BALANSOWANIE OBCIĄŻEŃ JEDNOSTEK SEKCYJNYCH

Tomasz PRIMKE

Streszczenie: Złożony problem konfiguracji wariantów gotowości może zostać rozwiązany poprzez dekompozycję na prostsze podproblemy. Jednym z takich podproblemów jest problem balansowania obciążeń jednostek sekcyjnych. W pracy został opisany model matematyczny tego problemu. Opisany został również algorytm heurystyczny LPT-split, bazujący na znanym z literatury algorytmie LPT.

Słowa kluczowe: algorytmy optymalizacji dyskretnej, zarządzanie zmianami organizacyjnymi, programowanie całkowitoliczbowe, szeregowanie zadań.

1. Wstęp

Problem optymalizacji konfiguracji wariantów gotowości (KWG) został opisany w [3], [2] i uogólniony w [1]. W problemie tym mamy do czynienia z systemem produkcyjnym składającym się z wielu jednakowych jednostek, będących elastycznymi modułami produkcyjnymi. W systemie tym musimy zrealizować pewne procesy produkcyjne. Zakładamy, że dla każdego procesu *on* mamy daną prognozę wielkości produkcji *Zk(on)*.

Zazwyczaj nie jest możliwa realizacja wszystkich procesów bez przezbrajania jednostek produkcyjnych. W problemie KWG zakładamy, że wszystkie jednostki będą przezbrajane jednocześnie. Sposób przezbrojenia jednostek, przypisanie tych jednostek do poszczególnych operacji w procesach, a także pogrupowanie jednostek (oraz procesów) w sekcje nazywane jest wariantem gotowości. Problem optymalizacji KWG polega na doborze takich wariantów, aby czas realizacji wszystkich procesów był jak najmniejszy.

Do rozwiązywania problemu KWG w [3] zaproponowano algorytm ewolucyjny CAE-KWG. W algorytmie tym dokonano dekompozycji złożonego problemu optymalizacji KWG na wiele podproblemów. Znane prognozy *Zk(on)* dzielone są na plany sekcyjne *Zoj(on)*. Tworzenie sekcji w wariancie gotowości w algorytmie CAE-KWG [3] polega na przydzielaniu do sekcji kolejnych planów sekcyjnych. Z każdym planem sekcyjnym związany jest pewien proces produkcyjny. Do sekcji przydzielane są również jednostki produkcyjne (nazywane w odniesieniu do sekcji jednostkami sekcyjnymi) w taki sposób, aby realizacja wszystkich procesów w sekcji była możliwa. Tak utworzone sekcje grupuje się następnie w warianty gotowości.

Tematem niniejszego artykułu jest problem balansowania obciążeń jednostek sekcyjnych. Polega on na przypisaniu jednostek do operacji w sekcjach w taki sposób, aby obciążenie sekcji było jak najmniejsze.

2. Sformułowanie problemu

W algorytmie CAE-KWG, po określeniu planów sekcyjnych oraz przydzieleniu jednostek do sekcji, trzeba przypisać poszczególne jednostki sekcyjne do realizacji wszystkich operacji w procesach. Nie jest to zadanie proste do rozwiązania i zostało podzielone na dwa podproblemy. W pierwszej kolejności jednostki sekcyjne zostają przypisane do operacji. Następnie określa się, w ramach jakich konkretnie procesów sekcyjnych będą one realizowane. W niniejszym punkcie zostanie opisany ten pierwszy podproblem.

Warto jest zwrócić uwagę na pewien fakt. W sekcji realizowane są pewne procesy, które składają się z operacji. Do tych operacji przypisane są jednostki sekcyjne. W pierwszym etapie, będącym właśnie problemem balansowania obciążeń jednostek sekcyjnych, bierze się pod uwagę tylko operacje. Decyzje dotyczące tego, w ramach jakiego procesu produkcyjnego ma zostać wykonana operacja na danej jednostce, zostają podjęte w kolejnym etapie algorytmu CAE-KWG.

