XV Krajowa Konferencja Automatyki

Tom I

KKA 2005

Redaktorzy:
Zdzisław Bubnicki
Roman Kulikowski
Janusz Kacprzyk

XV Krajowa
Konferencja Automatyki
Tom I

KKA
2005

Redaktorzy:
Zdzisław BUBNICKI
Roman KULIKOWSKI
Janusz KACPRZYK

ORGANIZATOR
Komitet Automatyki i Robotyki Polskiej Akademii Nauk
Instytut Badań Systemowych Polskiej Akademii Nauk

WSPÓŁORGANIZATORZY
Politechnika Warszawska
Przemysłowy Instytut Automatyki i Pomiarów
Polskie Stowarzyszenie Pomiarów, Automatyki i Robotyki

ORGANIZATOR

Komitet Automatyki i Robotyki Polskiej Akademii Nauk
Instytut Badań Systemowych Polskiej Akademii Nauk

WSPÓŁORGANIZATORZY

Politechnika Warszawska
Przemysłowy Instytut Automatyki i Pomiarów
Polskie Stowarzyszenie Pomiarów, Automatyki i Robotyki

KOMITET PROGRAMOWY

Przewodniczący	Zdzisław BUBNICKI
Zastępca Przewodniczącego	Roman KULIKOWSKI

CZŁONKOWIE

Stanisław BAŃKA
Mikołaj BUSŁOWICZ
Ryszard GESSING
Jakub GUTENBAUM
Stanisław KACZANOWSKI
Janusz KACPRZYK
Józef KORBICZ
Krzysztof KOZŁOWSKI
Krzysztof KUŹMIŃSKI
Krzysztof MALINOWSKI
Antoni NIEDERLIŃSKI
Tadeusz PUCHAŁKA
Stanisław SKOCZOWSKI
Jerzy ŚWIĄTEK
Ryszard TADEUSIEWICZ
Krzysztof TCHOŃ
Jan WĘGLARZ
Michał BIAŁKO
Władysław FINDEISEN
Henryk GÓRECKI
Jerzy JÓZEFCZYK
Tadeusz KACZOREK
Jerzy KLAMKA
Zbigniew KOWALSKI
Juliusz L. KULIKOWSKI
Kazimierz MALANOWSKI
Wojciech MITKOWSKI
Władysław PEŁCZEWSKI
Leszek RUTKOWSKI
Roman SŁOWIŃSKI
Andrzej ŚWIERNIAK
Piotr TATJEWSKI
Leszek TRYBUS
Andrzej P. WIERZBICKI

KOMITET ORGANIZACYJNY

Przewodniczący	Roman KULIKOWSKI
Zastępcy Przewodniczącego	Janusz KACPRZYK Stanisław KACZANOWSKI Tadeusz KACZOREK Krzysztof MALINOWSKI
Członkowie	Roman OSTROWSKI Tadeusz PUCHAŁKA Dariusz WAGNER
Sekretarze naukowi	Jan STUDZIŃSKI Jan W. OWSIŃSKI

ISBN 83-89475-00-6

Druk: ARGRAF, Warszawa

REFERATY PLENARNE

ZASTOSOWANIA ZMIENNYCH NIEPEWNYCH W PROBLEMACH STEROWANIA SYSTEMAMI KOMPUTEROWYMI

Zdzisław BUBNICKI *

* Instytut Informatyki Technicznej, Politechnika Wrocławska
ul. Wyb. Wyspiańskiego 27, 50-370 Wrocław, e-mail: zdzislaw.bubnicki@pwr.wroc.pl

Streszczenie: Praca dotyczy wybranych problemów sterowania systemami komputerowymi na podstawie niepewnej wiedzy. Systemy komputerowe są specyficznymi obiektami podejmowania decyzji i sterowania, dla których szczególnie uzasadnione i przydatne jest stosowanie nieprobabilistycznych opisów niepewności formułowanych przez eksperta. W ostatnich latach rozwinięto formalizm tzw. zmiennych niepewnych jako narzędzie do analizy i podejmowania decyzji w systemach niepewnych charakteryzowanych przez eksperta. Celem niniejszej pracy jest krótki przegląd możliwości zastosowań zmiennych niepewnych w wybranych przypadkach sterowania systemami komputerowymi. Część pierwsza dotyczy sterowania rozdziałem obciążeń (alokacją zadań) w grupie równolegle pracujących procesorów. Część druga poświęcona jest sterowaniu ruchem w sieciach komputerowych, a w szczególności sterowaniu przeciwdziałającemu przeciążeniom.

Słowa kluczowe: Zmienne niepewne, systemy niepewne, systemy komputerowe, sterowanie systemami komputerowymi, systemy z niepewną wiedzą.

1. WSTĘP

Wprowadzona i rozwijana w ostatnich latach koncepcja tzw. zmiennych niepewnych jest szczególnie przydatna dla problemów podejmowania decyzji i sterowania w systemach niepewnych opisywanych zarówno tradycyjnymi modelami matematycznymi, jak i relacyjnymi reprezentacjami wiedzy traktowanymi jak uogólnienie klasycznych modeli funkcjonalnych [6, 5, 7, 8, 14, 17, 21, 22, 19, 20, 31]. Zmienna niepewna jest opisana za pomocą tzw. rozkładu pewności podanego przez eksperta i charakteryzującego jego wiedzę dotyczącą przybliżonych wartości zmiennej. Zmienne niepewne są w pewnym sensie podobne do zmiennych losowych i rozmytych, istnieją jednak zasadnicze różnice omówione w książkach cytowanych powyżej. Zestaw podstawowych koncepcji formalizacji niepewności i „obliczeń miękkich" (obejmujący zmienne niepewne) znaleźć można także w książkach [33, 32]. Opracowano również zastosowania zmiennych niepewnych m.in. do sterowania procesem montażu i innymi procesami wytwarzania w systemach produkcyjnych, sterowania transportem, rozpoznawania i diagnostyki oraz rozdziału nakładów na badania naukowe [10, 35, 16], a także do wybranych przypadków sterowania systemami komputerowymi (m.in. [4, 9,12, 15, 18, 27, 28, 36, 37, 38]).

Systemy komputerowe są specyficznymi obiektami podejmowania decyzji i sterowania, dla których szczególnie uzasadnione i przydatne jest stosowanie nieprobabilistycznych opisów niepewności formułowanych przez eksperta. **Sterowanie systemami komputerowymi na podstawie niepewnej wiedzy** jest aktualnie jednym z podstawowych kierunków badawczych w Instytucie Informatyki Technicznej Politechniki Wrocławskiej, obejmującym zastosowania nie tylko zmiennych niepewnych (np. [34]) i związanym z pracami dotyczącymi metod i technik sterowania ruchem w sieciach komputerowych [29]. Celem niniejszego opracowania jest krótki przegląd możliwości zastosowań zmiennych niepewnych w wybranych przypadkach sterowania systemami komputerowymi, ze szczególnym uwzględnieniem rezultatów dotyczących sterowania rozdziałem obciążeń w systemie wieloprocesorowym (p. 3 i 4). Rozpoczniemy od krótkiego przedstawienia zmiennych niepewnych i podstawowego problemu decyzyjnego. Szczegóły znaleźć można w książkach autora [8, 17, 21, 22] i w innych cytowanych tu pracach.

