przepływowej FS

W rozdziale przedstawiono model problemu produkcji przepływowej z deadlineami i magazynami międzyoperacyjnymi o nieograniczonej pojemności. Celem optymalizacji jest jak najszybsze wykonanie wszystkich zadań. Rozwiązanie to bazuje na meta-modelu algebraiczno-logicznym $ALMM$.

Przedstawiony model algebraiczno-logiczny produkcji przepływowej wraz z kryterium jakości został opracowany we współautorstwie autorki z E. Dudek-Dyduch oraz E. Kucharską i przedstawiony w Grobler-Dębska et al. (2013). Model ten stanowi podstawę do zamodelowania rozważanych dalej w pracy złożonych problemów produkcji przepływowej z brakami jakościowymi.

4.1. Opis problemu produkcji przepływowej FS

W rozdziale tym przedstawiony jest opis rozważanego problemu produkcji przepływowej. Niech $\mathcal{M} = \{M_1, M_2, \dots, M_m\}$ oznacza zbiór maszyn dedykowanych w marszrucie technologicznej, $\mathcal{J} = \{J_1, J_2, \dots, J_n\}$ oznacza zbiór zadań, które mają być przetworzone na tych maszynach. Każde J_j -te zadanie $1 \leq j \leq n$, musi być poddane m operacjom. Wszystkie zadania przetwarzane są przez wszystkie maszyny w odpowiedniej kolejności od pierwszej do ostatniej maszyny według marszrut technologicznej (M_1, M_2, \dots, M_m) . Przed każdą maszyną znajduje się magazyn międzyoperacyjny, w którym znajdują się zadania przetworzone przez poprzednią maszynę i czekające na przydzielenie ich do obróbki na tej maszynie. Zatem niech $\mathcal{W} = \{W_1, W_2, \dots, W_m\}$ oznacza zbiór magazynów, gdzie W_i oznacza magazyn przed maszyną M_i . Rozważany jest również magazyn końcowy oznaczony jako W_0 , w którym znajdują się poprawnie wykonane zadania (o odpowiedniej wymaganej jakości). Przy czym zakłada się, że pojemność magazynów jest nieograniczona. Ponadto dany jest wymagany termin końcowy każdego zadania. Termin końcowy dla zadania J_j oznaczany jest przez $\bar{d}(j)$. Zatem należy tak ustalić harmonogram produkcji, aby ukończyć wszystkie zadania z odpowiednią jakością przed upłynięciem terminu końcowego.

4.2. Model algebraiczno-logiczny problemu

Według podejścia $ALMM$ tworzenie modelu należy rozpocząć od określenia informacji, które składają się na stan systemu oraz zaproponowania sposobu formalnego zapisu tych informacji, a więc pewnego schematu kodowania ich w postaci wyodrębnionych współrzędnych stanu. Należy podkreślić, że

możliwe jest stworzenie wielu różnych, ale równoważnych modeli zagadnienia, a wybór postaci stanu systemu ma duży wpływ na dalsze budowanie modelu algebraiczno-logicznego.

Dla problemu produkcji przepływowej z deadlineami i magazynami międzyoperacyjnymi zdefiniowano poszczególne elementy modelu algebraiczno-logicznego. Najpierw przedstawiono postać stanu systemu, zbiory stanów docelowych oraz stanów niedopuszczalnych. Przeprowadzono również analizę postaci stanu systemu, mającą na celu wyróżnienie pewnych charakterystycznych klas stanów. Dla danego stanu wyodrębniono pewne zbiory elementów systemu (maszyn, magazynów i zadań) o wspólnych cechach, istotne dla definiowania pozostałych składników modelu. Dzięki odpowiedniej postaci stanu przedstawiony model uwzględnia możliwość oczekiwania maszyny na ukończenie pewnego zadania oraz oczekiwanie zadania na przydzielenie do maszyny, która w danej chwili jest zajęta. Następnie przedstawiono postać decyzji, zbiór decyzji możliwych do podjęcia w poszczególnych stanach, zbiór decyzji dopuszczalnych. Na koniec budowy modelu podano funkcję przejścia, czyli pokazano sposób wyznaczenia momentu wystąpienia kolejnego stanu oraz podano szczegółowy wzór na określenie wartości współrzędnych stanu właściwego.

Dla rozważanego problemu produkcji przepływowej stan procesu w danej chwili uwzględnia wszystkie informacje o stopniu wykonania zadań, o magazynach międzyoperacyjnych i dostępnych maszynach oraz o przydzielonych im zadaniach. Stany niedopuszczalne dotyczą sytuacji, kiedy jakieś zadanie nie jest wykonane, a jego termin wykonania minął. Natomiast stan docelowy procesu to stan, kiedy wykonane są wszystkie zadania, przy czym zadania ukończono przed upływem wymaganych terminów.

Po ustaleniu struktury stanu procesu należy ustalić strukturę decyzji. Decyzje polegają na wyznaczeniu zadań, które mają być przydzielone poszczególnym maszynom do przetwarzania.

Ostatnim etapem tworzenia modelu jest budowa funkcji przejścia z jednego stanu procesu do kolejnego.

4.2.1. Stan systemu

Stan procesu w danej chwili musi zawierać wszystkie informacje o stopniu wykonanych zadań, o przepływie zadań w ramach marszruty technologicznej, o przydzielonych zadaniach do maszyn oraz o zadaniach czekających na obróbkę na danej maszynie. Stany niedopuszczalne dotyczą sytuacji, kiedy jakieś zadanie nie jest wykonane, a jego wymagany termin wykonania minął. Natomiast stan docelowy procesu to sytuacja, kiedy wykonane są wszystkie zadania, przy czym przy czym zadania ukończono przed upływem wymaganych terminów.

Aby móc określić stopień wykonania zadań i przepływ zadań w ramach marszruty technologicznej wystarczy odpowiednio zawrzeć te informacje w postaci stanu magazynów i maszyn. Do opisanego postaci stanu magazynu są potrzebne informacje, które zadania znajdują się w danym magazynie, zaś do opisanego stanu maszyny są potrzebne następujące informacje:

- działanie jakie maszyna wykonuje: wykonywanie zadania czy postój (brak przydzielonego zadania do wykonania),
- które zadanie jest wykonywane przez maszynę,

- ile czasu zostało maszynie do zakończenia wykonywania zadania.

W wyniku analizy można stwierdzić, że stan systemu $s = (x, t)$ w danej chwili t musi zawierać wszystkie informacje o stanie maszyny z ustalonej marszruty technologicznej, stanie magazynu początkowego oraz stanach magazynów międzyoperacyjnych. Celowo w stanie systemu nie zostaną uwzględnione informacje o stanie magazynu końcowego, aby model nie był nadmiarowy.

Zatem stan właściwy procesowi jest zdefiniowany jako n -tka:

$$x = \underbrace{(x^1, x^2, \dots, x^m)}_{\text{stany magazynów}}, \underbrace{(x^{m+1}, \dots, x^{2m})}_{\text{stany maszyn}} \quad (4.1)$$

gdzie:

- x^1 – oznacza stan magazynu początkowego,
- x^i – oznacza stan W_i -tego magazynu międzyoperacyjnego (pomiędzy maszynami o indeksach M_{i-1} i M_i dla $i = 2, 3, \dots, m$,
- x^{m+i} – oznacza stan M_i -tej maszyny dla $i = 1, 2, \dots, m$.

Kodowanie stanu poszczególnych magazynów jest następujące:

- współrzędna x^1 jest to zbiór zadań do wykonania na pierwszej maszynie, tzn. zadań których obróbka nie została rozpoczęta,
- współrzędne x^i , dla $i = 2, 3, \dots, m$, to zbiory zadań, które zostały obrabiane na maszynie M_{i-1} i aktualnie są dostępne do obróbki na M_i -tej maszynie, ale nie zostały jeszcze przydzielone do tej maszyny.

Natomiast kodowanie stanu poszczególnych maszyn jest następujące:

$$x^i = (\beta, \tau) \quad (4.2)$$

gdzie:

- $\beta \in \{0, 1, \dots, n\}$ – oznacza numer zadania, które jest aktualnie obrabiane na danej maszynie (0 oznacza, że żadne zadanie nie zostało przydzielone do maszyny),
- $\tau \in [0, \infty)$ – czas do ukończenia obróbki aktualnego zadania na maszynie (∞ oznacza, że maszyna nie pracuje).

W celu ułatwienia dalszej prezentacji, wprowadzone zostaną następujące oznaczenia:

- $x(s)$ - oznaczenie stanu właściwego danego stanu s ,
- β^i i τ^i - oznaczenia odpowiednich wartości współrzędnych wektora stanu dla M_i -tej maszyny.

Łatwo zobaczyć, że w tym modelu możemy obliczyć zbiór zakończonych zadań \mathcal{FJ} w stanie s . Jest on równy różnicy zbioru wszystkich zadań i sumy zbiorów zadań znajdujących się w magazynie początkowym, magazynach międzyoperacyjnych i przetwarzanych przez maszyny:

$$\mathcal{FJ}(s) = \mathcal{J} \setminus \bigcup_{i=1}^m (x^i(s) \cup \{\beta^{i+m}(s)\}). \quad (4.3)$$

W związku z tym stan magazynu końcowego odzwierciedlający zbiór zadań wykonanych przez wszystkie maszyny, czyli zbiór zadań zakończonych nie został ujęty w postaci stanu systemu.

Charakterystyczne stany maszyn Analizując postać stanu systemu można wyróżnić następujące charakterystyczne klasy stanów maszyny M_i , przydatne w budowaniu pozostałych składników rozważanego modelu:

- M_i -ta maszyna nie pracuje (jest wolna) w stanie s :

$$x^{m+i}(s) = (0, \infty), \quad (4.4)$$

- M_i -ta maszyna pracuje (jest zajęta) w stanie s :

$$x^{m+i}(s) = (\beta \neq 0, \tau > 0). \quad (4.5)$$

Na tej podstawie można wyróżnić w danym stanie s dwa zbiory maszyn: maszyn wolnych $\mathcal{M}_W(s)$ oraz maszyn zajętych $\mathcal{M}_Z(s)$.

Stan początkowy systemu $s_0 = (x_0, t_0)$ W stanie początkowym ($t_0 = 0$), zakłada się, że wszystkie zadania znajdują się w magazynie początkowym, nie rozpoczęto obróbki żadnego zadania i żadne zadanie nie zostało jeszcze przydzielone do pierwszej maszyny. Stąd stan początkowy $s_0 = (x_0, t_0)$ jest następujący:

$$x_0 = (x_0^1, x_0^2, \dots, x_0^m, x_0^{m+1}, \dots, x_0^{2m}) \quad (4.6)$$

gdzie poszczególne współrzędne stanów magazynów i maszyn przyjmują następujące wartości:

- $x_0^1 = \mathcal{J}$,
- $x_0^i = \emptyset$ dla $i = 2, 3, \dots, m$,
- $x_0^{m+i} = (0, \infty)$ dla $i = 1, 2, \dots, m$.

Zbiór stanów niedopuszczalnych S_N Stan niedopuszczalny jest to stan, w którym istnieje niedokończony zadanie, a jego deadline $\bar{d}(j)$ wykonania minął. Zatem zbiór stanów niedopuszczalnych wynosi: $S_N = \{s = (x, t) : \exists j \in \mathcal{J} j \notin \mathcal{FJ}(s) \wedge \bar{d}(j) < t\}$

Zbiór stanów docelowych S_G Stan $s = (x, t)$ nazywamy docelowym, wtedy gdy wszystkie zadania zostały ukończone, zaś ich deadline $\bar{d}(j)$ wykonania nie minął. Stąd zbiór stanów docelowych wynosi: $S_G = \{s = (x, t) : \mathcal{FJ}(s) = \mathcal{J} \wedge \forall j \in \mathcal{J} \bar{d}(j) \geq t\}$

4.2.2. Decyzje

W danym stanie $s = (x, t)$ podejmowana jest decyzja, na podstawie której system przeprowadzany jest do następnego stanu. Wyznaczona decyzja $u(s)$ musi należeć do zbioru $U_p(s)$ decyzji możliwych w danym stanie.

W rozważanym problemie przyjęto, iż decyzja polega na przydzieleniu zadań do konkretnych maszyn w tym samym czasie.

Odnośnie decyzji przyjmowane są następujące założenia:

- decyzja o przydzieleniu następnego zadania dla maszyny może być podjęta dopiero po wykonaniu poprzednio przydzielonego jej zadania,
- maszynie można przydzielić tylko zadanie wcześniej wykonane przez maszynę poprzedzającą w marszrucie technologicznej, czyli zadanie znajdujące się w magazynie międzyoperacyjnym bezpośrednio przed daną maszyną,
- podjęta decyzja nie ulega zmianie.

Zatem dla maszyny $M_i \in \mathcal{M}_Z(s)$ (zajętej, czyli realizującej wcześniej podjętą decyzję) dozwolona jest tylko decyzja o kontynuacji działania. Maszyna ta wykonuje wcześniej przydzielone jej zadanie. Natomiast maszynie $M_i \in \mathcal{M}_W(s)$ (wolnej w danym stanie) można przydzielić tylko zadanie spełniające warunek, że zostało wykonane przez poprzednią maszynę w marszrucie technologicznej. Gdy wolnej maszynie M_i nie zostanie przydzielony żadne zadanie, to maszyna nadal jest wolna.

Uwzględniając powyższe rozważania w danym stanie $s = (x, t)$ należy podjąć decyzję $u \in U = U^1 \times U^2 \times \dots \times U^m$ i na podstawie tej decyzji wyznaczany jest kolejny stan systemu. Poszczególne współrzędne decyzji $u^i \in U^i$ reprezentują poszczególne oddzielne decyzje dla M_i -tej maszyny. Zatem struktura decyzji jest wektorem

$$u = (u^1, u^2, \dots, u^m) \quad (4.7)$$

gdzie wartości współrzędnych mogą być następujące:

- $u^i = j$ dla $(j = 1, \dots, n)$ oznacza przypisanie nowego J_j -tego zadania do M_i -tej maszyny,
- $u^i = 0$ oznacza kontynuację przez maszynę wcześniej przypisanego zadania.

Zatem współrzędne decyzji mogą przyjmować tylko wartości indeksu zadania lub 0 ($u^i \in \mathcal{J} \cup \{0\}$)

Oczywiście nie wszystkie decyzje u mogą być podjęte w stanie (x, t) . Decyzje u muszą należeć do zbioru decyzji możliwych $U_p(x, t)$. Możliwa decyzja zależy od stanu poszczególnej maszyny. Stąd wartość poszczególnej współrzędnej decyzji u^i jest następująca:

- jeśli zadanie J_j jest przetwarzane przez maszyną M_i , wtedy jedyną możliwą decyzją jest kontynuacja obróbki $u^i = 0$,
- jeśli maszyna M_i jest wolna i nie istnieje żadne zadanie, które można by było przypisać maszynie tzn. magazyn przed maszyną jest pusty $x^i(s) = \emptyset$, to jedyną możliwą decyzją jest nieprzypisywanie żadnego zadania $u^i = 0$,

- jeśli maszyna M_i jest wolna i magazyn W_i nie jest pusty (tzn. istnieje zadanie J_j w magazynie) wtedy możliwą decyzją jest przypisanie zadania maszynie $u^i = j$ lub nie przypisywanie żadnego zadania $u^i = 0$.

Zatem decyzja $u \in U_p(s)$, która należy do zbioru decyzji możliwych $U_p^i(s)$ dla maszyny M_i w stanie $s = (x, t)$ jest określona następująco:

$$\forall_{i=1,2,\dots,m} : u^i(s) = \begin{cases} 0 & \text{dla } x^{i+m}(s) \neq (0, \infty) \\ 0 & \text{dla } x^{i+m}(s) = (0, \infty) \text{ jeśli } x^i(s) = \emptyset \\ 0 \vee j & \text{dla } x^{i+m}(s) = (0, \infty) \text{ jeśli } j \in x^i(s) \end{cases} \quad (4.8)$$

Zatem kompletna definicja zbioru decyzji możliwych jest następująca:

$$U_p(s) = U_p^1(s) \times U_p^2(s) \times \dots \times U_p^m(s) \setminus H(s) \quad (4.9)$$

gdzie $H(s)$ - zbiór zawierający decyzje nie przypisania żadnego zadania do żadnej maszyny $u = (0, 0, \dots, 0)$, kiedy wszystkie maszyny w danym stanie są wolne i nie wszystkie zadania zostały ukończone.

$$H(s) = \{u : (\forall_{M_i \in \mathcal{M}} : u^i(s) = 0 \wedge x^{i+m}(s) = (0, \infty)) \wedge \mathcal{FJ} \neq \mathcal{J}\} \quad (4.10)$$

Kolejno podjęte decyzje tworzą ciąg decyzyjny $\tilde{u} = (u_1, u_2, \dots, u_c)$, gdzie c jest liczbą podjętych decyzji. Ciąg decyzyjny wyznacza jednoznacznie jedną z trajektorii systemu.

4.2.3. Funkcja przejścia

Na podstawie aktualnego stanu $s = (x, t)$ i decyzji u podjętej w tym stanie, generowany jest następny stan $s' = (x', t')$ przy pomocy funkcji przejścia f .

$$(x', t') = f(u, x, t) \quad (4.11)$$

Jak już wcześniej podano, funkcja przejścia jest zdefiniowana przy pomocy dwóch funkcji $f = (f_x, f_t)$, gdzie f_x określa następny stan właściwy, a f_t określa następny moment czasu.

Wyznaczanie funkcji przejścia jest zadaniem kombinatorycznym. Należy rozważyć różne możliwe podzbiory stanów i przejścia pomiędzy nimi. Konieczne jest określenie jakie stany możemy otrzymać w wyniku realizacji konkretnej decyzji oraz w jaki sposób należy obliczać wartości poszczególnych składników następnego stanu.

Wyznaczenie kolejnych momentów czasowych W pierwszej kolejności należy wyznaczyć moment t' wystąpienia następnego stanu, czyli najbliższy moment, w którym co najmniej jedna maszyna zakończy obróbkę przydzielonego zadania. W tym celu dla każdej maszyny M_i obliczany jest czas ukończenia t_{M_i} niezbędny na zrealizowanie podjętej dla niej decyzji. Najmniejszy z tych czasu posłuży do wyznaczenia czasu t' .

$$t' = t + \Delta t, \quad \text{gdzie: } \Delta t = \min_{M_i \in \mathcal{M}} t_{M_i} \quad (4.12)$$

gdzie wielkość oznaczana t_{M_i} jest wyznaczana następująco:

- jeśli maszyna M_i jest wolna i zadanie J_j jest do niej przypisane tzn. $u^i(s) = j$, to czas zakończenia pracy przez maszynę wynosi p^{ij} , czyli $t_{M_i} = p^{ij}$
- jeśli maszyna M_i jest wolna i żadne zadanie nie jest do niej przypisane, to czas ukończenia jest równy nieskończoność $t_{M_i} = \infty$,
- jeśli maszyna M_i jest zajęta, czyli obrabia wcześniej przypisane zadanie J_j , to jedyną decyzją jest kontynuowanie obróbki tego zadania przez maszynę. Wtedy maszyna ukończy wykonywanie zadania z czasem równym τ^{i+m} (czas dokończenia zadania szczytany ze stanu maszyny), czyli $t_{M_i} = \tau^{i+m}$.

W związku z tym Δt wynosi:

$$\Delta t = \min_{i=1, \dots, m} \begin{cases} p^{ij} & \text{dla } x^{i+m}(s) : \beta^{i+m} = j \wedge u^i(s) = j \\ \infty & \text{dla } x^{i+m}(s) = (0, \infty) \wedge u^i(s) = 0 \\ \tau^{i+m} & \text{dla } x^{i+m}(s) : \beta^{i+m} = j \wedge u^i(s) = 0 \end{cases} \quad (4.13)$$

Po wyznaczeniu chwili t' wystąpienia następnego stanu, obliczane są nowe wartości współrzędnych właściwych tego stanu.

Wyznaczenie współrzędnych stanu magazynu Współrzędne reprezentujące stany magazynów w następujący sposób:

- współrzędna x^1 (stan magazynu początkowego - zbiór zadań do obróbki na pierwszej maszynie) jest pomniejszana o zadanie przydzielone do obróbki na pierwszej maszynie:

$$(x^1(s))' = x^1(s) \setminus \{j : u^1(s) = j\}, \quad (4.14)$$

- współrzędna x^i dla $i = 2, 3, \dots, m$, która odpowiada stanowi magazynu międzyoperacyjnego W_i (zbiór zadań ukończonych przez maszynę M_{i-1} i nie przydzielonych do obróbki przez M_i -tą maszynę), jest zwiększana przez zadanie wykonane przez maszynę M_{i-1} tzn. $\exists J_j : x^{i-1+m}(s) = (j, \Delta t)$ i pomniejszana o zadanie przypisane na podstawie decyzji $u^i(s)$ do M_i -tej maszyny z ciągu technologicznego tzn. $\exists J_j : u^i(s) = j$:

$$(x^i(s))' = x^i(s) \cup (\{j : x^{i-1+m}(s) = (j, \Delta t)\} \setminus \{j : u^i(s) = j\}), \quad (4.15)$$

Wyznaczenie współrzędnych stanu maszyn Kolejne współrzędne reprezentujące stany poszczególnych maszyn M_i $x^{i+m} = (\beta, \tau)$ dla $i = 1, 2, \dots, m$, również są przeprowadzane do nowego stanu $(x^{i+m})' = (\beta', \tau')$ i zależą od podjętej decyzji w następujący sposób:

1. jeśli maszyna M_i jest wolna ($x^{i+m} = (0, \infty)$) w danym stanie i decyzja $u^i(s) = j$ przypisuje zadanie J_j do tej maszyny, to:

$$\begin{aligned} \beta' &= \begin{cases} 0 & \text{dla } \tau^{i+m} = \Delta t \\ j & \text{dla } \tau^{i+m} > \Delta t \end{cases} \\ \tau' &= \begin{cases} \infty & \text{dla } \tau^{i+m} = \Delta t \\ p^{ij} - \Delta t & \text{dla } \tau^{i+m} > \Delta t \end{cases} \end{aligned} \quad (4.16)$$

2. jeśli maszyna M_i jest wolna $x^{i+m} = (0, \infty)$ w danym stanie i podjęta decyzja $u^i(s) = 0$ nie przypisuje żadnego zadania do tej maszyny, to: $\beta' = 0, \tau' = \infty$,
3. jeśli maszyna M_i jest zajęta $x^{i+m} = (j, \tau)$ w danym stanie i podjęta została jedyna możliwa decyzja $u^i(s) = 0$ o kontynuowaniu przetwarzania wcześniej przydzielanego zadania J_j , to:

$$\begin{aligned} \beta' &= \begin{cases} 0 & \text{dla } \tau^{i+m} = \Delta t \\ j & \text{dla } \tau^{i+m} > \Delta t \end{cases} \\ \tau' &= \begin{cases} \infty & \text{dla } \tau^{i+m} = \Delta t \\ \tau^{i+m} - \Delta t & \text{dla } \tau^{i+m} > \Delta t \end{cases} \end{aligned} \quad (4.17)$$

4.3. Kryterium jakości

Celem optymalizacji rozważanego problemu jest jak najszybsze wykonanie wszystkich zadań. Minimalizowanym kryterium Q jest zatem całkowity czas wykonywania wszystkich zadań.

Szukany jest skończony dopuszczalny ciąg decyzji $\tilde{u}^* \in \tilde{U}_d$ taki, że:

$$Q(\tilde{u}^*) = \min_{\tilde{u} \in \tilde{U}_d} Q(\tilde{u}) \quad (4.18)$$

gdzie:

- \tilde{u}^* oznacza optymalny skończony ciąg decyzji
- \tilde{U}_d oznacza zbiór wszystkich dopuszczalnych ciągów decyzyjnych, które wyznaczają dopuszczalne trajektorie należące do zbioru S , przy czym zbiór decyzji dopuszczalnych w danym stanie definiowany jest następująco: $U_d(s) = \{u \in U_p(s) : f(u, s) \notin S_N\}$.

Całkowita wartość funkcji kryterialnej, to suma przyrostów funkcji kryterialnej w kolejnych stanach:

$$Q(\tilde{u}) = \Delta Q(u_0, x_0, t_0) + \dots + \Delta Q(u_i, x_i, t_i) + \dots + \Delta Q(u_{\omega-1}, x_{\omega-1}, t_{\omega-1}) \quad (4.19)$$

gdzie:

- $\Delta Q(u_i, x_i, t_i)$ to przyrost czasu wynikający ze zrealizowania decyzji u_i w stanie $s_i = (x_i, t_i)$,
- ω oznacza numer ostatniego stanu trajektorii.

W związku z tym, w rozważanym problemie przyrost $\Delta Q(u_i, x_i, t_i)$ jest równy przyrostowi czasu Δt_i pomiędzy kolejnymi stanami systemu:

$$\Delta Q(u_i, x_i, t_i) = \Delta t_i \quad (4.20)$$

Postać kryterium jakości wpływa na postać modelu, jego struktury stanów, decyzji i funkcji przejścia.

Ciąg decyzji minimalizujący kryterium Q wyznaczany jest na podstawie odpowiednio zaprojektowanego algorytmu optymalizacyjnego.

5. Problemy szeregowania zadań w produkcji przepływowej z brakami jakościowymi

W rozdziale tym rozważana jest klasa problemów produkcji przepływowej, w której występują braki jakościowe (defekty) elementów produkowanych i konieczność ich naprawiania. W tym przypadku czas wykonania zlecenia jest zależny od ilości wykrytych defektów, zatem nie można określić go a priori. Występowanie defektów jest wykrywane podczas kontroli jakości.

Zaprezentowana klasa problemów należy do problemów, w których pewne istotne zmienne decyzyjne lub parametry są zależne od aktualnego stanu systemu. Przykładowo na całkowity czas i koszt produkcji może mieć wpływ konieczność naprawy w trakcie procesu produkcyjnego wykrytego braku jakościowego elementu produkowanego. Problemy tego typu mogą występować w różnych dziedzinach, przykładowo w procesach lakierowania, produkcji mebli lub wytwarzania elektroniki.

5.1. Opis procesu wykrywania braków jakościowych

Zanim przedstawiony zostanie opis rozpatrywanej klasy problemów produkcji przepływowej rozważmy w jaki sposób kontrola jakości może odbywać się na różnych etapach procesu produkcyjnego. Oprócz kontroli jakości na zakończenie procesu, tzn. kontroli wyrobu gotowego, możliwe są również kontrole po wykonaniu operacji na:

- jednej konkretnej maszynie,
- na wielu określonych maszynach,
- na wszystkich maszynach.

Kontrola jakości może mieć różny zakres i mierzyć poziom dokładności wykonania zadania (produkowanego elementu). Te parametry określone są dla konkretnego procesu produkcyjnego i różnią się dla różnych procesów. Od przebiegu procesu produkcyjnego i wyniku kontroli jakości uzależnione są dalsze czynności (operacje) marszruty technologicznej. W wyniku pozytywnej kontroli jakości (spełnione wymogi jakościowe), kontrolowane zadanie (element) przekazywane jest do obróbki przez kolejną maszynę w marszrucie technologicznej. W wyniku negatywnej kontroli jakości (nie spełnione wymogi jakościowe) mogą nastąpić następujące sytuacje:

- a. element jest całkowicie wybrakowany - należy go wyrzucić (złomować), a z magazynu surowców pobrać elementy uzupełniające,

- b. element jest częściowo wybrakowany – zadanie jest **niepodzielne**:
- i. poprawa na wskazanej maszynie lub maszynach w ciągu technologicznym, powrót na stanowisko kontroli i kontynuowanie marszruty technologicznej,
 - ii. ponowna obróbka (ponowne przejście marszruty produkcyjnej) od pierwszej maszyny lub od wskazanej maszyny,
 - iii. operacja naprawy elementu na dodatkowej maszynie lub maszynach spoza marszruty technologicznej i ponowna obróbka od pierwszej maszyny lub od wskazanej maszyny z ciągu technologicznego.
- c. element jest częściowo wybrakowany – zadanie jest **podzielne** (np. jest to partia):
- i. część prawidłowo wykonana czeka, a część wybrakowana jest przekazana do ponownej obróbki (z wariantami takimi jak w punkcie b., po czym kontynuowana jest obróbka całej partii,
 - ii. część prawidłowo wykonana kontynuuje obróbkę na kolejnej maszynie (nie czeka), a część wybrakowana jest przekazana do ponownej obróbki (z wariantami takimi jak w punkcie b.).
- W tym przypadku zadanie dzielimy na wyodrębnione części i na nich wykonujemy odpowiednie operacje.

W niniejszej pracy rozważana jest klasa problemów, w której zadanie jest równoważne partii składającej się z kilku elementów i **częściowo wybrakowane** zadanie dzielone jest na dwie części. Prawidłowo wykonana część (partia składająca się z elementów prawidłowo wykonanych) kontynuuje obróbkę na kolejnej maszynie. Część wybrakowana (partia składająca się z elementów z defektami) jest przekazywana do naprawy na dodatkowej maszynie lub maszynach spoza marszruty technologicznej i ponownej obróbki od wskazanej maszyny z ciągu technologicznego. W dalszej części pracy będą używane następujące określenia dla tak podzielonych partii: subpartia prawidłowa i subpartia wybrakowana. Trudnością tych problemów jest fakt, że nie jest znany a priori czas wykonania całego zadania, ponieważ nie wiadomo jak duża część zadania będzie wybrakowana i odesłana do ponownej obróbki. Stąd nieznanym jest czas i koszt całkowity wytworzenia zadania. Poniżej przedstawiono dokładny opis rozważanych problemów.

5.2. Opis wyróżnionych problemów produkcji przepływowej z brakami jakościowymi

Niech $\mathcal{M} = \{M_1, M_2, \dots, M_m\}$ oznacza zbiór maszyn dedykowanych w marszrucie technologicznej, $\mathcal{J} = \{J_1, J_2, \dots, J_n\}$ oznacza zbiór zadań, które mają być przetworzone na tych maszynach. Zadanie jest równoważne partii składającej się z kilku elementów oraz liczba elementów partii może być różna dla każdego zadania. Ponadto zadanie jest podzielne na mniejsze zadania. Każde j -te zadanie $1 \leq j \leq n$, musi być poddane m operacjom. Określmy dla każdej operacji czas przetwarzania konkretnego zadania jako $p^{i,j}$, gdzie i numer operacji i j numer zadania. Dalej oznaczmy przez $\mathcal{M}_Q \subset \mathcal{M}$ zbiór maszyn z kontrolą jakości. Do czasu przetwarzania operacji $p^{q,j}$ na maszynie z kontrolą jakości doliczony jest czas samej kontroli. Wszystkie zadania przetwarzane są przez wszystkie maszyny w odpowiedniej

kolejności od pierwszej do ostatniej maszyny według marszruty technologicznej (M_1, M_2, \dots, M_m) . Przed każdą maszyną znajduje się magazyn, w którym znajdują się zadania przetworzone przez poprzednią maszynę i czekające na przydzielenie ich do obróbki na tej maszynie. Przy czym magazyn przed pierwszą maszyną jest magazynem początkowym. Oczywiście rozważany jest również magazyn końcowy, w którym znajdują się poprawnie (odpowiedniej jakości) wykonane zadania. Niech zatem $\mathcal{W} = \{W_0, W_1, \dots, W_m\}$ oznacza zbiór magazynów, gdzie magazyn W_0 oznacza magazyn końcowy, zaś W_i oznacza magazyn przed maszyną M_i . Przy czym zakłada się, że pojemność magazynów nie jest ograniczona. Dalej oznaczmy przez $\mathcal{M}_{\mathcal{D}} = \{M_{m+1}, \dots, M_{md}\}$ taki, że $\mathcal{M}_{\mathcal{D}} \cap \mathcal{M} = \emptyset$ zbiór maszyn dodatkowych, służących do naprawy wadliwych elementów. Przed każdą maszyną naprawczą również znajduje się magazyn. Oznaczmy zbiór tych magazynów przez $\mathcal{W}_{\mathcal{D}} = \{W_{m+1}, \dots, W_{md}\}$. Po naprawie zadania na maszynie naprawczej, zadanie powraca do ponownej obróbki na konkretną wybraną maszynę w ciągu technologicznym. Ponadto dany jest wymagany termin końcowy każdego zadania. Termin końcowy dla zadania J_j oznaczany jest przez $\bar{d}(j)$. Zatem należy tak ustalić harmonogram produkcji, aby ukończyć wszystkie zadania z odpowiednią jakością przed upłynięciem terminu końcowego (o ile możliwe). Dana jest również liczba elementów w partii zadania J i oznaczona $e(j)$.