Konsekwencją takiej dekompozycji tego problemu jest sumowanie na poszczególnych jednostkach planów operacji, a nie procesów, we wzorze (2). Jest to różnica w stosunku do modeli problemu KWG opisanych w [3] i [1]. Plany produkcyjne, a właściwie prognozy, są jednak określone dla procesów. Dlatego w opisywanym problemie należy wyznaczyć plany produkcyjne *Zo(o)* dla wszystkich rodzajów operacji $o \in \boldsymbol{O_s}$ zgodnie z następującym wzorem:

$$Zo(o) = \sum_{on \in \boldsymbol{ON}_s} Zoj(on) \cdot Lop(o,on) \tag{1}$$

gdzie: $\boldsymbol{ON_s}$ – zbiór procesów produkcyjnych, które mają zostać zrealizowane w sekcji
$\boldsymbol{O_s}$ – zbiór rodzajów operacji, które mają zostać zrealizowane w sekcji
Lop(o,on) – liczba wykonań operacji rodzaju o przypadająca na jednokrotne wykonanie procesu on

Zbiór $\boldsymbol{O_s}$ jest oczywiście wyznaczany na podstawie zbioru $\boldsymbol{ON_s}$, w oparciu o znany w problemie KWG podział procesów produkcyjnych na operacje.

Dany jest więc zbiór rodzajów operacji $\boldsymbol{O_s}$ oraz zbiór jednostek $\boldsymbol{J_s}$. Dla każdego rodzaju operacji $o \in \boldsymbol{O_s}$ dany jest czas wykonywania *To(o)* oraz (wspomniany wcześniej) plan produkcji *Zo(o)*. Plan ten może zostać zrealizowany na wszystkich jednostkach $j \in \boldsymbol{J_s}$. Dla każdej jednostki określona jest maksymalna liczba rodzajów operacji *OJM(j)*, do których ta jednostka może zostać przypisana.

Oznaczmy plan wykonań operacji rodzaju *o* na jednostce *j* przez *Zo(o,j)*. Aby wszystkie plany produkcji *Zo(o)* zostały wykonane, musi być spełniony warunek:

$$\sum_{j \in \boldsymbol{J_s}} Zo(o,j) = Zo(o), \quad dla\ o \in \boldsymbol{O_s} \tag{2}$$

przy czym każda zmienna $Zo(o,j) \geq 0$.

Jednocześnie żadna jednostka *j* nie może zostać przypisana do większej liczby rodzajów operacji, niż *OJM(j)*:

$$\sum_{o \in \boldsymbol{O_s}} B(o,j) = OJM(j), \quad dla\ j \in \boldsymbol{J_s} \tag{3}$$

przy czym

$$B(o,j) = \begin{cases} 0 & gdy\ Zo(o,j) = 0 \\ 1 & gdy\ Zo(o,j) > 0 \end{cases} \tag{4}$$

Warunki (2) i (3), przy uwzględnieniu dziedzin zmiennych decyzyjnych *Zo(o,j)* oraz *B(o,j)*, pozwalają wyznaczyć rozwiązanie dopuszczalne problemu balansowania obciążeń jednostek sekcyjnych. Zależy nam jednak na uzyskaniu możliwie najmniejszego obciążenia całej sekcji. W tym celu wprowadzamy do problemu zmienne *Xj(j)*, których wartości będą równe obciążeniom (czasowym) jednostek *j* i zmienne *Xj(o,j)*, których wartości będą równe obciążeniom jednostek *j* konkretnym rodzajem operacji *o*.

Związek pomiędzy zmiennymi *Xj(o,j)* i zmiennymi *Zo(o,j)* jest następujący:

$$Xj(o,j) = Zo(o,j) \cdot To(o), \quad dla\, o \in \boldsymbol{O_s}\ i\ j \in \boldsymbol{J_s} \tag{5}$$

Obciążenie całej jednostki można wyrazić jako sumę obciążeń tej jednostki poszczególnymi rodzajami operacji:

$$Xj(j) = \sum_{o \in \boldsymbol{O_s}} Xj(o,j), \quad dla\ j \in \boldsymbol{J_s} \tag{6}$$

Problem balansowania jednostek sekcyjnych polega na znalezieniu takich wartości zmiennych decyzyjnych *Zo(o,j)* (oraz powiązanych z nimi zmiennych *B(o,j)* i *Xj(o,j)*), aby obciążenie sekcji określone wzorem:

$$X_s = \max_{j \in \boldsymbol{J_s}} Xj(j) \tag{7}$$

było jak najmniejsze.