2. ZMIENNE NIEPEWNE I PODSTAWOWY PROBLEM DECYZYJNY

W definicji zmiennej niepewnej $\bar{x}$ występują dwie własności miękkie (tj. takie własności $\varphi(x)$, że dla ustalonego x wartość logiczna $v[\varphi(x)] \in [0,1]$): „$\bar{x} \cong x$" co oznacza, że „$\bar{x}$ jest w przybliżeniu równe x" lub „x jest przybliżoną wartością $\bar{x}$" oraz „$\bar{x} \,\tilde{\in}\, D_x$" co oznacza, że „$\bar{x}$ w przybliżeniu należy do zbioru D_x" lub „przybliżona wartość $\bar{x}$ należy do D_x". *Zmienna niepewna* $\bar{x}$ jest zdefiniowana przez zbiór wartości X (przestrzeń wektorów liczbowych o składowych rzeczywistych), funkcję $h(x) = v(\bar{x} \cong x)$ czyli *wskaźnik pewności*, że $\bar{x} \cong x$ podany przez eksperta oraz następujące określenia dla $D_x, D_1, D_2 \subseteq X$:

$$\nu(\bar{x} \mathrel{\tilde{\in}} D_x) = \begin{cases} \max\limits_{x \in D_x} h(x) & \text{dla} \quad D_x \neq \varnothing \\ 0 & \text{dla} \quad D_x = \varnothing, \text{ (zbiór pusty).} \end{cases}$$

$$\nu(\bar{x} \mathrel{\tilde{\notin}} D_x) = 1 - \nu(\bar{x} \mathrel{\tilde{\in}} D_x) ,$$

$$\nu(\bar{x} \mathrel{\tilde{\in}} D_1 \vee \bar{x} \mathrel{\tilde{\in}} D_2) = \max\{\nu(\bar{x} \mathrel{\tilde{\in}} D_1), \nu(\bar{x} \mathrel{\tilde{\in}} D_2)\},$$

$$\nu(\bar{x} \mathrel{\tilde{\in}} D_1 \wedge \bar{x} \mathrel{\tilde{\in}} D_2)$$

$$= \begin{cases} \min\{\nu(\bar{x} \mathrel{\tilde{\in}} D_1), \nu(\bar{x} \mathrel{\tilde{\in}} D_2)\} & \text{dla} \quad D_1 \cap D_2 \neq \varnothing \\ 0 & \text{dla} \quad D_1 \cap D_2 = \varnothing. \end{cases}$$

Funkcję $h(x)$ nazywamy rozkładem pewności.

Zmienna niepewna typu C jest zdefiniowana przez zbiór wartości X, funkcję $h(x) = \nu(\bar{x} \cong x)$ podaną przez eksperta oraz następujące określenia:

$$\nu_c(\bar{x} \mathrel{\tilde{\in}} D_x) = \frac{1}{2}[\nu(\bar{x} \mathrel{\tilde{\in}} D_x) + 1 - \nu(\bar{x} \mathrel{\tilde{\in}} \overline{D}_x)] , \qquad (1)$$

gdzie $\overline{D}_x = X - D_x$,

$$\nu_c(\bar{x} \mathrel{\tilde{\notin}} D_x) = 1 - \nu_c(\bar{x} \mathrel{\tilde{\in}} D_x) ,$$

$$\nu_c(\bar{x} \mathrel{\tilde{\in}} D_1 \vee \bar{x} \mathrel{\tilde{\in}} D_2) = \nu_c(\bar{x} \mathrel{\tilde{\in}} D_1 \cup D_2) ,$$

$$\nu_c(\bar{x} \mathrel{\tilde{\in}} D_1 \wedge \bar{x} \mathrel{\tilde{\in}} D_2) = \nu_c(\bar{x} \mathrel{\tilde{\in}} D_1 \cap D_2) .$$

Zastosowanie zmiennej typu C oznacza lepsze wykorzystanie wiedzy eksperta, lecz może być bardziej skomplikowane.

Rozpatrzmy obiekt statyczny z wektorem wejściowym $u \in U$ i wektorem wyjściowym $y \in Y$, opisany relacją $R(u, y; x) \subset U \times Y$, gdzie x oznacza wektor nieznanych parametrów, o którym zakładamy, że jest wartością zmiennej niepewnej $\bar{x}$ opisanej rozkładem pewności $h(x)$ podanym przez eksperta. Jeśli relacja R nie jest funkcją, to wartość u determinuje zbiór możliwych wyjść

$$D_y(u; x) = \{y \in Y : (u, y) \in R(u, y; x)\} .$$

Dla własności $y \in D_y \subset Y$ wymaganej przez użytkownika możemy sformułować następujący **problem decyzyjny**: Dla danych R, $h(x)$ i D_y należy znaleźć decyzję u^* maksymalizującą wskaźnik pewności własności: „zbiór możliwych wyjść w przybliżeniu należy do D_y" czyli „zbiór możliwych wyjść należy do D_y dla przybliżonej wartości zmiennej $\bar{x}$". Zatem

$$u^* = \arg \max_{u \in U} \nu[D_y(u; \bar{x}) \mathrel{\tilde{\subseteq}} D_y] = \arg \max_{u \in U} \max_{x \in D_x(u)} h(x), \qquad (2)$$

gdzie $D_x(u) = \{x \in X : D_y(u; x) \subseteq D_y\}$.

3. STEROWANIE ROZDZIAŁEM (ALOKACJĄ) ZADAŃ W SYSTEMIE WIELOPROCESOROWYM

Formalizm zmiennych niepewnych zastosować można w wybranych problemach sterowania alokacją zadań i zasobów w systemie wieloprocesorowym (wielokomputerowym), traktowanych jak specyficzne problemy alokacji zadań i zasobów w kompleksie operacji opisanym relacyjną reprezentacją wiedzy z nieznanymi parametrami [9, 11, 17]. Rozważania w obecnym punkcie ograniczymy do stosunkowo prostego problemu alokacji polegającej na rozdziale zbioru dużej liczby elementarnych zadań (elementarnych programów lub części programów) pomiędzy równolegle pracujące procesory [4, 15]. Załóżmy zatem, że globalne zadanie obliczeniowe może być zdekomponowane na N części (zadań elementarnych), które mogą być wykonywane równocześnie przez oddzielne procesory. Każde zadanie elementarne jest scharakteryzowane przez czas realizacji τ_i na i-ym procesorze ($i = 1,2,...,k$) i wartości τ_i są jednakowe dla wszystkich zadań elementarnych. Problem decyzyjny (problem sterowania) polega tu na wyznaczeniu liczb zadań elementarnych $n_1, n_2, ..., n_k$ przydzielonych do poszczególnych procesorów, przy uwzględnieniu czasu realizacji wszystkich zadań elementarnych

$$T = \max\{T_1, T_2, ..., T_k\} , \qquad (3)$$

gdzie T_i jest czasem realizacji zadań przydzielonych i-emu procesorowi; $n_1 + n_2 + ... + n_k = N$. Jeśli N jest dostatecznie duże, możemy wyznaczyć decyzje u_i jako dowolne liczby rzeczywiste spełniające ograniczenia

$$\bigwedge_i u_i \geq 0, \qquad \sum_{i=1}^{k} u_i = N , \qquad (4)$$

i następnie otrzymać n_i przez zaokrąglenie u_i do najbliższej liczby całkowitej nieujemnej.

