mgr inż. Irena Bach
dr inż. Grzegorz Bocewicz
prof. dr hab. inż. Zbigniew Banaszak
Katedra Podstaw Informatyki i Zarządzania
Politechnika Koszalińska

# PLANOWANIE PRZEDSIĘWZIĘĆ W ŚRODOWISKU WIELOPROJEKTOWYM

*Planowanie produkcji w środowisku wieloprojektowym, typowym dla przedsiębiorstw klasy MSP, wymaga wsparcia ze strony interaktywnych, dedykowanych systemów wspomagania decyzji. Przedstawione podejście, implementujące techniki programowania z ograniczeniami, ilustruje możliwości budowy tego typu systemów zorientowanych na wybrany zbiór pytań rutynowych, pytań odnoszących się do warunków spełnienie których gwarantuje jakość planowanego przedsięwzięcia.*

## PROJECT PLANNING IN MULTIPRODUCT ENVIRONMENT

*Project planning in multi-product environment of SMEs has to be supported by interactive, task oriented decision support systems. The approach proposed employing the constraint concerning sufficient conditions guaranteeing required project management quality.*

## 1. WSTĘP

Najistotniejsza potrzebą małych i średnich przedsiębiorstw (MŚP) jest osiągnięcie przewagi rynkowej. W tym też kontekście oczekiwania **MŚP** wiążą się z wykorzystaniem technologii informacyjnych (ang. Information Technology (IT)), oczekiwania związane z zaspokajaniem potrzeb monitorowania i archiwizacji , a przede wszystkim przetwarzania danych na potrzeby wspomagania decyzji (np. planowania i sterowania przepływem produkcji) [1]. Przykładowe zastosowanie technologii IT wiąże się z wykorzystaniem systemów wspomagania decyzji (ang. Decision Support Systems DSS), w szczególności w zadaniach planowania inwestycji projektowych wymagających zarządzania zasobami różnej natury i charakteru w zmiennym środowisku wieloprojektowym [5], [6]. Zarządzanie takie wiąże się z koniecznością podejmowanie decyzji w warunkach związanych z występowaniem zakłóceń, niepewnością danych, decyzji integrujących różne problemy cząstkowe, związane np. przydziałem zasobów, marszrutowaniem i harmonogramowania zadań itp.

Przedstawioną sytuację charakteryzują operacje oraz zasoby, a także łączące je związki (określające jakie zasoby i w jakiej ilości są niezbędne do wykonania poszczególnych operacji, ograniczenia np. zakładające niewywłaszczalność i/lub współdzielenie zasobów, ciągły lub dyskretny sposób zużycia zasobów itp.). Łatwo zauważyć, że przyjęcie jednego rodzaju założeń zakładającego np. dyskretny sposób zużycia zasobu wyklucza pozostałe przyjmujące np. ciągły charakter dostępnych zasobów. Oznacza to, że uwzględnienie wszystkich możliwości jest równoznaczne z przyjęciem do rozważań najbardziej ogólnego modelu sytuacji problemowej, modelu tworzącego hybrydę modeli „prostych" przyjmujących jeden rodzaj ograniczenia i/lub założeń dla wszystkich operacji i zasobów.

Naturalnej, programowej implementacji powyższego typu modeli hybrydowych dostarczają języki programowania ograniczeń (ang. constraints programming (CP)) [2]. Celem ilustracji tej możliwości rozważmy przypadek poniższego modelu cząstkowego.

## 2. PROBLEM SPEŁNIANIA OGRANICZEŃ

Programowanie z ograniczeniami jest obszarem technologii oprogramowania bazującym na specyfikacji ograniczeń zmiennych decyzyjnych rozwiązywanych problemów. Istotną cechę ograniczeń stanowi ich deklaratywny charakter. Oznacza to, że ograniczenia specyfikują jedynie postać wymaganych relacji, nie podają natomiast sposobu gwarantującego ich zachodzenie. W tym kontekście, specyfikację problemu stanowią zbiory zmiennych i ich dziedzin oraz zbiór ograniczeń wiążących wybrane zmienne decyzyjne. Poszukiwanym rozwiązaniem jest zbiór wartości zmiennych spełniających przyjęte ograniczenia.

W zależności od charakteru dziedzin zmiennych decyzyjnych, programowanie ograniczeń obejmuje badania: związane z oceną spełniania ograniczeń dla zmiennych, wartości których należą do skończonych dziedzin dyskretnych oraz rozwiązywaniem ograniczeń dla zmiennych należących do dziedzin nieskończonych. Warto zauważyć, że ponad 95 % wszystkich problemów decyzyjnych w obszarze produkcji i usług należy do pierwszej z ww. kategorii, tzn. problemów spełniania ograniczeń (ang. Constraint Satisfaction Problem (CSP)) [2].

Problem spełniania ograniczeń $CS = ((V,D),C)$ określa skończony zbiór zmiennych $V = \{v_1,v_2,...,v_n\}$, rodzina dziedzin zmiennych $D = \{D_i \mid D_i = [d_{i,1}, d_{i,2}, ..., d_{i,j}, ..., d_{i,m}], i = 1 ... n\}$ oraz skończony zbiór ograniczeń $C = \{C_i \mid i = 1..L\}$ limitujących wartości zmiennych decyzyjnych. Poszukiwane jest rozwiązanie bądź to dopuszczalne, tzn. rozwiązanie w którym wartości wszystkich zmiennych spełniają wszystkie ograniczenia (zwykle jedno – najwcześniej uzyskane), bądź też rozwiązanie optymalne ekstremalizujące funkcję celu określoną na wybranym podzbiorze zmiennych decyzyjnych. W poszukiwaniu rozwiązania optymalnego wykorzystywane są metody podziału i ograniczeń bazujące na heurystycznych funkcjach oceny przestrzeni rozwiązań dopuszczalnych.