Podsumowując, w problemie balansowania obciążeń jednostek sekcyjnych dane są:

- zbiór rodzajów operacji $\boldsymbol{O_s}$
- zbiór jednostek sekcyjnych $\boldsymbol{J_s}$

Dla każdego rodzaju operacji $o \in \boldsymbol{O_s}$ dane są:

- plan produkcyjny *Zo(o)*
- czas wykonywania *To(o)*

Ponadto dla każdej jednostki produkcyjnej $j \in \boldsymbol{J_s}$ dana jest maksymalna liczba operacji *OJM(j)*, do której jednostka może zostać przypisana.

Zmiennymi decyzyjnymi są:

- plany wykonań operacji rodzaju *o* na jednostce *j Zo(o,j)*, dla $o \in \boldsymbol{O_s}$ i $j \in \boldsymbol{J_s}$
- zmienne pomocnicze (binarne) *B(o,j)*, dla $o \in \boldsymbol{O_s}$ i $j \in \boldsymbol{J_s}$
- obciążenia (czasowe) jednostek poszczególnymi rodzajami operacji *Xj(o,j)*, dla $o \in \boldsymbol{O_s}$ i $j \in \boldsymbol{J_s}$
- obciążenia jednostek *Xj(j)*, dla $j \in \boldsymbol{J_s}$
- obciążenie sekcji X_s

3. Algorytm LPT-Split

Aby rozwiązać problem balansowania obciążeń jednostek sekcyjnych, w [3] zaproponowano dwa różne algorytmy. Jednym z nich był algorytm LPT-split, który zostanie dokładnie opisany w tym artykule.

Opisany w poprzednim punkcie problem jest podobny do problemów szeregowania zadań na maszynach równoległych. Najważniejszą różnicą jest występowanie ograniczenia na maksymalną liczbę rodzajów operacji *OJM(j)*, do jakiej można przypisać daną jednostkę *j*. Z tego względu nie można wykorzystać znanych z literatury algorytmów listowych bez ich uprzedniej modyfikacji.

W algorytmie listowym LPT, w każdym kroku wybiera się operację o największym czasie wykonywania i przydziela się ją do najmniej obciążonej jednostki. W problemie balansowania obciążeń jednostek sekcyjnych za odpowiednik czasu wykonywania operacji rodzaju *o* można przyjąć wartość *To(o)* · *Zo(o)*, natomiast obciążenie jednostki sekcyjnej *j* zostało oznaczone przez *Xj(j)*. W pierwszej fazie algorytmu LPT-split wykorzystuje się te podobieństwa, aby wstępnie przypisać jednostki do operacji, podobnie jak w znanym z literatury algorytmie LPT.

Celem drugiej fazy algorytmu jest lepszy podział obciążenia operacjami poszczególnych jednostek sekcyjnych, przy równoczesnym spełnieniu warunku (3). Wszystkie kroki tej fazy powtarzane są w pętli, aż do momentu zakończenia algorytmu.

Krok 1: Wyznacz jednostkę j_{max}, cechującą się największym obciążeniem w sekcji.

W dalszych krokach będziemy dążyli do zmniejszenia obciążenia jednostki j_{max} kosztem zwiększenia obciążenia innej jedostki w sekcji (oznaczmy ją przez j_x). Trzeba będzie zrobić to w taki sposób, aby obciążenie sekcji się zmniejszyło.

Krok 2: Utwórz zbiór pomocniczy $\boldsymbol{ZJ_X}$.