Załóżmy, że procesory scharakteryzowane są nierównościami

$$T_i \leq x_i u_i , \qquad i = 1,2,...,k , \qquad (5)$$

nieznany parametr x_i (górne ograniczenie jednostkowego czasu realizacji) jest wartością zmiennej niepewnej opisanej przez rozkład pewności $h_i(x_i)$ podany przez eksperta oraz, że zmienne $\bar{x}_1, ..., \bar{x}_k$ są niezależne. Rozpatrywany problem przydziału elementarnych programów do procesorów jest przykładem ogólniejszego problemu alokacji zadań w kompleksie operacji równoległych, traktowanym jak specyficzny obiekt decyzyjny opisany w p. 2 (rys. 1), w którym y jest czasem realizacji T, $x = (x_1, ..., x_k)$, $u = (u_1, ..., u_k) \in \overline{U}$, zbiór $\overline{U} \subset R^k$ jest określony ograniczeniami (4), relacja $R(u, y; x)$ jest określona nierównościami (5) i funkcją (3). Zgodnie z ogólnym sformułowaniem problemu decyzyjnego w p. 2 – dla podanego przez użytkownika wymagania $T \leq \alpha$, czyli $y \in [0,\alpha]$ – problem alokacji można sformułować jako problem optymalizacji polegający na wyznaczeniu alokacji u^* maksymalizującej wskaźnik pewności własności miękkiej: „zbiór możliwych wartości T w przybliżeniu należy do $[0,\alpha]$" (tj. należy do $[0,\alpha]$ dla przybliżonej wartości zmiennej $\bar{x}$), czyli wymaganie dla zadanej wartości $\alpha > 0$ jest w przybliżeniu spełnione.

Problem optymalnej alokacji (rozdziału): Dla danych φ_i, h_i $(i \in \overline{1,k})$, N oraz α należy wyznaczyć

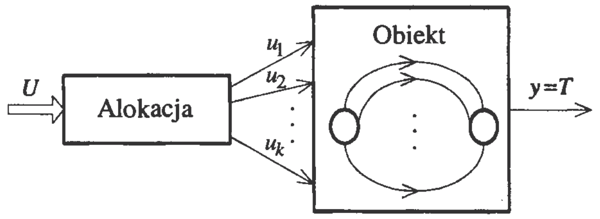
$$u^* = \arg\max_{u \in \overline{U}} v(u),$$

gdzie

$$v(u) = v\{D_T(u;\bar{x}) \,\tilde{\subseteq}\, [0,\alpha]\} = v[T(u,\bar{x}) \,\tilde{\leq}\, \alpha].$$

Własność miękka „$D_T(u;\bar{x}) \tilde{\subseteq} [0,\alpha]$" została tu oznaczona przez „$T(u,\bar{x}) \tilde{\leq} \alpha$", natomiast $D_T(u;x)$ oznacza zbiór możliwych wartości T dla ustalonego u, określony nierównością

$$T \leq \max_i x_i u_i . \qquad (6)$$



Rys. 1. Kompleks operacji równoległych jako obiekt decyzyjny.

Zgodnie z (5) i (6)

$$v(u) = v\{[T_1(u_1,\bar{x}_1) \tilde{\leq} \alpha] \wedge [T_2(u_2,\bar{x}_2) \tilde{\leq} \alpha] \wedge \ldots \wedge [T_k(u_k,\bar{x}_k) \tilde{\leq} \alpha]\}.$$

Zatem

$$u^* = \arg\max_{u \in \overline{U}} \min_i v_i(u_i), \qquad (7)$$

gdzie

$$v_i(u_i) = v[T_i(u_i,\bar{x}_i) \tilde{\leq} \alpha] = v[\bar{x}_i u_i \tilde{\leq} \alpha] = v[\bar{x}_i \,\tilde{\in}\, D_i(u_i)],$$
$$D_i(u_i) = [0, \alpha u_i^{-1}].$$

Ostatecznie, zgodnie z (2)

$$v_i(u_i) = \max_{x_i \in D_i(u_i)} h_i(x_i) \qquad (8)$$

i

$$u^* = \arg\max_{u \in \overline{U}} \min_i \max_{x_i \in D_i(u_i)} h_i(x_i).$$

Procedura wyznaczania optymalnej alokacji n_1^*, n_2^*,..., n_k^* jest zatem następująca (rys. 2)

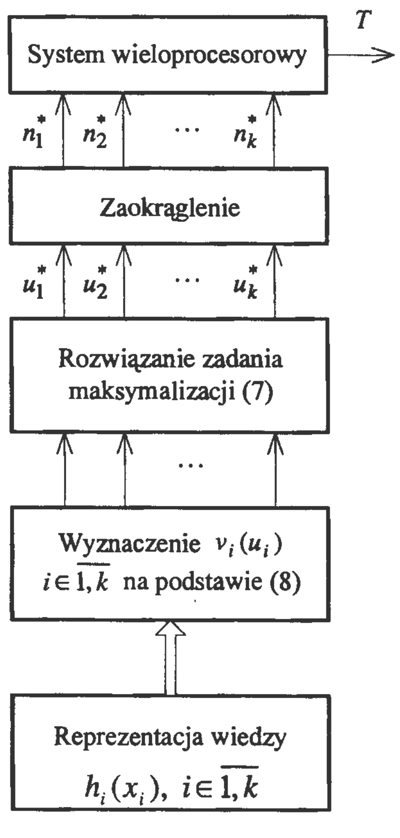
1. Wyznaczenie wskaźników pewności $v_i(u_i)$ według (8).
2. Wyznaczenie u^* w wyniku maksymalizacji (7).
3. Wyznaczenie $n_1^*,\ldots,n_k^*$ w wyniku zaokrąglenia $u_1^*,\ldots,u_k^*$.

Warto zauważyć, że jeśli α jest odpowiednio duże (wymaganie jest odpowiednio słabe), to $v(u^*) \triangleq v^* = 1$. Jeśli natomiast α jest zbyt małe (wymaganie jest za ostre), to $v(u) = 0$ dla każdego $u \in \overline{U}$, czyli nie istnieje alokacja, dla której wskaźnik pewności $v(u)$ jest większy od zera. Łatwo wykazać następujące

Twierdzenie.

Jeśli $0 < v(u) < 1$, to optymalna alokacja u spełnia układ równań

$$v_1(u_1) = v_2(u_2) = \ldots = v_k(u_k) \triangleq v. \quad \square \qquad (9)$$



Rys. 2. Struktura rozpatrywanego systemu sterowania rozdziałem obciążeń na podstawie reprezentacji wiedzy.