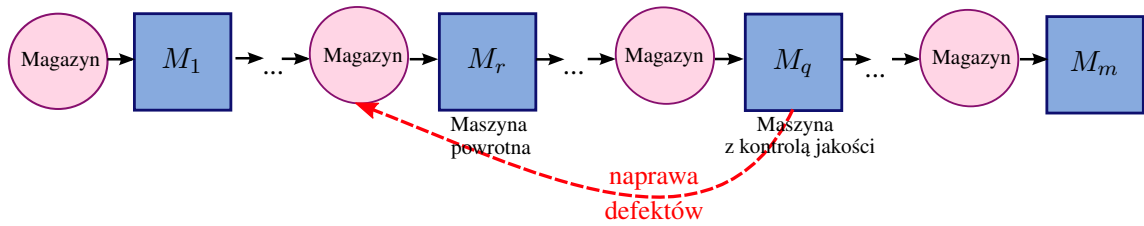
Dla tak rozważanej klasy problemów produkcyjnych wyróżnione są trzy grupy **maszyn specjalnych**:

- maszyny ze stanowiskiem kontroli jakości z marszruty technologicznej M_q ,
- wybrane maszyny z marszruty technologicznej, na które powracają po naprawie wadliwe elementy, tzw. maszyny powrotne M_r ,
- maszyny naprawcze spoza marszruty technologicznej M_d .

Na podstawie liczby występowania maszyn specjalnych, w ramach tej klasy problemów wyróżniono następujące problemy:

5.2.1. Problem z jedną maszyną z kontrolą jakości, brak maszyny naprawczej oraz jedną maszyną powrotną ($FS-M_qM_r$)

W tym problemie w marszrucie technologicznej występuje tylko jedna (ustalona) maszyna ze stanowiskiem kontroli jakości ($|\mathcal{M}_{\mathcal{Q}}| = 1$). Naprawa elementów polega na ponownej obróbce od wskazanej (ustalonej) maszyny z marszruty technologicznej i nie ma konieczności rozważania dodatkowej maszyny naprawczej w celu usunięcia braków jakościowych. Zatem wadliwe elementy powracają do ponownej obróbki przez maszyny z marszruty technologicznej począwszy od tylko jednej konkretnie wybranej maszyny. Oznaczmy maszynę ze stanowiskiem kontroli jakości jako $M_q \in \mathcal{M}_{\mathcal{Q}} \subset \mathcal{M}$ oraz maszynę (wcześniejszą), na którą wracają zadania z defektem jako $M_r \in \mathcal{M}$, gdzie $r \leq q$. W szczególnym przypadku naprawiane elementy mogą powtarzać całą marszrutę technologiczną, wtedy $r = 1$. Na rysunku 5.1 przedstawiono przykładową linię produkcyjną dla tak określonego problemu.



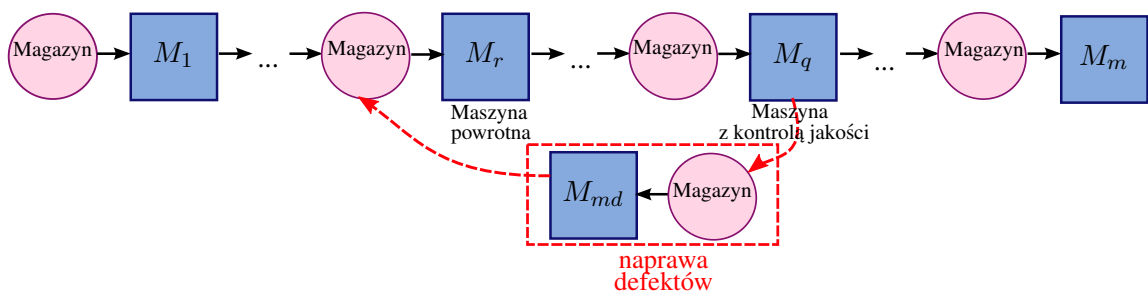
Rysunek 5.1: Przykład z maszyną z kontrolą jakości, maszyną powrotną oraz bez maszyny naprawczej

5.2.2. Problem z jedną maszyną z kontrolą jakości, jedną maszyną naprawczą oraz jedną maszyną powrotną ($FS-M_qM_dM_r$)

W tym problemie w marszrucie technologicznej występuje tylko jedna maszyna ze stanowiskiem kontroli jakości ($|\mathcal{M}_Q| = 1$) oraz tylko jedna maszyna dodatkowa, do naprawy wadliwych elementów ($|\mathcal{M}_D| = 1$ i $M_{md} = M_{m+1}$). Poza tym naprawione elementy powracają do ponownej obróbki przez maszyny z marszrutu technologicznej, począwszy od tylko jednej konkretnie wybranej maszyny. Oznaczmy maszynę ze stanowiskiem kontroli jakości jako $M_q \in \mathcal{M}_Q \subset \mathcal{M}$ oraz maszynę, na którą wracają poprawione zadania jako $M_r \in \mathcal{M}$. Można rozważyć następujące przypadki:

- $r < q$ - zadania wracają na maszynę w ciągu technologicznym przed maszyną z kontrolą jakości, co oznacza, że oprócz operacji naprawy, elementy będą poddane ponownej obróbce od wskazanej maszyny w ciągu technologicznym (M_r, \dots, M_q) i ponownie przejdą kontrolę jakości, w szczególności naprawione elementy mogą powtarzać całą marszrutę technologiczną, wtedy $r = 1$,
- $r = q$ - elementy po naprawie wracają na maszynę z kontrolą jakości,
- $r > q$ - elementy naprawione wracają na dalszą maszynę w marszrucie technologicznej, przykładowo tylko na ostatnią maszyną, w celu zapakowania.

Na rysunku 5.2 przedstawiono przykładową linię produkcyjną dla tak określonego problemu.



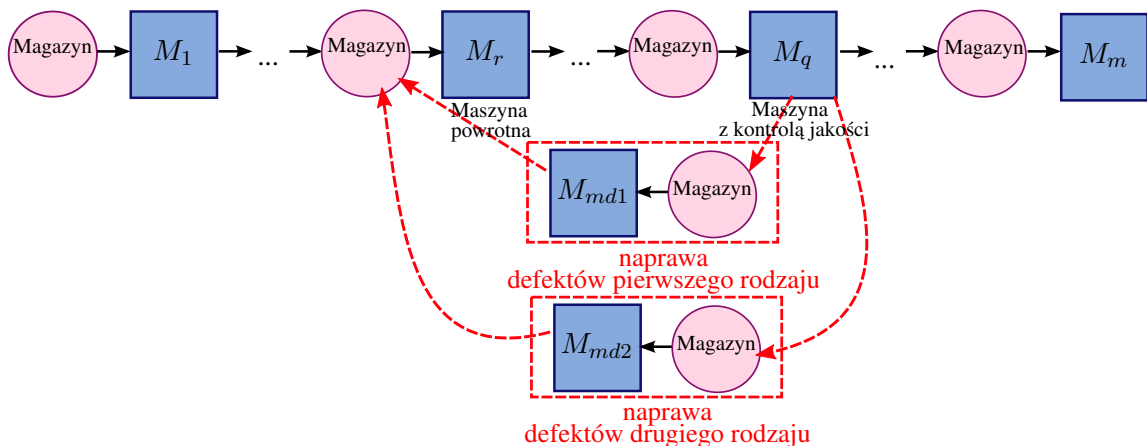
Rysunek 5.2: Przykład z maszyną z kontrolą jakości, maszyną powrotną oraz z maszyną naprawczą

5.2.3. Problem z jedną maszyną z kontrolą jakości, więcej niż jedną maszyną naprawczą oraz jedną maszyną powrotną ($FS-M_qmM_dM_r$)

W tym problemie w marszrucie technologicznej występuje tylko jedna maszyna ze stanowiskiem kontroli jakości ($|\mathcal{M}_Q| = 1$) oraz więcej niż jedna maszyna dodatkowa, do naprawy wadliwych elementów ($|\mathcal{M}_D| > 1$ oraz $\mathcal{M}_D = \{M_{m+1}, \dots, M_{md}\}$). Wybór maszyny do naprawy jest zależny od stopnia uszkodzenia wadliwych elementów i jest wynikiem kontroli jakości. Naprawione elementy powracają do ponownej obróbki przez maszyny z marszruty technologicznej począwszy od konkretnej tylko jednej maszyny. Oznaczmy maszynę ze stanowiskiem kontroli jakości jako $M_q \in \mathcal{M}_Q \subset \mathcal{M}$ oraz maszynę, na którą wracają poprawione zadania jako $M_r \in \mathcal{M}$. Można rozważyć następujące przypadki:

- $r < q$ - zadania wracają na maszynę w ciągu technologicznym przed maszyną z kontrolą jakości, co oznacza, że oprócz operacji naprawy, elementy będą poddane ponownej obróbce od wskazanej maszyny w ciągu technologicznym (M_r, \dots, M_q) i ponownie przejdą kontrolę jakości, w szczególnym przypadku naprawione elementy mogą powtarzać całą marszrutę technologiczną, wtedy $r = 1$,
- $r = q$ - elementy po naprawie wracają na maszynę z kontrolą jakości,
- $r > q$ - elementy naprawione wracają na dalszą maszynę w marszrucie technologicznej, przykładowo tylko na ostatnią maszynę, w celu zapakowania.

Na rysunku 8.1 przedstawiono przykładową linię produkcyjną dla tak określonego problemu.

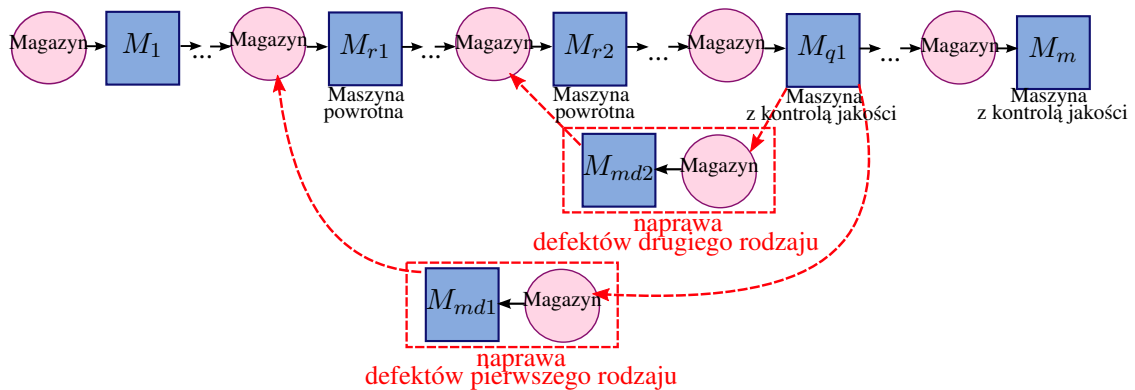


Rysunek 5.3: Przykład z maszyną z kontrolą jakości, maszyną powrotną oraz dwoma maszynami naprawczymi

5.2.4. Problem z jedną maszyną z kontrolą jakości, więcej niż jedną maszyną naprawczą oraz więcej niż jedną maszyną powrotną ($FS-M_qmM_dM_r$)

W tym problemie w marszrucie technologicznej występuje tylko jedna maszyna ze stanowiskiem kontroli jakości ($|\mathcal{M}_Q| = 1$) oraz więcej niż jedna maszyna dodatkowa, do naprawy wadliwych elementów ($|\mathcal{M}_D| > 1$ i $\mathcal{M}_D = \{M_{m+1}, \dots, M_{md}\}$). Wybór maszyny do naprawy jest zależny od stopnia

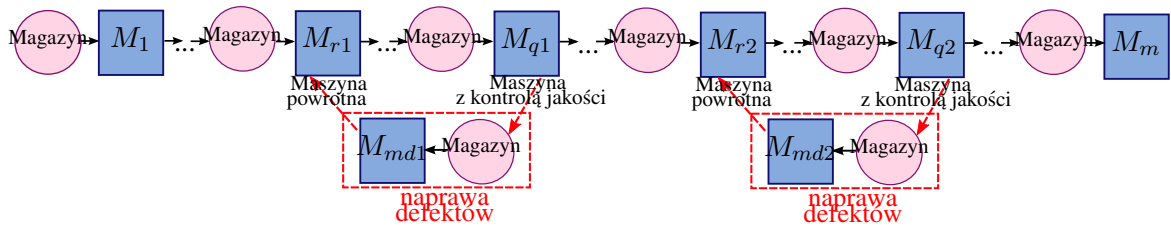
lub rodzaju uszkodzenia wadliwych elementów i jest wynikiem kontroli jakości. Naprawione elementy powracają na różne konkretnie wybrane (określone) maszyny w marszrucie technologicznej. Nie jest wykluczana sytuacja, w której z więcej niż jednej maszyny dodatkowej poprawione zadania wracają na jedną konkretną maszynę w ciągu technologicznym. Natomiast wykluczona jest sytuacja, w której z jednej maszyny dodatkowej poprawione zadania wracają na więcej niż jedną maszyną z marszrutę technologiczną. Oznaczmy maszynę ze stanowiskiem kontroli jakości jako $M_q \in \mathcal{M}_Q \subset \mathcal{M}$ oraz indeksy maszyn, na które wracają poprawione zadania jako $r_1 < \dots < r_b \leq q$ i $M_{r_1}, \dots, M_{r_b} \in \mathcal{M}$. Na rysunku 8.3 przedstawiono przykładową linię produkcyjną dla tak określonego problemu.



Rysunek 5.4: Przykład z maszyną z kontrolą jakości oraz dwiema maszynami powrotnymi i naprawczymi

5.2.5. Problem z więcej niż jedną maszyną z kontrolą jakości, więcej niż jedną maszyną naprawczą oraz więcej niż jedną maszyną powrotną ($FS-mM_qmM_dM_r$)

W tym problemie w marszrucie technologicznej występuje więcej niż jedna maszyna ze stanowiskiem kontroli jakości ($1 < |\mathcal{M}_Q| \leq |\mathcal{M}|$) oraz więcej niż jedna maszyna dodatkowa, do naprawy wadliwych elementów ($|\mathcal{M}_D| > 1$ i $\mathcal{M}_D = \{M_{m+1}, \dots, M_{md}\}$). Wybór maszyny do naprawy jest zależny od stopnia lub rodzaju uszkodzenia wadliwych elementów i jest wynikiem kontroli jakości. Zatem w wyniku każdej kontroli otrzymujemy informację, na którą konkretną maszynę ma być przekazane zadanie. Wykluczona jest sytuacja, w której z dowolnych dwóch stanowisk kontroli, elementy z defektem odsyłane są na tą samą maszynę dodatkową. Naprawione elementy powracają na różne konkretnie wybrane maszyny w marszrucie technologicznej. Nie jest wykluczana sytuacja, w której z więcej niż jednej maszyny dodatkowej poprawione zadania wracają na jedną konkretną maszynę w ciągu technologicznym. Natomiast wykluczona jest sytuacja, w której z jednej maszyny naprawczej poprawione zadania wracają na więcej niż jedną maszyną z marszrutę technologiczną. Oznaczmy indeksy maszyn ze stanowiskiem kontroli jakości jako $q_1 < \dots < q_d \in \mathcal{M}_Q \subset \mathcal{M}$ oraz maszyny, na które wracają poprawione zadania jako: $r_1 < \dots < r_b \in \mathcal{M}$. Z tym, że jeśli z maszyny ze stanowiskiem kontroli jakości M_{q_i} , gdzie: $i \in \{1, \dots, d\}$, po naprawie zadanie wraca na maszynę M_{r_i} , gdzie: $i \in \{1, \dots, b\}$, to musi zachodzić zależność $r_i \leq q_i$. Na rysunku 8.4 przedstawiono przykładową linię produkcyjną dla tak określonego problemu.



Rysunek 5.5: Przykład z dwiema maszynami z kontrolą jakości

5.3. Opis procesu technologicznego w malarni proszkowej jako przykład produkcji przepływowej z brakami jakościowymi

Proces produkcyjny napyłania farby proszkowej jest przykładem problemu produkcyjnego 5.2.4, w którym rozważane jest jedno stanowisko kontroli jakości oraz dwie różne operacje spoza marszruty technologicznej służące do naprawy wadliwych elementów.

Malowanie proszkowe polega na nakładaniu farby proszkowej na malowane elementy metalowe. W malowaniu proszkowym stosuje się technikę natrysku elektrostatycznego lub elektrokinetycznego. Po nałożeniu (napyleniu) farby elementy są wypalane w specjalnym piecu. W wyniku wypalania farba proszkowa ulega stopieniu i polimeryzacji. Uzyskana powłoka lakiernicza jest odporna na korozję, chemikalia, wysoką temperaturę i uszkodzenia mechaniczne oraz odznacza się bardzo wysokimi walorami estetyczno-dekoracyjnymi.

Elementy są malowane partiami. Wszystkie elementy partii umieszczane są w specjalnym koszu (o określonej pojemności) i w nim przechodzą kolejno przez wszystkie etapy procesu technologicznego. Proces technologiczny malowania proszkowego podzielony jest na następujące etapy:

- przygotowanie powierzchni do lakierowania,
- napyłanie farby proszkowej,
- obróbka termiczna,
- pakowanie.

W etapie przygotowania powierzchni elementy metalowe poddawane są procesom chemicznym odtłuszczenia, trawienia, bądź piaskowania. Ten etap jest bardzo ważny ze względu na uzyskaną jakość gotowego wyrobu (jakość otrzymanej powierzchni). Rozważamy chemiczne przygotowanie powierzchni, które odbywa się metodą zanurzeniową. Realizowane jest ono na dwóch stanowiskach roboczych. Pierwszym są wanny ze specjalnymi kąpielami chemicznymi. Na drugim stanowisku następuje suszenie elementów.

Etap napyłania realizowany jest w specjalnej kabynie, która wyposażona jest między innymi w urządzenia umożliwiające transport elementów oraz elektryzujące proszek farby, a także pistolety do napyłania farby. Naelektryzowane cząsteczki farby, unoszone w strumieniu powietrza, po skierowaniu ich na uziemiony przedmiot osiadają na jego powierzchni. Wydajność napyłania zależy od zastosowanego typu kabiny i ilości zastosowanych w niej pistoletów. Ponadto w zależności od typu kabiny proszek

który osiada na ścianach kabiny można odzyskiwać do ponownego wykorzystania lub nie, wtedy jest utylizowany.

Następnie malowane elementy poddawane są obróbce termicznej w piecu komorowym w temperaturze około $200^{\circ}C$. Czas jednego cyklu zależy od rodzaju malowanych elementów i wynosi 15-45 minut.

Po obróbce termicznej następuje kontrola jakości, w trakcie której sprawdzane są pomalowane elementy. W zależności od wyniku kontroli realizowane są różne czynności. Gdy jakość jest zadowalająca, wynik kontroli jest pozytywny. Wtedy elementy kierowane są na stanowisko pakowania i przygotowania do wydania odbiorcy. Jeśli jakość nie jest zadowalająca to określany jest rodzaj braku. Jeśli powłoka jest za gruba lub za cienka albo gdy w powłoce znajdują się zanieczyszczenia, ale czynniki te występują w nieznacznym stopniu, to wadliwe elementy podlegają operacji matowienia i wraca na stanowisko napyłania, po czym kontynuowany jest proces technologiczny. Gdy stopień błędów jest znaczny, to wadliwy element musi zostać w całości odlakierowany i powtórzony musi zostać cały proces technologiczny, począwszy od kąpieli chemicznej.

5.3.1. Złożoność obliczeniowa problemu przepływowego z kryterium C_{max}

Problem szeregowania zadań w systemie przepływowym $F_m || C_{max}$ jest problemem silnie NP-trudnym (Pinedo (2008)). W pracy Pinedo (2008) został przedstawiony dowód twierdzenia, że problem $F_3 || C_{max}$ jest już problemem silnie NP-trudnym. Co oznacza, że dla tego problemu optymalizacji nie istnieją dokładne algorytmy pseudo-wielomianowe. W literaturze najczęściej rozważany jest permutacyjny problem przepływowy $F_m | pmu | C_{max}$, ze względu na to, że dla danej permutacji J_1, J_2, \dots, J_n czas zakończenia wykonania zadania J_k na maszynie M_i wynosi:

$$C_{i,k} = \max(C_{i-1,k}, C_{i,k-1}) + p_{i,k}$$

gdzie:

$$C_{1,k} = \sum_{l=1}^k p_{1,l} \quad k = 1, \dots, n$$

$$C_{i,1} = \sum_{l=1}^i p_{l,1} \quad i = 1, \dots, m.$$

Ponadto dla problemu $F_m | pmu, p_{ij} = p_j | C_{max}$, w którym dla danego zadania J_j czas przetwarzania przez każdą z maszyn jest taki sam, znane jest rozwiązanie optymalne:

Twierdzenie 1 Dla $F_m | pmu, p_{ij} = p_j | C_{max}$ minimalny czas wykonywania wszystkich zadań jest niezależny od uszeregowania i wynosi

$$C_{max} = \sum_{j=1}^n p_j + (m-1) \max(p_1, \dots, p_n).$$

Twierdzenie to podano w Pinedo (2008) bez dowodu. W dodatku C zaproponowano taki dowód. Ze względu na to, że dla problemu $F_m | pmu, p_{ij} = p_j | C_{max}$ znane jest optymalne rozwiązanie i rozwiązanie to jest łatwo obliczalne.

6. Modelowanie problemu $FS-M_qM_r$ za pomocą podejścia ALMM z zastosowaniem metody przełączania

W rozdziale tym przedstawiono rozwiązanie dla problemu produkcji przepływowej z brakami jakościowymi $FS-M_qM_r$, gdzie występowanie braków jakościowych wykrywane jest na jednym stanowisku kontroli jakości M_q oraz naprawa elementów wybrakowanych polega na ponownej obróbce od wskazanej maszyny M_r w marszrucie technologicznej, czyli nie jest wymagana żadna dodatkowa operacja naprawcza. Problem ten należy do klasy problemów re-entrant.

Opis wyróżnionego problemu został przedstawiony w rozdziale 5.2.1. Celem optymalizacji rozważanego problemu jest jak najszybsze wykonanie wszystkich zadań.

Rozwiązanie problemu $FS-M_qM_r$ będzie polegać na przełączaniu modeli algebraiczno-logicznych. Przedstawiono kolejno etapy rozwiązywania problemu z uwzględnieniem występowania braków jakościowych: analizę problemu, podział problemu niedeterministycznego na deterministyczne problemy pomocnicze oraz modele ALM dla problemów pomocniczych. Modele algebraiczno-logiczne przedstawione w tym rozdziale, to modyfikacja modeli opracowanych we współautorstwie autorki z E. Dudek-Dyduch oraz E. Kucharską i przedstawionych w pracy Grobler-Dębska et al. (2013). W dalszej części rozdziału przedstawiono wyróżnione dla rozważanego problemu: rodzaje przełączania, reguły przełączania oraz zaproponowany został algorytm przełączania dla problemu $FS-M_qM_r$.

6.1. Analiza problemu

W rozważanym problemie występują dwie charakterystyczne maszyny w marszrucie technologicznej. Jedną z wyróżnionych maszyn jest maszyna M_q ze stanowiskiem kontroli jakości, zaś drugą M_r maszyna powrotna. Na wyróżnioną maszynę powrotną M_r trafiają zadania do naprawy, wskazane w wyniku negatywnej kontroli jakości. Naprawa wadliwych zadań polega na ponownym wykonaniu operacji od wskazanej maszyny w marszrucie technologicznej M_r . W związku z tym należy zauważyć, że maszyna powrotna M_r powinna mieć numer nie większy niż maszyna z kontrolą jakości M_q , czyli $r \leq q$. Do naprawy wadliwych elementów nie jest wymagana żadna dodatkowa maszyna. Tak zdefiniowany problem będzie należał do klasy problemów re-entrant, przy czym nie zakłada się, że wskazana maszyna powrotna, jest zawsze pierwsza na linii produkcyjnej. A takie założenie często przyjmuje się dla problemów re-entrant.

Ponadto w odróżnieniu od klasycznych problemów re-entrant, gdzie całe zadanie (partia) wraca do ponownej obróbki, w rozważanym problemie również dopuszcza się częściowe wybrakowanie zadań.

Zatem w wyniku wykrycia w zadaniu (partii) nowych braków jakościowych zadanie dzielone jest na dwie subpartie. Prawdłowo wykonana subpartia kontynuuje obróbkę na kolejnej maszynie, zaś wybrakowana subpartia (partia składająca się z elementów z defektami) jest przekazywana do ponownej obróbki od wskazanej maszyny z ciągu technologicznego.

W związku z powyższym w tym problemie będą występować jedynie zaburzenia w zbiorze zadań. Mianowicie z powodu podziału partii na dwie subpartie zwiększeniu ulegnie zbiór zadań do wykonania o jedno zadanie. Proponuje się by subpartia wykonana prawidłowo zachowała indeks skontrolowanej partii. Zatem ze zbioru zadań nie będzie usuwane żadne zadanie, natomiast zmianie ulegnie liczba elementów w zadaniu, czyli zmieniają się parametry tego zadania. W przypadku, w którym całe zadanie jest wybrakowane otrzymuje ono nowy indeks.

6.2. Podział problemu na problemy pomocnicze

W rozważanym problemie dany jest stały zbiór maszyn $\mathcal{M} = \{M_1, \dots, M_m\}$ ułożony w ciąg technologiczny (M_1, \dots, M_m) wraz z wyróżnioną jedną maszyną ze stanowiskiem kontroli jakości $M_q \in \mathcal{M}$ i jedną maszyną $M_r \in \mathcal{M}$, od której następuje ponowna obróbka. Zmianie ulega zbiór zadań \mathcal{J} . W związku z tym rozważany będzie jeden problem pomocniczy produkcji przepływowej, w którym wyróżniona jest maszyna z ciągu technologicznego ze stanowiskiem kontroli jakości. Aby uwzględnić zmiany na zbiorze zadań zmieniana będzie instancja problemu pomocniczego (zwiększenie zbioru zadań). Aby uwzględnić zmiany instancji problemu po wykryciu braku jakościowego, *model będzie przełączany na model o dokładnie takiej samej strukturze*, natomiast inna będzie wartość stanu początkowego. W związku z tym przedstawiony będzie jeden model problemu. Celem jest jak najszybsze wykonanie wszystkich zadań.

6.3. Model algebraiczno-logiczny problemu pomocniczego

Model algebraiczno-logiczny dla problemu pomocniczego jest budowany na bazie modelu algebraiczno-logicznego dla klasycznego problemu produkcji przepływowej z z deadlineami i magazynami międzyoperacyjnymi o nieograniczonej pojemności. Model ten został przedstawiony w rozdziale 4. Dla zdefiniowania procesu za pomocą modelu algebraiczno-logicznego przedstawiona została najpierw postać stanu systemu, zbiory stanów docelowych oraz stanów niedopuszczalnych. Następnie określona została postać decyzji, zbiór decyzji możliwych do podjęcia w poszczególnych stanach, zbiór decyzji dopuszczalnych oraz zaprezentowana została funkcja przejścia.

Stan systemu Stan systemu $s = (x, t)$ w danej chwili t można opisać analogicznie jak w rozdziale 4 poprzez aktualny stan maszyn z marszruty technologicznej, stan magazynu początkowego i stan magazynów międzyoperacyjnych.

Z punktu widzenia metody przełączania istotne jest, aby w modelu pomocniczym ALM_A wyróżnić z marszruty technologicznej maszynę z kontrolą jakości M_q i maszynę powrotną M_r , gdzie $r \leq q$, oraz magazyn międzyoperacyjny przed maszyną powrotną W_r . Współrzędne stanu właściwego maszyny

powrotnej i magazynu są takie same jak pozostałych maszyn i magazynów w marszrucie technologicznej. Natomiast w stanie systemu zmodyfikowany został stan maszyny M_q ze stanowiskiem kontroli jakości.

Zatem stan właściwy procesu jest zdefiniowany jako n -tka:

$$x = \underbrace{(x^1, x^2, \dots, x^m)}_{\text{stany magazynów}}, \underbrace{(x^{m+1}, \dots, x^{m+q}, \dots, x^{2m})}_{\text{stany maszyn}} \quad (6.1)$$

gdzie:

x^1 – oznacza stan magazynu początkowego,

x^i – oznacza stan i -tego magazynu międzyoperacyjnego W_i (pomiędzy $i - 1$ i i -tą maszyną) dla $i = 2, 3, \dots, m$,

x^{m+i} – oznacza stan i -tej maszyny dla $i = 1, 2, \dots, m$ i $i \neq q$,

x^{m+q} – oznacza stan wyróżnionej maszyny z kontrolą jakości M_q .

Analogicznie jak w modelu w rozdziale 4 współrzędne stanu magazynów są następujące. Współrzędna x^1 jest to zbiór zadań do wykonania na pierwszej maszynie, tzn. zadań których obróbka nie została rozpoczęta. Współrzędne x^i , dla $i = 2, 3, \dots, m$, to zbiory zadań, które zostały obrobione na maszynie $i - 1$ i aktualnie są dostępne do obróbki na i -tej maszynie, ale nie zostały jeszcze przydzielone do tej maszyny.

Stan właściwy maszyny bez stanowiska z kontrolą jakości Struktura stanu maszyny jest następująca:

$$x^{m+i} = (\beta, \tau) \quad (6.2)$$

gdzie:

$\beta \in \{0, 1, \dots, n\}$ – oznacza numer zadania, które jest aktualnie obrabiane na danej maszynie (0 oznacza, że żadne zadanie nie zostało przydzielone do maszyny),

$\tau \in [0, \infty)$ – czas do ukończenia obróbki aktualnego zadania na maszynie (∞ oznacza, że maszyna nie pracuje).

Stan właściwy wyróżnionej maszyny ze stanowiskiem kontroli jakości Struktura stanu wyróżnionej maszyny ze stanowiskiem kontroli jest rozszerzona o dwa parametry związane z wynikiem kontroli jakości:

$$x^{m+q} = (\beta, \tau, QC, \hat{\beta}) \quad (6.3)$$

gdzie: β i τ – oznaczają odpowiednio i przyjmują wartości numeru zadania oraz czasu do ukończenia zadania, tak jak w przypadku kodowania stanów maszyn bez stanowiska kontroli jakości (6.2),

$QC \in [0, 100]$ – oznacza procent wybrakowanych elementów w partii (0 oznacza pozytywny wynik kontroli jakości, czyli wszystkie elementy w partii są wykonane poprawnie),

$\hat{\beta} \in \{0, 1, \dots, n\}$ – oznacza indeks zadania, w którym wykryto braki jakościowe podczas ostatniej kontroli jakości.