Zbiór $\boldsymbol{ZJ_X}$ powinien zawierać te jednostki, które można przypisać do jednej z operacji przypisanych do jednostki j_{max}, zmniejszając obciążenie sekcji. Zbiór ten można zdefiniować następująco:

$$\boldsymbol{ZJ_X} = \left\{ j \in \boldsymbol{J_s} : j \neq j_{max} \wedge Xj(j) + To_{min} < Xj(j_{max}) \right\} \tag{8}$$

przy czym

$$To_{min} = \min_{o \in \boldsymbol{O}(j_{max})} To(o) \tag{9}$$

gdzie: ***O**(j)* – zbiór rodzajów operacji, do których została przypisana jednostka *j*

Chcąc zmniejszyć obciążenie jednostki j_{max}, jednocześnie zwiększając obciążenie innej jednostki j_x, trzeba przypisać jednostkę j_x do jednej z tych operacji, do których jest przypisana jednostka j_{max}. Obciążenie $Xj(j_{max})$ zmniejszy się przynajmniej o wartość $To(o_x)$ dla wybranej operacji $o_x \in \boldsymbol{O}(j_{max})$. Wartość To_{min} określa więc najmniejszą możliwą zmianę obciążenia jednostki j_{max}. Ponieważ w wyniku zmiany obciążeń obydwu jednostek obciążenie całej sekcji powinno się zmniejszyć, obciążenie $Xj(j_x)$ powinno być mniejsze od obciążenia $Xj(j_{max})$ przynajmniej o wartość $To_{min} + 1$.

Krok 3: Utwórz zbiory pomocnicze $\boldsymbol{ZJ_O}$ oraz $\boldsymbol{ZJ_{OJM}}$.

Istnieją tylko dwie możliwości zmiany obciążeń jednotki j_{max} i innej, wybranej jednostki j_x. Spośród wszystkich rodzajów operacji, do których jest przypisana jednostka j_{max}, można wybrać taki, do którego jest również przypisana jednostka j_x. Zbiór wszystkich jednostek *j* spełniających taki warunek zdefiniowany jest następująco:

$$\boldsymbol{ZJ_O} = \{ j \in \boldsymbol{J_s} : j \neq j_{max} \wedge \exists_{o \in \boldsymbol{O}(j)} o \in \boldsymbol{O}(j_{max}) \} \tag{10}$$

Drugą możliwością jest wybór takiego rodzaju operacji o_x, do którego jednostka j_x nie jest przypisana. W tym przypadku trzeba pamiętać o spełnieniu warunku (3). Zbiór wszystkich jednostek j spełniających ten warunek zdefiniowany jest następująco:

$$\boldsymbol{ZJ}_{\boldsymbol{OJM}} = \{ j \in \boldsymbol{J}_{\boldsymbol{s}} : j \neq j_{max} \wedge \sum_{o \in \boldsymbol{O}_{s}} B(o, j) < OJM(j) \} \tag{11}$$

Znając zbiory $\boldsymbol{ZJ_X}$, $\boldsymbol{ZJ_O}$ oraz $\boldsymbol{ZJ_{OJM}}$, można wyznaczyć zbiór $\boldsymbol{ZJ}$:

$$\boldsymbol{ZJ} = \boldsymbol{ZJ}_{\boldsymbol{X}} \cap (\boldsymbol{ZJ}_{\boldsymbol{O}} \cup \boldsymbol{ZJ}_{\boldsymbol{OJM}}) \tag{12}$$

Jeśli zbiór $\boldsymbol{ZJ}$ nie jest zbiorem pustym, to znaczy, że można wybrać taki rodzaj operacji $o_x \in \boldsymbol{O}(j_{max})$ i taką jednostkę $j_x \in \boldsymbol{ZJ}$, że obciążenie $Xj(j_{max})$ zostanie zmniejszone kosztem zwiększenia obciążenia $Xj(j_x)$ przy jednoczesnym zmniejszeniu obciążenia całej sekcji.

W przypadku gry zbiór $\boldsymbol{ZJ}$ jest zbiorem pustym, należy zakończyć algorytm.

Krok 4: Wybierz jednostkę j_x.

Jako jednostkę j_x wybiera się najmniej obciążoną jednostkę ze zbioru ZJ. Zmieniając obciążenie jednostek j_{max} oraz j_x najlepiej byłoby doprowadzić do sytuacji, w której obciążenia tych jednostek byłyby sobie równe. Optymalna zmiana wartości obciążeń wynosi więc

$$dX_{opt} = \frac{Xj(j_{max}) - Xj(j_x)}{2} \tag{13}$$

Krok 5: Wybierz operację o_x.

Znając już j_x oraz wartość dX_{opt}, można dokonać wyboru takiego rodzaju operacji $o_x \in \boldsymbol{O}(j_{max})$, aby ostateczna zmiana obciążenia tych jednostek była jak najbardziej zbliżona do wartości dX_{opt}.