Twierdzenie wynika z faktu, że $x_i u_i$ jest rosnącą funkcją u_i, a zatem $v_i(u_i) = v(\bar{x}_i u_i \tilde{\leq} \alpha)$ jest rosnącą (ogólniej – niemalejącą) funkcją u_i. W konsekwencji decyzję u^* można otrzymać jako rozwiązanie układu równań (9) oraz $u_1 + u_2 + \ldots + u_k = N$. **Procedura** wyznaczania u^* dla α takiego, że $0 < v(u) < 1$ jest zatem następująca:

1. Dla otrzymanych poprzednio funkcji $v_i(u_i) \triangleq g_i(u_i)$ wyznaczyć funkcję odwrotną $u_i = g_i^{-1}(v_i)$.
2. Rozwiązać względem v równanie

$$\sum_{i=1}^{k} g_i^{-1}(v) = N. \tag{10}$$

3. Wstawić rozwiązanie v^* równania (10) do $g_i^{-1}(v_i)$ w miejsce v_i i wyznaczyć $u_i^* = g_i^{-1}(v^*)$.

W przypadku zmiennej niepewnej typu C rozważania są podobne (lecz bardziej skomplikowane), ze wskaźnikiem v_c, który należy wyznaczyć zgodnie z (1).

Dla przyjętej wartości $\bar{v}$, czyli wymaganego przez użytkownika *poziomu pewności* – wyznaczyć można najostrzejsze wymaganie, tj. minimalną możliwą wartość $\alpha \triangleq \bar{\alpha}$, dla której wymaganie $T \le \alpha$ jest w przybliżeniu spełnione z zadaną wartością wskaźnika pewności $\bar{v}$. W tym celu należy rozwiązać względem α równanie $v^*(\alpha) = \bar{v}$ i wyznaczyć optymalną alokację

$$\bar{u}_i = u_i^*(\bar{\alpha}), \qquad i = 1,2,...,k.$$

Procedura wyznaczania optymalnej alokacji dla zadanego poziomu pewności jest zatem następująca:
1. Wyznaczenie funkcji $u_i^*(\alpha)$ oraz $v^*(\alpha)$.
2. Rozwiązanie równania $v^*(\alpha) = \bar{v}$ względem α.
3. Wyznaczenie optymalnej alokacji $\bar{u}_i$ przez wstawienie $\bar{\alpha}$ do $u_i^*(\alpha)$ w miejsce α.

4. SPECJALNE PRZYPADKI. PRZYKŁADY

W realnych sytuacjach można się spodziewać, że ekspert scharakteryzuje czas wykonania elementarnego programu przez poszczególne procesory podając dla $i \in \overline{1,k}$ wartość x_i^*, dla której $h_i(x_i) = 1$ oraz przedział możliwych wartości przybliżonych: $x_i^* - d_i \le x_i \le x_i^* + d_i$. Można wówczas przyjąć trójkątny rozkład pewności przedstawiony na rys. 3, gdzie $d_i \le x_i^*$. Zgodnie z (8) łatwo wyznaczyć funkcje $v_i(u_i)$:

$$v_i(u_i) = \begin{cases} 1 & \text{dla} \quad u_i \le \dfrac{\alpha}{x_i^*} \\[2mm] \dfrac{1}{d_i}\left(\dfrac{\alpha}{u_i} - x_i^*\right) + 1 & \text{dla} \quad \dfrac{\alpha}{x_i^*} \le u_i \le \dfrac{\alpha}{x_i^* - d_i} \\[2mm] 0 & \text{dla} \quad u_i \ge \dfrac{\alpha}{x_i^* - d_i}. \end{cases} \tag{11}$$

Zastosowanie dla $k > 2$ procedur przedstawionych w p. 3 jest związane z trudnościami obliczeniowymi, które można częściowo pokonać stosując dekompozycję [17].

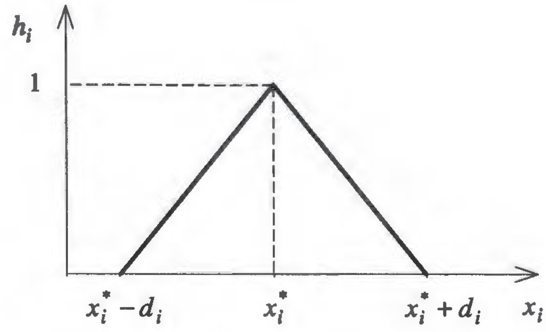
Problem upraszcza się przy założeniu

$$\frac{x_1^*}{d_1} = \frac{x_2^*}{d_2} = ... = \frac{x_k^*}{d_k} \triangleq \gamma. \tag{11a}$$

Łatwo pokazać, że zastosowanie procedury dla $0 < v(u) < 1$, obejmującej rozwiązanie równania (10), daje następujące rezultaty:

$$v^*(\alpha) = 1 + \gamma[\alpha N^{-1} \sum_{i=1}^{k} (x_i^*)^{-1} - 1], \tag{12}$$

$$u_i^*(\alpha) = \frac{\gamma\alpha}{x_i^*[v^*(\alpha) + \gamma - 1]}. \tag{13}$$



Rys. 3. Trójkątny rozkład pewności.

Korzystając z (12) można wyznaczyć warunek na α, przy spełnieniu którego $0 < v^* < 1$:

$$N(1-\gamma^{-1})[\sum_{i=1}^{k}(x_i^*)^{-1}]^{-1} < \alpha < N[\sum_{i=1}^{k}(x_i^*)^{-1}]^{-1}. \tag{14}$$

Równanie $v^*(\alpha) = \bar{v}$ daje rezultat

$$\bar{\alpha} = (\bar{v} + \gamma - 1)\gamma^{-1} N[\sum_{i=1}^{k}(x_i^*)^{-1}]^{-1}. \tag{15}$$

Wstawiając w (13) $\alpha = \bar{\alpha}$ oraz $v^*(\alpha) = \bar{v}$ otrzymamy optymalną alokację $\bar{u}_i$ dla przyjętego poziomu pewności $\bar{v}$.

Przykład 1

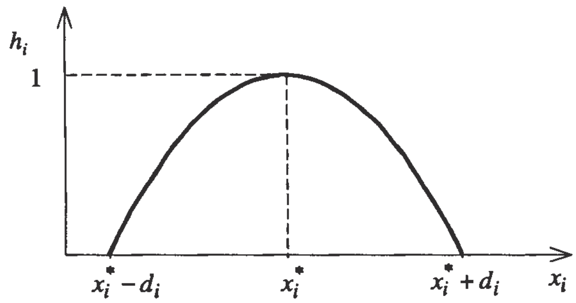
Dla trzech równolegle pracujących procesorów ($k = 3$) przyjmijmy następujące dane: $x_1^* = 1$, $x_2^* = \frac{1}{2}$, $x_3^* = \frac{1}{3}$, $\gamma = 2$, $N = 100$. Korzystając z (14) dostajemy $8.33 < \alpha < 16.67$, co oznacza, że wskaźnik pewności spełnienia wymagania $T \le 8.33$ równa się zero, natomiast przy odpowiedniej alokacji wskaźnik pewności spełnienia w przybliżeniu wymagania $T \le 16.67$ równa się 1.