W poprzednim rozdziale wyróżniono wzorami dwa charakterystyczne stany maszyn: stan maszyny, w którym maszyna nie pracuje (jest wolna 4.4) oraz stan maszyny, w którym maszyna pracuje (jest zajęta 4.5). Analogicznie dla maszyny ze stanowiskiem kontroli jakości można wyróżnić te stany, przy czym

dodatkowo należy uwzględnić wynik kontroli jakości (dwie ostatnie współrzędne stanu maszyny). W związku z tym dla maszyny M_q wyróżniono następujące charakterystyczne stany:

- M_q maszyna ze stanowiskiem kontroli jakości nie pracuje (jest wolna) w stanie s i wynik ostatniej kontroli jakości jest pozytywny:

$$x^{m+q}(s) = (0, \infty, 0, \widehat{\beta}), \quad (6.4)$$

- M_q maszyna ze stanowiskiem kontroli jakości nie pracuje (jest wolna) w stanie s i wynik ostatniej kontroli jakości jest negatywny:

$$x^{m+q}(s) = (0, \infty, QC > 0, \widehat{\beta} \neq 0), \quad (6.5)$$

- M_q maszyna ze stanowiskiem kontroli jakości pracuje (jest zajęta) w stanie s :

$$x^{m+q}(s) = (\beta \neq 0, \tau > 0, 0 \leq QC \leq 100, \widehat{\beta} \geq 0). \quad (6.6)$$

Stan początkowy systemu $s_0 = (x_0, t_0)$ W stanie początkowym, gdzie ($t_0 = 0$), tak samo jak w modelu rozważanym w rozdziale 4, wszystkie zadania są w magazynie początkowym, nie rozpoczęto obróbki żadnego zadania i żadne zadanie nie zostało jeszcze przydzielone do pierwszej maszyny. Stąd stan początkowy $s_0 = (x_0, t_0)$ jest następujący:

$$x_0 = (x_0^1, x_0^2, \dots, x_0^m, x_0^{m+1}, \dots, x_0^{m+q}, \dots, x_0^{2m}) \quad (6.7)$$

gdzie poszczególne współrzędne stanów magazynów i maszyn przyjmują następujące wartości:

$$x_0^1 = \mathcal{J},$$

$$x_0^i = \emptyset \text{ dla } i = 2, 3, \dots, m,$$

$$x_0^{m+i} = (0, \infty) \text{ dla } i = 1, 2, \dots, 2m \text{ i } i \neq q,$$

$$x_0^{m+q} = (0, \infty, 0, 0).$$

Zbiór stanów niedopuszczalnych S_N i **Zbiór stanów docelowych** S_G Zarówno zbiór stanów niedopuszczalnych S_N jak i zbiór stanów docelowych S_G mają dokładnie tę samą postać jak w rozdziale 4. Stan niedopuszczalny jest to stan, w którym istnieje niedokończony zadanie, a jego deadline wykonania minął. Stan $s = (x, t)$ nazywamy docelowym, wtedy gdy wszystkie zadania zostały ukończone, zaś ich deadline wykonania nie minął.

Decyzje W rozważanym problemie, tak samo jak w problemie rozważanym w rozdziale 4, przyjęto, iż decyzja polega na przydzieleniu zadań do konkretnych maszyn w tym samym czasie. W danym stanie $s = (x, t)$ należy podjąć decyzję $u \in U = U^1 \times U^2 \times \dots \times U^m$ i na podstawie tej decyzji wyznaczany jest kolejny stan systemu. Poszczególne współrzędne decyzji $u^i \in U^i$ reprezentują poszczególne oddzielne decyzje dla i -tej maszyny. Zatem decyzja jest wektorem:

$$u = (u^1, u^2, \dots, u^m) \quad (6.8)$$

gdzie wartość $u^i = j$ dla $(j = 1, \dots, n)$ oznacza przydzielenie nowego j -tego zadania do i -tej maszyny, zaś $u^i = 0$ oznacza kontynuację przez maszynę wcześniej przydzielonego zadania. Zatem współrzędne decyzji mogą przyjmować tylko wartości indeksu zadania lub 0 ($u^i \in \mathcal{J} \cup \{0\}$).

Dodatkowo uwzględnienie w problemie maszyny z kontrolą jakości, nie zmienia struktury decyzji u . Zatem decyzje u , które należą do zbioru decyzji możliwych $U_p(x, t)$ są dane wzorem (4.8).

Funkcja przejścia Na podstawie aktualnego stanu $s = (x, t)$ i decyzji u podjętej w tym stanie, generowany jest następny stan $s' = (x', t')$ przy pomocy funkcji przejścia f .

$$(x', t') = f(u, x, t) \quad (6.9)$$

Jak już wcześniej podano, funkcja przejścia jest zdefiniowana przy pomocy dwóch funkcji $f = (f_x, f_t)$, gdzie f_x określa następny stan właściwy, a f_t określa następny moment czasu.

W pierwszej kolejności należy wyznaczyć moment t' wystąpienia następnego stanu. Obliczane jest to analogicznie jak w rozdziale 4. Moment t' jest to najbliższy moment, w którym co najmniej jedna maszyna zakończy obróbkę przydzielonego zadania. W tym celu dla każdej maszyny obliczany jest czas ukończenia t_{M_i} niezbędny na zrealizowanie podjętej w stanie $s = (x, t)$ dla niej decyzji. Najmniejszy z tych czasu posłuży do wyznaczenia czasu t' .

$$t' = t + \Delta t, \quad \text{gdzie: } \Delta t = \min_{M_i \in \mathcal{M}} t_{M_i} \quad (6.10)$$

Analogicznie jak w rozdziale 4, ze względu na charakterystyczne stany maszyn oraz różne typy decyzji, czas ten dany jest wzorem (4.13).

Po wyznaczeniu chwili t' wystąpienia następnego stanu, obliczane są nowe wartości współrzędnych właściwych tego stanu.

Współrzędna x^1 (stan magazynu początkowego - zbiór zadań do obróbki na pierwszej maszynie) jest pomniejszany przez zadanie przydzielone do obróbki na pierwszej maszynie i dane jest wzorem:

$$(x^1(s))' = x^1(s) \setminus \{j : u^1(s) = j\}. \quad (6.11)$$

Współrzędna x^i dla $i = 2, 3, \dots, m$, (zbiór zadań ukończonych przez $i - 1$ maszynę i nieprzydzielonych do obróbki przez i -tą maszynę), jest zwiększana przez zadanie wykonane przez $i - 1$ maszynę i pomniejszane przez zadanie przypisane do i -tej maszyny z ciągu technologicznego:

$$(x^i(s))' = x^i(s) \cup (\{j : x^{i-1+m}(s) = (j, \Delta t)\} \setminus \{j : u^i(s) = j\}) . \quad (6.12)$$

Kolejne współrzędne $x^{i+m} = (\beta, \tau)$ dla $i = 1, 2, \dots, m$ i $i \neq q$, reprezentujące stany poszczególnych maszyn (bez maszyny ze stanowiskiem kontroli jakości) również są przeprowadzane do nowego stanu $(x^{i+m})' = (\beta', \tau')$ i zależą od podjętej decyzji w taki sposób jak zostało to przedstawione w rozdziale 4 i dane wzorami (4.16) – (4.17).

Współrzędna $x^{q+m} = (\beta, \tau, QC, \hat{\beta})$ reprezentująca stan maszyny ze stanowiskiem kontroli jakości jest również przeprowadzana do nowego stanu $(x^{q+m})' = (\beta', \tau', QC', \hat{\beta}')$ i nowy stan zależy od podjętej decyzji i wyniku kontroli jakości. Wynik kontroli jakości jest podany w momencie zakończenia wykonania zadania. Niech $QC(j)$ oraz $\hat{\beta}(j)$ oznaczają odpowiednio wynik kontroli jakości dla zadania J_j . W związku z tym funkcja przejścia dla stanu maszyny z kontrolą jakości jest następująca:

1. jeśli maszyna jest wolna i wynik kontroli jakości jest pozytywny ($x^{q+m} = (0, \infty, 0, \widehat{\beta})$) w danym stanie i decyzja $u^q(s) = j$ przypisuje j -te zadanie do tej maszyny, to:

i dla $\tau^{q+m} = \Delta t$, w kolejnym stanie maszyna z kontrolą jakości będzie wolna i sczytany zostanie nowy wynik kontroli jakości:

$$(x^{q+m})' = (0, \infty, QC(j), \widehat{\beta}(j)), \quad (6.13)$$

ii w przeciwnym przypadku gdy $\tau^{q+m} > \Delta t$, w kolejnym stanie maszyna będzie nadal przetwarzać zadanie j i czas do dokończenia zadania będzie wynosił $p^{qj} - \Delta t$ zaś wynik kontroli jakości będzie taki sam jak w stanie poprzednim:

$$(x^{q+m})' = (j, p^{qj} - \Delta t, 0, \widehat{\beta}). \quad (6.14)$$

2. jeśli q -ta maszyna jest wolna i wynik kontroli jakości jest pozytywny $x^{q+m} = (0, \infty, 0, \widehat{\beta})$ w danym stanie i podjęta decyzja $u^q(s) = 0$ nie przypisuje żadnego zadania do tej maszyny, to: $(x^{q+m})' = x^{q+m}$

3. jeśli q -ta maszyna jest zajęta $x^{q+m} = (j, \tau, QC, \widehat{\beta})$ w danym stanie i podjęta została jedyna możliwa decyzja $u^q(s) = 0$ o kontynuowaniu przetwarzania wcześniej przydzielanego zadania j , to:

i dla $\tau^{q+m} = \Delta t$, analogicznie jak w przypadku 1 w kolejnym stanie maszyna z kontrolą jakości będzie wolna i sczytany zostanie nowy wynik kontroli jakości, czyli kolejny stan będzie dany wzorem (6.13),

ii w przeciwnym przypadku, gdy $\tau^{q+m} > \Delta t$, to w kolejnym stanie maszyna będzie nadal przetwarzać zadanie j i czas do dokończenia zadania będzie wynosił $\tau - \Delta t$ zaś wynik kontroli jakości będzie taki sam jak w stanie poprzednim:

$$(x^{q+m})' = (j, \tau - \Delta t, QC, \widehat{\beta}). \quad (6.15)$$

W przypadku kiedy maszyna z kontrolą jakości jest wolna, a wynik kontroli jest negatywny, to stan następny nie jest obliczany za pomocą funkcji przejścia, tylko za pomocą funkcji przełączającej.

6.4. Funkcja przełączająca

W rozdziale tym przedstawiono reguły przełączania, wyróżniono zbiór stanów, w których następują przełączenia oraz algorytm przełączania.

6.4.1. Reguły przełączania

W rozważanym problemie, braki jakościowe wykrywane są podczas kontroli jakości, na jednej wybranej maszynie M_q i subpartie z wybrakowanymi elementami są poddawane ponownej obróbce od wybranej maszyny z marszruty technologicznej M_r , gdzie $r \leq q$.

Dla tak zdefiniowanego problemu tworzony jest problem pomocniczy i definiowany jest dla niego model algebraiczno-logiczny. Wykrycie braku jakościowego powoduje zmiany w zbiorze zadań (podział na subpartie), czyli zmianę instancji problemu. Zmiany te zostaną uwzględnione poprzez przełączenie na model o tej samej strukturze, ale o innej wartości stanu początkowego.

Oznaczmy przez J_y partię z wykrytym brakiem jakościowym. Zostanie ono podzielone na dwie subpartie. Subpartie poprawnie wykonaną, którą oznaczać będziemy dalej jako J_y . Subpartia poprawnie wykonana J_y kontynuuje obróbkę zgodnie z marszrutą technologiczną, zatem w związku ze zmniejszeniem elementów w partii zostaną przeliczone czasy obróbki zadania na kolejnych maszynach, czyli p^{iy} dla każdego $i > q$. Subpartię wadliwą oznaczać będziemy jako J_z . Subpartia wadliwa J_z jest poddawana ponownej obróbce na marszrucie technologicznej od maszyny M_r zatem dla tej subpartii zostaną obliczone wszystkie czasy obróbki p^{iz} dla każdego $i > r$

6.4.2. Zbiór stanów przełączania \mathcal{S}_{switch}

Kolejnym etapem metody przełączania jest określenie zbioru stanów przełączania oznaczanego przez \mathcal{S}_{switch} . Przełączenie następuje wtedy, gdy maszyna ze stanowiskiem kontroli jakości skończy pracę i wynik kontroli jakości jest negatywny, co świadczy o wykryciu braków jakościowych. Zatem do zbioru stanów przełączania należą te stany, w których maszyna ze stanowiskiem kontroli jakości jest wolna i wynik kontroli jakości jest negatywny:

$$\mathcal{S}_{switch} = \{s = (x, t) : \beta^{q+m} = 0, \tau^{q+m} = \infty, QC^{q+m} \neq 0, \hat{\beta}^{q+m} \neq 0\} \quad (6.16)$$

6.4.3. Algorytm przełączania

Poniżej przedstawiona jest funkcja przełączania pomiędzy dwoma modelami algebraiczno-logicznymi. Wprowadźmy pomocnicze oznaczenia:

- ALM_{now} oznacza aktualny model algebraiczno-logiczny, z którego nastąpi przełączenie,
- $(s_k)_{now}$ oznacza aktualny stan w modelu ALM_{now} , należący do zbioru stanów przełączania $(s_k)_{now} \in \mathcal{S}_{switch}$,
- ALM_{next} oznacza model algebraiczno-logiczny, na który nastąpi przełączenie,
- $(s_0)_{next}$ oznacza stan początkowy w modelu ALM_{next} , obliczony na podstawie funkcji przełączającej.

Zgodnie z przedstawioną regułą przełączania negatywna kontrola jakości powoduje zmiany w zbiorze zadań. W takim przypadku wystarczy dokonać przełączenia pomiędzy modelami algebraiczno-logicznymi o tej samej strukturze, obliczając stan początkowy $(s_0)_{next}$ na podstawie aktualnego stanu $(s_k)_{now}$ i wyniku negatywnej kontroli jakości. Podkreślimy więc, że nie jest generowany stan s' za pomocą funkcji przejścia, tylko obliczany jest stan początkowy $(s_0)_{next}$ modelu ALM_{next} :

$$(s_k)_{now} = (x_k, t_k)_{now} \longrightarrow (s_0)_{next} = (x_0, t_0)_{next}$$

. Zauważmy, że w tej sytuacji maszyna z kontrolą jakości M_q jest wolna w stanie s_k . W tym stanie, stan właściwy maszyny z kontrolą jakości wynosi $x^{m+q}(s_k) = (0, \infty, QC > 0, y)$, gdzie $QC > 0$ oznacza procentową ilość wadliwych elementów z zadania J_y , a y jest to numer zadania, w którym wykryto brak jakościowy.

Rozważane subparie J_y oraz J_z , gdzie $z = |\mathcal{J}| + 1$ powinny zostać przypisane do odpowiednich magazynów międzyoperacyjnych. Zadanie J_y przypiszemy do magazynu W_{q+1} przed następną maszyną M_{q+1} . Zadanie J_z przypiszemy do magazynu W_r przed maszyną powrotną z ciągu technologicznego M_r . Zaktualizować należy również stan maszyny z kontrolą jakości M_q , w celu zakodowania uwzględnienia negatywnego wyniku kontroli, czyli zresetować negatywny wynik kontroli jakości. Pozostałe współrzędne stanu się nie zmieniają. Zatem zbiór zadań nowej instancji problemu wynosi $|\mathcal{J}_{next}| = |\mathcal{J}_{now}| + 1$ oraz:

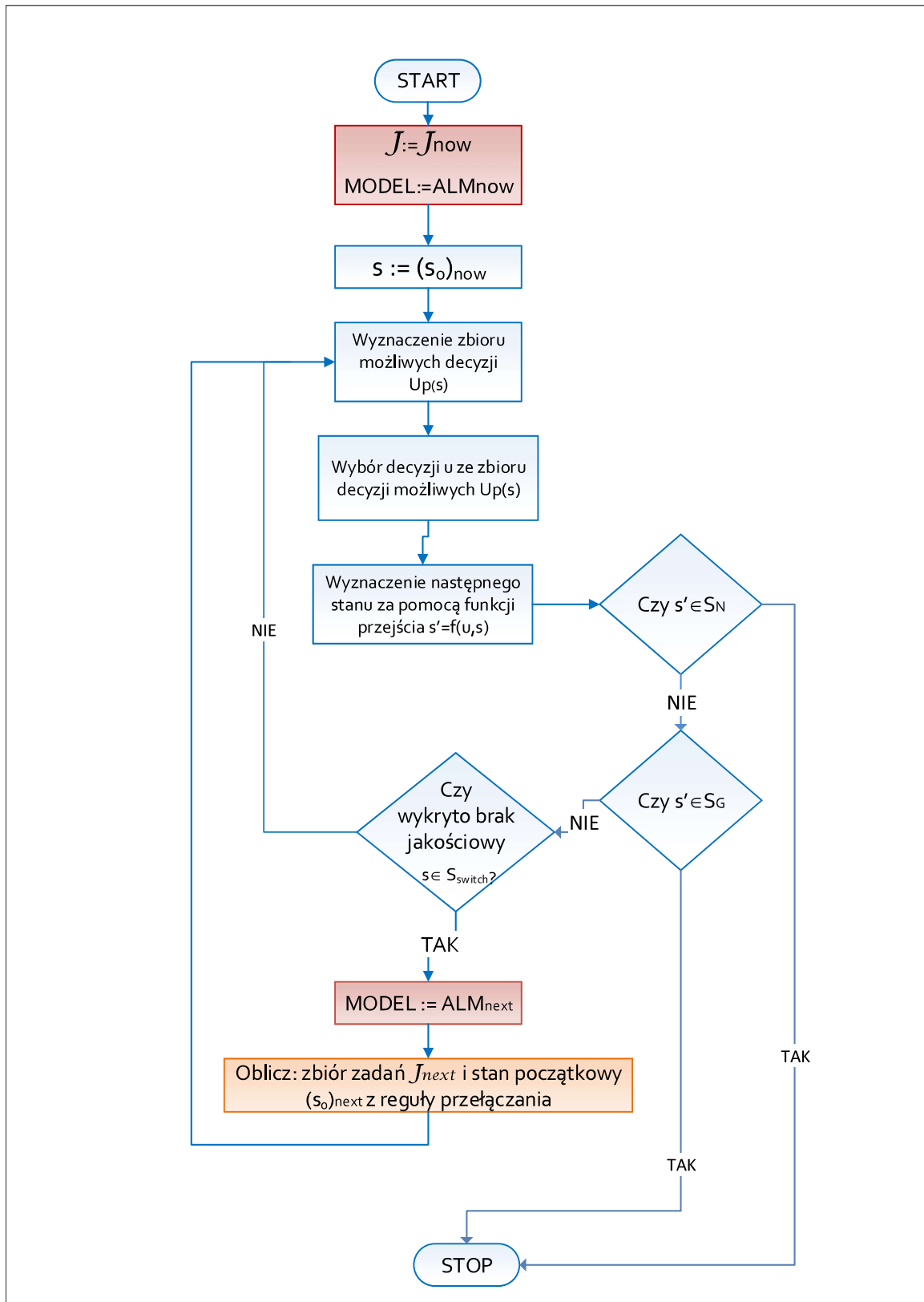
$$\begin{aligned}
 (x_0^i)_{next} &= (x_k^i)_{now} && \text{dla } i \neq q+1 \wedge i \neq r \wedge i = 1, 2, \dots, m \\
 (x_0^r)_{next} &= (x_k^r)_{now} \cup \{J_z\} \\
 (x_0^{q+1})_{next} &= (x_k^{q+1})_{now} \cup \{J_y\} \\
 (x_0^{m+i})_{next} &= (x_k^{m+i})_{now} && \text{dla } i = 1, 2, \dots, m, \wedge i \neq q \\
 (x_0^{m+q})_{next} &= (0, \infty, 0, 0) && \text{dla } i = q
 \end{aligned} \tag{6.17}$$

Następnie trajektoria procesu jest dalej konstruowana zaczynając od stanu $(s_0)_{next}$ za pomocą funkcji przejścia modelu ALM_{next} , aż do momentu wystąpienia kolejnego braku jakościowego. Wystąpienie kolejnego braku jakościowego (maszyna z kontrolą jakości jest wolna i wynik kontroli jest negatywny) spowoduje kolejne przełączenie według podanej funkcji przełączającej.

Algorytm przełączania dla problemu $FS-M_qM_r$ został przedstawiony na rysunku 6.1.

Określenie funkcji przełączającej kończy modelowanie rozważanego problemu $FS-M_qM_r$ metodą przełączania. Problem ten należy do klasy problemów przepływowych re-entrant. W kolejnym rozdziale rozważany jest problem, w którym dodatkowo występuje konieczność naprawy na dodatkowej maszynie M_d nie znajdującej się w ciągu technologicznym.

Natomiast, w celu rozwiązania problemu można zastosować jedną z opisanych w rozdziale 3 metod optymalizacji lokalnej bazującą na schemacie ALMM w połączeniu z metodą przełączania.



Rysunek 6.1: Algorytm przełączania dla problemu produkcji przepływowej z jedną maszyną kontroli jakości i bez maszyn naprawczych

7. Modelowanie problemu $FS-M_qM_dM_r$ za pomocą podejścia ALMM z zastosowaniem metody przełączania.

W rozdziale przedstawiono model algebraiczno-logiczny oraz metodę przełączania dla problemu $FS - M_qM_dM_r$ produkcji przepływowej z brakami jakościowymi wykrywanymi na jednym stanowisku kontroli jakości M_q , naprawianymi na dodatkowej maszynie M_d i ponownie obrabianymi począwszy od maszyny M_r . Stanowisko kontroli jakości znajduje się na maszynie w marszrucie technologicznej. Naprawa elementów wybrakowanych odbywa się na jednej dodatkowej maszynie naprawczej spoza marszruty technologicznej, następnie naprawione elementy powracają do ponownej obróbki na konkretną (tylko jedną) maszynę w marszrucie technologicznej, po czym realizowane są operacje w ciągu technologicznym od tej maszyny. Celem optymalizacji rozważanego problemu jest jak najszybsze wykonanie wszystkich zadań.

Idea zastosowania metody przełączania modeli algebraiczno-logicznych dla rozważanego problemu została przedstawiona we współautorstwie z E. Dudek-Dyduch oraz E. Kucharską i przedstawiona w pracy Grobler-Dębska et al. (2013), a następnie kontynuowana w Grobler-Dębska et al. (2014); Kucharska et al. (2017a,b). Jest to metoda, która została zaproponowana w celu umożliwienia odpowiedniego zamodelowania niedeterministycznego zdarzenia, jakim jest wykrycie braku jakościowego i konieczność naprawy tego braku. W rozdziale tym przedstawiona metoda została rozszerzona i doprecyzowana. Na podstawie analizy parametrów opisujących pracę maszyn, zmodyfikowano modele algebraiczno-logiczne. Mianowicie, aby móc wyróżnić w sposób formalny stany, w których następuje przełączenie między modelami, należy wyróżnić maszynę z kontrolą jakości w stanie systemu i zakodować wynik tej kontroli. W konsekwencji zmodyfikowane są modele algebraiczno-logiczne problemów pomocniczych i algorytm przełączania, a także zaproponowano przedstawienie w sposób formalny zbioru stanów przełączania i uszczegółowiono i doprecyzowano rodzaje przełączania.

Kolejno zostanie przedstawiona analiza problemu, podział problemu niedeterministycznego na deterministyczne problemy pomocnicze oraz opisane zostaną modele ALM dla problemów pomocniczych. W kolejnych punktach zostaną określone rodzaje przełączania, reguły przełączania oraz zaproponowany zostanie algorytm przełączania dla danego problemu.

7.1. Analiza problemu

W rozważanym problemie występują trzy charakterystyczne maszyny, dwie z nich w marszrucie technologicznej i jedna spoza marszruty. W marszrucie technologicznej jedną z wyróżnionych maszyn

jest maszyna M_q ze stanowiskiem kontroli jakości, zaś drugą M_r maszyna powrotna. Na wyróżnioną maszynę naprawczą M_d trafiają zadania do naprawy, wskazane w wyniku negatywnej kontroli jakości, a następnie przekazywana są do obróbki przez maszyny z marszruty technologicznej zaczynając od wyróżnionej maszyny powrotnej M_r . Należy zauważyć, że jeśli indeks maszyny powrotnej M_r jest mniejszy bądź równy indeksowi maszyny z kontrolą jakości $r \leq q$, to naprawa wadliwych zadań polega na wykonaniu dodatkowej operacji na maszynie M_d oraz ponownym wykonaniu operacji od wskazanej maszyny w marszrucie technologicznej M_r . Natomiast jeśli indeks maszyny powrotnej jest większy niż indeks maszyny z kontrolą jakości M_q , czyli $r > q$, to naprawa wadliwych elementów polega na wykonaniu dodatkowej operacji na maszynie M_d oraz i nie powtarza się ponownie operacji z marszruty technologicznej. Tak zdefiniowany problem nie należy do klasy problemów re-entrant, jest bardziej złożony.

Ponadto w odróżnieniu od klasycznych problemów re-entrant, gdzie całe zadanie (partia) wraca do ponownej obróbki, w rozważanym problemie również dopuszcza się częściowe wybrakowanie zadań. W rozważanym problemie w wyniku wykrycia w zadaniu (partii) nowych braków jakościowych zadanie dzielone jest na dwie subpartie. Prawidłowo wykonana subpartia kontynuuje obróbkę na kolejnej maszynie z marszruty technologicznej, zaś wybrakowana subpartia jest przekazywana do obróbki na maszynę naprawczą.

Z powodu podziału partii na dwie subpartie zmianie ulegnie zbiór zadań do wykonania: powstanie nowe zadanie składające się z elementów z defektami oraz zadanie, w którym wykryto elementy z brakami jakościowymi zostanie pomniejszone o te wybrakowane elementy.

W przypadku wykrycia braku jakościowego konieczne jest rozważanie dodatkowej maszyny naprawczej M_d , spoza ciągu technologicznego. Do obróbki na maszynie naprawczej mogą być przydzielone tylko subpartie wybrakowane. W sytuacji, w której wszystkie wybrakowane partie zostaną naprawione, nie ma konieczności rozważania maszyny naprawczej, oczywiście do momentu wykrycia kolejnych braków jakościowych. W związku z tym przyjęto, że zbiór maszyn w zależności od sytuacji będzie się zwiększał lub zmniejszał. Będzie, to miało szczególne znaczenie, w przypadku, gdy wykrycie braków jakościowych będzie występowało rzadko.

W związku z tym zmianie ulegać będzie zarówno zbiór zadań, jak i zbiór maszyn.

7.2. Podział problemu na problemy pomocnicze

W rozważanym problemie dany jest stały zbiór maszyn $\mathcal{M} = \{M_1, \dots, M_{m+1}\}$, gdzie pierwsze m maszyn ułożonych jest w ciąg technologiczny (M_1, \dots, M_m) zaś $m + 1$ maszyna jest to maszyna naprawcza spoza marszruty technologicznej $M_{md} = M_{m+1}$. Jak wcześniej w ciągu technologicznym wyróżniona jest maszyna powrotna $M_r \in \mathcal{M}$, na którą wracają naprawione subpartie oraz maszyna ze stanowiskiem kontroli jakości $M_q \in \mathcal{M}$.

Zmianie ulegać będzie rozważany zbiór maszyn \mathcal{M} , dlatego będą dwa problemy pomocnicze. Pierwszy problem będzie to problem produkcji przepływowej z wyróżnioną maszyną z kontrolą jakości M_q i maszyną powrotną M_r . Drugi problem będzie to problem produkcji przepływowej z wyróżnionymi maszynami: z kontrolą jakości, maszyną naprawczą spoza marszruty technologicznej oraz powrotną z

marszruty technologicznej, gdzie wracają do ponownej obróbki naprawione elementy. Problemy te będą reprezentowane przez dwa różne modele (modele o różnej strukturze).

Jak wcześniej w obu przypadkach zmianie ulegać będzie również zbiór zadań \mathcal{J} . Zatem będą zmieniane również instancje problemów pomocniczych. Celem jest jak najszybsze wykonanie wszystkich zadań.

7.3. Modele problemów pomocniczych

Pierwszy problem pomocniczy, czyli problem produkcji przepływowej z ograniczeniami czasowymi, maszyną ze stanowiskiem kontroli jakości, ale bez maszyny naprawczej, został wykorzystany również jako problem pomocniczy dla zaprezentowania metody przełączania problemu rozważanego rozdziale 6. Zatem jeden model algebraiczno-logiczny jest taki sam jak w podrozdziale 6.3. Oznaczmy ten model jako ALM_A . W związku z tym rozdziale tym przedstawiony zostanie tylko model drugiego problemu pomocniczego uwzględniającego dodatkową maszynę naprawczą. Oznaczmy ten model jako ALM_B .

Dla ułatwienia zapisu oznaczmy przez $U_A, S_A, (s_A)_0, f_A, (S_A)_N, (S_A)_G$ odpowiednio zbiór decyzji, zbiór stanów uogólnionych, uogólniony stan początkowy, funkcję przejścia, zbiór stanów niedopuszczalnych i zbiór stanów docelowych modelu ALM_A oraz niech $\mathcal{M}_A = \{M_1, \dots, M_m\}$ oznacza zbiór maszyn w produkcji przepływowej zaś \mathcal{J}_A zbiór zadań dla pierwszego problemu pomocniczego.

Istotnymi elementami modelu problemu pomocniczego ALM_A dla metody przełączania jest wyróżnienie struktury maszyny ze stanowiskiem kontroli M_q i dodanie do stanu właściwego tej maszyny dwóch parametrów związanych z wynikiem kontroli jakości tak jak w rozdziale 6. Stan maszyny z kontrolą jakości dany jest wzorem (6.3) i dla przypomnienia wynosi:

$$x^{m+q} = (\beta, \tau, QC, \hat{\beta})$$

gdzie:

$\beta \in \{0, 1, \dots, n\}$ – oznacza indeks zadania, które jest aktualnie obrabiane na danej maszynie (0 oznacza, że żadne zadanie nie zostało przydzielone do maszyny),

$\tau \in [0, \infty)$ – czas do ukończenia obróbki aktualnego zadania na maszynie (∞ oznacza, że maszyna nie pracuje),

$QC \in [0, 100]$ – oznacza procent wybrakowanych elementów w partii (0 oznacza pozytywny wynik kontroli jakości, czyli wszystkie elementy w partii są wykonane poprawnie),

$\hat{\beta} \in \{0, 1, \dots, n\}$ – oznacza indeks zadania, w którym wykryto braki jakościowe podczas ostatniej kontroli jakości.

Jak poprzednio istotne w modelu ALM_A jest, że w przypadku kiedy maszyna z kontrolą jakości jest wolna, a wynik kontroli jest negatywny, to stan następny nie jest obliczany za pomocą funkcji przejścia, tylko za pomocą funkcji przełączającej.

Również z punktu widzenia metody przełączania istotne było, aby w modelu pomocniczym ALM_A wyróżnić maszynę powrotną M_r z marszrutu technologicznej, gdzie $r \leq q$ oraz magazyn międzyoperacyjny przed tą maszyną W_r . Kodowanie stanu właściwego tej wyróżnionej maszyny i magazynu są takie same jak pozostałych maszyn i magazynów w marszrucie technologicznej.

7.3.1. Model algebraiczno-logiczny drugiego problemu pomocniczego z maszyną naprawczą

W drugim problemie pomocniczym występuje jedna maszyna ze stanowiskiem kontroli jakości M_q , jedna maszyna naprawcza (dodatkowa spoza marszruty technologicznej) M_d , do naprawy wadliwych elementów oraz naprawione elementy powracają do ponownej obróbki tylko na jedną konkretną maszynę M_r w marszrucie technologicznej. Model tego problemu jest odpowiednią modyfikacją modelu ALM_A o elementy związane z koniecznością naprawy wadliwych subpartii, czyli o maszynę naprawczą i o maszynę powrotną. Model ten oznaczono jako ALM_B .

Niech $U_B, S_B, (s_0)_B, f_B, (S_N)_B, (S_G)_B$ oznaczają odpowiednio zbiór decyzji, zbiór stanów uogólnionych, uogólniony stan początkowy, funkcję przejścia, zbiór stanów niedopuszczalnych i zbiór stanów docelowych modelu ALM_B oraz niech $\mathcal{M}_B = \{M_1, \dots, M_m, M_{m+1}\}$, gdzie $M_{m+1} = M_d$. Zatem $|\mathcal{M}_B| = |\mathcal{M}_A| + 1$.