Rozwiązanie problemu spełniania ograniczeń (PSO) uzyskiwane jest w wyniku systematycznego przeszukiwania możliwych przyporządkowań wartości zmiennych decyzyjnych. Wykorzystywane metody poszukiwania rozwiązań dzielą się na te, w których przeszukiwana jest cała przestrzeń wszystkich możliwych przyporządkowań (wykorzystywane są tutaj zarówno techniki przeszukiwania systematycznego jak i metody stochastyczne) oraz na te, w których przeszukiwana jest tylko część tej przestrzeni. Metody te implementowane są w językach klasy CP, np. Oz MOZART [8], ILOG [7].

## 3. MODEL PROBLEMU DECYZYJNEGO

Dany jest horyzont czasowy $H = [0, h]$, $H \subset N$, określający przedział czasowy realizacji zbioru projektów $P$. Dana jest $lz$ liczba zasobów odnawialnych. Znane są wielkości dostępnych zasobów odnawialnych $Zo = (zo_1, zo_2, \ldots, zo_{lz})$; $zo_i = (zo_{i,1}, zo_{i,2}, \ldots ,zo_{i,h})$ – jest to sekwencja określająca dopuszczalne wartości $i$-tego zasobu dla każdej jednostki czasu horyzontu $H$; $zo_{i,j}$ – dopuszczalna wartość i-tego zasobu w j-tej jednostce czasu. Dany jest zbiór projektów $P = \{P_1, P_2, ..., P_{lp}\}$. Każdy projekt $P_i$ składa się z $lo$ operacji $P_i=\{O_{i,1}, O_{i,2}, O_{i,3}, \ldots, O_{i,loi}\}$, gdzie: $O_{i,j} = (x_{i,j}, t_{i,j}, Tp_{i,j}, Tz_{i,j}, Dp_{i,j})$, przy czym:

$x_{i,j}$ – termin rozpoczęcia operacji $O_{i,j}$ liczony względem początku horyzontu $H$,

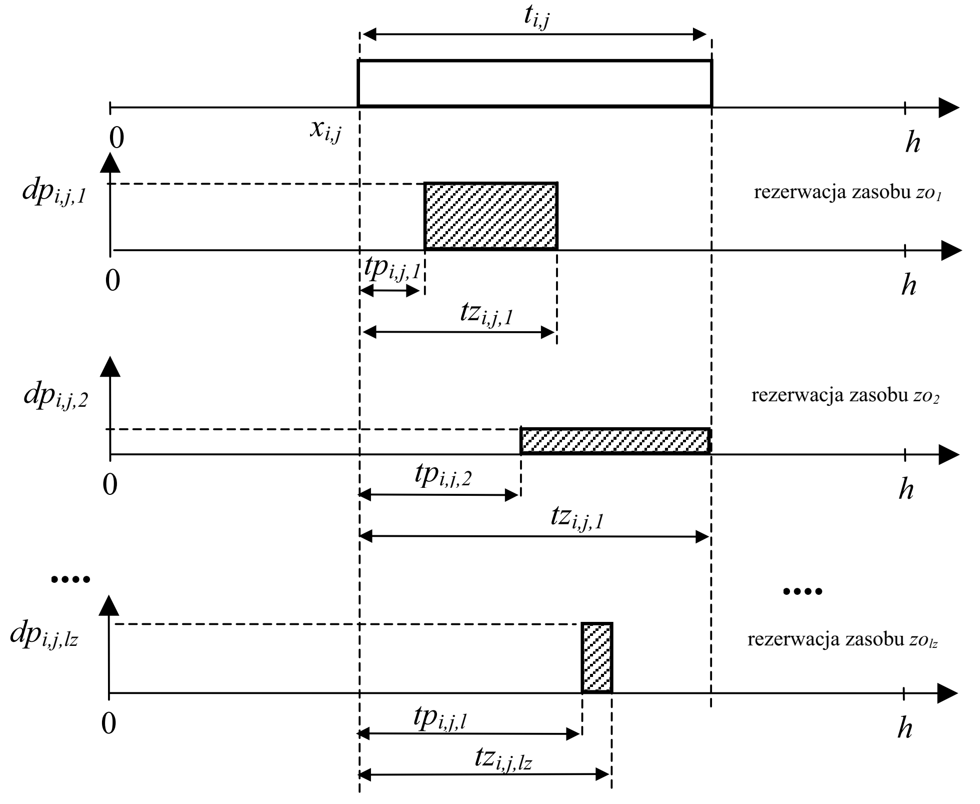
$t_{i,j}$ – czas trwania operacji $O_{i,j}$,

$Tp_{i,j} = (tp_{i,j,1}, tp_{i,j,2}, \ldots , tp_{i,j,lz})$ – sekwencja terminów pobrania przez operacje $O_{i,j}$ kolejnych zasobów odnawialnych: $tp_{i,j,k}$ – termin liczony względem $x_{i,j}$ pobrania przez operację $O_{i,j}$, $k$-tego zasobu odnawialnego w ilości $dp_{i,j,k}$. Przyjmuje się, że zasób jest pobierany w trakcie trwania operacji: $0 \le tp_{i,j,k} < t_{ij}$; $k = 1,2,\ldots,lz$.

$Tz_{i,j} = (tz_{i,j,1}, tz_{i,j,2}, \ldots , tz_{i,j,lz})$ – sekwencja terminów zwracania przez operacje $O_{i,j}$ kolejnych zasobów odnawialnych: $tz_{i,j,k}$ – termin liczony względem $x_{i,j}$ zwrócenia przez operację $O_{i,j}$, $k$-tego zasobu odnawialnego w ilości $dp_{i,j,k}$. Przyjmuje się, że zasób jest pobierany w trakcie trwanie operacji: $0 < tz_{i,j,k} \le t_{i,j}$; $k = 1,2,\ldots,lz$, oraz $tp_{i,j,k} < tz_{i,j,k}$ ; $k = 1, 2, \ldots, lz$.