Najlepsza byłaby oczywiście operacja takiego rodzaju o, dla którego obciążenie $Xj(j_{max},o) = dX_{opt}$. Definiujemy więc zbiór $\boldsymbol{O_{opt}}$:

$$\boldsymbol{O}_{\boldsymbol{opt}} = \{ o \in \boldsymbol{O}(j_{max}) : Xj(o, j_{max}) = dX_{opt} \} \tag{14}$$

Dodatkowo definiujemy zbiór $\boldsymbol{O_x}$:

$$\boldsymbol{O}_{\boldsymbol{x}} = \boldsymbol{O}_{\boldsymbol{opt}} \cap \boldsymbol{O}(j_x) \tag{15}$$

Jeśli zbiór $\boldsymbol{O_x}$ zawiera jakieś rodzaje operacji, to wybieramy dowolne $o_x \in \boldsymbol{O_x}$. Gdy zbiór $\boldsymbol{O_x}$ jest zbiorem pustym, to jednostkę j_x można przypisać do dowolnej operacji $o_x \in \boldsymbol{O_{opt}}$ tylko wówczas, gdy

$$OJM(j_x) > \sum_{o \in \boldsymbol{O}_s} B(o, j_x) \tag{16}$$

Znając rodzaj operacji o_x, jej obciążenie dX_{opt}, oraz jednostki j_{max} i j_x, możemy przypisać jednostkę j_x do operacji o_x, odciążając jednostkę j_{max} i odpowiednio modyfikując wartości zmiennych decyzyjnych $Xj(o_x,j_{max})$, $Xj(o_x,j_x)$, $Zo(o_x,j_{max})$ oraz $Zo(o_x,j_x)$. Po wykonaniu tych zmian należy powrócić do kroku pierwszego.

W sytuacji, gdy nie udało się wybrać żadnego rodzaju operacji, tworzymy zbiór $\boldsymbol{O}_+$:

$$\boldsymbol{O}_+ = \{o \in \boldsymbol{O}(j_{max}) : Xj(o, j_{max}) > dX_{opt}\} \tag{17}$$

Następnie wybieramy taki rodzaj operacji $o_x \in \boldsymbol{O}+$, dla którego obciążenie $Xj(o_x,j_{max})$ jest największe. Jeśli jednostki j_x nie można przypisać do operacji rodzaju o_x, usuwamy o_x ze zbioru $\boldsymbol{O}_+$. Powtarzając te czynności doprowadzimy do sytuacji, w której albo będziemy znali rodzaj operacji o_x, do którego można przypisać jednostkę j_x, albo zbiór $\boldsymbol{O}_+$ będzie zbiorem pustym.

W przypadku, gdy uda się wyznaczyć odpowiedni rodzaj operacji o_x, pozostaje wyznaczyć wartość obciążenia dXj_x, o które zmienią się obciążenia $Xj(j_{max})$ oraz $Xj(j_x)$. Wartość dXj_x będzie związana ze zmianą planów $Zo(o_x,j_{max})$ i $Zo(o_x,j_x)$ o wartość dZo_x:

$$\boldsymbol{ZO}_{\boldsymbol{x}} = \{z \in \boldsymbol{N} : To(o_x) \cdot z > dX_{opt}\}$$

$$dZox = To(o_x) \cdot \min_{i \in \boldsymbol{ZO}_x} i \tag{18}$$

gdzie: $\boldsymbol{N}$ – zbiór liczb naturalnych

Oczywiście $dXj_x = dZo_x \cdot To(o_x)$.

Po odpowiednim zmodyfikowaniu wartości zmiennych decyzyjnych $Xj(o_x,j_{max})$, $Xj(o_x,j_x)$, $Zo(o_x,j_{max})$ i $Zo(o_x,j_x)$, algorytm należy wykonać ponownie od kroku pierwszego.