Stan sytemu s_B Podobnie jak w modelu ALM_A , uogólniony stan procesu $s_B = (x_B, t)$ w danej chwili t jest opisany jako stan maszyn z marszruty, stan maszyny z kontrolą jakości, stan magazynu początkowego, stan magazynów międzyoperacyjnych oraz stan dodatkowego magazynu przed maszyną naprawczą W_d oraz stan maszyny naprawczej:

$$x_B = \underbrace{(x_B^1, \dots, x_B^m)}_{\text{stany magazynów}}, \underbrace{(x_B^{m+1}, \dots, x_B^{m+q}, \dots, x_B^{2m})}_{\text{stany maszyn z marszruty}}, \underbrace{(x_B^{2m+1}, x_B^{2m+2})}_{\substack{\text{stan magazynu} \\ \text{i maszyny naprawczej}}} \quad (7.1)$$

gdzie:

x_B^1 - stan magazynu początkowego,

x_B^i - stan i -tego magazynu międzyoperacyjnego W_i (pomiędzy $i - 1$ -wszą maszyną i i -tą maszyną) dla $i = 2, 3, \dots, m$,

x_B^{m+i} - stan i -tej maszyny dla $i = 1, 2, \dots, m$ i $i \neq q$,

x_B^{m+q} - stan q -tej maszyny ze stanowiskiem kontroli jakości dla $i = 1, 2, \dots, m$ i $i \neq q$,

x_B^{2m+1} - stan magazynu dodatkowego dla subpartii wybrakowanych W_d ,

x_B^{2m+2} - stan maszyny naprawczej M_d .

Kodowanie stanu właściwego magazynu początkowego, magazynów międzyoperacyjnych, maszyn z marszruty oraz wyróżnionej maszyny z kontrolą jakości, maszyny i magazynu powrotnego są takie jak w modelu opisanym w podrozdziale 6.3. Dodatkowo są rozważane w modelu współrzędne stanu właściwego magazynu dodatkowego dla subpartii wybrakowanych oraz maszyny naprawczej M_d . Kodowanie stanu maszyny naprawczej x^{2m+2} jest takie samo jak pozostałych maszyn w marszrucie technologicznej i dane jest wzorem (4.2).

Stan początkowy modelu $(s_0)_B = ((x_0)_B, (t_0))$ jest obliczany za pomocą funkcji przełączającej.

Zbiór stanów niedopuszczalnych $(S_N)_B$ ma taką samą strukturę jak zbiór stanów niedopuszczalnych $(S_N)_A$, należy tylko wziąć pod uwagę zbiór zadań \mathcal{J}_B , dodatkowy magazyn W_d i maszynę do naprawy braków M_d :

$$(S_N)_B = \{s_B = (x_B, t) : \exists j \in \mathcal{J}_B \ j \notin \mathcal{F}\mathcal{J}_B(s) \wedge \tilde{d}(j) < t\}.$$

Ponieważ stan uwzględniający naprawę elementów oznacza, że proces jest w trakcie (istnieją zadania niewykonane, w szczególnym przypadku jest to tylko jedno naprawiane zadanie), a stan ukończenia naprawy elementów należy do zbioru stanów przełączania dlatego zbiór stanów docelowych $(S_G)_B$ jest nieokreślony.

Decyzje u_B Analogicznie jak w modelu ALM_A przyjęto, iż decyzja polega na przydzieleniu zadań do konkretnych maszyn w tym samym czasie. Rozważając w problemie maszynę naprawczą M_d , należy rozszerzyć wektor u_B o dodatkową współrzędną u_B^{m+1} oraz zbiór zadań \mathcal{J}_B . Stąd wektor decyzji w modelu ALM_B ma postać:

$$u_B = (u_B^1, u_B^2, \dots, u_B^m, u_B^{m+1}) \text{ i } u_B^i \in \mathcal{J}_B \cup \{0\}, \text{ dla } i = 1, 2, \dots, m+1$$

Oczywiście nie wszystkie decyzje u_B mogą być podjęte w stanie (x_B, t) . Decyzje u_B muszą należeć do zbioru decyzji możliwych $(U_p)_B(x, t)$. Możliwa decyzja zależy od stanu poszczególnej maszyny. Stąd wartość poszczególnej współrzędnej decyzji u_B^i jest następująca:

- jeśli zadanie J_j jest przetwarzane przez maszyną M_i , wtedy jedyną możliwą decyzją jest kontynuacja obróbki $u_B^i = 0$,
- jeśli maszyna M_i jest wolna i nie istnieje żadne zadanie, które można by było przypisać maszynie tzn. magazyn przed maszyną jest pusty $x_B^i(s) = \emptyset$, to jedyną możliwą decyzją jest nieprzypisywanie żadnego zadania $u_B^i = 0$,
- jeśli maszyna M_i jest wolna i magazyn W_i nie jest pusty (tzn. istnieje zadanie J_j w magazynie) wtedy możliwą decyzją jest przypisanie zadania maszynie $u^i = j$ lub nie przypisywanie $u_B^i = 0$.

Funkcja przejścia f_B Moment t' jest wyznaczany dokładnie tak samo jak w modelu ALM_A , oczywiście biorąc pod uwagę również czas pracy maszyny naprawczej.

$$t' = t + \Delta t_B, \quad \text{gdzie: } \Delta t_B = \min_{M_i \in \mathcal{M}_B} t_{M_i} \quad (7.2)$$

gdzie wielkość oznaczana $t_{M_i \in \mathcal{M}_B}$ jest wyznaczana następująco:

- jeśli maszyna $M_i \in \mathcal{M}_B$ jest wolna i zadanie $J_j \in \mathcal{J}_B$ jest do niej przypisane tzn. $u^i(s) = j$, to czas zakończenia pracy przez maszynę wynosi p^{ij} ,
- jeśli maszyna $M_i \in \mathcal{M}_B$ jest wolna i żadne zadanie nie jest do niej przypisane, to czas ukończenia jest równy nieskończoność,
- jeśli maszyna M_i jest zajęta, czyli obrabia wcześniej przypisane zadanie $J_j \in \mathcal{J}_B$, to jedyną decyzją jest kontynuowanie obróbki tego zadania przez maszynę. Wtedy maszyna ukończy wykonywanie zadania z czasem równym τ^{i+m} (czas dokończenia zadania szczytany ze stanu maszyny).

W związku z tym Δt_B wynosi:

$$\Delta t_B = \min_{i=1, \dots, m+1} \begin{cases} p^{ij} & \text{dla } x^{i+m}(s) : \beta^{i+m} = j \wedge u^i(s) = j \\ \infty & \text{dla } x^{i+m}(s) = (0, \infty) \wedge u^i(s) = 0 \\ \tau^{i+m} & \text{dla } x^{i+m}(s) : \beta^{i+m} = j \wedge u^i(s) = 0 \end{cases} \quad (7.3)$$

Współrzędne stanów magazynów międzyoperacyjnych x_B^i dla: $i = 1, \dots, m$ zmieniają się za pomocą funkcji przejścia dokładnie w taki sam sposób jak w modelu ALM_A .

W funkcji przejścia współrzędna x_B^{2m+1} oznaczająca stan magazynu dodatkowego i wynosi:

$$(x_B^{2m+1}(s))' = x_B^{2m+1}(s) \setminus \{j : u_B^{m+1}(s) = j\} \quad (7.4)$$

Wartość współrzędnych oznaczających stan maszyn w marszrucie technologicznej $(x_B^{i+m})'$ dla: $i = 1, \dots, m$ zależą od podjętej decyzji i zmieniają się dokładnie w taki sam sposób jak w modelu ALM_A .

Natomiast stan maszyny naprawczej M_{m+1} (współrzędna x_B^{2m+2}) zmienia się w następujący sposób:

1. jeśli maszyna M_{m+1} jest wolna i została podjęta decyzja $u_B^{m+1}(s) = j$ o przypisaniu zadania J_j do maszyny (tzn. magazyn dodatkowy nie jest pusty), to:

$$\begin{aligned} \beta'_B &= \begin{cases} 0 & \text{dla } \tau_B^{2m+2} = \Delta t_B \\ j & \text{dla } \tau_B^{2m+2} > \Delta t_B \end{cases} \\ \tau'_B &= \begin{cases} \infty & \text{dla } \tau_B^{2m+2} = \Delta t_B \\ p^{(2m+2)j} - \Delta t_B & \text{dla } \tau_B^{2m+2} > \Delta t_B \end{cases} \end{aligned} \quad (7.5)$$

2. jeśli maszyna M_{m+1} jest wolna w danym stanie i podjęta decyzja $u_B^{m+1}(s) = 0$ jest o nie przypisywaniu żadnego zadania do maszyny, wtedy:

$$\beta'_B = 0$$

,

$$\tau'_B = \infty$$

3. jeśli maszyna M_{m+1} jest zajęta w danym stanie i podjęta decyzja $u_B^{m+1}(s) = 0$ jest o kontynuowaniu poprzednio przypisanego zadania J_j , to:

$$\begin{aligned} \beta'_B &= \begin{cases} 0 & \text{dla } \tau_B^{2m+2} = \Delta t_B \\ j & \text{dla } \tau_B^{2m+2} > \Delta t_B \end{cases} \\ \tau'_B &= \begin{cases} \infty & \text{dla } \tau_B^{2m+2} = \Delta t_B \\ \tau_B^{2m+2} - \Delta t_B & \text{dla } \tau_B^{2m+2} > \Delta t_B \end{cases} \end{aligned} \quad (7.6)$$

7.4. Funkcja przełączająca

W rozdziale tym zostały przeanalizowane przypadki, w których powinna nastąpić zmiana modeli. Wyróżniono rodzaje przełączeń, reguły przełączania oraz wyróżniono zbiór stanów, w którym następują przełączania między modelami dla problemów pomocniczych. Pod koniec rozdziału zaproponowano funkcję przełączającą wraz z algorytmem.

7.4.1. Rodzaje i reguły przełączania \mathcal{R}_{switch}

Tak jak już wspomiano w rozważanym problemie z brakami jakościowymi i ich naprawą na dodatkowej maszynie, braki jakościowe wykrywane są na podstawie kontroli jakości na jednej wybranej maszynie M_q i subpartie z wybrakowanymi elementami są poddawane w pierwszej kolejności naprawie na dodatkowej maszynie M_{m+1} spoza marszruty i następnie poddawane ponownej obróbce od wybranej maszyny z marszruty technologicznej M_r , gdzie $r \leq q$.

Ponieważ dla rozważanego problemu tworzone są dwa problem pomocnicze ALM_A i ALM_B . Pierwszy problem pomocniczy produkcji przepływowej z maszyną ze stanowiskiem kontroli jakości ALM_A oraz drugi problem pomocniczy produkcji przepływowej z maszyną z kontrolą jakości oraz maszyną naprawczą spoza marszruty technologicznej ALM_B . Należy rozważyć przełączenie pomiędzy tymi modelami. Modele te mają różną strukturę. Należy również rozważyć czy nie jest konieczne zdefiniowanie także reguł przełączania między modelami o tej samej strukturze. Mówiąc o modelu o tej samej strukturze należy rozumieć modele różniące się między sobą tylko wartością stanu początkowego.

Rozważmy osobno dwa następujące przypadki. W pierwszym przypadku, w którym rozważany w aktualnym stanie jest model ALM_A mogą nastąpić następujące sytuacje:

- A1 - Wykryto po raz pierwszy zadanie z brakami jakościowymi. Co oznacza, że do tego momentu nie było konieczności rozważania maszyny naprawczej w modelu, więc do tej chwili posługiwaliśmy się modelem ALM_A . W takiej sytuacji następuje przełączenie z modelu ALM_A na model ALM_B , w którym rozważana jest dodatkowa maszyna naprawcza i magazyn przed tą maszyną. Ponadto nastąpił podział zadania, w którym wykryto elementy z brakami jakościowymi na dwa zadania: poprawnie wykonane i zadanie z brakami do naprawy.
- A2 - Wykryto kolejne zadanie z brakami jakościowymi, ale wszystkie wszystkie wcześniej wykryte braki jakościowe zostały już naprawione i są poddawane ponownej obróbce zgodnie z marszrutą. Co oznacza, że w danym stanie nie była rozważana maszyna naprawczej w modelu, więc do tej chwili posługiwaliśmy się modelem ALM_A . W takiej sytuacji następuje przełączenie z modelu ALM_A na model ALM_B , w którym rozważana jest dodatkowa maszyna i magazyn przed tą maszyną oraz pojawienie się dodatkowej partii.

Zauważmy, że sytuacje A1 i A2 z punktu widzenia metody przełączania są równoważne, ponieważ nie była rozważana maszyna naprawcza, a dopiero teraz jest konieczne jej rozważenie.

W drugim przypadku, w którym rozważany w aktualnym stanie jest model ALM_B mogą nastąpić następujące sytuacje:

- B1 - Wykryto zadanie z brakami jakościowymi, w przypadku kiedy wcześniej wystąpił już brak jakościowy i nie został do tej pory naprawiony. To oznacza, że do tego momentu konieczne było uwzględnienie maszyny naprawczej w modelu, więc do tej chwili posługiwaliśmy się modelem ALM_B . W takiej sytuacji mamy do czynienia ze zmianą w zbiorze zadań (nastąpił podział zadania, w którym wykryto elementy z brakami jakościowymi na dwa zadania: poprawnie wykonane i zadanie z brakami do naprawy). W takiej sytuacji następuje przełączenie z modelu ALM_B na

model o takiej samej strukturze, w którym należy zmodyfikować parametry. Oznaczmy go poprzez $ALM_{B'}$.

- B2 - Wykryto zadanie z brakami jakościowymi, w przypadku kiedy wcześniej wystąpił brak jakościowy i został on właśnie naprawiony. W takiej sytuacji nowo wykryte zadanie z brakami jakościowymi dzielone jest na dwa zadania: poprawnie wykonane i zadanie do naprawy oraz zadanie, które zostało właśnie naprawione należy umieścić do wykonania w marszrucie technologicznej zaczynając od wyróżnionej maszyny powrotnej. To oznacza, że do tego momentu konieczne było uwzględnienie maszyny naprawczej w modelu (model ALM_B). W takiej sytuacji następuje przełączenie z modelu ALM_B na model $ALM_{B'}$, w którym należy zmodyfikować parametry.
- B3 - Zakończono naprawę wybrakowanego zadania, ale są jeszcze zadania do naprawy w magazynie przed maszyną naprawczą. To oznacza, że do tego momentu konieczne było uwzględnienie maszyny naprawczej (stosowano model ALM_B), zadanie właśnie naprawione należy umieścić do ponownej obróbki w marszrucie technologicznej od wskazanej maszyny M_r . W takiej sytuacji następuje przełączenie z modelu ALM_B na model $ALM_{B'}$, w którym należy zmodyfikować parametry.
- B4 - Zakończono naprawę wybrakowanej partii i nie ma innych partii do naprawy w magazynie przed maszyną naprawczą oraz nie wykryto żadnego nowego braku (model ALM_B). W takiej sytuacji następuje przełączenie z modelu ALM_B na model o strukturze ALM_A o zmienionych parametrach, czyli ALM'_A .

Zatem wyróżnione zostaną trzy rodzaje przełączeń:

- I Przełączenie z modelu ALM_A na model ALM_B w przypadku wykrycia po raz pierwszy przez kontrolę jakości zadania z brakami jakościowymi.
- II Przełączenie z modelu ALM_B na model o tej samej strukturze tylko ze zmienionymi parametrami, czyli o strukturze ALM'_B w przypadkach, gdy kontrola jakości wykryła kolejne zadanie z brakami jakościowymi, a poprzednio wykryte zadanie nie zostało jeszcze naprawione lub właśnie zostało naprawione albo przy braku wykrycia kolejnego zadania z brakami jakościowymi, któreś poprzednio wykryte zadanie z brakami zostało naprawione, a inne wcześniej wykryte czekają na naprawę.
- III Przełączenie modelu o strukturze ALM_B na model o strukturze ALM_A ze zmienionymi parametrami, czyli ALM'_A , w przypadku gdy ostatnie z zadań naprawianych zostało właśnie naprawione.

7.4.2. Zbiór stanów przełączania

Przełączenie modeli algebraiczno-logicznych następuje wtedy, gdy maszyna ze stanowiskiem kontroli jakości skończy pracę i wynik kontroli jakości jest negatywny lub też maszyna naprawcza skończy pracę i należy umieścić naprawione zadanie do ponownej obróbki na marszrutę technologiczną. Zatem

do zbioru stanów przełączania należą te stany, w których maszyna ze stanowiskiem kontroli jakości jest wolna i wynik kontroli jakości był negatywny lub maszyna naprawcza jest wolna:

$$\mathcal{S}_{switch} = \{s = (x, t) : (\beta^{q+m} = 0, \tau^{q+m} = \infty, QC^{q+m} \neq 0, \hat{\beta}^{q+m} \neq 0) \vee (\beta^{2m+2} = 0, \tau^{2m+2} = \infty)\} \quad (7.7)$$

7.4.3. Algorytm przełączania

Analogicznie jak w poprzednim problemie, aby zapisać formalnie algorytm przełączania wprowadźmy oznaczenia. Niech J_y oznacza zadanie, w którym wykryto brakami jakościowe na maszynie z kontrolą jakości. Zadanie to zostanie podzielone na dwa zadania (dwie subpartie):

- zadanie z elementami poprawnie wykonywanymi, które może być dalej obrabiane przez kolejne maszyny z marszruty technologicznej, oznaczmy je również jako J_y
- zadanie z elementami do naprawy, które zostanie oznaczone jako J_z .

Dla zadania J_y , pomniejszonego o elementy wybrakowane i kontynuującego obróbkę zgodnie z marszrutą technologiczną zostaną przeliczone czasy obróbki zadania na kolejnych maszynach, czyli p_{iy} dla każdego $i > q$ na podstawie ilości elementów i zdolności produkcyjnym maszyn. Zadanie wadliwe J_z jest poddawane naprawie na maszynie M_{m+1} oraz później ponownej obróbce na maszynach z marszruty technologicznej od maszyny M_r . Zatem dla tego zadania zostaną obliczone wszystkie czasy obróbki p_{m+1z} oraz p_{iz} dla każdego $i > r$ na podstawie ilości elementów w zadaniu i zdolności produkcyjnym maszyn. Ponadto oznaczmy przez J_r zadanie naprawione przez maszynę naprawczą, które należy poddać ponownej obróbce na marszrucie technologicznej od maszyny M_r .

W przypadku wykrycia nowego braku jakościowego, należy również pamiętać o aktualizacji stanu maszyny z kontrolą jakości M_q . Aktualizacja stanu ma na celu zresetowanie negatywnego wyniku kontroli jakości i zakodowania w ten sposób, że wynik ten został uwzględniony.

Zaproponowany algorytm został skonstruowany na podstawie reguł przełączania podanych w podrozdziale 7.4.1.

1. Na podstawie reguł [A1 i A2] - wykrycie nowego zadania J_y z brakami jakościowymi, w przypadku kiedy nie występują inne zadania z brakami.

To oznacza, że do tego momentu nie było konieczności rozważania maszyny naprawczej, więc do tej chwili posługiwaliśmy się modelem ALM_A . W takiej sytuacji następuje przełączenie z modelu ALM_A na model ALM_B , w którym rozważana jest dodatkowa maszyna i magazyn przed tą maszyną. Należy również rozważyć podział zadania J_y na zadanie z poprawnie wykonanymi elementami również J_y i na zadanie z wybrakowanymi elementami J_z . Zatem stan początkowy $(s_0)_B$ modelu ALM_B jest obliczany na podstawie aktualnego stanu $(s_k)_A$ modelu ALM_A i wyniku kontroli jakości: $(s_k)_A = (x_k, t_k) \longrightarrow (s_0)_B = (x_0, t_0)$.

Zauważmy, że w tej sytuacji maszyna z kontrolą jakości M_q jest wolna w stanie k i zwrócony jest negatywny wynik kontroli jakości $(x_k^{m+q})_A = (0, \infty, QC > 0, \hat{\beta} \neq 0)$, w związku z tym należy zaktualizować jej stan. Rozważane zadania nowe J_y i J_z powinny zostać przypisane do

odpowiedniego magazynu międzyoperacyjnego. Zadanie J_y przypisujemy do zbioru zadań, które są w magazynie przed następną maszyną M_{q+1} . Zadanie J_z przypisujemy do magazynu przed maszyną naprawczą M_{m+1} . Pozostałe współrzędne stanu się nie zmieniają:

$$\begin{aligned}
(x_0^i)_B &= (x_k^i)_A && \text{dla } i \neq q+1 \wedge i = 1, 2, \dots, m \\
(x_0^{q+1})_B &= (x_k^{q+1})_A \cup \{J_y\} \\
(x_0^{m+i})_B &= (x_k^{m+i})_A && \text{dla } i = 1, 2, \dots, m \wedge i \neq q \\
(x_0^{m+q})_B &= (0, \infty, 0, 0) && \text{dla } i = q \\
(x_0^{2m+1})_B &= \{J_z\} \\
(x_0^{2m+2})_B &= (0, \infty).
\end{aligned} \tag{7.8}$$

2. Na podstawie reguły [B1] - wykrycie nowego zadania z brakami jakościowymi J_y , w przypadku kiedy wcześniej wystąpił brak jakościowy i nie został do tej pory naprawiony.

To oznacza, że do tego momentu konieczne było uwzględnienie maszyny naprawczej, więc do tej chwili posługiwaliśmy się modelem ALM_B . W takiej sytuacji następuje przełączenie z modelu ALM_B na model $ALM_{B'}$, w którym należy zmodyfikować parametry. Mianowicie należy uwzględnić podział zadania J_y na dwa zadania J_y i J_z oraz odpowiednio zmodyfikować zawartość magazynów tzn zadania J_y i J_z powinny zostać przypisane do odpowiedniego magazynu międzyoperacyjnego. Zadanie J_y przypisujemy do zbioru zadań, które są w magazynie przed następną maszyną M_{q+1} . Zadanie J_z przypisujemy do magazynu przed maszyną naprawczą M_{m+1} . Ponadto maszyna z kontrolą jakości M_q jest wolna i zwrócony jest negatywny wynik kontroli jakości $(x_k^{m+q})_B = (0, \infty, QC > 0, \hat{\beta} \neq 0)$, więc należy zaktualizować jej stan. Stąd stan początkowy $(s_0)_{B'}$ modelu $ALM_{B'}$ jest obliczany na podstawie aktualnego stanu $(s_k)_B$ modelu ALM_B i rezultatu kontroli jakości. Współrzędne stanu początkowego $(s_0)_{B'}$ są następujące:

$$\begin{aligned}
(x_0^i)_{B'} &= (x_k^i)_B && \text{dla } i \neq q+1 \wedge i = 1, 2, \dots, m \\
(x_0^{q+1})_{B'} &= (x_k^{q+1})_B \cup \{J_y\} \\
(x_0^{m+i})_{B'} &= (x_k^{m+i})_B && \text{dla } i = 1, 2, \dots, m \wedge i \neq q \\
(x_0^{m+q})_{B'} &= (0, \infty, 0, 0) && \text{dla } i = q \\
(x_0^{2m+1})_{B'} &= (x_k^{2m+1})_B \cup \{J_z\} \\
(x_0^{2m+2})_{B'} &= (x_k^{2m+2})_B.
\end{aligned} \tag{7.9}$$

3. Na podstawie reguły [B2] - wykrycie nowego zadania z brakami jakościowymi, w przypadku kiedy wcześniej wystąpiło już zadanie wybrakowane i zostało właśnie naprawione.

To oznacza, że do tego momentu konieczne było uwzględnienie maszyny naprawczej (stosowany był model ALM_B). W takiej sytuacji następuje przełączenie z modelu ALM_B na model $ALM_{B'}$, w którym należy zmodyfikować parametry. Mianowicie należy uwzględnić podział zadania J_y na dwa zadania J_y i J_z oraz odpowiednio zmodyfikować zawartość magazynu przed kolejną maszyną w ciągu technologicznym oraz zawartość magazynu przed maszyną naprawczą. Ponadto należy umieścić naprawione zadanie J_r w magazynie międzyoperacyjnym przed maszyną powrotną M_r , od której powtarzane jest przetwarzanie w ciągu technologicznym oraz zaktualizować stan

maszyny z kontrolą jakości M_q . W związku z tym stan początkowy $(s_0)_{B'}$ modelu $ALM_{B'}$ jest następujący:

$$\begin{aligned}
(x_0^i)_{B'} &= (x_k^i)_B && \text{dla } i \neq q+1, r \wedge i = 1, 2, \dots, m \\
(x_0^r)_{B'} &= (x_k^r)_B \cup \{J_r\} \\
(x_0^{q+1})_{B'} &= (x_k^{q+1})_B \cup \{J_y\} \\
(x_0^{m+i})_{B'} &= (x_k^{m+i})_B && \text{dla } i = 1, 2, \dots, m \wedge i \neq q \\
(x_0^{m+q})_B &= (0, \infty, 0, 0) && \text{dla } i = q \\
(x_0^{2m+1})_{B'} &= (x_k^{2m+1})_B \cup \{J_z\} \\
(x_0^{2m+2})_{B'} &= (x_k^{2m+2})_B.
\end{aligned} \tag{7.10}$$

4. Na podstawie reguły [B3] - zakończenie naprawy wybrakowanego zadania, ale są jeszcze zadania do naprawy w magazynie przed maszyną naprawczą.

To oznacza, że do tego momentu konieczne było uwzględnienie maszyny naprawczej (stosowany był model ALM_B). W takiej sytuacji następuje przełączenie z modelu ALM_B na model $ALM_{B'}$, w którym należy zmodyfikować parametry. Mianowicie należy umieścić naprawione zadanie J_r w magazynie międzyoperacyjnym przed maszyną M_r od której powtarzane jest przetwarzanie w ciągu technologicznym. W tym przypadku, przełączenie nie wynika z negatywnego wyniku kontroli jakości maszyny M_q , więc stan jej nie jest zmieniany. Zatem stan początkowy $(s_0)_{B'}$ modelu $ALM_{B'}$ jest następujący:

$$\begin{aligned}
(x_0^i)_{B'} &= (x_k^i)_B, && \text{dla } i \neq r \wedge i = 1, 2, \dots, m \\
(x_0^r)_{B'} &= (x_k^r)_B \cup \{J_r\} \\
(x_0^{m+i})_{B'} &= (x_k^{m+i})_B && \text{dla } i = 1, 2, \dots, m \\
(x_0^{2m+1})_{B'} &= (x_k^{2m+1})_B \\
(x_0^{2|M|+2})_{B'} &= (x_k^{2m+2})_B.
\end{aligned} \tag{7.11}$$

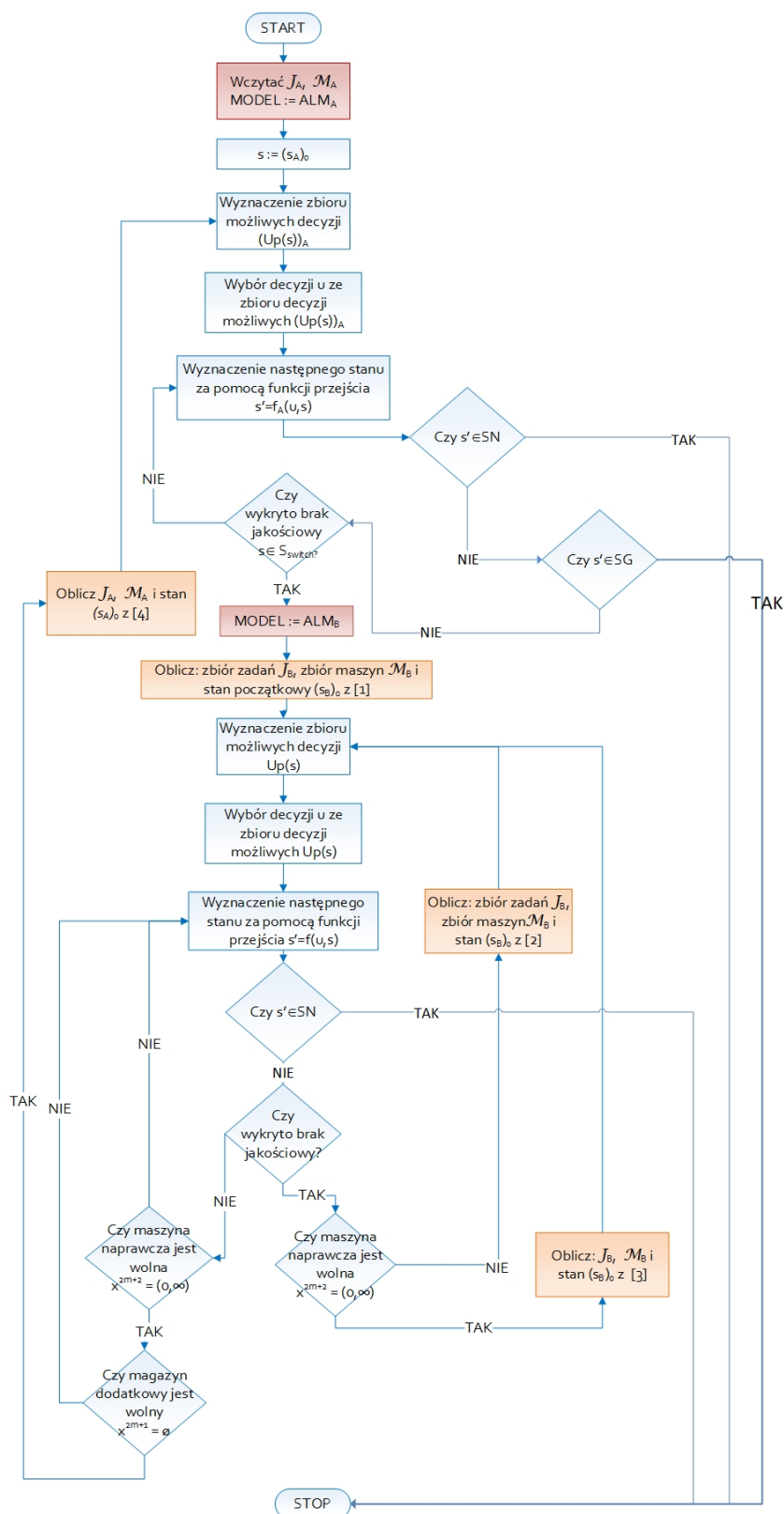
5. Na podstawie reguły [B4] - zakończenie naprawy wybrakowanego zadania i nie ma innych zadań do naprawy w magazynie przed maszyną naprawczą oraz nie wykryto żadnego nowego braku.

W takiej sytuacji następuje przełączenie z modelu ALM_B na model ALM'_A . Zatem należy umieścić naprawione zadanie J_r w magazynie międzyoperacyjnym przed maszyną, od której powtarzane jest przetwarzanie w ciągu technologicznym. Wtedy stan początkowy $(s_0)'_A$ modelu ALM'_A jest następujący:

$$\begin{aligned}
(x_0^i)'_A &= (x_k^i)_B, && \text{dla } i \neq r \wedge i = 1, 2, \dots, m \\
(x_0^r)'_A &= (x_k^r)_B \cup \{J_z\} \\
(x_0^{m+i})'_A &= (x_k^{m+i})_B && \text{dla } i = 1, 2, \dots, m.
\end{aligned} \tag{7.12}$$

Powyższy algorytm przełączania został przedstawiony na rysunku 7.1.

Określenie funkcji przełączającej kończy modelowanie rozważanego problemu $FS-M_qM_dM_r$ metodą przełączania. Ten problem nie należy do klasy klasycznych problemów re-entrant ze względu na



Rysunek 7.1: Algorytm przełączania dla problemu produkcji przepływowej z naprawą braków na jednej maszynie naprawczej.

występowanie dodatkowej maszyny naprawczej. Natomiast, w celu rozwiązania problemu można zastosować jedną z opisanych w rozdziale 3 metodę optymalizacji lokalnej bazującą na schemacie ALMM w połączeniu z metodą przełączania.