$Dp_{i,j} = (dp_{i,j,1}, dp_{i,j,2},\ldots, dp_{i,j,lz})$ – sekwencja ilości pobieranych przez operacje $O_{i,j}$ zasobów odnawialnych: $dp_{i,j,k}$ – ilość $k$-tego zasobu pobieranego przez operację $O_{i,j}$. Przyjmuje się, że: $0 \le dp_{i,j,k} \le zo_k$; $k = 1, 2, \ldots, lz$.

Interpretację graficzną przedstawionych wielkości ilustruje rys 1.




Rys. 1. Ilustracja graficzna wielkości: $x_{i,j}$, $t_{i,j}$, $Tp_{i,j}$, $Tz_{i,j}$, $Dp_{i,j}$ - dla zasobów odnawialnych

Zakłada się, że operacje są niepodzielne w czasie oraz mogą rezerwować dowolną liczbę zasobów. Przykładowo, sekwencja $Dp_{1,2} = (1, 2, 1, 0, 0)$, oznacza, że realizacja operacji $O_{1,2}$, wymaga tylko pierwszych trzech zasobów; zasoby 4, 5 nie są wykorzystywane. Przyjmuje się również, że każdy zasób w danej operacji może być wykorzystany tylko jednokrotnie; ilości danego zasobu wykorzystywanego przez daną operację nie może ulec zmianie, nie może zostać przydzielona do innej operacji. Warunkiem rozpoczęcia operacji jest dostęp do żądanej liczby zasobów odnawialnych w zadanych terminach $Tp_{i,j}$.

Wielkości $x_{i,j}$, $t_{i,j}$, oraz elementy sekwencji $Tp_{i,j}$, $Tz_{i,j}$, $Dp_{i,j}$, gdzie $x_{i,j}$, $t_{i,j}$, $tp_{i,k,j}$, $tz_{i,k,j}$, $dp_{i,k,j} \in N$, są elementami następujących sekwencji:

- sekwencji terminów rozpoczęcia operacji projektu $P_i$:

  $X_i = (x_{i,1}, x_{i,2},\ldots, x_{i,lo_i})$, $\quad 0 \leq x_{i,j} < h;\ i = 1, 2,\ldots, lp;\ j = 1, 2,\ldots, lo_i$,

- sekwencji czasów trwania operacji projektu $P_i$:

  $T_i = (t_{i,1}, t_{i,2}, \ldots, t_{i,lo_i})$,

- sekwencji terminów pobierania $j$-tego zasobu podczas trwania operacji projektu $P_i$:

  $TP_{i,j} = (tp_{i,1,j}, tp_{i,2,j},\ldots, tp_{i,lo_i,j})$,

- sekwencji terminów zwalniania $j$-tego zasobu podczas trwania operacji projektu $P_i$:

  $TZ_{i,j} = (tz_{i,1,j}, tz_{i,2,j},\ldots, tz_{i,lo_i,j})$,

- sekwencji ilości pobieranego $j$-tego zasobu w trakcie realizacji operacji projektu $P_i$:

  $DP_{i,j} = (dp_{i,1,j}, dp_{i,2,j},\ldots, dp_{i,lo_i,j})$.

Kolejność realizacji operacji poszczególnych przedsięwzięć określa sieć czynności, wierzchołki której to operacje $O_{i,j}$, a łuki wskazują porządek ich realizacji. Rozważane są następujące ograniczenia:

- dla operacji występujący po sobie:

$$x_{i,j} + t_{ij} \leq x_{i,k}, \tag{1}$$

- dla wielu poprzedników:

$$x_{i,j}+t_{i,j}\leq x_{i,k},\ x_{i,j+1}+t_{i,j+1} \leq x_{i,k},\ x_{i,j+2}+t_{i,j+2}\leq x_{i,k}, \ldots, x_{i,j+n}+t_{i,j+n}\leq x_{i,k}, \tag{2}$$

- dla wielu następników:

$$x_{i,k}+t_{i,k} \leq x_{i,j},\ x_{i,k}+t_{i,k} \leq x_{i,j+1},\ x_{i,k}+t_{i,k} \leq x_{i,j+2}, \ldots, x_{i,k}+t_{i,k} + \leq x_{i,j+n}. \tag{3}$$

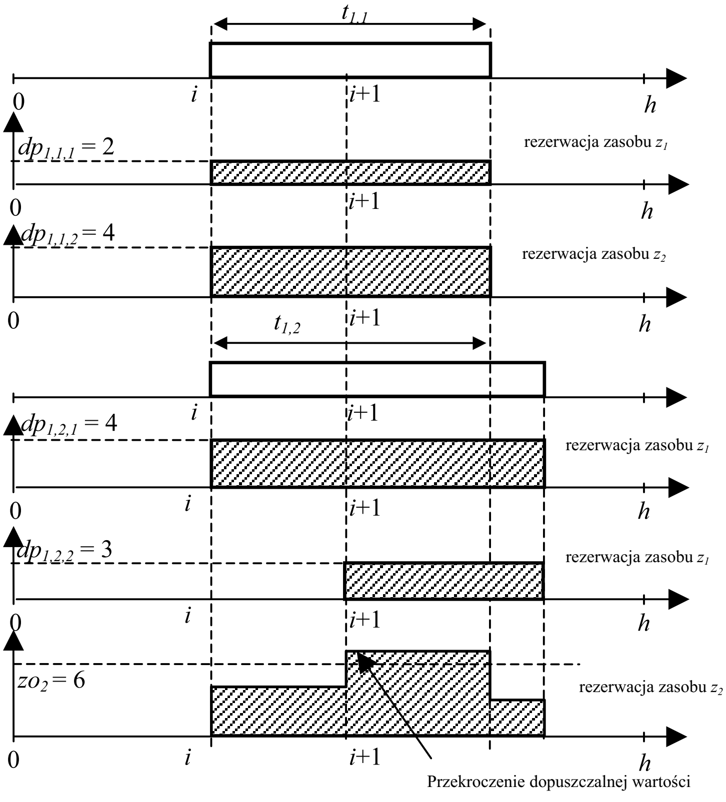
Przyjmuje się również ograniczenia określające związki pomiędzy zasobami rezerwowanymi przez operację $O_{i,j}$:

$$dp_{i,j,1}+dp_{i,j,2}+dp_{i,j,3}+\ldots+dp_{i,j,lz} = rm_{i,j}, \tag{4}$$

gdzie: $lz$ – liczba zasobów, $i$=1,2,…,$lp$; $j$ = 1, 2,…, $lo_i$. Inaczej mówiąc, suma zasobów pobieranych przez operację $O_{i,j}$ jest ograniczona przez wartość $rm_{i,j}$. Wielkości $rm_{i,j}$ dla projektu $P_i$ tworzą sekwencję: $RM_i = (rm_{1,1},\ rm_{1,2},\ \ldots\ rm_{i,loi})$.

Łatwo zauważyć, że ograniczona pula zasobów może prowadzić do występowania konfliktów zasobowych, związanych z koniecznością rozstrzygania pierwszeństwa przydziału limitowanych zasobów. Sytuacje tego typu występują w chwilach gdy kontynuacja równolegle realizowanych operacji wymaga przydziału danego zasobu w ilości przekraczającej jego limit. Rozstrzyganie konfliktów zasobowych ma wpływ na występowanie blokad. Dwie operacje $O_{1,1}$ i $O_{1,2}$ w chwili i-tej korzystają z dwóch zasobów $z_1$ i $z_2$. Limity zasobów wynoszą 6 jednostek każdy dla chwil $i$ ($zo_{1,i}$, $zo_{2,i}$ = 6) i dla chwili $i$+1 ($zo_{1,i+1}$, $zo_{2,i+1}$ = 6). Jeżeli w i-tej chwili operacja $O_{1,1}$ wykorzystuje 2 jednostki zasobu $z_1$ i 4 jednostki zasobu $z_2$, podczas gdy operacja $O_{1,2}$ wykorzystuje 4 jednostki zasobu $z_1$, a w chwili $i$+1-ej do swoich kontynuacji potrzebują odpowiednio: operacja $O_{1,1}$ 2 jednostki zasobu $z_1$ i 4 jednostki zasobu $z_2$, a operacja $O_{1,2}$ wykorzystuje 4 jednostki zasobu $z_1$, i 3 jednostki zasobu $z_2$, wówczas rozstrzygnięcie konfliktów zasobowych w ten sposób, że do operacji $O_{1,1}$

przydzielonych zostaje 4 jednostki zasobu $z_2$, i do operacji $O_{1,2}$ 3 jednostki zasobu $z_2$, prowadzi do blokady operacji. Operacje $O_{1,1}$ i $O_{1,2}$ nie będą mogły się wykonać z uwagi na brakujące: 1 jednostkę zasobu $z_2$ operacji $O_{1,2}$.. Ilustrację takiej sytuacji przedstawiono na rys. 2.




Rys. 2. Przykład blokady

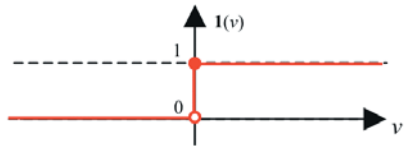
Uniknięcie sytuacji blokadowy przedstawionych na rys. 2 uzyskuje się poprzez wprowadzanie dodatkowych ograniczeń gwarantujących, że wartości terminów rozpoczęcia operacji $X_i$ nie doprowadzą do sytuacji blokadowej [3], [4]. Poniżej przedstawiono propozycję tego typu ograniczeń. Ograniczenie tego typu powinno zapewniać, że każdej chwili horyzontu $H$ suma zarezerwowanych zasobów odnawialnych nie przekracza dopuszczalnych wartości $Zo$. W szczególności, dla $k$-tego zasobu, w chwili $v \in H$, ograniczenie to przyjmuje postać nierówności (5):

$$\sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot \bar{1}(v, x_{i,j}+tp_{i,j,k}, x_{i,j}+tz_{i,j,k})\right] \le zo_{k,v} \tag{5}$$

gdzie: $lp$ – liczba projektów, $lo_i$ – liczba operacji w $i$-tym projekcie,
$dp_{i,j,k}$ – liczba $k$-tego zasobu rezerwowana przez operację $O_{i,j}$,
$\bar{\mathbf{1}}(v,a,b)$ – jednostkowa funkcja czasu rezerwacji zasobu:
$\bar{\mathbf{1}}(v,a,b) = \mathbf{1}(v-a) - \mathbf{1}(v-b)$, gdzie: $\mathbf{1}(v)$ - funkcja jednostkowa

Funkcja jednostkowa definiowana jest następująco (rys. 3):

$$\mathbf{1}(v) = \begin{cases} 0 \text{ dla } v < 0 \\ 1 \text{ dla } v \geq 0 \end{cases}$$




Rys. 3. Funkcja jednostkowa $\mathbf{1}(v)$

Funkcja $\bar{\mathbf{1}}(v,a,b)$ jest wykorzystywana do opisu rezerwacji zasobu przez określoną operację. Można wykazać [9], że dla zbioru projektów $P$ zależność (5) prowadzi do ograniczeń postaci układu nierówności (6).