Przyjmijmy poziom pewności $\bar{v} = 0.8$. Wówczas z (15) otrzymujemy $\bar{\alpha} = 15$, a po wstawieniu do (13):

$\bar{u}_1 \approx 16.67$, $\bar{u}_2 \approx 33.33$, $\bar{u}_3 \approx 50$. Po zaokrągleniu $n_1 = 17$, $n_2 = 33$, $n_3 = 50$. Dla tak wyznaczonej alokacji elementarnych programów do procesorów wymaganie na czas realizacji $T \le 15$ będzie w przybliżeniu spełnione z wskaźnikiem pewności 0.8. Dla każdej innej alokacji oraz $\alpha = 15$ wskaźnik pewności będzie niższy niż 0.8, lub dla wskaźnika $v = 0.8$ wartość α będzie większa niż 15. □

Dla pokazania zależności procedury wyznaczania alokacji od kształtu rozkładu pewności rozpatrzmy paraboliczny rozkład pewności przedstawiony na rys. 4. Wyznaczone zgodnie z (8) funkcje $v_i(u_i)$ są następujące:

$$v_i(u_i) = \begin{cases} 1 & \text{dla} \quad u_i \le \dfrac{\alpha}{x_i^*} \\ -\dfrac{1}{d_i^2}\left(\dfrac{\alpha}{u_i} - x_i^*\right)^2 + 1 & \text{dla} \quad \dfrac{\alpha}{x_i^*} \le u_i \le \dfrac{\alpha}{x_i^* - d_i} \\ 0 & \text{dla} \quad u_i \ge \dfrac{\alpha}{x_i^* - d_i}. \end{cases}$$



Rys. 4. Paraboliczny rozkład pewności.

Łatwo zauważyć, że obecnie stosowanie procedury opisanej w p. 5 dla $k > 2$ jest związane z trudnościami obliczeniowymi (nie można otrzymać analitycznego rozwiązania równania (10)). Można pokazać, że wyniki dla $k = 2$ są tu następujące:

1. Dla

$$\alpha \le \frac{N(x_1^* - d_1)(x_2^* - d_2)}{x_1^* - d_1 + x_2^* - d_2}$$

$v(u) = 0$ dla każdego u_1.

2. Dla

$$\frac{N(x_1^* - d_1)(x_2^* - d_2)}{x_1^* - d_1 + x_2^* - d_2} \le \alpha \le \frac{N x_1^* x_2^*}{x_1^* + x_2^*},$$

u_1^* jest pierwiastkiem równania

$$\frac{1}{d_1}\left(\frac{\alpha}{u_1} - x_1^*\right) = -\frac{1}{d_2}\left(\frac{\alpha}{N - u_1} - x_2^*\right)$$

spełniającym warunek

$$\frac{\alpha}{x_1^*} \le u_1^* \le \frac{\alpha}{x_1^* - d_1},$$

oraz $v(u^*) = v_1(u_1^*)$.

3. Dla

$$\alpha \ge \frac{N x_1^* x_2^*}{x_1^* + x_2^*}$$

$v(u^*) = 1$ dla dowolnego u_1 spełniającego warunek

$$N - \frac{\alpha}{x_2^*} \le u_1 \le \frac{\alpha}{x_1^*}.$$

Przy założeniu (11a) równanie z przypadku 2 przyjmuje postać

$$\frac{\alpha}{u_1 x_1^*} + \frac{\alpha}{(N - u_1)x_2^*} = 2$$

i w rezultacie

$$v(u^*) \triangleq v^* = 1 - \gamma^2\left(\frac{\alpha}{x_1^* u_1^*} - 1\right)^2.$$

Przykład 2

Dla $N = 40$, $\alpha = 30$, $x_1^* = 1.5$, $x_2^* = 2$, $\gamma = 1.5$, korzystając z rezultatów przedstawionych w przypadku 2 otrzymujemy $n_1^* = 28$, $n_2^* = 12$ oraz $v^* = 0.8$, co oznacza, że do pierwszego procesora należy przydzielić 28 zadań elementarnych, a do drugiego – 12. Wówczas wymaganie dotyczące czasu realizacji będzie w przybliżeniu spełnione ze wskaźnikiem pewności 0.8.

5. STEROWANIE PRZYDZIAŁEM ZADAŃ DO PROCESORÓW

Rozpatrzmy obecnie problem rozdziału pomiędzy równolegle pracujące procesory różnych programów (o różnych czasach realizacji), których nie można dzielić na elementarne części, a więc nie można stosować zaokrąglania, o którym była mowa w p. 3 i 4. Jest to zatem *dyskretny* problem przydziału programów (zadań) do procesorów, czyli problem szeregowania bez ograniczeń kolejnościowych. Przydział jest zdeterminowany macierzą

$$[c_{ij}]_{\substack{i=1,\dots,k \\ j=1,\dots,M}} \triangleq P,$$

gdzie $c_{ij} = 1$ jeśli j-e zadanie jest przydzielone i-temu procesorowi, $c_{ij} = 0$ w przeciwnym razie. Znany problem deterministyczny polega na wyznaczeniu P^* minimalizującego czas wykonania wszystkich programów, czyli

$$P^* = \arg\min_P \max_i T_i = \arg\min_P \max_i \sum_{j=1}^{M} c_{ij}\tau_{ij}, \quad (16)$$

gdzie τ_{ij} oznacza czas realizacji j-go programu na i-tym procesorze, T_i – czas realizacji programów przydzielonych i-temu procesorowi. Uwzględnienie niepewności i zastosowanie zmiennych niepewnych analogiczne jak w poprzednim punkcie – opiera się na założeniu, że $\tau_{ij} \le x_i \delta_{ij}$ czyli $T_i \le x_i \gamma_i$, gdzie

$$\gamma_i = \sum_{j=1}^{M} c_{ij} \delta_{ij} ,$$

wartości δ_{ij} są znane, x_i jest wartością zmiennej niepewnej scharakteryzowanej przez eksperta w formie rozkładu pewności $h_i(x_i)$. W konsekwencji problem polega na wyznaczeniu P^* maksymalizującego wskaźnik pewności tego, że wymaganie $T = \max T_i \le \alpha$ jest „w przybliżeniu spełnione", czyli

$$P^* = \arg \max_P \min_i v_i(P) , \tag{17}$$

gdzie

$$v_i(P) = v[T_i(\bar{x}_i, P) \mathrel{\tilde{\le}} \alpha] = \max_{0 \le x_i \le \alpha \gamma_i^{-1}(P)} h_i(x_i) .$$