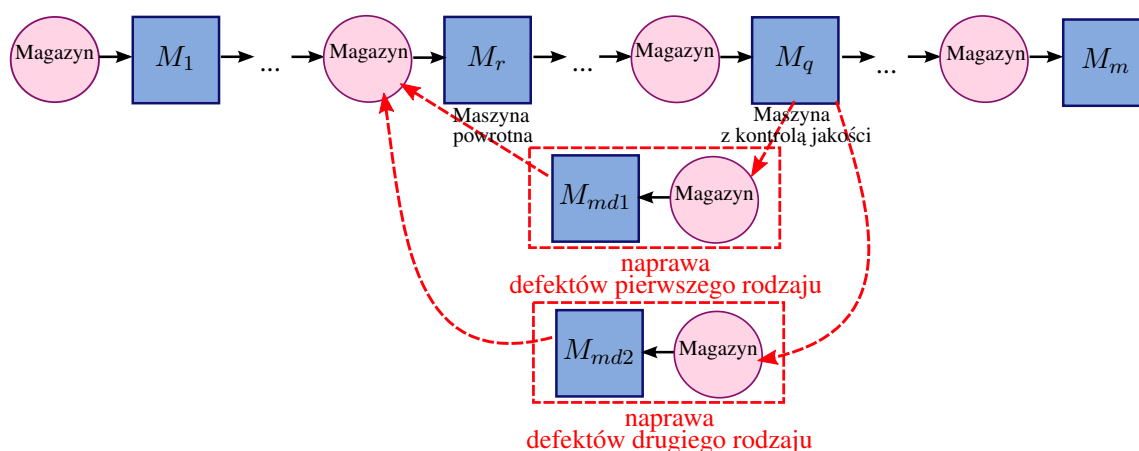
W kolejnym rozdziale przedstawiono schemat projektowania metody przełączania dla problemów produkcji przepływowej w bardziej skomplikowanych systemach 5.2.3-5.2.5.

8. Metoda przełączania modeli algebraiczno-logicznych dla problemów z brakami jakościowymi $FS-M_qmM_dM_r$

W rozdziale tym zostaną omówione etapy modelowania i konstruowania metody przełączania dla problemów produkcji przepływowej z brakami jakościowymi, w bardziej złożonych systemach kontroli jakości i naprawy. Rozważane są problemy, które opisano kolejno w rozdziałach 5.2.3-5.2.5.

8.1. Modelowanie problemu $FS-M_qmM_dM_r$

Rozważmy problem $FS-M_qmM_dM_r$ produkcji przepływowej z brakami jakościowymi, wykrywanymi na jednym stanowisku kontroli jakości M_q , naprawianymi na różnych maszynach dodatkowych ($|\mathcal{M}_D| > 1$ i $\mathcal{M}_D = \{M_{m+1}, \dots, M_{md}\}$) w zależności od rodzaju wymaganej naprawy oraz ponownie obrabianymi począwszy od jednej wyróżnionej maszyny powrotnej M_r z marszruty technologicznej. Wybór maszyny do naprawy jest uzależniony od stopnia uszkodzenia wadliwych elementów i jest wynikiem kontroli jakości. Naprawione elementy powracają do ponownej obróbki przez maszyny z marszruty technologicznej począwszy od konkretnej tylko jednej wybranej maszyny M_r niezależnie od rodzaju naprawy. Opis problemu został przedstawiony w 5.2.3, na rysunku przedstawiono przykład linii produkcyjnej z dwoma stanowiskami do dwóch różnych typów napraw.



Rysunek 8.1: Przykład linii produkcyjnej z maszyną z kontrolą jakości, maszyną powrotną oraz dwoma różnymi maszynami naprawczymi.

8.1.1. Analiza problemu

W rozważanym problemie, w wyniku kontroli jakości wykrywane są różnego typu braki jakościowe. Typ braku implikuje jednoznacznie wybór maszyny naprawczej. W problemie rozważane jest tyle maszyn naprawczych ile występuje typów (rodzajów) braków jakościowych.

W związku z tym występują trzy charakterystyczne typy maszyn, analogicznie jak w problemie $FS-M_qM_dM_r$ rozważanym w rozdziale 7 z tym, że maszyny naprawczych jest więcej niż jedna i są one spoza marszruty $|\mathcal{M}_D| > 1$. W marszrucie technologicznej jedną z wyróżnionych maszyn jest maszyna M_q ze stanowiskiem kontroli jakości, zaś drugą M_r maszyna powrotna.

Postępowanie podczas wykrycia nowego braku jest następujące: w wyniku wykrycia w zadaniu (partii) nowych braków jakościowych zadanie dzielone jest na subpartie. Liczba subpartii zależy od liczby typów wykrytych braków jakościowych w danym stanie. Prawidłowo wykonana subpartia kontynuuje obróbkę na kolejnej maszynie z marszruty technologicznej, zaś wybrakowana subpartia (partia składająca się z elementów z defektami) lub subpartie są przekazywane do obróbki na konkretne, dedykowane dla danego typu braku, maszyny naprawcze, wskazane w wyniku kontroli jakości. Zadanie naprawione dla danego typu braku, maszyny naprawcze, wskazane w wyniku kontroli jakości. Zadanie naprawione wraca do obróbki na konkretną, daną maszyną powrotną z marszruty produkcyjnej, niezależnie od tego, która maszyna wykonywała naprawę.

Z powodu podziału partii na subpartie zmianie ulega zbiór zadań do wykonania: powstaną nowe zadanie lub zadania składające się z elementów z defektami odpowiedniego typu oraz zadanie, w którym wykryto elementy z brakami jakościowymi. W szczególnym przypadku, gdy zostanie wykryty brak jakościowy tylko jednego typu, zamiast jednego zadania, w którym wykryto ten brak powstaną dwa zadania: nowe zadanie składające się z elementów z defektami oraz zadanie, w którym wykryto elementy z brakami jakościowymi pomniejszone o te wybrakowane elementy.

W przypadku wykrycia konkretnego typu braku jakościowego konieczne jest rozważanie dodatkowej maszyny naprawczej, spoza ciągu technologicznego. Do obróbki na maszynie naprawczej mogą być przydzielone tylko subpartie wybrakowane o danym typie braku. W sytuacji, w której nie występują braki jakościowe danego typu, nie ma konieczności rozważania w problemie maszyny do naprawy danego typu braku. W sytuacji, w której wszystkie wybrakowane partie zostaną naprawione, nie ma konieczności rozważania maszyny naprawczej, oczywiście do momentu wykrycia kolejnych braków.

W związku z tym należy zaprojektować podczas modelowania kodowanie stanu maszyny z kontrolą jakości, w taki sposób by uwzględnić różne typy braków jakościowych oraz wynikające z tego kodowania stany dodatkowych maszyn naprawczych.

Należy zauważyć, że jeśli indeks maszyny powrotnej M_r jest mniejszy bądź równy indeksowi maszyny z kontrolą jakości $r \leq q$, to naprawa wadliwych zadań konkretnego typu polega na wykonaniu dodatkowej operacji na maszynie dodatkowej M_{d_γ} danego typu γ oraz ponownym wykonaniu operacji od wskazanej maszyny w marszrucie technologicznej M_r . Natomiast jeśli indeks maszyny powrotnej jest większy niż indeks maszyny z kontrolą jakości M_q , czyli $r > q$, to naprawa wadliwych elementów polega danego typu γ na wykonaniu dodatkowej operacji na maszynie M_{d_γ} danego typu γ i nie powtarza się ponownie operacji z marszruty technologicznej. Tak zdefiniowany problem nie należy do klasy problemów re-entrant, jest to problem bardziej złożony.

8.1.2. Podział problemu na problemy pomocnicze

Proponuje się następujący sposób podziału problemu na problemy pomocnicze. Wyróżnione zostają dwa problemy pomocnicze analogicznie jak w problemie $FS-M_qM_dM_r$ opisanym w rozdziale 7.

Pierwszy problem pomocniczy to podstawowy problem produkcji przepływowej z ograniczeniami czasowymi, z wyróżnioną maszyną ze stanowiskiem kontroli jakości, ale bez żadnej maszyny naprawczej M_d . Problem ten jest to problemem wejściowym, który determinuje pewne założenia oraz ograniczenia. Drugi problem pomocniczy to problem produkcji przepływowej z wyróżnionymi maszynami specjalnymi: z kontrolą jakości, maszynami naprawczymi różnego typu spoza marszruty technologicznej oraz powrotną z marszruty technologicznej, gdzie wracają do ponownej obróbki naprawione elementy. Liczba dodatkowych maszyn naprawczych zależy od liczby typów braków jakościowych jakie wystąpiły w trakcie kontroli jakości. Dwa występujące tu szczególne przypadki to po pierwsze wystąpienie tylko jednego typu braku jakościowego, wtedy problem redukuje się do problemu $FS-M_qM_dM_r$ gdyż potrzebna jest tylko jedna maszyna naprawcza, ale rodzaj maszyny zależy od typu braku jakościowego. Drugi przypadek to wystąpienie wszystkich typów braków jakościowych w jednym stanie. Wtedy w problemie pomocniczym wystąpią wszystkie dodatkowe maszyny naprawcze i odpowiednio magazyny przez nimi. Tak więc w drugim problemie pomocniczym będzie się zmieniała liczba dodatkowych maszyn naprawczych.

W takiej sytuacji pierwszy i drugi problem pomocniczy będą reprezentowane przez dwa różne modele ALM_A i ALM_B (modele o różnej strukturze).

Ponadto w przypadkach wystąpienia ponownie braku jakościowego danego typu, podczas gdy poprzedni nie został jeszcze naprawiony, zmianie ulega również zbiór zadań \mathcal{J} . Zatem będą zmieniane instancje problemów pomocniczych.

8.1.3. Modele problemów pomocniczych

Niezależnie od przypadku, pierwszy problem pomocniczy jest to problem produkcji przepływowej z maszyną kontroli jakości, ale bez żadnej maszyny naprawczej. Został on wykorzystany również jako problem pomocniczy dla zaprezentowania metody przełączania problemów $FS-M_qM_r$ i $FS-M_qM_dM_r$ rozważanego odpowiednio w rozdziałach 6 i 7. Natomiast ze względu na występowanie różnych typów braków jakościowych kontrola jakości wygląda inaczej, więc inaczej będzie wyglądało kodowanie stanu właściwego maszyny z kontrolą jakości M_q . Zatem pierwszy model algebraiczno-logiczny jest taki sam jak w podrozdziale 6.3, z tym, że zmienione jest kodowanie stanu właściwego maszyny z kontrolą jakości, tak aby uwzględnić różne typy braków jakościowych.

Podobnie jak w poprzednio rozważanych problemach, oznaczymy przez $U_A, S_A, (s_A)_0, f_A, (S_A)_N, (S_A)_G$ odpowiednio zbiór decyzji, zbiór stanów uogólnionych, uogólniony stan początkowy, funkcję przejścia, zbiór stanów niedopuszczalnych i zbiór stanów docelowych modelu ALM_A oraz niech $\mathcal{M}_A = \{M_1, \dots, M_m\}$ oznacza zbiór maszyn w produkcji przepływowej zaś \mathcal{J}_A zbiór zadań dla pierwszego problemu pomocniczego.

Istotnymi elementami modelu problemu pomocniczego ALM_A dla metody przełączania jest wyróżnienie struktury maszyny ze stanowiskiem kontroli M_q i dodanie do stanu właściwego tej maszyny

parametrów związanych z uwzględnieniem wyróżniania typów braków w procesie kontroli jakości. Zatem stan właściwy maszyny z kontrolą jakości jest następujący:

$$x^{m+q} = (\beta, \tau, QC_1, \dots, QC_{md}, \hat{\beta}) \quad (8.1)$$

gdzie:

$\beta \in \{0, 1, \dots, n\}$ – oznacza indeks zadania, które jest aktualnie obrabiane na danej maszynie (0 oznacza, że żadne zadanie nie zostało przydzielone do maszyny),

$\tau \in [0, \infty)$ – czas do ukończenia obróbki aktualnego zadania na maszynie (∞ oznacza, że maszyna nie pracuje),

$QC_\gamma \in [0, 100]$ – oznacza procent wybrakowanych elementów w partii dla typu γ , gdzie $\gamma = 1, \dots, md$, a md jest liczbą maszyn dodatkowych, służących do naprawy danego typu braku $|\mathcal{M}_D| = md$, suma procentów wynosi 100, (0 oznacza pozytywny wynik kontroli jakości, czyli wszystkie elementy w partii są wykonane poprawnie),

$\hat{\beta} \in \{0, 1, \dots, n\}$ – oznacza indeks zadania, w którym wykryto braki jakościowe podczas ostatniej kontroli jakości

Ponadto istotne w modelu ALM_A jest, że w przypadku kiedy maszyna z kontrolą jakości jest wolna tuż po zakończeniu wykonywania zadania, a wynik kontroli jest negatywny, to stan następny nie jest obliczany za pomocą funkcji przejścia, tylko za pomocą funkcji przełączającej.

Również z punktu widzenia metody przełączania istotne jest, aby w modelu pomocniczym ALM_A wyróżnić maszynę powrotną M_r z marszruty technologicznej, gdzie $r \leq q$ oraz magazyn międzyoperacyjny przed tą maszyną W_r . Kodowanie stanu właściwego tej wyróżnionej maszyny i magazynu są takie same jak pozostałych maszyn i magazynów w marszrucie technologicznej i podane w rozdziale 6.

Model problemu pomocniczego z maszynami naprawczymi

Drugi problem pomocniczy reprezentowany jest przez ALM_B i uwzględnia jedną lub więcej dodatkowych maszyn naprawczych. Rozpatrzmy przypadek szczególny, gdy wystąpią braki jakościowe wszystkich typów. Wtedy w drugim problemie pomocniczym występuje jedna maszyna ze stanowiskiem kontroli jakości M_q , wszystkie dodatkowe maszyny naprawcze M_{m+1}, \dots, M_{md} spoza marszruty technologicznej, odpowiadające naprawie odpowiednich typów braków oraz wszystkie naprawione elementy powracają do ponownej obróbki tylko na jedną konkretną maszynę M_r w marszrucie technologicznej. Model tego problemu jest odpowiednią modyfikacją modelu ALM_A o elementy związane z koniecznością naprawy wadliwych subpartii, czyli o maszyny naprawcze i o maszynę powrotną. Model ten oznaczono jako ALM_B .

Niech $U_B, S_B, (s_0)_B, f_B, (S_N)_B, (S_G)_B$ oznaczają odpowiednio zbiór decyzji, zbiór stanów uogólnionych, uogólniony stan początkowy, funkcję przejścia, zbiór stanów niedopuszczalnych i zbiór stanów docelowych modelu ALM_B oraz niech $\mathcal{M}_B = \{M_1, \dots, M_m, M_{m+1}, \dots, M_{md}\}$. Zatem $|\mathcal{M}_B| = |\mathcal{M}_A| + |\mathcal{M}_D|$.

Stan systemu s_B Podobnie jak w modelu ALM_A , uogólniony stan procesu $s_B = (x_B, t)$ w danej chwili t jest opisany jako stan maszyn z marszruty, stan maszyny z kontrolą jakości (według wzoru (8.1)), stan magazynu początkowego, stan magazynów międzyoperacyjnych oraz stan dodatkowych magazynów przed odpowiadającymi im maszynami naprawczymi W_{m+1}, \dots, W_{md} oraz stany maszyn naprawczych M_{m+1}, \dots, M_{md} :

$$x_B = \left(\underbrace{x_B^1, \dots, x_B^m}_{\text{stany magazynów}}, \underbrace{x_B^{m+1}, \dots, x_B^{m+q}, \dots, x_B^{2m}}_{\text{stany maszyn z marszruty}}, \underbrace{x_B^{2m+1}, x_B^{2m+2}}_{\substack{\text{stan magazynu} \\ \text{i maszyny naprawczej typu 1}}}, \dots, \underbrace{x_B^{2m+2md-1}, x_B^{2m+2md}}_{\substack{\text{stan magazynu} \\ \text{i maszyny naprawczej typu md}}} \right) \quad (8.2)$$

gdzie:

x_B^1 - stan magazynu początkowego,

x_B^i - stan i -tego magazynu międzyoperacyjnego W_i (pomiędzy $i - 1$ -wszą maszyną i i -tą maszyną) for $i = 2, 3, \dots, m$,

x_B^{m+i} - stan i -tej maszyny dla $i = 1, 2, \dots, m$ i $i \neq q$,

x_B^{m+q} - stan q -tej maszyny ze stanowiskiem kontroli jakości,

$x_B^{2m+2\gamma-1}$ - stan magazynu dodatkowego $W_{m+\gamma}$ dla subpartii wybrakowanych o odpowiednich typach braków γ , gdzie $\gamma = 1, \dots, md$,

$x_B^{2m+2\gamma}$ - stan maszyny naprawczej $M_{m+\gamma}$ dla subpartii wybrakowanych o odpowiednich typach braków γ , gdzie $\gamma = 1, \dots, md$.

Kodowanie stanu właściwego magazynu początkowego, magazynów międzyoperacyjnych, maszyn z marszruty oraz wyróżnionej maszyny z kontrolą jakości, maszyny i magazynu powrotnego są takie jak w modelu ALM_A . Dodatkowo rozważane są w modelu współrzędne stanu właściwego magazynów dodatkowych oraz dedykowanych maszyn naprawczych. Kodowanie tych stanów jest takie samo jak magazynów międzyoperacyjnych i maszyn w marszrucie technologicznej i dane jest wzorem (4.2).

Stan początkowy modelu $(s_0)_B = ((x_0)_B, t_0)$ jest obliczany za pomocą funkcji przełączającej. Zbiór stanów niedopuszczalnych $(S_N)_B$ ma taką samą strukturę jak zbiór stanów niedopuszczalnych $(S_N)_A$, należy tylko wziąć pod uwagę zbiór zadań \mathcal{J}_B , dodatkowe magazyny W_{m+1}, \dots, W_{md} i maszyny do naprawy braków M_{m+1}, \dots, M_{md} :

$$(S_N)_B = \{s_B = (x_B, t) : \exists j \in \mathcal{J}_B \ j \notin \mathcal{F}\mathcal{J}_B(s) \wedge \tilde{d}(j) < t\}.$$

Ponieważ stan uwzględniający naprawę elementów oznacza, że proces jest w trakcie (istnieją zadania niewykonane, w szczególnym przypadku jest to tylko jedno naprawiane zadanie), a stan ukończenia naprawy elementów należy do zbioru stanów przełączania dlatego zbiór stanów docelowych $(S_G)_B$ jest nieokreślony.

Decyzje u_B Struktura decyzji modelu ALM_B dla tego problemu będzie analogiczna jak dla modelu ALM_B dla problemu $FS-M_qM_dM_r$ (sekcja 7.3.1). Rozważając w problemie maszyny naprawcze $M_{m+1} \dots M_{md}$, należy rozszerzyć wektor u_B o dodatkowe współrzędne $u_B^{m+\gamma}$, dla $\gamma = \{1, \dots, md\}$. Stąd wektor decyzji w modelu ALM_B ma postać:

$$u_B = (u_B^1, u_B^2, \dots, u_B^m, u_B^{m+1}, \dots, u_B^{m+md}) \text{ i } u_B^i \in \mathcal{J}_B \cup \{0\}, \text{ dla } i = 1, 2, \dots, m + md.$$

Oczywiście nie wszystkie decyzje u_B mogą być podjęte w stanie (x_B, t) . Możliwe wartości współrzędnych są takie same jak dla modelu ALM_B problemu $FS-M_qM_dM_r$ (sekcja 7.3.1).

Funkcja przejścia f_B Funkcja przejścia jest taka sama jak dla modelu ALM_B problemu $FS-M_qM_dM_r$ (sekcja 7.3.1). Moment t' jest wyznaczany dokładnie tak samo, oczywiście biorąc pod uwagę również czas pracy maszyn naprawczych.

Współrzędne stanów magazynów międzyoperacyjnych x_B^i dla: $i = 1, \dots, md$ zmieniają się za pomocą funkcji przejścia dokładnie w taki sam sposób jak w modelu ALM_A .

W funkcji przejścia współrzędne $x_B^{2m+2\gamma-1}$ dla $\gamma = 1, \dots, md$ oznaczające stan magazynów dodatkowych i wynoszą odpowiednio:

$$(x_B^{2m+2\gamma-1}(s))' = x_B^{2m+2\gamma-1}(s) \setminus \{j : u_B^{m+\gamma}(s) = j\}, \quad \gamma = 1, \dots, md \quad (8.3)$$

Wartość współrzędnych oznaczających stan maszyn w marszrucie technologicznej $(x_B^{i+m})'$ dla: $i = 1, \dots, m$ zależą od podjętej decyzji i zmieniają się dokładnie w taki sam sposób jak w modelu ALM_A .

Natomiast stan maszyn naprawczych M_{m+1}, \dots, M_{md} , współrzędne $x_B^{2m+2\gamma}$ dla $\gamma = 1, \dots, md$, zmieniają się dokładnie w taki sam sposób jak stan maszyny naprawczej M_d w modelu ALM_B problemu $FS-M_qM_dM_r$ w sekcji (7.3.1).

Funkcja przełączająca Ze względu na to, że podział problemu na problemy pomocnicze jest analogiczny jak dla problemu $FS-M_qM_dM_r$ opisanego w rozdziale 7, to funkcja przełączająca jest podobna jak w sekcji 7.4, przy czym konieczne jest uwzględnienie występowania braków jakościowych różnego typu:

1. Wyróżnione są trzy rodzaje przełączeń:

- i Przełączenie z modelu ALM_A na model ALM_B w przypadku wykrycia braku jakościowego dowolnego typu (negatywny wynik kontroli jakości), w sytuacji gdy nie występuje w problemie zadanie ze wcześniej wykrytymi brakami jakościowymi,
- ii Przełączenie z modelu ALM_B na model o tej samej strukturze tylko ze zmienionymi parametrami, czyli o strukturze ALM'_B w przypadkach, gdy kontrola jakości wykryła kolejne zadanie z brakami jakościowymi dowolnego typu, a poprzednio wykryte zadanie (zadania) nie zostało jeszcze naprawione lub właśnie zostało naprawione albo przy braku wykrycia kolejnego zadania z brakami jakościowymi, któreś poprzednio wykryte zadanie z brakami zostało naprawione, a inne wcześniej wykryte czekają na naprawę.
- iii Przełączenie modelu o strukturze ALM_B na model o strukturze ALM_A ze zmienionymi parametrami, czyli ALM'_A , w przypadku gdy ostatnie z zadań naprawianych zostało właśnie naprawione.

2. Zbiór stanów przełączania jest zdefiniowany jako zbiór stanów, w których maszyna ze stanowiskiem kontroli jakości M_q skończy pracę i wynik kontroli jakości jest negatywny tzn. wykryto dowolny, co najmniej jednego typu błąd jakościowy lub też dowolna maszyna naprawcza skończy

pracę M_γ , gdzie $\gamma = m + 1, \dots, m + md$ i należy umieścić naprawione zadanie do ponownej obróbki na marszrutę technologiczną:

$$\mathcal{S}_{switch} = \{s = (x, t) : x^{q+m} = (0, \infty, QC_1 \dots, QC_{md}, \hat{\beta}^{q+m} \neq 0) \wedge \exists \gamma = 1, \dots, md : QC_\gamma \neq 0 \\ \vee \exists \gamma = 1, \dots, md : x^{2m+2\gamma-1} = (0, \infty)\} \quad (8.4)$$

3. Reguły przełączania \mathcal{R}_{switch} służą do określenia w jaki sposób należy przeliczyć parametry modelu przy przełączaniu. Reguły te określa się analogicznie jak dla problemu $FS-M_qM_dM_r$, co przedstawiono w podrozdziale 7.4.1. Nie mniej jednak w regułach tych należy uwzględnić, że przy wykryciu braku jakościowego typu γ reguła ma określić zmiany parametrów dla maszyny naprawczej dedykowanej dla danego typu braku. Pojawienie się braku jakościowego typu γ nie ma wpływu na działanie pozostałych maszyn naprawczych.

W związku powyższym algorytm przełączania będzie następujący:

- KROK 1 Określenie wejściowego zbioru zadań \mathcal{J}_A , zbioru maszyn \mathcal{M}_A
- KROK 2 Określenie modelu ALM_A .
- KROK 3 Wyznaczenie stanu początkowego $(s_0)_A$.
- KROK 4 Wyznaczenie następnego stanu systemu s' za pomocą funkcji przejścia aktualnego modelu $s' = f_A(u, s)$.
- KROK 5 Sprawdzenie warunków stopu. Jeżeli stan s' należy do stanów docelowych $(S_G)_A$ lub stanów niedopuszczalnych $(S_N)_A$, to generowanie trajektorii zostaje zakończone i wypisywany jest uzyskany wynik. W przeciwnym przypadku należy przejść do KROKU 6
- KROK 6 Sprawdzenie, czy stan należy do zbioru stanów przełączenia $s \in \mathcal{S}_{switch}$. Jeśli wykryto nowy brak jakościowy dowolnego typu γ , to należy przejść do KROKU 7. W przeciwnym przypadku należy przejść do KROKU 4.
- KROK 7 Określenie modelu ALM_B uwzględniającego magazyny i maszyny dodatkowe, w tym magazyn dodatkowy i maszynę dodatkową do naprawy braku jakościowego typu γ .
- KROK 8 Obliczenie zbioru zadań \mathcal{J}_B , zbioru maszyn \mathcal{M}_B oraz stanu $(s_0)_B$ na podstawie odpowiedniej reguły przełączenia R_{switch} .
- KROK 9 Wyznaczenie następnego stanu systemu s' za pomocą funkcji przejścia aktualnego modelu $s' = f_B(u, s)$.
- KROK 10 Sprawdzenie warunku stopu. Jeżeli stan s' należy do stanów niedopuszczalnych $(S_N)_B$, to generowanie trajektorii zostaje zakończone i wypisywany jest uzyskany wynik. W przeciwnym przypadku należy przejść do KROKU 11.
- KROK 11 Sprawdzenie, czy stan należy do zbioru stanów przełączenia $s \in \mathcal{S}_{switch}$. Jeśli nie, to należy przejść do KROKU 9. W przeciwnym przypadku:

1. jeśli wykryto nowy brak jakościowy dowolnego typu γ , to należy przejść do KROKU 7.
2. jeśli maszyna naprawcza dowolnego typu γ zakończyła naprawę, to w przypadku, w którym istnieją jeszcze braki jakościowe dowolnego typu nie naprawione, to należy przejść do KROKU 8. W przypadku, w którym nie występują już nowe braki należy przejść do KROKU 12.

KROK 12 Obliczenie zbioru zadań \mathcal{J}_A , zbioru maszyn \mathcal{M}_A oraz stanu $(s_0)_A$ na podstawie odpowiedniej reguły przełączenia R_{switch} i przejście do KROKU 4.

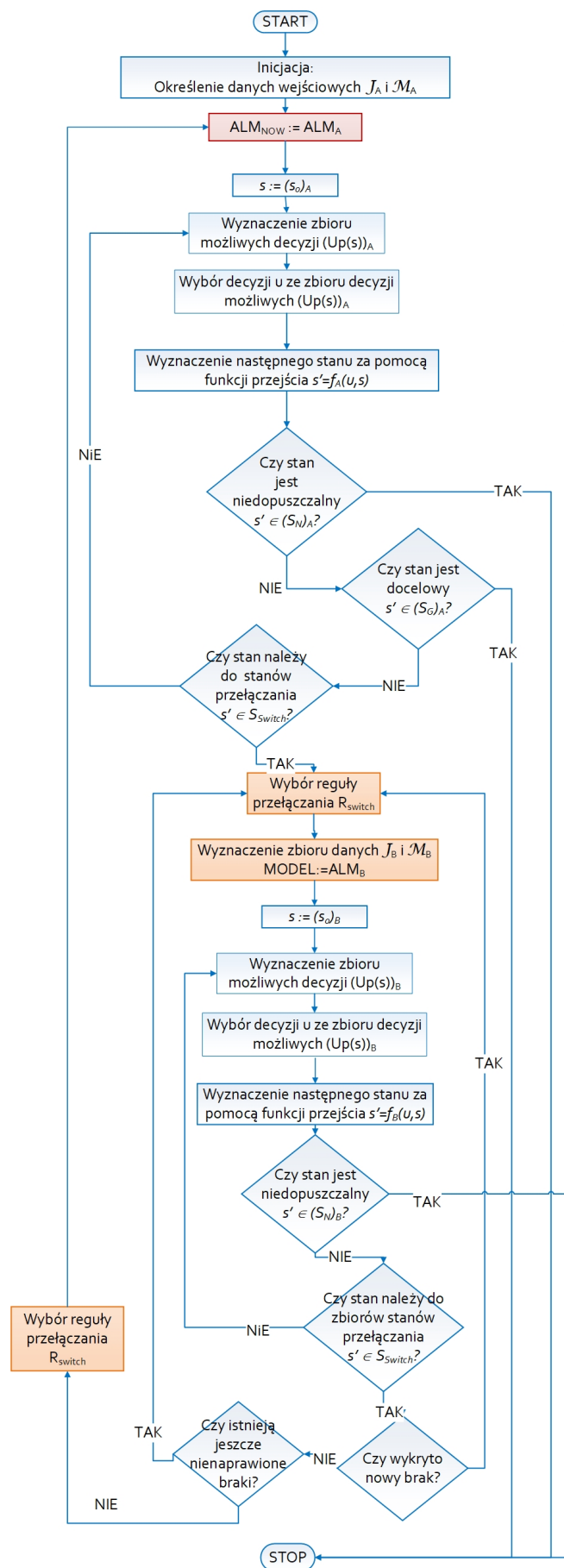
Na rysunku 8.2 przedstawiono powyższy algorytm.

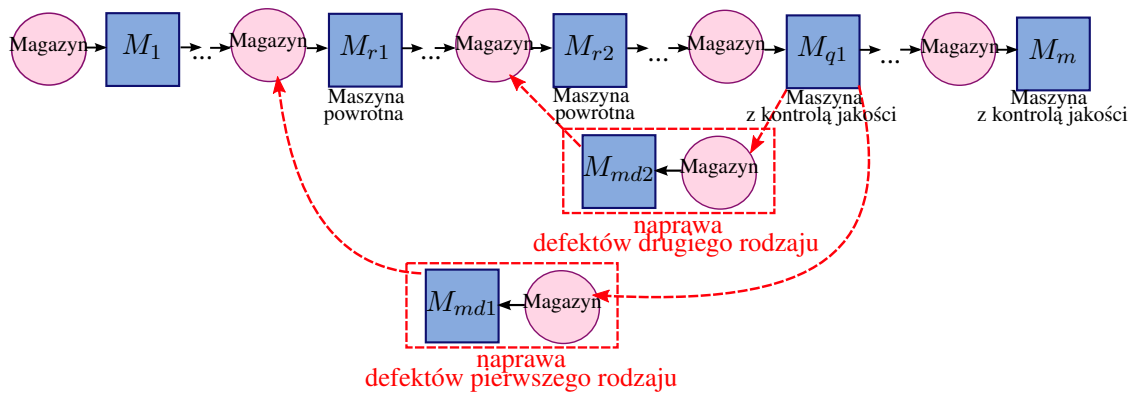
8.2. Modelowanie problemu $FS-M_qmM_d mM_r$

W problemie $FS-M_qmM_d mM_r$ w odróżnieniu od wyżej rozważanego problemu, powrót naprawionych detali następuje nie na jedną ale na wiele maszyn powrotnych. Zatem w marszrucie technologicznej występuje tylko jedna maszyna ze stanowiskiem kontroli jakości ($|\mathcal{M}_Q| = 1$), więcej niż jedna maszyna dodatkowa, do naprawy wadliwych elementów ($|\mathcal{M}_D| > 1$ i $\mathcal{M}_D = \{M_{m+1}, \dots, M_{md}\}$) oraz więcej niż jedna maszyna powrotna $|\mathcal{M}_R| > 1$. Wybór maszyny do naprawy jest zależny od stopnia lub rodzaju uszkodzenia wadliwych elementów i jest wynikiem kontroli jakości. Naprawione elementy powracają na różne konkretnie wybrane (określone) maszyny w marszrucie technologicznej. Nie jest wykluczana sytuacja, w której z więcej niż jednej maszyny dodatkowej poprawione zadania wracają na jedną konkretną maszynę w ciągu technologicznym. Natomiast wykluczona jest sytuacja, w której z jednej maszyny dodatkowej poprawione zadania wracają na więcej niż jedną maszyną z marszrutu technologicznej. Oznaczmy maszynę ze stanowiskiem kontroli jakości jako $M_q \in \mathcal{M}_Q \subset \mathcal{M}$ oraz indeksy maszyn, na które wracają poprawione zadania jako $r_1 < \dots < r_b$ i $M_{r_1}, \dots, M_{r_b} \in \mathcal{M}$. Na rysunku 8.3 przedstawiono przykładową linię produkcyjną dla tak określonego problemu.