$$\begin{cases}
\sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(x_{1,1}+tp_{1,1,k},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\leq zo_{k,x_{1,1}} \\
\sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(x_{1,2}+tp_{1,2,k},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\leq zo_{k,x_{1,2}} \\
\qquad\qquad\qquad\qquad \ldots \\
\sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(x_{1,lo_1}+tp_{1,lo_1,k},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\leq zo_{k,x_{1,lo_1}} \\
\sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(x_{2,1}+tp_{2,1,k},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\leq zo_{k,x_{2,1}} \\
\sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(x_{2,2}+tp_{2,2,k},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\leq zo_{k,x_{2,2}} \\
\qquad\qquad\qquad\qquad \ldots \\
\sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(x_{2,lo_2}+tp_{2,lo_2,k},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\leq zo_{x_{2,lo_2}} \\
\qquad\qquad\qquad\qquad \ldots \\
\sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(x_{lp,1}+tp_{lp,1,k},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\leq zo_{k,x_{lp,1}} \\
\sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(x_{lp,2}+tp_{lp,2,k},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\leq zo_{k,x_{lp,2}} \\
\qquad\qquad\qquad\qquad \ldots \\
\sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(x_{lp,lo_{lp}}+tp_{lp,lo_{lp},k},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\leq zo_{k,lo_{lp}}
\end{cases} \qquad (6)$$

dla: $k = 1, 2, \ldots, lz$, gdzie $lz$ – liczba zasobów odnawialnych.

W ograniczeniach (6), wartość sumy rezerwowanych zasobów wyznaczane są tylko dla chwil odpowiadających terminom pobrania zasobów $x_{i,j} + tp_{i,j,}$ przez kolejne operacje.

W horyzoncie $H$ dopuszcza się zmienność wartości posiadanych zasobów. Zmienność ta wyrażana jest poprzez sekwencje $zo_1$, $zo_2$, …, $zo_{lz}$. Aby zagwarantować by w rozważanym horyzoncie $H$ suma rezerwowanych zasobów nie przekroczyła dopuszczalnych wartości należy dodatkowo wprowadzić ograniczenia (odpowiadające ograniczeniu (5)) dla chwil $vp_{k,i}$, w których dochodzi do zmian dopuszczalnych wartości zasobów odnawialnych (7):

$$\begin{cases} \sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(vp_{k,1},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\le zo_{k,vp_{k,1}} \\ \sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(vp_{k,2},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\le zo_{k,vp_{k,2}} \\ \ldots \\ \sum_{i=1}^{lp}\sum_{j=1}^{lo_i}\left[dp_{i,j,k}\cdot 1(vp_{k,q},x_{i,j}+tp_{i,j,k},x_{i,j}+tz_{i,j,k})\right]\le zo_{k,vp_{k,q}} \end{cases} \tag{7}$$

dla: $k$ = 1,2, …, $lz$, gdzie $lz$ – liczba zasobów odnawialnych,

gdzie: $vp_{k,i}$ – $i$-ta chwila, w której nastąpiła zmiana dopuszczalnych wartości zasobu $k$,

$zo_{k,(vp_{k,i}-1)} - zo_{k,vp_{k,i}} \neq 0$, Wielkości $vp_{k,i}$ dla $k$-tego zasobu tworzą sekwencję:

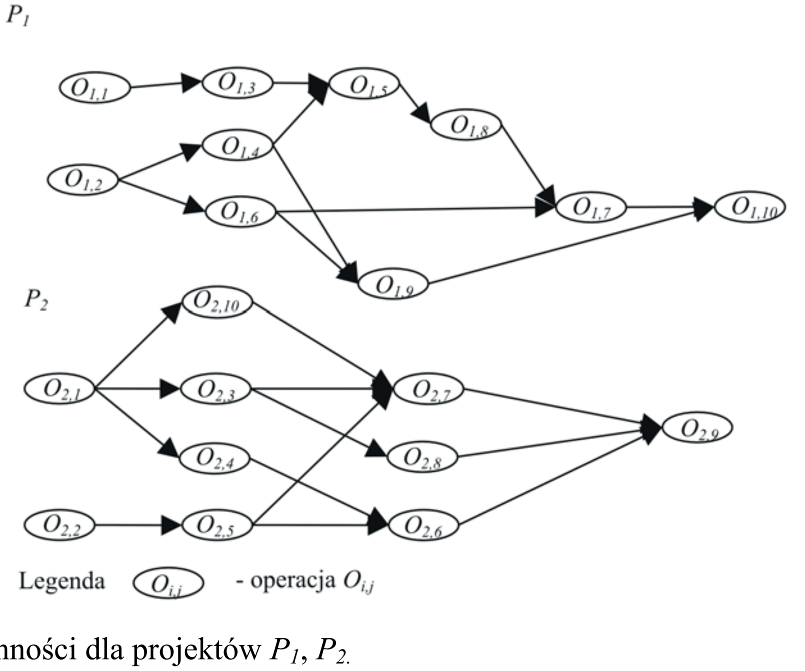
$Vp_k = (vp_{k,1}, vp_{k,2}, \ldots , vp_{k,q})$,

Sekwencja $Vp_k$ zawiera terminy, w których doszło do zmiany wartości kolejnych elementów sekwencji $zo_k$, $q$ – liczba elementów sekwencji $Vp_k$.

W rozważanym modelu, dla danego horyzontu $H$, zbioru projektów $P$, dopuszczalnej wartości zasobów $Zo$, oraz znanych sekwencji $T_i$ , $TP_{i,j}$ , $TZ_{i,j}$. Poszukiwana jest odpowiedź na pytanie: Czy istnieją takie przydziały zasobów do wybranych operacji przy, których możliwa jest realizacji projektów $P$ w zadanym horyzoncie $H$?

## 4. PRZYKŁAD ILUSTRACYJNY

Celem ilustracji analizowanego zagadnienia rozważmy zbiór przedsięwzięć $P = \{P_1, P_2\}$. Każde z przedsięwzięć wymaga realizacji 15 operacji. Kolejność realizacji operacji określa sieć czynności z rys. 4.