Dla $0 < v_i < 1$

$$P^* = \arg \max_P \min_i h_i(\alpha \gamma_i^{-1}) \tag{18}$$

przy założeniu, że $h_i(x_i)$ jest funkcją rosnącą dla $x_i \in [x_{\min}, x_i^*]$, gdzie $x_{\min}$, x^* są takie, że $h_i(x_i^*) = 1$, $h_i(x_i) = 0$ dla $x_i \le x_{\min}$. Problemy (16) oraz (17) lub (18) są podobne i mogą być rozwiązane przez zastosowanie odpowiednich algorytmów programowania dyskretnego. Podobieństwo jest większe, a procedura otrzymania rozwiązania na ogół łatwiejsza dla innej wersji problemu, w której zakłada się, że $\tau_{ij} \ge x_i \delta_{ij}$ i należy minimalizować wskaźnik pewności $\hat{v}$ tego, że nierówność $T \ge \alpha$ jest „w przybliżeniu spełniona", czyli

$$P^* = \arg \min_P v[T(\bar{x}, P) \mathrel{\tilde{\ge}} \alpha] = \arg \max_P \hat{v}(P) ,$$

gdzie

$$\hat{v}(P) = v[(T_1(\bar{x}_1, P) \mathrel{\tilde{\ge}} \alpha) \vee \ldots \vee (T_k(\bar{x}_k, P) \mathrel{\tilde{\ge}} \alpha)] .$$

Zatem

$$P^* = \arg \min_P \max_i \hat{v}_i(P) ,$$

gdzie

$$\hat{v}_i(P) = v[T_i(\bar{x}_i, P) \mathrel{\tilde{\ge}} \alpha] = \max_{x_i \ge \alpha \gamma_i^{-1}(P)} h_i(x_i) .$$

Przy odpowiednich założeniach

$$P^* = \arg \min_P \max_i h_i(\alpha \gamma_i^{-1}) .$$

W przypadku deterministycznym ($T_i = x_i \gamma_i$, x_i jest znane) (16) sprowadza się do

$$P^* = \arg \min_P \max_i x_i \gamma_i .$$

Tak więc $x_i \gamma_i$ w przypadku deterministycznym zastąpione jest przez $h_i(\alpha \gamma_i^{-1})$ w rozpatrywanym przypadku z niepewnością. W obu przypadkach można zatem stosować ten sam algorytm do wyznaczania

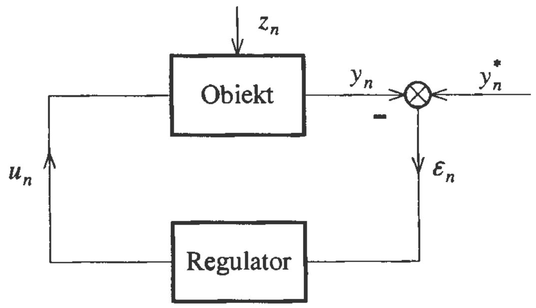
$$P^* = \arg \min_P \max_i w_i(P) .$$

Dla ustalonego P, w pierwszym przypadku wyznacza się $w_i(P) = x_i \gamma_i$, a w drugim – $w_i(P) = h_i(\alpha \gamma_i^{-1})$. Warto zauważyć, że uwzględnienie zarówno $v_i(P)$ jak i $\hat{v}_i(P)$ oznacza zastosowanie zmiennej niepewnej typu C (por. (1)), czyli wskaźnika pewności

$$v_{ci}(P) = \frac{1}{2}[v_i(P) + 1 - \hat{v}_i(P)].$$

6. STEROWANIE RUCHEM W SIECIACH KOMPUTEROWYCH

Celem sterowania ruchem w sieci komputerowej jest odpowiednie wykorzystanie zasobów sieci oraz przeciwdziałanie powstawaniu stanów niepożądanych (przeciążeń i blokad) prowadzących do degradacji jakości usług [29]. Dotyczy to w szczególności sieci pakietowych (szerzej – z komutacją jednostek danych). Znaczenie bardzo już dzisiaj obszernej problematyki sterowania ruchem rośnie wraz z rozwojem Internetu. Jest to z reguły sterowanie w systemie zamkniętym z zadanym wyjściem obiektu (rys. 5), w którym wielkości u_n, y_n, z_n i poszczególne bloki składające się na obiekt i regulator mogą mieć różną interpretację praktyczną w zależności od konkretnej funkcji i mechanizmu sterowania. Niepewność parametryczna oznacza tu wystąpienie nieznanych parametrów w modelu obiektu, a zastosowanie zmiennych niepewnych opiera się na założeniu, że parametry te charakteryzowane są w sposób subiektywny przez eksperta w formie rozkładów pewności. Opis obiektu (który może mieć charakter relacyjny odbiegający od tradycyjnych modeli) wraz z rozkładami pewności składa się na reprezentację wiedzy w rozpatrywanej sytuacji, a w konsekwencji problem decyzyjny jest tu problemem *sterowania ruchem w sieci na podstawie niepewnej wiedzy*, której opis wykorzystuje formalizm zmiennych niepewnych.



Rys. 5. Schemat blokowy zamkniętego systemu sterowania.

6.1. Stabilność

Do analizy stabilności i stabilizacji systemu sterowania ruchem w sieci wykorzystać można ogólną metodę analizy stabilności i stabilizacji z zastosowaniem zmiennych niepewnych [3, 11]. Niech $x \in X$ oznacza niepewny parametr wektorowy w obiekcie, $b \in B$ – wektorowy parametr algorytmu sterowania, i niech własności $W(x,b)$, $G(x,b)$ oznaczają odpowiednio wystarczający i konieczny warunek stabilności. Wówczas wskaźnik pewności v_s tego, że system jest „w przybliżeniu stabilny" (tzn. stabilny dla przybliżonych wartości parametrów obiektu) można oszacować za pomocą nierówności $v_w \le v_s \le v_g$, gdzie v_w i v_g oznaczają odpowiednio wskaźniki pewności spełnienia w przybliżeniu własności W i G, tj.

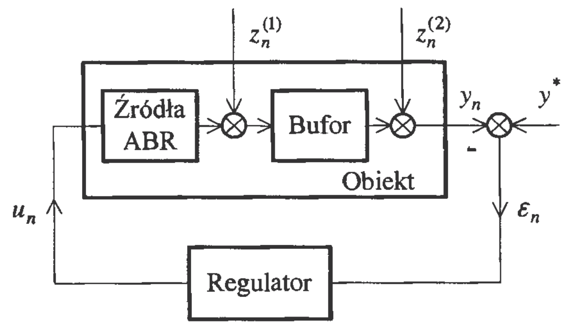
$$v_w(b) = \max_{x \in D_w(b)} h(x), \qquad v_g(b) = \max_{x \in D_g(b)} h(x),$$

gdzie $D_w(b) = \{x \in X : W(x,b)\}, D_g(b) = \{x \in X : G(x,b)\}$. Przez odpowiedni dobór b można wpływać na górne i dolne oszacowanie v_s. Rozpatrzmy kilka przykładów.

W pracy [2] (i w wielu późniejszych) analizuje się stabilność systemu sterowania przeciążeniem w sieci ATM (rys. 6), w którym u_n jest szybkością transmisji źródła ABR, y_n – liczbą komórek ABR w kolejce wyjściowej portu węzła sieci, $z_n^{(1)}$ – liczbą komórek trafiających do bufora nie podlegających sterowaniu, $z_n^{(2)}$ – liczbą komórek opuszczających bufor w n-tym takcie sterowania. Obiekt i regulator opisywane są odpowiednimi równaniami różnicowymi (w różnych wersjach). W pracach [36, 37, 38] zastosowano do analizy stabilności i stabilizacji formalizm zmiennych niepewnych i zbadano wpływ parametrów rozkładów pewności na wartości v_w i v_g dla różnych przypadków równań obiektu i regulatora.