8.2.1. Analiza problemu

W rozważanym problemie w wyniku kontroli jakości wykrywane są różnego typu braki jakościowe, na jednej maszynie z kontrolą jakości. Rodzaj braku implikuje jednoznacznie wybór maszyny naprawczej. W problemie rozważane jest tyle maszyn naprawczych ile typów (rodzajów) braków jakościowych. Postępowanie podczas wykrycia nowego braku jest następujące: w wyniku wykrycia w zadaniu (partii) nowych braków jakościowych zadanie dzielone jest na subpartie. Prawidłowo wykonana subpartia kontynuuje obróbkę na kolejnej maszynie z marszrutu technologicznej, zaś wybrakowana subpartia (partia składająca się z elementów z defektami) jest przekazywana do obróbki na dedykowaną dla danego typu braku, maszynę naprawczą, wskazaną w wyniku kontroli jakości. Zadanie naprawione wraca do obróbki na ustaloną maszynę powrotną, w zależności od tego, która maszyna wykonywała naprawę. Tzn. jest rozważane więcej maszyn powrotnych, natomiast jest ich co najwyżej tyle ile maszyn naprawczych. Rodzaj maszyny naprawczej jednoznacznie określa wybór maszyny powrotnej.

Rysunek 8.2: Algorytm przełączania dla problemu $FS-M_qmM_dM_r$



Rysunek 8.3: Przykład linii z maszyną z kontrolą jakości oraz dwiema maszynami powrotnymi i naprawczymi

Z powodu podziału partii na subpartie zmianie ulegnie zbiór zadań do wykonania: powstaną nowe zadania składające się z elementów z defektami oraz zadanie, w którym wykryto elementy z brakami jakościowymi zostanie pomniejszone o te wybrakowane elementy.

W przypadku wykrycia konkretnego typu braku jakościowego konieczne jest rozważanie dodatkowej maszyny naprawczej, spoza ciągu technologicznego. Do obróbki na maszynie naprawczej mogą być przydzielone tylko subpartie wybrakowane o danym typie braku. W sytuacji, w której nie występują braki jakościowe danego typu, nie ma konieczności rozważania w problemie maszyny do naprawy danego typu braku. W sytuacji, w której wszystkie wybrakowane partie zostaną naprawione, nie ma konieczności rozważania żadnej maszyny naprawczej, oczywiście do momentu wykrycia kolejnych braków.

W problemie tym rozważane są zatem trzy charakterystyczne typy maszyn, przy czym występuje tylko jedna maszyna z kontrolą jakości M_q w marszrucie technologicznej, więcej niż jedna maszyna powrotna $M_{r1}, \dots, M_{rb} \in \mathcal{M}$ w marszrucie technologicznej i więcej niż jedna maszyna naprawcza $\mathcal{M}_{\mathcal{D}} = \{M_{m+1}, \dots, M_{md}\}$ spoza marszrut. Dla danego typu braku jakościowego γ określona jest jednoznacznie maszyna naprawcza $M_{m+\gamma}$ oraz maszyna powrotna $M_{r\gamma}$.

W związku z tym należy zaprojektować podczas modelowania kodowanie stanu maszyny z kontrolą jakości, w taki sposób by uwzględnić różne typy braków jakościowych oraz wynikające z tego kodowania stany dodatkowych maszyn naprawczych, a także uwzględnienie więcej niż jednej wyróżnionej maszyny powrotnej.

8.2.2. Podział problemu na problemy pomocnicze

W opisywanym przypadku podział na problemy pomocnicze będzie identyczny jak w problemie $FS-M_qmM_dM_r$ opisanym w sekcji 8.1, ponieważ liczba wyróżnionych maszyn powrotnych należących do podstawowego procesu technologicznego (marszrutu technologicznego) nie ma znaczenia.

Pierwszy problem pomocniczy jest to problem produkcji przepływowej z maszyną kontroli jakości, ale bez żadnej maszyny naprawczej. Został on wykorzystany również jako problem pomocniczy dla wszystkich wcześniej rozważanych problemów. Natomiast ze względu na występowanie różnych typów braków jakościowych kontrola jakości wygląda analogicznie jak w problemie $FS-M_qmM_dM_r$

przedstawionym w podrozdziale 8.1.2, więc kodowanie stanu właściwego maszyny z kontrolą jakości M_q będzie również dane wzorem 8.1. Zatem pierwszy model algebraiczno-logiczny jest taki sam jak w podrozdziale 6.3, z tym, że zmienione jest kodowanie stanu właściwego maszyny z kontrolą jakości, tak aby uwzględnić różne typy braków jakościowych.

Podobnie jak w poprzednio rozważanych problemach, oznaczmy przez $U_A, S_A, (s_A)_0, f_A, (S_A)_N, (S_A)_G$ odpowiednio zbiór decyzji, zbiór stanów uogólnionych, uogólniony stan początkowy, funkcję przejścia, zbiór stanów niedopuszczalnych i zbiór stanów docelowych modelu ALM_A oraz niech $\mathcal{M}_A = \{M_1, \dots, M_m\}$ oznacza zbiór maszyn w produkcji przepływowej zaś \mathcal{J}_A zbiór zadań dla pierwszego problemu pomocniczego.

Jak wcześniej podano istotnymi elementami modelu problemu pomocniczego ALM_A dla metody przełączania jest wyróżnienie struktury maszyny ze stanowiskiem kontroli M_q i dodanie do stanu właściwego tej maszyny parametrów związanych z uwzględnieniem wyróżniania typów braków w procesie kontroli jakości oraz dodatkowo wyróżnienie wielu a nie tylko jednej maszyny powrotnej.

Istotne w modelu ALM_A jest, że w przypadku kiedy maszyna z kontrolą jakości jest wolna tuż po zakończeniu wykonywania zadania, a wynik kontroli jest negatywny, to stan następny nie jest obliczany za pomocą funkcji przejścia, tylko za pomocą funkcji przełączającej.

Również z punktu widzenia metody przełączania istotne jest, aby w modelu pomocniczym ALM_A wyróżnić maszyny powrotne $M_{r_1}, \dots, M_{r_b} \in \mathcal{M}$ z marszruty technologicznej, gdzie M_{r_γ} dla $\gamma = 1, \dots, b$ oznacza maszynę powrotną dla odpowiedniego typu braku M_{r_γ} . Należy również wyznaczyć odpowiednie magazyny międzyoperacyjne przed wyróżnionymi maszynami powrotnymi W_{r_1}, \dots, W_{r_b} . Kodowanie stanu właściwego tej wyróżnionej maszyny i magazynu są takie same jak pozostałych maszyn i magazynów w marszrucie technologicznej i podane w rozdziale 6.

Drugi problem pomocniczy, reprezentowany przez ALM_B jest analogiczny jak dla problemu powyższego z wyróżnieniem więcej niż jednej maszyny powrotnej (podrozdział 8.1.3). Ze względu na to, że stan maszyn powrotnych jest zakodowany tak samo jak stan wszystkich maszyn z marszruty technologicznej, to wyróżnienie więcej niż jednej maszyny powrotnej nie ma znaczenia dla struktury stanu systemu s_B , struktury decyzji u_B ani funkcji przejścia f_B , ma dopiero znaczenie przy definiowaniu funkcji przełączającej.

Funkcja przełączająca Ze względu na to, że podział problemu na problemy pomocnicze jest analogiczny jak dla problemu $FS-M_qmM_dM_r$ opisanego w podrozdziale 8.1, to funkcja przełączająca jest niemal identyczna jak w sekcji 8.1.3. Należy tylko wziąć pod uwagę, że występują braki jakościowe różnego typu analogicznie jak dla problemu $FS-M_qmM_dM_r$:

1. Wyróżnione są trzy rodzaje przełączeń:

- i Przełączenie z modelu ALM_A na model ALM_B w przypadku wykrycia braku jakościowego dowolnego typu (negatywny wynik kontroli jakości), w sytuacji gdy nie występuje w problemie zadanie ze wcześniej wykrytymi brakami jakościowymi,
- ii Przełączenie z modelu ALM_B na model o tej samej strukturze tylko ze zmienionymi parametrami, czyli o strukturze ALM'_B w przypadkach, gdy kontrola jakości wykryła kolejne

zadanie z brakami jakościowymi dowolnego typu, a poprzednio wykryte zadanie (zadania) nie zostało jeszcze naprawione lub właśnie zostało naprawione albo przy braku wykrycia kolejnego zadania z brakami jakościowymi, któreś poprzednio wykryte zadanie z brakami zostało naprawione, a inne wcześniej wykryte czekają na naprawę.

iii Przełączenie modelu o strukturze ALM_B na model o strukturze ALM_A ze zmienionymi parametrami, czyli ALM'_A , w przypadku gdy ostatnie z zadań naprawianych zostało właśnie naprawione.

2. Zbiór stanów przełączania jest zdefiniowany jako zbiór stanów, w których maszyna ze stanowiskiem kontroli jakości M_q skończy pracę i wynik kontroli jakości jest negatywny tzn. wykryto dowolny, co najmniej jednego typu brak jakościowy lub też dowolna maszyna naprawcza skończy pracę M_γ , gdzie $\gamma = m + 1, \dots, m + md$ i należy umieścić naprawione zadanie do ponownej obróbki na marszrucie technologicznym:

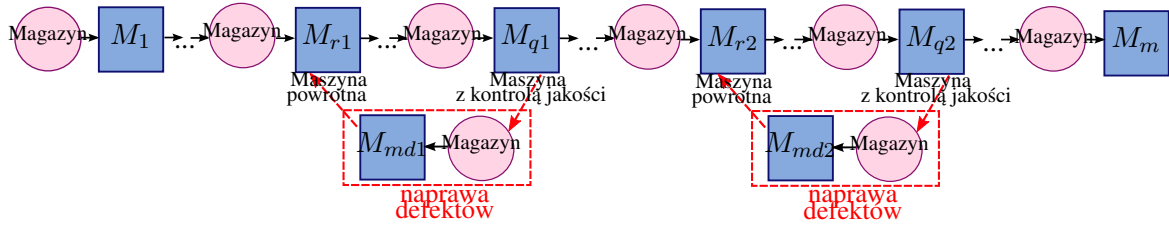
$$\mathcal{S}_{switch} = \{s = (x, t) : x^{q+m} = (0, \infty, QC_1 \dots, QC_{md}, \hat{\beta}^{q+m} \neq 0) \wedge \exists \gamma = 1, \dots, md : QC_\gamma \neq 0 \vee \exists \gamma = 1, \dots, md : x^{2m+2\gamma-1} = (0, \infty)\} \quad (8.5)$$

3. Reguły przełączania \mathcal{R}_{switch} służą do określenia w jaki sposób należy przeliczyć parametry modelu przy przełączaniu. W związku z tym muszą dodatkowo uwzględniać, że przy zakończeniu naprawy braku jakościowego typu γ przez maszynę naprawczą $M_{m+\gamma}$ należy przeliczyć stan magazynu $W_{r\gamma}$ przed dedykowaną maszyną powrotną $M_{r\gamma}$ dla tego typu braku oraz nie ulegają zmianie stany maszyn powrotnych dedykowanych dla innych typów braków. Ponadto przy wykryciu nowego braku jakościowego typu γ reguła ma określić zmiany parametrów dla maszyny naprawczej dedykowanej dla danego typu braku. Pojawienie się braku jakościowego typu γ nie ma wpływu na zmianę stanu innych maszyn naprawczych.

8.3. Modelowanie problemu $FS-mM_qmM_d mM_r$

W tym problemie w marszrucie technologicznej występuje więcej niż jedna maszyna ze stanowiskiem kontroli jakości ($1 < |\mathcal{M}_Q| \leq |\mathcal{M}|$), więcej niż jedna maszyna dodatkowa, do naprawy wadliwych elementów ($|\mathcal{M}_D| > 1$ i $\mathcal{M}_D = \{M_{m+1}, \dots, M_{md}\}$) oraz więcej niż jedna maszyna powrotna $|\mathcal{M}_R| > 1$. Wybór maszyny do naprawy jest zależny od rodzaju uszkodzenia wadliwych elementów i jest wynikiem kontroli jakości na konkretnej maszynie z kontrolą jakości dedykowanej do wykrywania braków jednego, konkretnego rodzaju. Zatem w wyniku każdej kontroli otrzymujemy informację, czy wykryto brak jakościowy, w jakiej liczbie i na którą konkretną maszynę ma być przekazane zadanie. Wykluczona jest sytuacja, w której z dowolnych dwóch stanowisk kontroli, elementy z defektem odsyłane są na ta samą maszynę dodatkową. Naprawione elementy powracają na różne konkretnie wybrane maszyny w marszrucie technologicznej. Nie jest wykluczana sytuacja, w której z więcej niż jednej maszyny dodatkowej poprawione zadania wracają na jedną konkretną maszynę w ciągu technologicznym. Natomiast wykluczona jest sytuacja, w której z jednej maszyny naprawczej poprawione zadania wracają

na więcej niż jedną maszyną z marszruty technologicznej. Oznaczmy indeksy maszyn ze stanowiskiem kontroli jakości jako $q_1 < \dots < q_d \in \mathcal{M}_Q \subset \mathcal{M}$ oraz maszyny, na które wracają poprawione zadania jako: $r_1 < \dots < r_b \in \mathcal{M}$. Na rysunku 8.4 przedstawiono przykładową linię produkcyjną dla tak określonego problemu.



Rysunek 8.4: Przykład linii z dwiema maszynami z kontrolą jakości

8.3.1. Analiza problemu

W rozważanym problemie w wyniku kontroli jakości wykrywane są różnego rodzaju braki jakościowe, na więcej niż jednej maszynie z kontrolą jakości. Na jednej maszynie z kontrolą jakości może być wykryty tylko jeden rodzaj braku jakościowego, rozróżnia się tyle rodzajów braków jakościowych ile maszyn z kontrolą jakości. Rodzaj braku implikuje jednoznacznie wybór maszyny naprawczej. W problemie rozważane jest tyle maszyn naprawczych ile typów (rodzajów) braków jakościowych. Postępowanie podczas wykrycia nowego braku jest następujące: w wyniku wykrycia w zadaniu (partii) nowych braków jakościowych zadanie dzielone jest na dwie subpartie. Prawidłowo wykonana subpartia kontynuuje obróbkę na kolejnej maszynie z marszruty technologicznej, zaś wybrakowana subpartia (partia składająca się z elementów z defektami) jest przekazywana do obróbki na dedykowaną dla danego typu braku, maszynę naprawczą. Zadanie naprawione wraca do obróbki na ustaloną maszynę powrotną, w zależności od tego, która maszyna wykonywała naprawę. Zatem jest rozważane więcej maszyn powrotnych, natomiast jest ich co najwyżej tyle ile maszyn naprawczych. Rodzaj maszyny naprawczej jednoznacznie określa wybór maszyny powrotnej.

Z powodu podziału partii na subpartie zmianie ulegnie zbiór zadań do wykonania: powstaje nowe zadanie składające się z elementów z defektami oraz zadanie, w którym wykryto elementy z brakami jakościowymi zostanie pomniejszone o te wybrakowane elementy.

W przypadku wykrycia braku jakościowego, przez konkretną maszynę ze stanowiskiem kontroli jakości, konieczne jest rozważanie dodatkowej maszyny naprawczej, spoza ciągu technologicznego. Do obróbki na maszynie naprawczej mogą być przydzielone tylko subpartie wybrakowane o danym typie braku (wykryte na konkretnej maszynie z kontrolą jakości). W sytuacji, w której nie występują braki jakościowe danego typu, nie ma konieczności rozważania w problemie maszyny do naprawy danego typu braku. W sytuacji, w której wszystkie wybrakowane partie zostaną naprawione, nie ma konieczności rozważania żadnej maszyny naprawczej, oczywiście do momentu wykrycia kolejnych braków.

W problemie tym rozważane są zatem trzy charakterystyczne typy maszyn, maszyny ze stanowiskiem kontroli jakości \mathcal{M}_Q ($1 < |\mathcal{M}_Q| \leq |\mathcal{M}|$) w marszrucie technologicznej, więcej niż jedna maszyna powrotna $M_{r1}, \dots, M_{rb} \in \mathcal{M}$ w marszrucie technologicznej i więcej niż jedna maszyna naprawcza

$\mathcal{M}_{\mathcal{D}} = \{M_{m+1}, \dots, M_{md}\}$ spoza marszruty. Dla danego typu braku jakościowego γ , wykrytego na maszynie z kontrolą jakości $M_{q\gamma}$ określona jest jednoznacznie maszyna naprawcza $M_{m+\gamma}$ i $|\mathcal{M}_{\mathcal{Q}}| = |\mathcal{M}_{\mathcal{D}}|$ oraz maszyna powrotna $M_{r\gamma}$, przy czym nie wyklucza się sytuacji, że różnego rodzaju braki jakościowe, naprawione na różnych maszynach naprawczych wracają na tą samą maszynę powrotną.

W związku z tym należy zaprojektować podczas modelowania wyróżnienie kilku maszyn ze stanowiskiem kontroli jakości oraz kodowanie stanu maszyny z kontrolą jakości, w taki sposób by uwzględnić jeden typ braków jakościowych wykrywany na danej maszynie oraz wynikające z tego liczby maszyn ze stanowiskiem kontroli jakości kodowania stanów dodatkowych maszyn naprawczych, a także uwzględnienie więcej niż jednej wyróżnionej maszyny powrotnej.

8.3.2. Podział problemu na problemy pomocnicze

W opisywanym problemie podział na problemy pomocnicze będzie analogicznie jak w poprzednio rozważanych problemach $FS-M_qmM_dM_r$ i $FS-M_qmM_dM_r$ opisanych w sekcjach 8.1 i 8.2, ponieważ istotna jest liczba maszyn naprawczych, zaś liczba wyróżnionych maszyn z kontrolą jakości i powrotnych należących do podstawowego procesu technologicznego (marszruty technologicznej) nie ma znaczenia dla tego podziału.

Pierwszy problem pomocniczy jest to problem produkcji przepływowej z więcej niż jedną maszyną kontroli jakości, ale bez żadnej maszyny naprawczej. Liczba maszyn z kontrolą jakości determinuje liczbę rodzajów braków jakościowych, na jednej maszynie z kontrolą jakości wykrywany jest jeden rodzaj braków jakościowych, więc kodowanie stanu właściwego maszyn z kontrolą jakości $M_{q\gamma}$ będzie takie samo jak w problemach opisanych w rozdziałach 6 oraz 7 i jest dane wzorem (6.3). Zatem pierwszy model algebraiczno-logiczny jest analogiczny jak w podrozdziale 6.3, z tym, że wyróżnione jest więcej niż jedna maszyna z kontrolą jakości i więcej niż jedna maszyna powrotna.

Podobnie jak w poprzednio rozważanych problemach, oznaczmy przez $U_A, S_A, (s_A)_0, f_A, (S_A)_N, (S_A)_G$ odpowiednio zbiór decyzji, zbiór stanów uogólnionych, uogólniony stan początkowy, funkcję przejścia, zbiór stanów niedopuszczalnych i zbiór stanów docelowych modelu ALM_A oraz niech $\mathcal{M}_A = \{M_1, \dots, M_m\}$ oznacza zbiór maszyn w produkcji przepływowej zaś \mathcal{J}_A zbiór zadań dla pierwszego problemu pomocniczego.

Istotnymi elementami modelu problemu pomocniczego ALM_A dla metody przełączania jest wyróżnienie maszyn ze stanowiskiem kontroli jakości $M_{q\gamma}$, gdzie $\gamma > 1$ oraz dodatkowo wyróżnienie wielu a nie tylko jednej maszyny powrotnej.

Związku z tym uogólniony stan procesu $s_A = (x_A, t)$ w danej chwili t jest opisany jako stan maszyn z marszruty, stan maszyn z kontrolą jakości, stan magazynu początkowego oraz stan magazynów międzyoperacyjnych

$$x_A = (\underbrace{x_A^1, \dots, x_A^m}_{\text{stany magazynów}}, \underbrace{x_A^{m+1}, \dots, x_A^{m+q_1}, \dots, x_A^{m+i}, \dots, x_A^{m+q_d}, \dots, x_A^{2m}}_{\substack{\text{stany maszyn z marszruty} \\ \text{z wyróżnionymi maszynami z kontrolą jakości}}}) \quad (8.6)$$

gdzie:

x_A^1 - stan magazynu początkowego,

x_A^i - stan i -tego magazynu międzyoperacyjnego W_i (pomiędzy $i - 1$ -wszą maszyną i i -tą maszyną) for

$i = 2, 3, \dots, m$,

x_A^{m+i} - stan i -tej maszyny dla $i = 1, 2, \dots, m, i \neq q_\gamma$, gdzie $\gamma \in \{1, \dots, d\}$,

$x_B^{m+q_\gamma}$ - stan M_{q_γ} maszyny ze stanowiskiem kontroli jakości dla braków jakościowych rodzaju γ , gdzie $\gamma \in \{1, \dots, d\}$.

Kodowanie stanów właściwych zarówno wszystkich magazynów, jak i maszyn jest takie samo jak dla problemu przedstawionego w rozdziale 6 i dane wzorami (6.2) i (6.3).

Istotne w modelu ALM_A jest, że w przypadku kiedy przynajmniej jedna maszyna z kontrolą jakości jest wolna tuż po zakończeniu wykonywania zadania, a wynik kontroli jest negatywny, to stan następny nie jest obliczany za pomocą funkcji przejścia, tylko za pomocą funkcji przełączającej.

Model problemu pomocniczego z maszynami naprawczymi

Drugi problem pomocniczy reprezentowany jest przez ALM_B i jest budowanych analogicznie jak dla problemów $FS-M_qmM_d mM_r$ i $FS-mM_qmM_d mM_r$, tzn. oprócz marszrut technologicznej z wyróżnionymi stanowiskami kontroli jakości model uwzględnia jedną lub więcej dodatkowych maszyn naprawczych. Rozpatrzmy przypadek szczególny, gdy wystąpią braki jakościowe wszystkich typów. Wtedy w drugim problemie pomocniczym występują wszystkie dodatkowe maszyny naprawcze M_{m+1}, \dots, M_{md} spoza marszrut technologicznej, odpowiadające naprawie odpowiednich typów braków oraz wszystkie naprawione elementy powracają do ponownej obróbki na wskazaną konkretną maszynę M_{r_k} w marszrucie technologicznej. Model tego problemu jest odpowiednią modyfikacją modelu ALM_A o elementy związane z koniecznością naprawy wadliwych subpartii, czyli o maszyny naprawcze i stan właściwy x_B takiego systemu będzie następujący:

$$x_B = \underbrace{(x_B^1, \dots, x_B^m)}_{\text{stany magazynów}}, \underbrace{(x_B^{m+1}, \dots, x_B^{m+q_1}, \dots, x_B^{m+q_d}, \dots, x_B^{2m})}_{\substack{\text{stany maszyn z marszrut} \\ \text{z wyróżnionymi maszynami z kontrolą jakości}}}, \underbrace{(x_B^{2m+1}, x_B^{2m+2}, \dots)}_{\substack{\text{stan magazynu} \\ \text{i maszyny naprawczej typu 1}}}, \dots, \underbrace{(x_B^{2m+2md-1}, x_B^{2m+2md})}_{\substack{\text{stan magazynu} \\ \text{i maszyny naprawczej typu md}}} \quad (8.7)$$

gdzie:

x_A^1 - stan magazynu początkowego,

x_A^i - stan i -tego magazynu międzyoperacyjnego W_i (pomiędzy $i - 1$ -wszą maszyną i i -tą maszyną) for $i = 2, 3, \dots, m$,

x_A^{m+i} - stan i -tej maszyny dla $i = 1, 2, \dots, m, i \neq q_\gamma$, gdzie $\gamma \in \{1, \dots, d\}$,

$x_B^{m+q_\gamma}$ - stan M_{q_γ} maszyny ze stanowiskiem kontroli jakości dla braków jakościowych rodzaju γ , gdzie $\gamma \in \{1, \dots, d\}$

$x_B^{2m+2\gamma-1}$ - stan magazynu dodatkowego $W_{m+\gamma}$ dla subpartii wybrakowanych o odpowiednich typach braków γ , gdzie $\gamma = 1, \dots, md$,

$x_B^{2m+2\gamma}$ - stan maszyny naprawczej $M_{m+\gamma}$ dla subpartii wybrakowanych o odpowiednich typach braków γ , gdzie $\gamma = 1, \dots, md$.

Kodowanie stanu właściwego magazynu początkowego, magazynów międzyoperacyjnych, maszyn z marszrut oraz wyróżnionej maszyny z kontrolą jakości, maszyny i magazynu powrotnego są takie jak w modelu ALM_A . Dodatkowo rozważane są w modelu współrzędne stanu właściwego magazynów

dotychczasowych oraz dedykowanych maszyn naprawczych. Kodowanie tych stanów jest takie samo jak stan magazynów międzyoperacyjnych i maszyn w marszrucie technologicznej i dane jest wzorem (4.2).

Funkcja przełączająca Ze względu na to, że podział problemu na problemy pomocnicze jest analogiczny jak dla problemu $FS-M_qmM_dM_r$ opisanego w podrozdziale 8.1, to funkcja przełączająca jest niemal identyczna jak w sekcji 8.1.3. Należy tylko wziąć pod uwagę, że występują braki jakościowe różnego typu wykrywane przez różne maszyny z kontrolą jakości.

1. Wyróżnione są trzy rodzaje przełączeń:

- i Przełączenie z modelu ALM_A na model ALM_B w przypadku wykrycia przynajmniej jednego braku jakościowego dowolnego typu (negatywny wynik kontroli jakości przynajmniej jednej maszyny z kontrolą jakości), w sytuacji gdy nie występuje w problemie zadanie ze wcześniej wykrytymi brakami jakościowymi,
- ii Przełączenie z modelu ALM_B na model o tej samej strukturze tylko ze zmienionymi parametrami, czyli o strukturze ALM'_B w przypadkach, gdy przynajmniej jedna maszyna z kontrolą jakości wykryła kolejne zadanie z brakami jakościowymi, a poprzednio wykryte zadanie (zadania) nie zostało jeszcze naprawione lub właśnie zostało naprawione albo przy braku wykrycia kolejnego zadania z brakami jakościowymi, któreś poprzednio wykryte zadanie z brakami zostało naprawione, a inne wcześniej wykryte czekają na naprawę.
- iii Przełączenie modelu o strukturze ALM_B na model o strukturze ALM_A ze zmienionymi parametrami, czyli ALM'_A , w przypadku gdy ostatnie z zadań naprawianych przed dowolną maszyną naprawczą zostało właśnie naprawione.

2. Zbiór stanów przełączania jest zdefiniowany jako zbiór stanów, w których dowolna maszyna ze stanowiskiem kontroli jakości M_{q_k} skończy pracę i wynik kontroli jakości jest negatywny (wykryto dowolny brak jakościowy) lub też dowolna maszyna naprawcza skończy pracę M_γ , gdzie $\gamma = m + 1, \dots, m + md$ i należy umieścić naprawione zadanie do ponownej obróbki na marszrutę technologiczną:

$$\mathcal{S}_{switch} = \{s = (x, t) : \exists k = 1, \dots, d \ x^{m+q_k} = (0, \infty, QC \neq 0, \hat{\beta} \neq 0) \vee \exists \gamma = 1, \dots, md : x^{2m+2\gamma-1} = (0, \infty)\} \quad (8.8)$$

3. Reguły przełączania \mathcal{R}_{switch} służą do określenia w jaki sposób należy przeliczyć parametry modeli przy przełączaniu. W związku z tym muszą dodatkowo uwzględniać, że wystarczy aby dowolna maszyna ze stanowiskiem kontroli jakości zakończyła pracę z negatywnym wynikiem kontroli, aby należało uwzględnić odpowiednią maszynę naprawczą oraz przy zakończeniu naprawy dowolnego rodzaju braku jakościowego należy wskazać miejsce powrotu tego zadania na marszrutę technologiczną. W związku z tym przy wykryciu nowego braku jakościowego typu γ przez dedykowaną maszynę M_{q_γ} reguła ma określić zmiany parametrów dla maszyny naprawczej dedykowanej dla danego typu braku. Pojawienie się braku jakościowego typu γ nie ma wpływu na zmianę stanu

innych maszyn naprawczych dedykowanych dla innego rodzaju braków jakościowych. Zakończenie naprawy braku jakościowego typu γ przez maszynę naprawczą $M_{m+\gamma}$ powoduje konieczność przeliczenia stan magazynu $W_{r\gamma}$ przed dedykowaną maszyną powrotną $M_{r\gamma}$ dla tego typu braku oraz nie ulegają zmianie stany magazynów powrotnych dedykowanych dla innych typów braków.

Rozważanie bardziej złożonych systemów jest analogiczne jak to przedstawiono na powyższych problemu. Przykładowo w pracy tej nie rozważany jest przypadek, w którym naprawa braku jakościowego może polegać na obróbce w systemie kilku maszyn naprawczych równoległych. Problem taki jednak nie wiele różniłby się od podanych powyżej. W modelu pomocniczym wyróżniony byłby jeden wspólny magazyn dla wszystkich wykrytych braków danego typu dla równoległych maszyn naprawczych, a nie tak jak w powyższych przypadkach magazyn dedykowany dla danego typu braku i maszyny naprawczej. W wyniku przełączenia subpartia wybrakowana przydzielona byłaby do magazynu zaś przydział zadania do obróbki na maszynie podejmowany byłby na podstawie decyzji zgodnie z funkcją kryterialną.

8.4. Właściwości modeli problemów pomocniczych

Struktura modeli, pomiędzy którymi następuje przełączenie może być taka sama lub różna. Struktura zależy o typu zdarzeń zewnętrznych. W najprostszym przypadku można użyć do przełączania tylko jeden model. Na podstawie rozważonych problemów można wyróżnić następujące własności modeli pomocniczych, między którymi następuje przełączanie.

Własność 1 Jeśli naprawa braków jakościowych polega na wykonaniu dodatkowej operacji, ale zgodnie z marszrutą technologiczną (powtórzeniu operacji z marszruty), to rozważane są modele o tej samej strukturze.

Własność 2 Jeśli naprawa braków jakościowych polega na wykonaniu dodatkowej operacji spoza marszruty technologicznej, to rozważane są modele pomocnicze o różnej strukturze.

Własność 3 Liczba maszyn powrotnych nie wpływa na strukturę modeli pomocniczych.

Własność 4 Wystąpienie różnych typów braków jakościowych może być wykrywane na jednej lub na wielu maszynach z kontrolą jakości i nie ma to wpływu na strukturę modeli pomocniczych, tylko na sposób kodowania stanu maszyn ze stanowiskiem kontroli jakości, które jest jednakowe we wszystkich modelach pomocniczych.

Dla rozważanej klasy problemów, różna struktura modeli pomocniczych polega na zmianie długości wektora stanu systemu. Oznacza to, rozważana jest w danym stanie lub nie jest rozważana maszyna dodatkowa spoza marszruty technologicznej, natomiast kodowanie stanów konkretnych maszyn występujących w rozważanych modelach pomocniczych jest jednakowe.