Rys. 4. Sieć czynności dla projektów $P_1$, $P_2$.

Czasy trwania operacji projektów $P_1$, $P_2$, opisują sekwencje:

$T_1 =$ (3 4 5 3 4 2 3 1 2 4 ),
$T_2 =$ (1 4 4 3 5 5 4 6 8 3 ),

Do realizacji operacji wykorzystywane są dwa rodzaje zasobów odnawialnych. Zasoby są rezerwowane wraz z rozpoczęciem operacji i zwalniane w momencie jej ukończenia. Stąd odpowiednie sekwencje $TP_{1,1}$, $TP_{1,2}$, $TP_{2,1}$, $TP_{2,2}$, mają postać: $TP_{1,2}$=$TP_{1,1}$= $TP_{2,2}$=$TP_{2,1}$= (0,0, 0,0,0,0,0,0,0,0).

Sekwencje $TZ_{1,1}$, $TZ_{1,2}$, $TZ_{2,1}$, $TZ_{2,2}$, mają postać: $TP_{1,1} = T_1$, $TP_{1,2} = T_1$, $TP_{2,1} = T_2$, $TP_{2,2} = T_2$. Wymagane ilość zasobów dla operacji projektów $P_1$, $P_2$, ograniczone są poprzez (4) sumy których wartości są opisane w postaci sekwencji :

$RM_1$ = (4 2 3 4 2 3 3 4 4),
$RM_2$ = (4 2 3 4 2 3 3 4 4).

Dla zadanych wielkości zasobów odnawialnych $Zo = (zo_1, zo_2)$:

$zo_1$ = ( 8 8 8 8 8 8 8 8 8 8 10 10 10 10 5 5 5 5 5 5 5 5 5 5 )
$zo_2$ = (11 11 11 11 11 11 11 11 9 9 9 9 9 9 9 5 5 5 5 5 5 5 5 5 )

oraz horyzont $H = [0, 24]$, $H \subset N$, poszukiwana jest odpowiedź na pytanie: Czy istnieją takie przydziały zasobów do wybranych operacji, przy których możliwa jest realizacja projektów w zadanym horyzoncie $H$?
Odpowiedź na to pytanie wiąże się wyznaczeniem wartości elementów sekwencji $X_1$, $X_2$:

$X_1 = (x_{1,1}, x_{1,2},\ldots, x_{1,10})$, $X_2 = (x_{2,1}, x_{2,2},\ldots, x_{2,10})$,
gdzie: $0 \leq x_{i,j} < 24$; $i = 1,2$; $j = 1,2,\ldots,10$,

i sekwencji $DP_{1,1}$, $DP_{1,2}$, $DP_{2,1}$, $DP_{2,2}$:

$DP_{1,1} = (dp_{1,1,1}, dp_{1,2,1},\ldots, dp_{1,10,1})$, $DP_{1,2} = (dp_{1,1,2}, dp_{1,2,2},\ldots, dp_{1,10,2})$,
$DP_{2,1} = (dp_{2,1,1}, dp_{2,2,1},\ldots, dp_{2,10,1})$, $DP_{2,2} = (dp_{2,1,2}, dp_{2,2,2},\ldots, dp_{2,10,2})$,
gdzie: $0 \leq dp_{i,j,k} < 4$; $i = 1, 2$; $j = 1, 2, \ldots, 10$, $k = 1,2$;

Poszukiwane wartości sekwencji $X_1$, $X_2$, powinny spełniać ograniczenia kolejnościowe, ograniczenia narzucane na zasoby odnawialne oraz ograniczenia określające związki między zasobami:

- Ograniczenia kolejnościowe:
  Dla sieci czynności z rys. 4 ograniczenia kolejnościowe, budowane zgodnie z ograniczeniami (1), (2), (3), mają postać jak w Tabeli 1.

Tab. 1. Ograniczenia kolejnościowe dla sieci czynności z rys. 8.

| Projekt $P_1$ | Projekt $P_2$ |
|---|---|
| $x_{1,3} \geq x_{1,1} + t_{1,1}$ | $x_{2,3} \geq x_{2,1} + t_{2,1}$ |
| $x_{1,4} \geq x_{1,2} + t_{1,2}$ | $x_{2,4} \geq x_{2,1} + t_{2,1}$ |
| ... | ... |
| $x_{1,10} \geq x_{1,7} + t_{1,7}$ | $x_{2,9} \geq x_{2,8} + t_{2,8}$ |
| $x_{1,10} \geq x_{1,9} + t_{1,9}$ | $x_{2,10} \geq x_{2,10} + t_{2,10}$ |

- Ograniczenia określające związki między zasobami:
  Dla sieci czynności z rys. 4 ograniczenia określające związki między zasobami, budowane zgodnie z ograniczeniami (4) w oparciu o sekwencje $RM_1$, $RM_2$, mają postać jak w tab. 2.

Tab. 2 Ograniczenia określające związki między zasobami, budowane zgodnie z ograniczeniami (4).

| Projekt $P_1$ | Projekt $P_2$ |
|---|---|
| $dp_{1,1,1}+dp_{1,1,2} = 4$<br>$dp_{1,2,1}+dp_{1,2,2} = 2$<br>...<br>$dp_{1,9,1}+dp_{1,9,2} = 4$<br>$dp_{1,10,1}+dp_{1,10,2} = 4$ | $dp_{2,1,1}+dp_{2,1,2} = 4$<br>$dp_{2,2,1}+dp_{2,2,2} = 2$<br>...<br>$dp_{2,9,1}+dp_{2,9,2} = 4$<br>$dp_{2,10,1}+dp_{2,10,2} = 4$ |