Podobna koncepcja sterowania przeciążeniem opisana jest w [1]. Błąd regulacji $\varepsilon_n = y_n^* - y_n$ jest różnicą pomiędzy wymaganym i rzeczywistym obciążeniem (*traffic load*) i przy pominięciu zakłócenia ($y_n \approx u_n$) równanie systemu zamkniętego jest następujące: $u_{n+d+1} - u_{n+d} + bu_n = by_n^*$, gdzie b jest wzmocnieniem regulatora, d jest opóźnieniem, a u_n – wyznaczoną przez regulator szybkością transmisji. Zastosowanie kryterium Hurwitza do równania $z^{d+1} - z^d + b = 0$ (po odpowiedniej transformacji) daje następujący przybliżony warunek stabilności $b < 2(d+1)^{-1}$. Jeśli $d = x$ jest wartością zmiennej niepewnej $\bar{x}$ z rozkładem $h(x)$, to

$$v_s = v\{\bar{x} \,\tilde{\in}\, [0, 2b^{-1} - 1]\} = \max_{0 < x < 2b^{-1} - 1} h(x).$$



Rys. 6. Schemat blokowy rozpatrywanego systemu sterowania przeciążeniem w sieci.

Oczywiście x jest wartością dyskretnej zmiennej niepewnej i dane są wartości $h(x)$ dla liczb naturalnych. Rozważania można rozszerzyć dla wielu źródeł z opóźnieniami $d_1 = x_1, \ldots, d_k = x_k$ opisanymi odpowiednio przez $h_1(x_1), \ldots, h_k(x_k)$. Obecnie, zgodnie z [1], $d = \max\{d_1, d_2, \ldots, d_k\}$ i $v_s = \min_i v_{is}$, gdzie

$$v_{is} = v\{\bar{x}_i \,\tilde{\in}\, [0, 2b^{-1} - 1]\}.$$

Przytoczone przykłady (jak i wiele innych w literaturze) dotyczą w zasadzie sieci ATM, ale identyczna metodyka badania stabilności może być zastosowana dla sterowania przeciążeniem w przypadku TCP/IP. Np. w pracy [24] opisano model procesu sterowania przeciążeniem w sieci pakietowej, który może być stosowany zarówno dla stałych (ATM) jak i zmiennych (IP) rozmiarów pakietów. Przedstawione tam liniowe równanie zamkniętego systemu sterowania może być podstawą wyznaczania znanych warunków stabilności, a w konsekwencji analizy stabilności z zastosowaniem zmiennych niepewnych w przypadku nieznanych parametrów obiektu, charakteryzowanych przez eksperta.

6.2. Optymalizacja statyczna

Wiele zadań sterowania ruchem w sieci sprowadza się do problemów optymalnej alokacji zasobów, a więc do typowych problemów optymalizacji statycznej, które polegają na maksymalizacji określonej funkcji użyteczności z ograniczeniami uwzględniającymi ograniczenia w sieci oraz założony poziom jakości usług dostarczanych użytkownikom sieci. Postać funkcji użyteczności przyjmuje się zwykle mniej lub bardziej arbitralnie, a wyznaczenie odpowiednich decyzji wymaga dokładnej znajomości wartości parametrów tej funkcji. Zastosowanie zmiennych niepewnych dotyczy bardziej realnych sytuacji, gdy ekspert podaje zależności pomiędzy wartościami zasobów i odpowiadającymi im składnikami użyteczności w postaci nierówności (składających się na relacyjną reprezentację wiedzy o obiekcie sterowania), a dla nieznanych parametrów w tych nierównościach podaje rozkłady pewności. Wówczas zadanie decyzyjne można formułować i rozwiązywać podobnie, jak przedstawiono to w p. 2 i 3.

Rozpatrzmy dla przykładu problem opisany w [30] i dotyczący inżynierii ruchu w Internecie. Niech n oznacza liczbę typów żądań, n_i liczbę tras dostępnych dla żądań typu i oraz c_{ij} całkowitą szybkość transmisji danych dla żądań typu i używających trasy (ścieżki) j. Deterministyczny problem optymalizacji statycznej polega tu na wyznaczeniu dla wszystkich (i, j) wartości c_{ij} maksymalizujących funkcję użyteczności

$$y=\sum_{i=1}^{n}\varphi_i(c_{i1},\dots,c_{in_i})$$

z ograniczeniami uwzględniającymi pojemność łącz oraz ograniczeniami odpowiadającymi różnym klasom obsługi (CoS) dla różnych podzbiorów zbioru typów żądań $\{1,2,\dots,n\}$. Są to ograniczenia równościowe lub nierównościowe dotyczące sum składników c_{ij}, np. ograniczenia

$$\sum_{j=1}^{n_i} c_{ij}\ge C_i\,,$$

w przypadku klasy MRGS (Minimum Rate Guaranteed Service), dla wszystkich i należących do odpowiedniego podzbioru.

Podobny formalnie problem znaleźć można np. w pracy [25] dotyczącej przepływu wieloskładnikowego (*multicast flow*) i realizacji usług warstwy sesji. Problem alokacji sprowadza się tu do wyznaczenia wartości c_{ij} maksymalizujących funkcję użyteczności

$$y=\sum_{i\in S}\sum_{j\in R_i}\varphi_{ij}(c_{ij})$$

przy ograniczeniach

$$\sum_{i\in S_l}\max_{(i,j)\in V_{il}} c_{ij}\le C_l;\quad l\in\overline{1,L},\quad \forall i\in S,\ j\in R_i,$$

gdzie L oznacza liczbę łącz, C_l – pojemność l-go łącza, S – zbiór sesji, R_i – zbiór odbiorców odpowiadających sesji i, S_l – zbiór sesji przechodzących przez łącze l, V_{il} – zbiór wszystkich odbiorców sesji i korzystających z l, c_{ij} – szybkość transmisji danych.

Wymienione powyżej i liczne inne przykłady, w sytuacji niepewności, o której mowa powyżej (nierówności i niepewne parametry) – można ująć ogólnie następującą reprezentacją wiedzy:

$$y=y_1+y_2+\dots+y_k,\quad y_i\le\varphi_i(u_i,x_i),\quad i\in\overline{1,k}$$

uzupełnioną o podane przez eksperta rozkłady pewności $h_i(x_i)$. Wówczas (por. p. 2) można sformułować i rozwiązać problem decyzyjny polegający na wyznaczeniu optymalnej alokacji $u^*=(u_1^*,\dots,u_k^*)$ maksymalizującej wskaźnik pewności $v[y_{\min}(\overline{u})\,\tilde{\ge}\,\alpha]$, gdzie $y_{\min}(u)$ jest najmniejszą możliwą wartością y – z uwzględnieniem ograniczeń nałożonych na u. W konsekwencji można też wyznaczyć takie wartości α i u, dla których maksymalna wartość v (tj. największy wskaźnik pewności tego, że wymaganie $y\ge\alpha$ jest w przybliżeniu spełnione) równa się wymaganemu *poziomowi pewności* $\overline{v}$. Podejście takie dotyczy oczywiście ogólnego problemu alokacji zasobów (nie tylko w zadaniu sterowania ruchem w sieci) i jest rozpatrywane w [16].