9. Metoda przełączania modeli algebraiczno-logicznych dla produkcji przepływowej z brakami jakościowymi

Celem tego rozdziału jest zaproponowanie zastosowania koncepcji przełączania modeli algebraiczno-logicznych do modelowania klasy problemów produkcji przepływowej z brakami jakościowymi dla dowolnej liczby maszyn specjalnych. Podstawą do przedstawienia tej metody jest ogólny schemat meta-modelu algebraiczno-logicznego *ALMM* przytoczony w rozdziale 3 oraz ogólna koncepcja modelowania zakłóceń za pomocą metody przełączania modeli algebraiczno-logicznych, która została zaproponowana przez E. Dudek-Dyduch i przedstawiona po raz pierwszy w pracy Grobler-Dębska et al. (2013). E. Dudek-Dyduch zaproponowała w pracach Dudek-Dyduch (2015a,b) dwu-etapową metodę transformacji modeli algebraiczno-logicznych (*2SALMT*) (*ang. Two-Stage AL Model Transformation Method*). Metoda ta należy do metod sztucznej inteligencji i jej idea polega na tym, że w momencie wystąpienia zaburzenia w automatyczny sposób dokonywana jest transformacja modeli. Jest to ogólna metoda, która służy do modelowania problemów produkcji dyskretniej z różnego rodzaju niedeterministycznymi zakłóceniami.

Zastosowanie metody przełączania modeli algebraiczno-logicznych dla produkcji przepływowej z brakami jakościowymi umożliwia stworzenie formalnego zapisu skomplikowanych problemów produkcji przepływowej, w których w trakcie procesu występują niespodziewane zdarzenia takie jak wykrycie defektu produkowanego elementu, czy też jego uszkodzenia. W rzeczywistych procesach produkcyjnych do tego typu zdarzeń zaliczymy m.in nierównomierne nałożenie farby na lakierowany element, złe wyważenie sprężyn w produkowanym materacu. W trakcie procesu zazwyczaj nie da się przewidzieć a priori, kiedy wystąpią te zdarzenia, natomiast można przewidzieć skutki ich występowania np. pojawienie się nowego zadania do przetworzenia z powodu naprawy i konieczność wykonania dodatkowej operacji naprawczej, konieczność powtórzenia operacji lub ciągu operacji dla zadania.

W metodzie przełączania dla produkcji przepływowej z brakami jakościowymi rozważane problemy zapisywane są formalnie wykorzystując metodologię *ALMM*. Formalizm *ALMM*, poprzez mechanizm konstruowania rozwiązania sekwencyjnie podczas realizacji wieloetapowego procesu decyzyjnego MDP, pozwala w przypadku odpowiedniego zamodelowania problemu produkcji przepływowej na wykrycie wystąpienia zdarzenia. W zaproponowanych modelach przedstawionych w rozdziałach 6, 7 i 8, wykrycie zdarzenia następuje dzięki możliwości zamodelowania w stanie systemu stanowiska kontroli jakości. Zaś metoda przełączania tych modeli ma za zadanie umożliwić zmiany w procesie związane z konsekwencjami występowania niedeterministycznych zdarzeń, czyli konieczność odpowiedniej naprawy braku jakościowego wykrytego na stanowisku kontroli jakości.

Na tworzenie metody przełączania, dla problemu produkcji przepływowej z brakami jakościowymi, składają się następujące etapy:

1. analiza problemu niedeterministycznego, analiza zakłóceń
2. podział problemu niedeterministycznego na deterministyczne problemy pomocnicze,
3. budowa modeli *ALM* dla wszystkich wyróżnionych problemów,
4. określenie rodzajów przełączania między modelami,
5. ustalenie reguł przełączania,
6. zdefiniowanie zbioru stanów przełączania,
7. opracowanie algorytmu przełączającego.

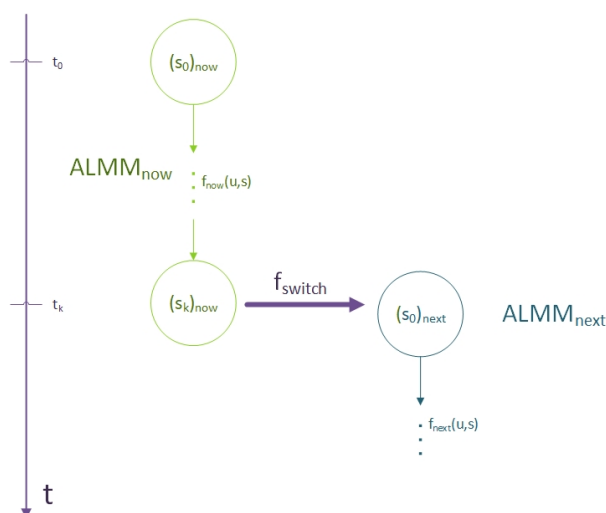
Każdy z tych etapów ma ogromne znaczenie. Analiza problemu ma na celu wskazanie istotnych elementów problemu tak, aby można było wychwycić zdarzenia, w tym przypadku wykrycie braku na stanowisku kontroli jakości. Podział problemu na problemy pomocnicze jest potrzebny w celu odzwierciedlenia możliwych konsekwencji wystąpienia zdarzenia. W tym problemie jest to konieczność odtworzenia procesu naprawczego braku jakościowego. Następnie na podstawie powyższych dwóch punktów należy zbudować modele w taki sposób, by uniknąć redundancji danych. Ostatnie cztery etapy składają się na opracowanie funkcji przełączającej. Określenie rodzajów przełączania ma na celu zdefiniowanie z którego modelu i na który może występować ewentualne przełączenie, a między którymi nie może. Następnie należy wypisać wszystkie reguły przełączania między modelami, wyróżnić stany przełączania i na tej podstawie stworzyć algorytm, który określa jak obliczać stan procesu uwzględniający zaburzenie na podstawie stanu wyróżnionego w zbiorze stanów przełączania w nowym modelu. Innymi słowy tworzone reguły określają w jaki sposób w wyniku zdarzenia aktualny stan procesu produkcyjnego reprezentowany przez s_k aktualnego modelu ALM_{now} przechodzi w kolejny stan procesu reprezentowanego jako stan początkowy s_0 najczęściej w innym modelu ALM_{next} . W nowym, tak wyznaczonym stanie są uwzględnione zmiany, które są konsekwencją zdarzenia. Zbiór tych reguł będzie określał funkcję przełączającą.

W dalszej części rozdziału omówiono szczegółowo etapy metody przełączania.

9.1. Analiza problemu

W trakcie procesu produkcyjnego mogą wystąpić różne zdarzenia, które dotyczą elementów procesu. W literaturze rozważane są zaburzenia dotyczące zarówno zadań jak i maszyn oraz pozostałych zasobów. W wyróżnionych problemach produkcji przepływowej z brakami jakościowymi, z powodu wykrycia braku jakościowego zmiany będą dotyczyły zadań (podział wybrakowanego zadania) i maszyn (zwiększenie zbioru zadań o maszyny naprawcze).

Mianowicie do zmian w zbiorze zadań należą wszelkie zmiany związane z dodaniem lub usunięciem jakiegoś zadania ze zbioru zadań oraz zmiany związane z parametrami konkretnego zadania takimi jak



Rysunek 9.1: Idea metody przełączania

przykładowo czas obróbki na konkretnej maszynie, deadline wykonania, ilość elementów w partii. W rozważanych problemach wyróżniono zmiany związane z dodaniem zadania oraz zmiany ilości elementów w zadaniu. W wyniku negatywnej kontroli jakości, w najprostszym przypadku kontrolowane zadanie dzielone jest na dwa: jedno zadanie składające się z poprawnie wykonanych elementów oraz zadanie do naprawy składające się z elementów z brakami. W związku z tym zwiększamy o jedno zadanie zbiór zadań do wykonania - zadanie z elementami wybrakowanymi jest traktowane jako nowe zadanie. Ponadto w obu zadach powstałych w wyniku podziału określana jest ilość elementów i czasy przetwarzania przez poszczególne maszyny. W ogólnym przypadku zadanie należy podzielić na tyle zadań, ile wykryto typów braków jakościowych podczas jednej kontroli jakości.

Do zmian w zbiorze maszyn należą wszelkie zmiany związane z dodaniem lub usunięciem maszyny ze zbioru rozważanych maszyn oraz zmiany związane z parametrami konkretnej maszyny takimi jak przykładowo wydajność maszyny, czas przebrojenia. W rozważanych problemach wybrakowane zadanie może podlegać naprawie na dodatkowej maszynie lub maszynach, więc rozważane są również zmiany w zbiorze maszyn. Zmiany te polegają na dodawaniu maszyn naprawczych do zbioru maszyn w przypadku wystąpienia wybrakowanych elementów w partii lub usunięciu ze zbioru rozważanych maszyn tej maszyny naprawczej, dla której zostały naprawione wszystkie wybrakowane zadania.

9.2. Rodzaje przełączania

Rodzaje zmian na zbiorach zadań i maszyn wpływają na rodzaje przełączania. Wyróżniono dwa rodzaje przełączeń:

1. Przełączanie pomiędzy instancjami w obrębie jednego problemu
2. Przełączania pomiędzy dwoma problemami

W pierwszym przypadku przełączanie związane jest ze zmianą danych/parametrów określających instancję problemu. Zmiana danych instancji jest konsekwencją uwzględnienia zaburzeń związanych tylko z zadaniami (np. dodanie lub usunięcie zadania ze zbioru zadań). W tym przypadku przy przełączaniu wykorzystywane są modele formalne *ALM* o tej samej strukturze stanu s i wektora decyzji u , natomiast różnią się wartością stanu początkowego s_0 .

W drugim przypadku uwzględniane są również zmiany na zbiorze maszyn. Konsekwencją dodania lub usunięcia maszyny ze zbioru maszyn jest ze zmiana rozważanego problemu. W tym przypadku przy przełączaniu wykorzystywane są modele *ALM* o różnej strukturze stanu uogólnionego s .

9.3. Podział problemu na problemy pomocnicze

Odpowiedni podział wyjściowego niedeterministycznego problemu na problemy pomocnicze jest kluczowy i jest związany z rodzajami zaburzeń. W przypadku problemów produkcji przepływowej z brakami jakościowymi wyjściowy problem obejmuje wszystkie rodzaje zaburzeń i dedykowane dla tych zaburzeń maszyny naprawcze. Problemy pomocnicze konstruowane są w następujący sposób:

- Pierwszy problem jest to klasyczny problem produkcji przepływowej, w której nie rozważa się występowania braków jakościowych, ale jest uwzględniona możliwość ich wykrywania, czyli rozważane jest stanowisko kontroli jakości (w ramach wyróżnionej maszyny M_q lub kilku maszyn z kontrolą jakości M_Q).
- Kolejne problemy są to problemy produkcji przepływowej, w których rozważa się naprawę konkretnych typów braków jakościowych i dedykowane maszyny naprawcze do naprawy tych typów braków.

Zatem podział na problemy pomocnicze ma na celu uwzględnienie niezbędnych zmian w zbiorze maszyn, w szczególności uwzględnienie konkretnej dedykowanej maszyny naprawczej (lub maszyn naprawczych) i uwzględnienie danej maszyny tylko wtedy kiedy jest to potrzebne.

Następnie należy skonstruować odpowiednie modele algebraiczno-logiczne dla problemów pomocniczych.

9.4. Ustalanie reguł przełączania R_{switch}

Reguły przełączania służą do określenia powiązań między wskazanymi problemami pomocniczymi. Innymi słowy reguły przełączania służą do ustalenia w jaki sposób wystąpienie zaburzeń spowoduje zmianę rozważanego w danym momencie modelu algebraiczno-logicznego: zmianę parametrów modelu czy zmian na inny model algebraiczno-logiczny i dla którego problemu pomocniczego.

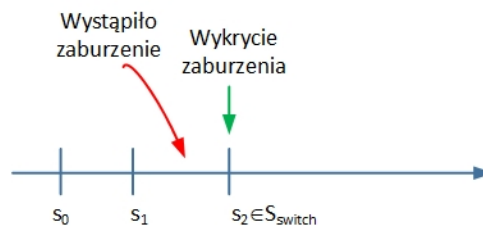
Określenie reguł przełączania sprowadza się do wyznaczenia we wszystkich modelach problemów pomocniczych charakterystycznych stanów, w których może dojść do wykrycia zaburzenia oraz wyznaczenia wszystkich sensownych kombinacji przełączeń między modelami.

Przełączenie modeli polega na tym, żeby stan procesu s , w którym zostanie wykryte zaburzenie zastąpić tak stanem uwzględniającym skutki zaburzenia, by móc kontynuować rozwiązywanie bez konieczności rozpoczynania od nowa. Zatem należy tak skonstruować modele, aby pewny stan s_k aktualnego modelu $ALM_{current}$ przechodził w stan początkowy s_0 nowego modelu ALM_{next} i w nowym stanie uwzględnione zostały zmiany, które są konsekwencją wystąpienia zdarzenia. Czyli funkcja przełączająca przyporządkowuje na podstawie aktualnego stanu systemu, aktualnego modelu i wyniku zdarzenia kolejny stan procesu w odpowiednim modelu, **co ważne bez podejmowania decyzji**.

Dla problemów produkcji przepływowej z brakami jakościowymi rodzaj zaburzenia jest związany z rodzajem defektu oraz koniecznością i sposobem naprawy wybrakowanego elementu.

9.5. Zbiór stanów przełączania \mathcal{S}_{switch}

Zaburzenie jest zdarzeniem wpływającym na zachowanie się systemu. Dla każdego rodzaju zaburzeń musi istnieć stan systemu (w rozważanych modelach), który umożliwi wykrycie tego zaburzenia. Wykrycie zaburzenia może nastąpić tylko w stanach systemu (stanach decyzyjnych) wygenerowanych za pomocą funkcji przejścia modelu algebraiczno-logicznego. W stanach, w których wykryto zaburzenie następuje przełączenie. Nie zawsze moment, w którym wystąpiło zaburzenie jest stanem systemu, w rozumieniu jego modelu algebraiczno-logicznego. W związku z tym jeśli stan, w którym wystąpiło zaburzenie jest stanem niedecyzyjnym (nie jest stanem systemu w rozumieniu modelu), to ten stan utożsamiamy z najbliższym kolejnym stanem decyzyjnym, w którym wykrywa się to zaburzenie. Sytuację tą ilustruje rysunek 9.2. Zbiór stanów decyzyjnych, w których następuje wykrycie zaburzeń będziemy oznaczać jako \mathcal{S}_{switch} oraz będziemy również nazywać wymiennie zbiorem stanów przełączania. Zbiór ten obejmuje wszystkie stany, w których następuje wykrycie zaburzenia we wszystkich rozważanych modelach pomocniczych problemu wyjściowego oraz stany, które są konsekwencją naprawy tych zaburzeń.



Rysunek 9.2: Stany umożliwiające wykrycie zaburzeń dla jednej trajektorii

Zbiór stanów przełączania dla produkcji przepływowej z brakami jakościowymi są to stany, w których pracę zakończy maszyna ze stanowiskiem kontroli jakości z negatywnym wynikiem kontroli oraz stany, w których pracę zakończy maszyna naprawcza i naprawione zadanie wraca na wskazaną maszynę powrotną w marszrucie technologicznej.

9.6. Funkcja przełączająca

Funkcja przełączająca dla metody przełączania dla problemu produkcji przepływowej z brakami jakościowymi jest zdefiniowana następująco:

Definicja 6 (Funkcja przełączająca) *Funkcja przełączająca jest to funkcja, która przyporządkowuje modelowi $ALM \in \mathcal{ALM}$ w aktualnym stanie należącym do zbioru stanów przełączania oraz danym i parametrom instancji problemu pomocniczego odpowiedni model (ze zdefiniowanym stanem początkowym) oraz dane i parametry odpowiedniej instancji problemu pomocniczego:*

$$f_{switch} : \mathcal{ALM} \times \mathcal{S}_{switch} \times \mathcal{D} \rightarrow \mathcal{ALM} \times \mathcal{D}. \quad (9.1)$$

Funkcja przełączająca jest zdefiniowana za pomocą dwóch funkcji:

$$f_{switch} = (f_{switch|_{\mathcal{ALM}}}, f_{switch|_{\mathcal{D}}}), \quad (9.2)$$

gdzie:

$f_{switch|_{\mathcal{ALM}}} : \mathcal{ALM} \times \mathcal{S}_{switch} \times \mathcal{D} \rightarrow \mathcal{ALM}$ określa następny model, czyli następny problem pomocniczy

$f_{switch|_{\mathcal{D}}} : \mathcal{ALM} \times \mathcal{S}_{switch} \times \mathcal{D} \rightarrow \mathcal{D}$ określa następny zbiór danych i parametrów dla instancji problemu pomocniczego.

Zatem możemy napisać

$$f_{switch|_{\mathcal{ALM}}}(ALM_{now}, s_{know}, D_{now}) = ALM_{next} \quad (9.3)$$

$$f_{switch|_{\mathcal{D}}}(ALM_{now}, s_{know}, D_{now}) = D_{next} \quad (9.4)$$

Funkcja przełączająca w odróżnieniu od funkcji przejścia służy do wyznaczenia kolejnego stanu systemu nie na podstawie poprzedniego stanu i podjętej decyzji, tylko na podstawie stanu systemu i wykrytego zaburzenia oraz służy do uwzględnienia na podstawie stanu i wykrytego zaburzenia zmian danych lub parametrów problemu **bez podejmowania decyzji w tym stanie**.

Ogólnie można zatem stwierdzić, że **Metoda przełączania dla problemów produkcji przepływowej** (ang. *Switching Method (SM) for FS*) polega na określeniu piątki $((\mathcal{D}, \mathcal{ALM}, R_{switch}, \mathcal{S}_{switch}, f_{switch}))$, gdzie:

- \mathcal{D} - zbiór danych problemu i parametrów
- \mathcal{ALM} - zbiór modeli rozważanych podproblemów,
- R_{switch} - zbiór reguł przełączania między modelami algebraiczno-logicznymi
- $\mathcal{S}_{switch} = (X_{switch}, T_{switch})$ jest zbiorem stanów, w których dochodzi do przełączania,
- f_{switch} jest funkcją przełączającą.

W związku z powyższym algorytm przełączania dla klasy problemów produkcji przepływowej z brakami jakościowymi można przedstawić w następujący sposób:

KROK 1 Wyznaczenie wszystkich składowych metody przełączania, czyli określenie:

- danych wejściowych \mathcal{D}_A dla pierwszego problemu pomocniczego. Czyli określenie zbioru zadań \mathcal{J}_A , zbioru maszyn \mathcal{M}_A oraz ich parametrów,
- modeli algebraiczno-logicznych problemów pomocniczych oraz ustalenie ALM_A modelu pierwszego problemu pomocniczego,
- reguł przełączania R_{switch}
- zbioru stanów przełączania S_{switch}
- funkcji przełączania f_{switch} .

KROK 2 Ustalenie aktualnych danych \mathcal{D}_{now} i aktualnego modelu ALM_{now} .

KROK 3 Podstawienie pod stan systemu stanu początkowego aktualnego modelu $(s_{now})_0$.

KROK 4 Wyznaczenie następnego stanu systemu s' za pomocą funkcji przejścia aktualnego modelu $s' = f(u, s)$.

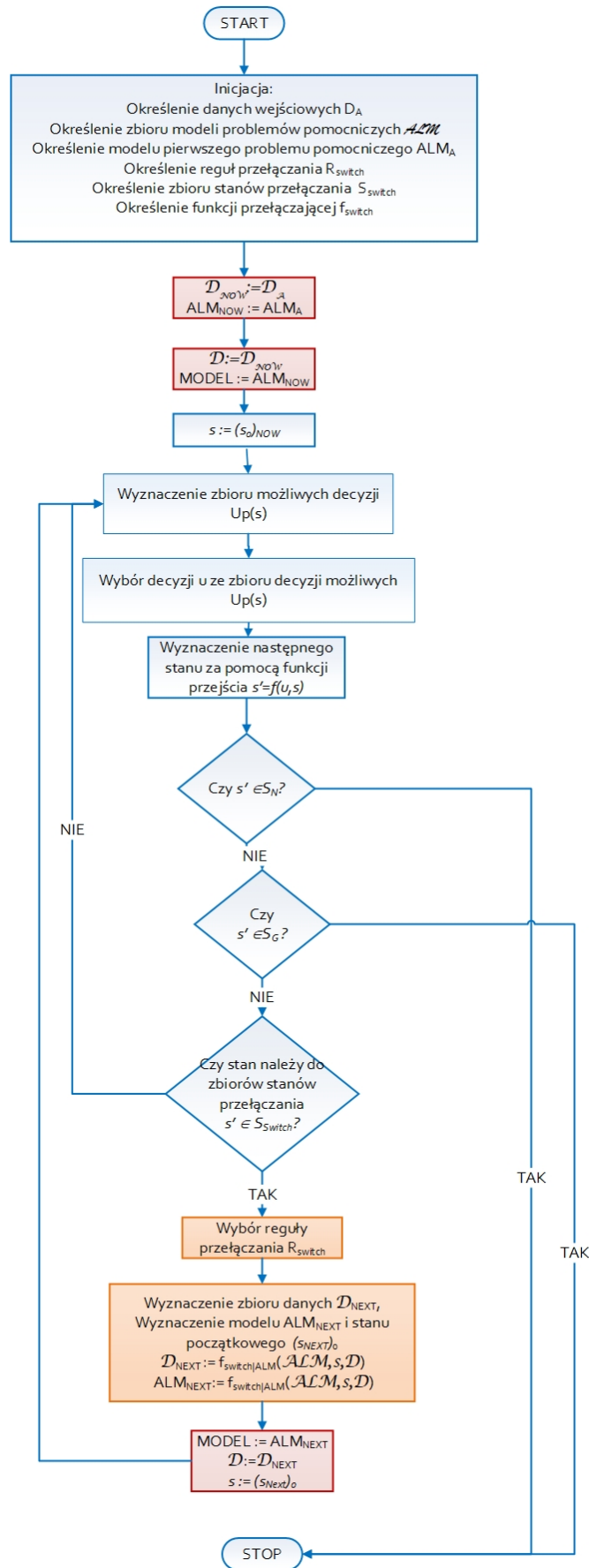
KROK 5 Sprawdzenie, czy stan należy do zbioru stanów przełączenia $s \in S_{switch}$. Jeśli nie, to wyznaczanie kolejnych stanów według KROKU 4. Jeśli tak, przejście do kolejnego KROKU 6.

KROK 6 Wybór reguły przełączenia R_{switch} .

KROK 7 Na podstawie funkcji przejścia wyznaczenie nowych danych \mathcal{D}_{next} , modelu ALM_{next} oraz stanu początkowego nowego modelu $(s_{next})_0$. Następnie należy przejść do KROKU 4.

Schemat algorytmu przedstawiono na rysunku 9.3

Natomiast, w celu rozwiązania problemu należy zastosować jedną z opisanych w rozdziale 3. Są to techniki bazujące na schemacie ALMM służące do wyboru decyzji.



Rysunek 9.3: Algorytm metody przełączania dla problemu produkcji przepływowej z brakami jakościowymi

Podsumowanie

W niniejszej pracy doktorskiej zaproponowano zastosowanie meta modelu algebraiczno-logicznego do modelowania i optymalizacji niedeterministycznego problemu dyskretnej produkcji przepływowej, w której występują defekty elementów produkowanych i konieczna jest ich naprawa. Do tej pory meta model algebraiczno-logiczny był stosowany przede wszystkim do rozwiązywania wieloetapowych dynamicznych problemów produkcyjnych i logistycznych bez uwzględniania niespodziewanych zakłóceń w podstawowym procesie technologicznym. W rozważanych problemach takim zakłóceniem jest nieudany proces technologiczny i konieczność zidentyfikowania oraz naprawienia źle wykonanego zadania. Takie i inne dodatkowe zakłócenia często występują w rzeczywistych problemach planowania produkcji i są coraz częstszym przedmiotem badań naukowych. Ogólna koncepcja modelowania zakłóceń za pomocą metody przełączania modeli algebraiczno-logicznych, która została zaproponowana przez E. Dudek-Dyduch i przedstawiona po raz pierwszy w pracy Grobler-Dębska et al. (2013). E. Dudek-Dyduch zaproponowała w pracach Dudek-Dyduch (2015a,b) dwu-etapową metodę transformacji modeli algebraiczno-logicznych (*2SALMT*) (*ang. Two-Stage AL Model Transformation Method*). Metoda ta należy do metod sztucznej inteligencji i jej idea polega na tym, że w momencie wystąpienia zaburzenia w automatyczny sposób dokonywana jest transformacja modeli. Jest to ogólna metoda, która służy do modelowania problemów produkcji dyskretnej z różnego rodzaju niedeterministycznymi zakłóceniami.

Za najważniejsze osiągnięcia autorka uważa się:

- przeanalizowane problemów z ponownym przetwarzaniem zadań (problemy re-entrant) wynikającego ze zdefiniowanego procesu produkcyjnego lub konieczności ponownego przetwarzania zadania z powodu niespełnienia kryterium jakości oraz wyróżnienie nowej klasy problemów, w których występują dodatkowe maszyny naprawcze (spoza linii technologicznej);
- przeanalizowanie problemów szeregowania zadań w produkcji przepływowej z brakami jakościowymi, uwzględniającą cechy procesów wykrywania braków jakościowych oraz naprawy wybrakowanych elementów oraz wyodrębnienie trzech rodzajów maszyn specjalnych: maszyn z kontrolą jakości \mathcal{M}_Q , maszyn naprawczych (dodatkowych) \mathcal{M}_D oraz maszyn powrotnych \mathcal{M}_R ;
- wyróżnienie pięciu klas problemów produkcji przepływowej ze względu na ilość i rodzaj maszyn specjalnych: $FS-M_qM_r$ problem z jedną maszyną kontroli jakości, brakiem maszyny naprawczej oraz jedną maszyną powrotną 5.2.1, $FS-M_qM_dM_r$ problem z jedną maszyną kontroli jakości, jedną maszyną naprawczą oraz jedną maszyną powrotną 5.2.2, $FS-M_qmM_dM_r$ - problem z jedną maszyną kontroli jakości, więcej niż jedną maszyną naprawczą oraz jedną maszyną powrotną

5.2.3, $FS-M_qmM_d m M_r$ problem z jedną maszyną kontroli jakości, więcej niż jedną maszyną naprawczą oraz więcej niż jedną maszyną powrotną 5.2.4, $FS-mM_qmM_d m M_r$ problem z więcej niż jedną maszyną z kontrolą jakości, więcej niż jedną maszyną naprawczą oraz więcej niż jedną maszyną powrotną 5.2.5;

- modyfikacja i opracowanie nowych modeli algebraiczno-logicznych wybranych problemów produkcji przepływowej z ograniczeniami czasowymi;
- opracowanie metody modelowania produkcji z brakami jakościowymi za pomocą metody przełączania modeli algebraiczko-logicznej dla klasy problemów produkcji przepływowej z brakami jakościowymi, zawierającej następujące elementy:
 - analiza zakłóceń
 - podział problemu na problemy pomocnicze
 - ustalenie reguł przełączania
 - definicja zbiorów stanów przełączania
 - konstruowanie funkcji przełączającej
- opracowanie metody przełączania wyróżnionych problemów produkcji przepływowej z brakami jakościowymi 5.2.1-5.2.5 uwzględniające specyficzne cechy problemów wynikające z rodzajów i konfiguracji maszyn specjalnych;

Tematyka szeregowania zadań z uwzględnieniem występujących zaburzeń jest obecnie intensywnie rozwijającą się dziedziną. Niestety niedostatkim w proponowanych rozwiązaniach jest odzwierciedlenie wyników gospodarczych systemu planowania tylko z klasycznymi miarami wydajności, takimi czas wykonania czy opóźnienie. Brakuje nadal rozwiązań, które uwzględniałyby uwzględnienie kosztów, jakie są ponoszone w związku z wprowadzaniem częstych zmian w harmonogramie (przykładowo dodatkowe koszty konfiguracji, koszty przeładunku materiałów, koszty przechowywania). Jest to ciekawa dziedzina badań, ważna z punktu widzenia efektywności i rentowności przedsiębiorstw. Będzie to przedmiotem moich dalszych prac.

A. Skorowidz ważniejszych oznaczeń

OZNACZENIA UŻYWANE W TEORII SZEREGOWANIA ZADAŃ

$\mathcal{J} = \{J_1, \dots, J_n\}$ - zbiór zadań do wykonania

$\mathcal{M} = \{M_1, \dots, M_m\}$ - zbiór maszyn, które będą wykonywać zadania

\mathcal{R} - zbiór dodatkowych zasobów potrzebnych do wykonywania tych zadań

φ - funkcja celu

J_j - j -te zadanie ze zbioru zadań

M_i - i -ta maszyna ze zbioru maszyn

n_j - liczba operacji w zadaniu

O_{ij} - i -ta operacja zadania o indeksie j

p_{ij} - czas wykonywania operacji O_{ij} (i -tej operacji zadania J_j)

r_j - najwcześniejszy możliwy termin rozpoczęcia wykonywania zadania (moment gotowości do wykonania)

d_j - pożądaný czas zakończenia wykonywania zadania (due date)

\bar{d}_j - termin krytyczny zakończenia wykonania zadania (deadline). Zadanie J_j musi zostać zakończone przed upływem tego terminu

w_j - waga (priorytet), którą można interpretować jako koszt oczekiwania na wykonanie zadania J_j w ciągu jednostki czasu.

c_{kj} - koszt przebrojenia pomiędzy zadaniem J_k a zadaniem J_j . Koszt ten, analogicznie do czasu przebrojeń, również może być sekwencyjnie zależny lub sekwencyjnie niezależny.

s_{kj} - czas przebrojenia (przygotowania maszyny) pomiędzy zadaniem J_k a zadaniem J_j . Czasy takich przebrojeń

v_{ij} - sposób wykonania i -tej operacji zadania J_j (przydzielenie do maszyn)

s_{ij} - termin rozpoczęcia wykonywania i -tej operacji zadania J_j

S_j – termin rozpoczęcia wykonywania zadania

C_j – termin zakończenia wykonywania zadania

L_j – nieterminowość zakończenia zadania

T_j – spóźnienie zakończenia zadania

E_j – przyspieszenie rozpoczęcia zadania

F_j – czas przepływu zadania przez system

W_j – czas przestoju zadania przy przepływie przez system

U_j – jednostkowe spóźnienie zadania.

C_{max} – czas zakończenia wykonania wszystkich zadań (długość uszeregowania)

L_{max} - maksymalna nieterminowość

T_{max} - maksymalne opóźnienie

c_{max} - maksymalny koszt wykonania zadania

$\sum w_j C_j$ – suma ważonych czasów zakończenia wykonania zadań

$\sum w_j T_j$ - suma ważonych opóźnień

$\sum c_j$ – całkowity koszt wykonania zadań

$\sum w_j U_j$ – suma ważonych opóźnień zadań

OZNACZENIA UŻYTE DO OPISU PROBLEMU FLOW SHOP

$\mathcal{J} = \{J_1, \dots, J_n\}$ - zbiór zadań do wykonania

$\mathcal{M} = \{M_1, \dots, M_{md}\}$ - zbiór maszyn, które będą wykonywać zadania

\mathcal{R} - zbiór dodatkowych zasobów potrzebnych do wykonywania tych zadań

φ - funkcja celu

J_j - j -te zadanie ze zbioru zadań

M_i - i -ta maszyna ze zbioru maszyn

(M_1, M_2, \dots, M_m) - ciąg maszyn ułożonych w marszrutę technologiczną,

$\mathcal{M}_Q \subset \mathcal{M}$ - zbiór maszyn z kontrolą jakości,

$\mathcal{M}_{\mathcal{D}} = \{M_{m+1}, \dots, M_{md}\}$, $\mathcal{M}_{\mathcal{D}} \cap \mathcal{M} = \emptyset$ zbiór maszyn dodatkowych, służących do naprawy wadliwych elementów,

p_{ij} - czas wykonywania operacji O_{ij} (i -tej operacji zadania J_j)

d_j - pożądany czas zakończenia wykonywania zadania (due date)

\bar{d}_j - termin krytyczny zakończenia wykonania zadania (deadline)

OZNACZENIA UŻYTE DO ALMM

P - proces dyskretny

U - zbiór decyzji sterujących,

$S = X \times T$ - zbiór stanów uogólnionych,

X - zbiór stanów właściwych,

$T \subset \mathbb{R}^+ \cup \{0\}$ - podzbiór nieujemnych liczb rzeczywistych reprezentujących chwile czasowe,

$s_0 = (x, t)$ - uogólniony stan początkowy $s_0 \in S$,

$f : U \times S \rightarrow S$ - jest funkcja przejścia

$f_x : U \times X \times T \rightarrow X$ - funkcja określająca następny stan właściwy,

$f_t : U \times X \times T \rightarrow T$ - funkcja określająca następny moment czasu,

$S_N \subset S$ - zbiór uogólnionych stanów niedopuszczalnych,

$S_G \subset S$ - zbiór uogólnionych stanów docelowych,

$S_E \subset S$ - zbiór stanów uogólnionych, dla których funkcja przejścia f jest nieokreślona

$U_p(s)$ - zbiór decyzji możliwych w danym stanie,

$u(s)$ - decyzja wybrana do realizacji w stanie s

\tilde{u} - ciąg decyzji wyznaczający trajektorię

\tilde{U} - zbiór wszystkich ciągów decyzyjnych wyznaczających trajektorie

\tilde{U}_d - podzbiór dopuszczalnych ciągów sterowań

s' - następny po stanie s stan uogólniony trajektorii

\tilde{s} - trajektoria procesu

\tilde{S} - zbiór wszystkich trajektorii procesu

$x(s)$ - stan właściwy danego stanu s

x^i - składowa stanu systemu

x^i dla $i \in \{1, \dots, m\}$ - stan magazynów międzyoperacyjnych

x^{m+i} dla $i \in \{1, \dots, m\}$ - stan maszyn z marszrutu technologicznej

x^{2m+1} - stan magazynu przed maszyną naprawczą

x^{2m+2} - stan maszyny naprawczej

$\beta \in \{0, 1, \dots, n\}$ - współrzędna stanu maszyny oznaczająca numer zadania, które jest aktualnie obrabiane na danej maszynie (0 oznacza, że żadne zadanie nie zostało przydzielone do maszyny),

$\tau \in [0, \infty)$ - współrzędna stanu maszyny oznaczająca czas do ukończenia obróbki aktualnego zadania na maszynie (∞ oznacza, że maszyna nie pracuje).