- Ograniczenia narzucane na zasoby odnawialne:
  Uwzględniając zerowe wartości sekwencji $TP_{i,j}$ oraz równość sekwencji $TZ_{i,j}$ z odpowiadającymi im sekwencjami $T_i$, ograniczenia na zasoby współdzielone dla rozważanego przykładu mają postać (zgodnie z (6)):

$$\begin{cases} \sum_{i=1}^{2}\sum_{j=1}^{10}\left[dp_{i,j,1}\cdot 1(x_{1,1},x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{1,x_{1,1}} \\ \dots \\ \sum_{i=1}^{2}\sum_{j=1}^{10}\left[dp_{i,j,1}\cdot 1(x_{1,10},x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{1,x_{1,20}} \\ \sum_{i=1}^{2}\sum_{j=1}^{10}\left[dp_{i,j,1}\cdot 1(x_{2,1},x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{1,x_{2,1}} \\ \dots \\ \sum_{i=1}^{2}\sum_{j=1}^{20}\left[dp_{i,j,1}\cdot 1(x_{2,10},x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{1,x_{2,10}} \end{cases} \quad (8)$$

$$\begin{cases} \sum_{i=1}^{2}\sum_{j=1}^{10}\left[dp_{i,j,2}\cdot 1(x_{1,1},x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{2,x_{1,1}} \\ \dots \\ \sum_{i=1}^{2}\sum_{j=1}^{10}\left[dp_{i,j,2}\cdot 1(x_{1,10},x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{2,x_{1,20}} \\ \sum_{i=1}^{2}\sum_{j=1}^{10}\left[dp_{i,j,2}\cdot 1(x_{2,1},x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{2,x_{2,1}} \\ \dots \\ \sum_{i=1}^{2}\sum_{j=1}^{20}\left[dp_{i,j,2}\cdot 1(x_{2,10},x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{2,x_{2,10}} \end{cases} \quad (9)$$

Ograniczenia zgodne z układem (7) mają postać:

$$\begin{cases} \sum_{i=1}^{2}\sum_{j=1}^{10}\left[dp_{i,j,1}\cdot 1(10,x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{1,10} \\ \sum_{i=1}^{2}\sum_{j=1}^{10}\left[dp_{i,j,1}\cdot 1(13,x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{1,13} \end{cases} \quad (10) \qquad \begin{cases} \sum_{i=1}^{2}\sum_{j=1}^{10}\left[dp_{i,j,2}\cdot 1(8,x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{2,8} \\ \sum_{i=1}^{2}\sum_{j=1}^{10}\left[dp_{i,j,2}\cdot 1(15,x_{i,j},x_{i,j}+t_{i,j})\right]\le zo_{2,15} \end{cases} \quad (11)$$

Liczba ograniczeń (8), (9) na zasoby współdzielone w rozważanym przypadku wynosi: *Logr* = 40. Każde z ograniczeń jest sumą 20 elementów. Terminy rozpoczęcia końcowych operacji projektów $O_{1,10}$, $O_{2,9}$, powinny spełniać ograniczenia: $x_{1,10}+t_{1,10} \le h$, $x_{2,9}+t_{2,9} \le h$.

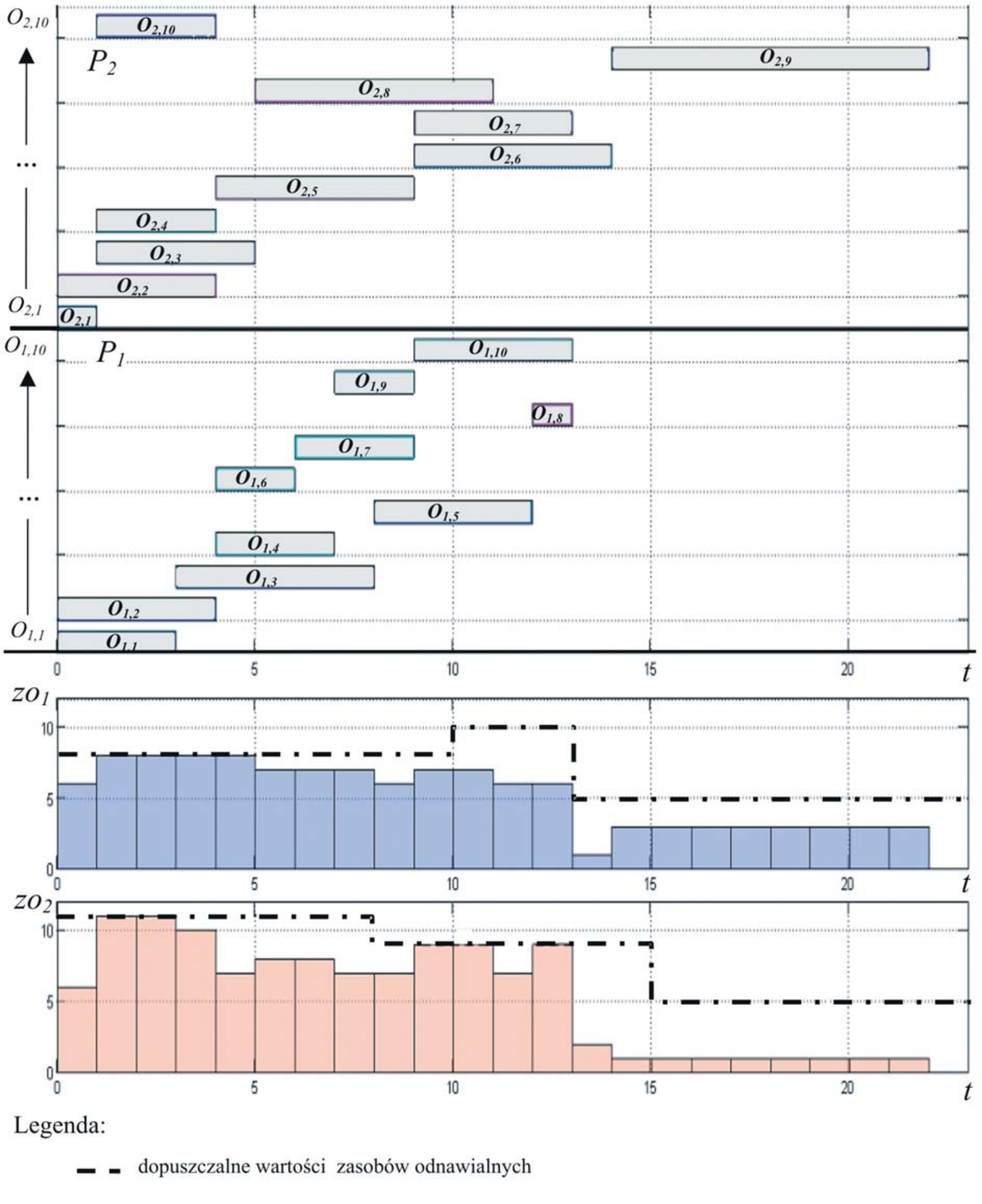
Powyższe ograniczenia zaimplementowane zostały w środowisku OzMozart. Pierwsze rozwiązanie dopuszczalne zostało znalezione po 36 krokach obliczeniowych. Czas trwania obliczeń na komputerze z procesorem AMD Athlon(tm)XP 2500+ 1,85 GHz i pamięcią RAM 1,00 GB wynosił: 32 m. Otrzymane rozwiązanie dopuszczalne:

$X_1 = (0\ 0\ 3\ 4\ 8\ 4\ 6\ 12\ 7\ 9)$, $\quad X_2 = (0\ 0\ 1\ 1\ 4\ 9\ 9\ 5\ 14\ 1)$;

$DP_{1,1} = (2\ 1\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2)$, $DP_{1,2} = (2\ 1\ 1\ 3\ 1\ 1\ 1\ 3\ 2\ 3)$;

$DP_{2,1} = (2\ 1\ 2\ 1\ 1\ 1\ 2\ 1\ 3\ 1)$, $DP_{1,2} = (2\ 1\ 1\ 3\ 1\ 2\ 1\ 2\ 1\ 3)$;

Rozwiązanie przedstawiono w postaci harmonogramu na rys. 5:




Rys. 5. Harmonogram realizacji operacji projektów $P_1$, $P_2$, oraz wykresy ilości zarezerwowanych zasobów $zo_1$, $zo_2$, w kolejnych jednostkach czasu.

Na rys. 5 przedstawiono 3 wykresy. Pierwszy ilustruje harmonogram realizacji operacji. Kolejne dwa ilustrują, ilość zarezerwowanych zasobów odpowiadającą wyznaczonemu harmonogramowi.

## 5. WNIOSKI

Przedstawione podejście ilustruje możliwości wykorzystania języków programowania z ograniczeniami do budowy zadaniowo zorientowanych, interakcyjnych systemów wspomagania decyzji, w szczególności w zakresie wariantowania realizacji portfeli projektów; wariantowania zorientowanego na wyznaczanie wartości zmiennych decyzyjnych gwarantujących zadane wartości funkcji celu. Aktualnie prowadzone badania związane z możliwościami rozwiązywania zadań alokacji zasobów odnawialnych i nieodnawialnych pracujących w systemie wielozadaniowym, uwzględniające nieprecyzyjne wartości zmiennych decyzyjnych stanowią naturalne rozszerzenie przedstawionych wyników.

## Literatura


1. Bach I., Tomczuk-Piróg I., Bzdyra K., Banaszak Z., Zarządzanie wiedzą MŚP (wspomaganie decyzji). W: Komputerowo Zintegrowane Zarządzanie, T. I, Knosla R., Oficyna Wydawnicza PTZP, Opole 2007, s.22-31
2. Barták R. Incomplete Depth-First Search Techniques: A Short Survey, Proceedings of the 6$^{th}$ Workshop on Constraint Programming for Decision and Control, Ed. Figwer J., 2004; 7-14.
3. Bocewicz G., Banaszak Z., Wójcik R.: ‘Design of admissible schedules for AGV systems with constraints: a logic-algebraic approach’, In: Agent and Multi-Agent Systems: Technologies and Applications, Nguyen N.T., Grzech A., Howlett R.J., Jain L.C. (Eds.), Lecture Notes in Artificial Intelligence 4496, Springer-Verlag, Berlin, Heidelberg, 2007, 578-587.
4. Bocewicz G., Bach I., Banaszak Z., Komputerowo wspomagane sterowanie dyspozytorskie pojazdami AGV. Zastosowania teorii systemów, seria: Monografie, nr 36, Wydział Inżynierii Mechanicznej i Robotyki AGH, Kraków 2007, 5-20.
5. Kis, T., G. Ërdos, A. Markus and J. Vancza, A Project-Oriented Decision Support System for Production Planning in Make-to-Order Manufacturing. ERCIN News, Vol. 58, 2004; 44-52.
6. Martinez, E. C., D. Duje and G.A. Perez, On performance modeling of project-oriented production. Computers and Industrial Engineering, Vol. 32, 1997; 509-527.
7. Puget J-F.: *A C++* Implementations of CLP, Proceeding of SPICS 94,
8. Schulte CH., Smolka G., Wurtz J.: Finite Domain Constraint Programming in Oz, DFKI OZ documentation series, German Research Center for Artificial Intelligence, Stuhlsaltzenhausweg 3, D-66123 Saarbrucken, Germany, 1998.
9. Wójcik R., Bach., Banaszak Wariantowanie portfeli przedsięwzięć w wieloprojek-towym środowisku MŚP. W: Komputerowo Zintegrowane Zarządzanie, T. I, Knosla R., Oficyna Wydawnicza PTZP, Opole 2008, (w druku).