W sytuacji, gdy zbiór $\boldsymbol{O}_+$ jest zbiorem pustym, tworzymy zbiór $\boldsymbol{O}_-$:

$$\boldsymbol{O}_- = \{o \in \boldsymbol{O}(j_{max}) : Xj(o, j_{max}) < dX_{opt}\} \tag{19}$$

Następnie wybieramy taki rodzaj operacji $o_x \in \boldsymbol{O}_-$, dla którego obciążenie $Xj(o_x,j_{max})$ jest największe. Jeśli jednostki j_x nie można przypisać do operacji rodzaju o_x, usuwamy o_x ze zbioru $\boldsymbol{O}_-$. Powtarzając te czynności doprowadzimy do sytuacji, w której albo będziemy znali rodzaj operacji o_x, do którego można przypisać jednostkę j_x, albo zbiór $\boldsymbol{O}_-$ będzie zbiorem pustym.

W przypadku, gdy uda się wyznaczyć odpowiedni rodzaj operacji o_x, pozostaje wyznaczyć wartość obciążenia dXo_x, o które zmienią się obciążenia $Xj(o_x,j_{max})$ oraz $Xj(o_x,j_x)$. Wartość ta będzie równa po prostu $Xj(o_x,j_{max})$. Wartości zmiennych decyzyjnych $Zo(o_x,j_{max})$ i $Zo(o_x,j_x)$ trzeba będzie zmienić w taki sposób, aby warunek (5) był spełniony. Po wykonaniu tych zmian cały algorytm należy powtórzyć od kroku pierwszego.

Gdy zbiór $\boldsymbol{O}_-$ będzie zbiorem pustym, należy usunąć jednostkę j_x ze zbioru $\boldsymbol{ZJ}$ i powrócić do kroku czwartego.

4. Podsumowanie

W literaturze znanych jest wiele problemów klasy szeregowania zadań na maszynach równoległych. Problemu balansowania obciążeń jednostek sekcyjnych nie można zaliczyć do tej klasy. Rozwiązując takie zadanie nie mamy do czynienia z układaniem harmonogramu, gdyż nie odpowiadamy na pytania kiedy mają zostać wykonane poszczególne operacje.

Istnieje jednak dużo podobieństw pomiędzy balansowaniem obciążeń jednostek sekcyjnych i problemami klasy szeregowania zadań na maszynach równoległych, w których za kryterium optymalizacji przyjęto długość uszeregowania. Wielkości tej w oczywisty sposób odpowiada obciążenie sekcji. Ponadto długość uszeregowania nie zależy od kolejności wykonywania operacji na maszynach, a tylko od tego, jakie maszyny zostały przypisane do jakich operacji.

Istotną różnicą pomiędzy problemem balansowania obciążeń jednostek sekcyjnych a problemami szeregowania zadań na maszynach równoległych opisywanymi w literaturze jest ograniczenie (3) (które zostało podane w bardziej ogólnej postaci w [1]). Ze względu na występowanie tego ograniczenia, nie jest możliwe zastosowanie algorytmów heurystycznych opisywanych w literaturze do rozwiązywania problemu będącego przedmiotem niniejszego artykułu bez wprowadzenia odpowiednich modyfikacji. Z tego względu podjęto próbę opracowania algorytmu LPT-split.

Praca finansowana ze środków przewidzianych na BK-218/RAu1/2009 t.5.

Literatura

1. Zaborowski M.: Sterowanie nadążne zasobami przedsiębiorstwa. Wydawnictwo Pracowni Komputerowej Jacka Skalmierskiego, Gliwice, 2008.
2. Zaborowski M., Primke T.: Optymalizacja konfiguracji wariantów gotowości w pewnym systemie produkcyjnym. W: Knosala R. (red.) „Komputerowo Zintegrowane Zarządzanie", Oficyna Wydawnicza Polskiego Towarzystwa Zarządzania Produkcją, Opole 2008, tom II, str. 595-603.
3. Primke T.: Analiza efektywności algorytmu ewolucyjnego w wybranych problemach sterowania dyskretnymi procesami produkcji. Rozprawa doktorska (nieopublikowana), Politechnika Śląska, Wydział Automatyki, Elektroniki i Informatyki, Gliwice 2008.

Dr inż. Tomasz PRIMKE
Instytut Automatyki
Politechnika Śląska
44-100 Gliwice, ul. Akademicka 16a
tel./fax.: (0-32) 237 28 99
e-mail: Tomasz.Primke@polsl.pl