β^i - oznacza wartość współrzędnej β dla poszczególnej i -tej maszyny

τ^i - oznacza wartość współrzędnej τ dla poszczególnej i -tej maszyny

Q - kryterium jakości

Q_k - częściową wartość kryterium Q , obliczoną dla K -tego stanu rozważanej trajektorii

ΔQ_k - przyrost kryterium Q w k -tym stanie ustalonej trajektorii procesu P

$\Delta Q(u, x, t)$ - przyrost wartości wskaźnika jakości w wyniku podjętej decyzji u w stanie (x, t)

$\widehat{Q}(u, x, t)$ - oszacowanie wartości wskaźnika jakości końcowego odcinka trajektorii po zrealizowaniu decyzji u

$q_i(u, x, t)$ - składniki kryterium lokalnego, odzwierciedlające dodatkowe ograniczenia lub dodatkowe wymagania w przestrzeni stanów

\mathcal{ALM} - zbiór modeli rozważanych problemów pomocniczych,

\mathcal{S}_{switch} - zbiór stanów uogólnionych, w których dochodzi do przełączania,

\mathcal{R}_{switch} - zbiór reguł przełączania

f_{switch} funkcja przełączająca

$\mathcal{D} = (\mathcal{J}, \mathcal{M}, \mathcal{R}, \varphi)$ oznacza zbiór danych i parametrów rozważanych instancji problemów pomocniczych

B. Skorowidz ważniejszych nazw

- Wieloetapowy proces decyzyjny - Multistage decision process (MDP)
- Dyskretny proces produkcyjny - Discrete manufacturing process (DMP)
- Meta-model algebraiczno-logiczny (ALMM) - Algebraic-logical meta-models
- Metoda zadań zastępczych (MZZ)- Substitution tasks method (STM)
- Metoda gromadzenia informacji dla potrzeb sterowania (GIPS) - Gathering information during search process method (GISP)
- Metoda uczenia na bazie GIPS - Learning method based on GISP
- Metoda przełączania modeli algebraiczno-logicznych dla problemów produkcji przepływowej - Switching method of ALM models for flow shop system

C. Dowód twierdzenia

Dla problemu $Fm|prmu, p_{ij} = p_j|C_{max}$, w którym dla danego zadania J_j czas przetwarzania przez każdą z maszyn jest taki sam, znane jest rozwiązanie optymalne. Twierdzenie, to podane jest w pracy Pinedo (2008) bez dowodu:

Twierdzenie 2 Dla $Fm|prmu, p_{ij} = p_j|C_{max}$ minimalny czas wykonywania wszystkich zadań jest niezależny od uszeregowania i wynosi

$$C_{max} = \sum_{j=1}^n p_j + (m - 1) \max(p_1, \dots, p_n).$$

Poniżej przedstawiono propozycję dowodu:

Dowód: Dane jest n zadań i m maszyn. Rozważmy uszeregowane tych zadań rosnąco według czasów przetwarzania p_j . Takie uszeregowanie jest optymalne, ponieważ przetwarzanie każdego zadania w całym ciągu technologicznym będzie odbywać się płynnie, bez przestojów. Inaczej mówiąc żadne zadanie, po zakończeniu przetwarzania na danej maszynie, nie będzie czekać na rozpoczęcie przetwarzania na kolejnej maszynie.

Niech τ_{ij} oznacza czas rozpoczęcia przetwarzania zadania J_j na maszynie M_i . Rozważmy czasy rozpoczęcia zadań dla kolejnych maszyn.

Dla maszyny pierwszej czasy rozpoczęcia przetwarzania zadań wynoszą:

$$\tau_{11} = 0$$

$$\tau_{12} = p_1$$

$$\tau_{13} = p_1 + p_2$$

...

$$\tau_{1n} = \sum_{k=1}^{n-1} p_k$$

Dla drugiej maszyny czasy rozpoczęcia przetwarzania zadań wynoszą:

$$\tau_{21} = p_1$$

$$\tau_{22} = p_1 + p_2$$

$$\tau_{23} = p_1 + p_2 + p_3$$

...

$$\tau_{2n} = \sum_{k=1}^n p_k$$

Dla trzeciej maszyny czasy rozpoczęcia przetwarzania zadań wynoszą:

$$\tau_{31} = p_1 + p_1$$

$$\tau_{32} = p_1 + p_2 + p_2$$

$$\tau_{33} = p_1 + p_2 + p_3 + p_3$$

...

$$\tau_{3n} = \sum_{k=1}^n p_k + p_n$$

W konsekwencji dla maszyny M_m czasy rozpoczęcia przetwarzania zadań wynoszą:

$$\tau_{m1} = p_1 + (m - 2)p_1$$

$$\tau_{m2} = p_1 + p_2 + (m - 2)p_2$$

$$\tau_{m3} = p_1 + p_2 + p_3 + (m - 2)p_3$$

...

$$\tau_{mn} = \sum_{k=1}^n p_k + (m - 2)p_n$$

Stąd C_{max} czas zakończenia ostatniego zadania J_n przez wszystkie maszyny wynosi:

$$\tau_{mn} + p_n = \sum_{k=1}^n p_k + (m - 2)p_n + p_n = \sum_{k=1}^n p_k + (m - 1)p_n = C_{max}$$

A ponieważ uszeregowanie było rosnące, to $p_n = \max(p_1, \dots, p_n)$. Stąd optymalna wartość wynosi

$$C_{max} = \sum_{j=1}^n p_j + (m - 1)\max(p_1, \dots, p_n).$$

Spis rysunków

2.1	Systematyka problemów szeregowania. Opracowanie własne na podstawie Karkula (2013).	28
5.1	Przykład z maszyną z kontrolą jakości, maszyną powrotną oraz bez maszyny naprawczej	56
5.2	Przykład z maszyną z kontrolą jakości, maszyną powrotną oraz z maszyną naprawczą . .	56
5.3	Przykład z maszyną z kontrolą jakości, maszyną powrotną oraz dwoma maszynami naprawczymi	57
5.4	Przykład z maszyną z kontrolą jakości oraz dwiema maszynami powrotnymi i naprawczymi	58
5.5	Przykład z dwiema maszynami z kontrolą jakości	59
6.1	Algorytm przełączania dla problemu produkcji przepływowej z jedną maszyną kontroli jakości i bez maszyn naprawczych	69
7.1	Algorytm przełączania dla problemu produkcji przepływowej z naprawą braków na jednej maszynie naprawczej.	82
8.1	Przykład linii produkcyjnej z maszyną z kontrolą jakości, maszyną powrotną oraz dwoma różnymi maszynami naprawczymi.	85
8.2	Algorytm przełączania dla problemu $FS-M_qmM_dM_r$	93
8.3	Przykład linii z maszyną z kontrolą jakości oraz dwiema maszynami powrotnymi i naprawczymi	94
8.4	Przykład linii z dwiema maszynami z kontrolą jakości	97
9.1	Idea metody przełączania	105
9.2	Stany umożliwiające wykrycie zaburzeń dla jednej trajektorii	107
9.3	Algorytm metody przełączania dla problemu produkcji przepływowej z brakami jakościowymi	110

Bibliografia

- M. Abdolzadeh and H. Rashidi. An approach of cellular learning automata to job shop scheduling problem. *International Journal of Simulation: Systems, Science and Technology*, 11(2):56–64, 2010.
- C. Almeder and R. Hartl. A metaheuristic optimization approach for a real-world stochastic flexible flow shop problem with limited buffer. *International Journal of Production Economics*, 145(1):88–95, 2013. doi: 10.1016/j.ijpe.2012.09.014.
- A. Antczak, P. Antczak, and T. Witkowski. Using of evolving cellular automata for flexible job shop with makespan criterion. *Proceedings of the 5th IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS'2009*, pages 297–302, 2009. doi: 10.1109/IDAACS.2009.5342977.
- H. Aytug, M. Lawley, K. McKay, S. Mohan, and R. Uzsoy. Executing production schedules in the face of uncertainties: A review and some future directions. *European Journal of Operational Research*, 161(1):86–110, 2005. doi: 10.1016/j.ejor.2003.08.027.
- K. R. Baker. *Introduction to Sequencing and Scheduling*. Wiley, New York, 1974. ISBN 9780471045557.
- J. Behnamian. Survey on fuzzy shop scheduling. *Fuzzy Optimization and Decision Making*, 15(3): 331–366, 2016. ISSN 1573-2908. doi: 10.1007/s10700-015-9225-5.
- J. Blazewicz, K. H. Ecker, E. Pesch, G. Schmidt, and J. Weglarz. *Scheduling computer and manufacturing processes*. Springer, Berlin, Heidelberg, 2001. doi: <https://doi.org/10.1007/978-3-662-04363-9>.
- C. Blum and M. Sampels. An ant colony optimization algorithm for shop scheduling problems. *Journal of Mathematical Modelling and Algorithms*, 3(3):285–308, 2004. doi: 10.1023/B:JMMA.0000038614.39977.6f.
- W. Bożejko and J. Pempera. *Optymalizacja dyskretna w informatyce, automatyce i robotyce*. Oficyna Wydawnicza Politechniki Wrocławskiej, 2012. ISBN 9788374937436. URL <https://books.google.pl/books?id=LhzpnQEACAAJ>.
- P. Brucker. *Scheduling algorithms*. Springer, Berlin, Heidelberg, 2007. doi: 10.1007/978-3-540-69516-5.
- K. M. M. A. Bukkur, M. Shukri, and O. M. E. Elmardi. A review for dynamic scheduling in manufacturing. *Global Journal of Research In Engineering*, 2018. ISSN 2249-4596.

- K. Chakhlevitch and C. Glass. Scheduling reentrant jobs on parallel machines with a remote server. *Computers & Operations Research*, 36(9):2580 – 2589, 2009. ISSN 0305-0548. doi: <https://doi.org/10.1016/j.cor.2008.11.007>.
- S. Choi and K. Wang. Flexible flow shop scheduling with stochastic processing times: A decomposition-based approach. *Computers and Industrial Engineering*, 63(2):362–373, 2012. doi: 10.1016/j.cie.2012.04.001.
- Y. Chu, F. You, and J. Wassick. Integrated planning and scheduling under production uncertainties: Bi-level model formulation and hybrid solution method. *Computers & Chemical Engineering*, 72: 255–272, 2015.
- K. Chudzik and A. Janiak. Algorytm tabu search dla wybranych problemów szeregowania zadań na pojedynczej maszynie z przebrojeniami. *Zeszyty Naukowe. Automatyka / Politechnika Śląska*, z. 123: 97–109, 1998.
- L. Danping and C. Lee. *A review of the research methodology for the re-entrant scheduling problem*, volume 49. 2011. doi: 10.1080/00207541003720350.
- C. Desprez, F. Chu, and C. Chu. Minimising the weighted number of tardy jobs in a hybrid flow shop with genetic algorithm. *International Journal of Computer Integrated Manufacturing*, 22(8):745–757, 2009. doi: 10.1080/09511920902810938.
- E. Dudek-Dyduch. Formal bases of classification of discrete production processes control problems, zeszyty naukowe agh nr 1313. *Automatyka*, 49(8):189–200, 1989.
- E. Dudek-Dyduch. Formalizacja i analiza problematyki dyskretnych procesów produkcyjnych. *Zeszyty Naukowe AGH, Automatyka*, 54, 1990.
- E. Dudek-Dyduch. Wieloetapowe procesy decyzyjne w optymalizacji sterowania dyskretnych procesów produkcyjnych. *Zeszyty Naukowe AGH z.*, 59:173–180, 1991.
- E. Dudek-Dyduch. Discrete determinable processes - compact knowledge-based model. *Notas de Matematica*, 137, 1993.
- E. Dudek-Dyduch. Learning-based algorithms in scheduling. *Journal of Intelligent Manufacturing*, 11 (2):135–143, 2000. doi: 10.1023/A:1008934615349.
- E. Dudek-Dyduch. Modeling manufacturing processes with disturbances - a new method based on algebraic-logical meta-models. In *ICAISC proceedings 14th International Conference Part II*, pages 353–363, 2015a.
- E. Dudek-Dyduch. Modeling manufacturing processes with disturbances - two-stage al model transformation method. In *20th International Conference on Methods and Models in Automation and Robotics*, pages 782–787. MMAR proceedings, 2015b. doi: 10.1109/MMAR.2015.7283975.

- E. Dudek-Dyduch. Algebraic logical meta-model of decision processes - new metaheuristics. In *14th International Conference, ICAISC proceedings, Part I*, pages 541–554, 2015c.
- E. Dudek-Dyduch. Intelligent almm system for discrete optimization problems—the idea of knowledge base application. In *International Conference on Information Systems Architecture and Technology*, pages 3–12. Springer, 2017.
- E. Dudek-Dyduch and L. Dutkiewicz. Substitution tasks method for discrete optimization. In *Artificial Intelligence and Soft Computing: 12th International Conference, ICAISC proceedings*, 2013.
- E. Dudek-Dyduch and T. Dyduch. Scheduling some class of discrete processes. In *Proc. of 12th IMACS World Congress*, Paris, 1988.
- E. Dudek-Dyduch and T. Dyduch. Intelligent search algorithm in scheduling. *Cybernetics and Systems*, pages 1228–1232, 1996.
- E. Dudek-Dyduch and T. Dyduch. Learning algorithms for scheduling using knowledge based model. In L. Rutkowski, R. Tadeusiewicz, L. Zadeh, and J. M. Żurada, editors, *ICAISC 2006. LNCS (LNAI)*, pages 1091–1100. Springer-Verlag, Heidelberg, 2006a.
- E. Dudek-Dyduch and T. Dyduch. Hybrid learning method for discrete manufacturing control using knowledge based model. In *Proc. of the Third Int Conf. on Informatics Control, Automation and Robotics*, pages 160–166, Setubal, Portugal, 2006b.
- E. Dudek-Dyduch and S. Fuchs-Seliger. Approximate algorithms for some tasks in management and economy. *System, Modeling, Control*, 7(1), 1993.
- E. Dudek-Dyduch and E. Kucharska. Learning method for co-operation. In P. Jędrzejowicz and K. Nguyen, N Hoang, editors, *ICCCI 2011, Part II. LNCS*, pages 290–300, Heidelberg, 2011a. Springer-Verlag.
- E. Dudek-Dyduch and E. Kucharska. Optimization learning method for discrete process control. *ICINCO*, 1:24–33, 2011b.
- E. Dudek-Dyduch and J. Wąs. Knowledge representation of pedestrian dynamics in crowd: formalism of cellular automata. In *ICAISC 2006: 8th International Conference, Zakopane, Poland, J*, pages 1101–1110. Springer, Berlin Heidelberg, 2006.
- E. Dudek-Dyduch, L. Dutkiewicz, and E. Kucharska. Formalny model symulacji procesów decyzyjnych jako model algebraiczno-logiczny. *Zeszyty naukowe Politechniki Białostockiej*, 2005.
- E. Dudek-Dyduch, Z. Gomolka, B. Twarog, and E. Zesławska. Intelligent almm system-implementation assumptions for its knowledge base. In *ITM Web of Conferences*, volume 21, page 00002. EDP Sciences, 2018.

- E. Dudek-Dyduch, Z. Gomolka, B. Twarog, and E. Zeslowska. The concept of the almm solver knowledge base retrieval using protégé environment. In *International Conference on Dependability and Complex Systems*, pages 177–185. Springer, 2019.
- F. Dugardin, F. Yalaoui, and L. Amoleo. *New multi-objective method to solve reentrant hybrid flow shop scheduling problem*. European Journal of Operational Research, 2010.
- L. Dutkiewicz. *Dwupoziomowe algorytmy optymalizacji procesów wytwarzania z zasobami zależnymi od stanu systemu*. PhD thesis, Rozprawa doktorska 2005, 2005.
- L. Dutkiewicz and E. Dudek-Dyduch. Substitution tasks method for co-operation. In A. Badica, B. Trawinski, and N. Nguyen, editors, *Recent Developments in Computational Collective Intelligence*, pages 103–113. Springer, Heidelberg, 2014.
- L. Dutkiewicz and E. Kucharska. Metody optymalizacyjne oparte na ogólnym schemacie modelu algebraiczno-logicznego. *Pomiary, Automatyka, Robotyka*, 15:178–182, 2011.
- L. Dutkiewicz, E. Kucharska, K. Rączka, and K. Grobler-Dębska. St method-based algorithm for the supply routes for multilocation companies problem. In J. Kacprzyk, editor, *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, pages 123–135. 2016.
- K. Fang, N. A. Uhan, F. Zhao, and J. W. Sutherland. Flow shop scheduling with peak power consumption constraints. *Annals of Operations Research*, 206(1):115–145, 2013.
- M. Flasiński. *Wstęp do sztucznej inteligencji*. Wydawnictwo Naukowe PWN, Warszawa, 2011.
- E. C. Freuder. Modeling: The final frontier. In *Proceedings PACLP99 the 1st International Conference on the Practical Applications of Constraint Technologies and Logic Programming*, 15-21, 1999.
- S. Gawiejnowicz. *Time-Dependent Scheduling*. Springer-Verlag, Berlin, 2008.
- E. González-Neira, J. Montoya-Torres, and D. Barrera. Flow-shop scheduling problem under uncertainties: Review and trends. *International Journal of Industrial Engineering Computations*, 8(4):399–426, 2017.
- J. Grabowski, E. Nowicki, and C. Smutnicki. *Metoda blokowa w zagadnieniach szeregowania zadań*. Akademicka Oficyna Wydawnicza EXIT,, 2003.
- R. L. Graham, E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annals of Discrete Mathematics*, 5:287–326, 1979.
- K. Grobler-Dębska, E. Kucharska, and E. Dudek-Dyduch. Idea of switching algebraic-logical models in flow-shop scheduling problem with defects. In *The 18th International Conference on Methods and Models in Automation and Robotics MMAR proceedings*, pages 532–537, 2013.

- K. Grobler-Dębska, E. Kucharska, and M. Jagodziński. Almm-based switching method for fss problem with defects. In *The 19th International Conference on Methods and Models in Automation and Robotics MMAR proceedings*, pages 814–819, 2014.
- J. N. Gupta and E. F. Stafford. Flowshop scheduling research after five decades. *European Journal of Operational Research*, 169(3):699 – 711, 2006. ISSN 0377-2217. doi: <https://doi.org/10.1016/j.ejor.2005.02.001>.
- M. Hekmatfar, S. F. Ghomi, and B. Karimi. Two stage reentrant hybrid flow shop with setup times and the criterion of minimizing makespan. *Applied Soft Computing*, 11(8):4530 – 4539, 2011. ISSN 1568-4946. doi: <https://doi.org/10.1016/j.asoc.2011.08.013>.
- R. Hinze. A lot streaming model for a re-entrant flow shop scheduling problem with missing operations. In D. Mattfeld, T. Spengler, J. Brinkmann, and M. Grunewald, editors, *Logistics Management*, pages 149–158, Cham, 2016. Springer International Publishing. ISBN 978-3-319-20863-3.
- R. Hinze and D. Sackmann. An iterated local search for a re-entrant flow shop scheduling problem. In M. Lübbecke, A. Koster, P. Letmathe, R. Madlener, B. Peis, and G. Walther, editors, *Operations Research Proceedings*, pages 221–226, Cham, 2014. Springer International Publishing. ISBN 978-3-319-28697-6.
- R. Hinze, D. Sackmann, U. Buscher, and G. Aust. A contribution to the reentrant flow-shop scheduling problem. *IFAC Proceedings*, 46(9):718–723, 2013.
- S. Horng, S. Lin, and S. Yang. Evolutionary algorithm for stochastic job shop scheduling with random processing time. *Expert Systems with Applications*, 39:3603–3610, 2012.
- C. Hyun-Seon. Scheduling algorithms for two-stage reentrant hybrid flow shops: minimizing makespan under the maximum allowable due dates. *The International Journal of Advanced Manufacturing Technology*, 2009.
- A. Janiak. *Wybrane problemy i algorytmy szeregowania zadań i rozdziału zasobów*. Akademicka Oficyna Wydawnicza PLJ, Warszawa, 1999.
- A. Janiak, T. Krysiak, and M. Lichtenstein. Szeregowanie zadań z przezbrotzeniami i czasami wykonywania zadań zależnymi od zasobów - minimalizacja sumy ważonych czasów zakończenia wykonywania zadań. *Zeszyty Naukowe. Automatyka / Politechnika Śląska*, z. 134:209–218, 2002.
- S. M. Johnson. Optimal two- and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, 1(1):61–68, 1954.
- J. Kacprzyk and J. Węglarz, editors. *Modelowanie i optymalizacja : metody i zastosowania*. Akademicka Oficyna Wydawnicza EXIT, Warszawa, 2002.
- A. H. G. R. Kan. *Machine Scheduling Problems: Classification, Complexity and Computations*. Springer Science & Business Media, The Hague, 2012.

- M. Karkula. *Modelowanie i symulacja procesów logistycznych*. Wydawnictwa AGH, Kraków, 2013.
- K. Katragini, E. Vallada, and R. Ruiz. Rescheduling flowshops under simultaneous disruptions. In *2015 International Conference on Industrial Engineering and Systems Management (IESM)*, pages 84–91, Oct 2015. doi: 10.1109/IESM.2015.7380141.
- E. Kucharska. *Wykorzystanie modelu algebraiczno-logicznego do optymalizacji problemów szeregowania z czasem przebrojeń zależnym od stanu*. PhD thesis, AGH, 2006.
- E. Kucharska and E. Dudek-Dyduch. Extended learning method for designation of cooperation. In N. Nguyen, editor, *TCCI XIV 2014. LNCS*,, pages 136–157. Springer, Heidelberg, 2014.
- E. Kucharska, K. Grobler-Dębska, K. Rączka, and L. Dutkiewicz. Cellular automata approach for parallel machine scheduling problem. *SIMULATION*, 92(2):165–178, 2016. doi: 10.1177/0037549715625120.
- E. Kucharska, K. Grobler-Dębska, and K. Rączka. Almm-based methods for optimization makespan flow-shop problem with defects. In *Information Systems Architecture and Technology: Proceedings of 37th International Conference on Information Systems Architecture and Technology - ISAT 2016 - Part I*, pages 41–53, 2017a.
- E. Kucharska, K. Grobler-Dębska, and K. Rączka. Algebraic-logical meta-model based approach for scheduling manufacturing problem with defects removal. *Advances in Mechanical Engineering*, 9(4): 1–18, 2017b.
- E. Kucharska, K. Grobler-Dębska, and R. Klimek. Collective decision making in dynamic vehicle routing problem. In *MATEC Web of Conferences*, volume 252, page 03003. EDP Sciences, 2019.
- J. Kwiecień. *Algorytmy stadne w rozwiązywaniu wybranych zagadnień optymalizacji dyskretnej i kombinatoryczn*. Wydawnictwa AGH, 2015, 2015. ISBN 8374648376.
- C. K. M. Lee, L. Danping, W. Ho, and W. Zhang. Design of a genetic algorithm for bi-objective flow shop scheduling problems with reentrant job. *The International Journal of Advanced Manufacturing Technology*, 56:1105–1113, 2011.
- Z. C. Li, B. Qian, R. Hu, and X. H. Zhu. A hybrid population-based incremental learning algorithm for m-machine reentrant permutation flow-shop scheduling. *Advanced Materials Research*, 2013.
- Z. Lian, X. Gu, and B. Jiao. A similar particle swarm optimization algorithm for permutation flowshop scheduling to minimize makespan. *Applied Mathematics and Computation*, 175(1):773 – 785, 2006. ISSN 0096-3003. doi: <https://doi.org/10.1016/j.amc.2005.07.042>.
- C.-J. Liao, C.-T. Tseng, and P. Luarn. A discrete version of particle swarm optimization for flowshop scheduling problems. *Computers & Operations Research*, 34(10):3099–3111, 2007.
- D. Lin, C. Lee, and Z. Wu. Integrated ga and ahp for re-entrant flow shop scheduling problem. *2011 IEEE International Conference on Quality and Reliability, ICQR 2011*, pages 496–500, 2011. doi: 10.1109/ICQR.2011.6031769.

- J. T. Lin and C. M. Chen. Simulation optimization approach for hybrid flow shop scheduling problem in semiconductor back-end manufacturing. *Simulation Modelling Practice and Theory*, 51:100–114, 2015.
- L. Liu. Outsourcing and rescheduling for a two-machine flow shop with the disruption of new arriving jobs: A hybrid variable neighborhood search algorithm. *Computers & Industrial Engineering*, 130: 198 – 221, 2019. ISSN 0360-8352. doi: <https://doi.org/10.1016/j.cie.2019.02.015>.
- M. Nouiri, A. Bekrar, A. Jemai, D. Trentesaux, A. C. Ammari, and S. Niar. Two stage particle swarm optimization to solve the flexible job shop predictive scheduling problem considering possible machine breakdowns. *Computers & Industrial Engineering*, 112:595 – 606, 2017. ISSN 0360-8352. doi: <https://doi.org/10.1016/j.cie.2017.03.006>.
- J. M. Novas and G. P. Henning. Integrated scheduling of resource-constrained flexible manufacturing systems using constraint programming. *Expert Systems with Applications*, 41:2286–2299, 2014.
- D. Ouelhadj and S. Petrovic. A survey of dynamic scheduling in manufacturing systems. *Journal of Scheduling*, 12(4):417, Oct 2008. ISSN 1099-1425. doi: 10.1007/s10951-008-0090-8.
- Q.-K. Pan, M. F. Tasgetiren, P. N. Suganthan, and T. J. Chua. A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem. *Information sciences*, 181(12):2455–2468, 2011.
- J. Perl. *Heuristics: Intelligent Search Strategies for Computer Problem Solving*. Addison-Wesley Publishing Company, 1984.
- M. Pinedo. *Scheduling theory, algorithms, and systems*,. Prentice Hall, New Jersey, USA, 2008.
- J. F. Puget. Constraint programming next challenge: Simplicity of use. In M. Wallace, editor, *Principles and Practice of Constraint Programming - CP 2004, LNCS*. Springer, 2004.
- P. Pujo, M. Pedetti, and N. Giambiasi. Formal dev's modelling and simulation of a flow-shop relocation method without interrupting the productio, simulation modelling practice and theory. *Elsevier*, 14: 817–842, 2006.
- U. Ritzinger, J. Puchinger, and R. F. Hartl. A survey on dynamic and stochastic vehicle routing problems. *International Journal of Production Research*, 54(1):215–231, 2016. doi: 10.1080/00207543.2015.1043403. URL <https://doi.org/10.1080/00207543.2015.1043403>.
- D. A. Rossit, F. Tohmé, and M. Frutos. The non-permutation flow-shop scheduling problem: A literature review. *Omega*, 77:143 – 153, 2018. ISSN 0305-0483. doi: <https://doi.org/10.1016/j.omega.2017.05.010>.
- T. Sawik. *Badania operacyjne dla inżynierów zarządzania*. Wydawnictwo AGH, Kraków, 1998.
- F. Serebinski and A. Y. Zomaya. Sequential and parallel cellular automata-based scheduling algorithms. *IEEE Transactions on Parallel and Distributed Systems*, 13(10):1009–1023, Oct 2002. ISSN 1045-9219. doi: 10.1109/TPDS.2002.1041877.

- D. Shabtay. The just-in-time scheduling problem in a flow-shop scheduling. *European Journal of Operational Research*, pages 521–532, 2012.
- J. Shen and Y. Zhu. Uncertain flexible flow shop scheduling problem subject to breakdowns. *Journal of Intelligent & Fuzzy Systems*, 32(1):207–214, 2017.
- C. Smutnicki. *Algorytmy szeregowania*. Akademicka Oficyna Wydawnicza EXIT, Warszawa, 2002.
- A. Swiecicka and F. Serebinski. Cellular automata approach to scheduling problem. In *Proceedings International Conference on Parallel Computing in Electrical Engineering. PARELEC 2000*, pages 29–33, Aug 2000. doi: 10.1109/PCEE.2000.873596.
- A. Swiecicka, F. Serebinski, and A. Y. Zomaya. Multiprocessor scheduling and rescheduling with use of cellular automata and artificial immune system support. *IEEE Transactions on Parallel and Distributed Systems*, 17(3):253–262, March 2006. ISSN 1045-9219. doi: 10.1109/TPDS.2006.38.
- M. Sysło, N. DEO, and J. KOWALIK. *Algorytmy optymalizacji dyskretnej*. Warsaw: Wydawnictwo Naukowe PWN., 1995.
- D. Tang, M. Dai, M. A. Salido, and A. Giret. Energy-efficient dynamic scheduling for a flexible flow shop using an improved particle swarm optimization. *Computers in Industry*, 81:82 – 95, 2016. ISSN 0166-3615. doi: <https://doi.org/10.1016/j.compind.2015.10.001>.
- G. E. Vieira, J. W. Herrmann, and E. Lin. Rescheduling manufacturing systems: A framework of strategies, policies, and methods. *Journal of Scheduling*, 6(1):39–62, Jan 2003. ISSN 1099-1425. doi: 10.1023/A:1022235519958.
- N. Yalaoui, M. Camara, L. Amodeo, F. Yalaoui, and H. Mahdi. New heuristic for scheduling re-entrant production lines. In *2009 International Conference on Computers Industrial Engineering*, pages 199–204, July 2009. doi: 10.1109/ICCIE.2009.5223776.
- B. Zeigler. *Teoria modelowania i symulacji*. Wydawnictwo Naukowe PWN, Warszawa, 1984.
- B.-h. Zhou, L.-m. Hu, and Z.-y. Zhong. A hybrid differential evolution algorithm with estimation of distribution algorithm for reentrant hybrid flow shop scheduling problem. *Neural Computing and Applications*, 30(1):193–209, Jul 2018. ISSN 1433-3058. doi: 10.1007/s00521-016-2692-y.

OŚWIADCZENIE AUTORA PRACY

OŚWIADCZAM, ŚWIADOMY ODPOWIEDZIALNOŚCI KARNEJ ZA POŚWIADCZENIE NIEPRAWDY, ŻE NINIEJSZĄ PRACĘ DYPLOMOWĄ WYKONAŁEM OSOBIŚCIE I SAMODZIELNIE, I NIE KORZYSTAŁEM ZE ŹRÓDEŁ INNYCH NIŻ WYMIENIONE W PRACY.

.....

PODPIS