6.3. Optymalizacja dynamiczna

Równania różnicowe, o których mówiliśmy w pierwszej części obecnego punktu mogą służyć nie tylko do analizy stabilności, lecz również do oceny jakości i projektowania parametrycznego (wyboru parametru regulatora b, uwzględniającego jakość) odpowiedniego systemu sterowania ruchem w sieci, z wykorzystaniem ogólnego podejścia opartego na zmiennych niepewnych [7, 21, 17, 23]. Dla wskaźnika jakości

$$Q=\sum_{n=0}^{\infty}\varepsilon_n^2\triangleq\Phi(b,x)$$

i wymagania $Q\le\alpha$ można sformułować i rozwiązać problem decyzyjny przedstawiony w p. 2, tj. wyznaczyć $b^*(\alpha)$ maksymalizujące

$$v\{\Phi(b,\overline{x})\,\tilde{\in}\,[0,\alpha]\}=v[\Phi(b,\overline{x})\,\tilde{\le}\,\alpha]\triangleq v(b,\alpha)$$
$$=\max_{x\in D_x(b)} h(x)\,,$$

gdzie $D_x(b)=\{x\in X:\Phi(b,x)\le\alpha\}$. Wskaźnik $v(b,\alpha)$ jest rosnącą (ogólnie – niemalejącą) funkcją α. Dla zadanego *poziomu pewności* $\overline{v}$ można zatem wyznaczyć najmniejszą wartość $\alpha\triangleq\overline{\alpha}$ (najmocniejsze wymaganie) rozwiązując względem α równanie $v[b^*(\alpha),\alpha]=\overline{v}$, a następnie wyznaczyć optymalny parametr regulatora $\overline{b}=b^*(\overline{\alpha})$.

Dynamiczny obiekt sterowania ruchem w sieci może być również opisany za pomocą wektora stanu. Rozpatrzmy dla przykładu znany problem sterowania w systemie, w którym losowy charakter żądań dostępu przez wielu użytkowników (źródeł) do współdzielonego kanału transmisyjnego może powodować kolizje i konieczność retransmisji (prosty i szczelinowy schemat ALOHA oraz odpowiednie metody dostępu stosowane powszechnie w sieciach komputerowych, takie jak CSMA, tj. *Carrier Sense Multiple Access*). Na przykład w pracy [26] rozpatruje się szczelinowy schemat ALOHA, w którym probabilistyczny obiekt sterowania opisany jest w formie prawdopodobieństw warunkowych

$$P(x_{n+1}=i\,|\,x_n=j;u_n) \triangleq p_{ij}(u_n), \qquad i,j \in \overline{1,M},$$

gdzie x_n jest jednowymiarowym stanem oznaczającym liczbę użytkowników „z zaległościami" (*backlogged or blocked users*) na początku n-ej szczeliny, $u_n \in [0,1]$ jest jednowymiarową decyzją sterującą dotyczącą retransmisji („prawdopodobieństwo" retransmisji, według terminologii stosowanej w [26]), M jest liczbą możliwych stanów, p_{ij} oznacza prawdopodobieństwo przejścia ze stanu j do stanu i. Po wprowadzeniu chwilowego wskaźnika jakości $\varphi(x_n)$ oraz wskaźnika globalnego

$$Q_N=\sum_{n=1}^{N}\varphi(x_n)$$

otrzymujemy dobrze znany probabilistyczny problem optymalizacji wieloetapowego procesu decyzyjnego, polegający na wyznaczeniu ciągu decyzji $u_0,...,u_{N-1}$ minimalizujących wartość oczekiwaną $\mathrm{E}(Q_N)$. Stosując programowanie dynamiczne w wersji probabilistycznej wyznaczyć można algorytm sterowania w systemie zamkniętym $u_n=\Psi_n(x_n)$. W pracy [26] stosuje się znane metody teorii sterowania i decyzji dla różnych wersji tego problemu (dla *optimal power and retransmission control* i dla przypadku, gdy wartość x_n należy estymować na podstawie dostępnej obserwacji wyjścia obiektu). Słabością tego typu podejść jest konieczność znajomości wymienionych powyżej prawdopodobieństw warunkowych. Zastosowanie zmiennych niepewnych oznacza tu zastąpienie opisu probabilistycznego przez formułowany przez eksperta opis w formie warunkowych rozkładów pewności $h(x_{n+1}\,|\,x_n;u_n)$ $=v(\bar{x}_{n+1}\cong x_{n+1}\,|\,\bar{x}_n=x_n;u_n)$, czyli $h(i\,|\,j;u_n)$ $=v(\bar{x}_{n+1}\cong i\,|\,\bar{x}_n=j)$ oraz minimalizację wartości średniej Q_N.

7. UWAGI KOŃCOWE

Do sterowania systemami komputerowymi stosować można znane ogólne metody podejmowania decyzji i sterowania, z uwzględnieniem specyfiki występujących tu obiektów i funkcji sterowania. Na wybranych przykładach i zadaniach pokazano przydatność formalizmu zmiennych niepewnych do sterowania systemami komputerowymi opartego na niepewnej wiedzy charakteryzowanej przez eksperta. Przykłady te (których wybór jest z pewnością dyskusyjny) wskazują na możliwości opisywanych podejść w przypadkach bardziej złożonych i być może bardziej reprezentatywnych dla aktualnej techniki komputerowej. Na koniec warto zwrócić uwagę na możliwość i celowość zastosowania w rozpatrywanych przypadkach koncepcji uczenia w powiązaniu ze zmiennymi niepewnymi oraz rozszerzenia rozważań dla systemów o złożonej strukturze i rozproszonej wiedzy [17, 9, 18].

APPLICATIONS OF UNCERTAIN VARIABLES TO CONTROL PROBLEMS IN COMPUTER SYSTEMS

Abstract: The paper is concerned with selected problems of the control of computer systems, based on an uncertain knowledge. The computer systems may be specific control plants for which applications of non-probabilistic descriptions of the uncertainty, given by an expert are especially needed and useful. In recent years, theory of so called uncertain variables has been developed as a tool for the analysis and decision making in uncertain systems characterized by an expert. The purpose of this paper is to present a brief review of the applications of the uncertain variables in selected examples of computer systems control. The first part of the paper concerns the load control (tasks allocation) in a group of parallel processors. The second part is devoted to the traffic control in computer networks, in particular – to the congestion control.

Literatura

[1] Aweya J., Ouellette M., Montuno D.Y. (2004) Design and stability analysis of a rate control algorithm using the Routh-Hurwitz stability criterion. *IEEE/ACM Trans. on Networking*, 12, 719-732.

[2] Benhomed L., Meerkov S. (1993) Feedback control of congestion in packet switching networks: the case of single congested node. *IEEE/ACM Trans. on Networking*, 1, 839-848.

[3] Bubnicki Z. (2000) General approach to stability and stabilization for a class of uncertain discrete non-linear systems. *Int. J. Control*, 73, 1298-1306.

[4] Bubnicki Z. (2000) Learning process in an expert system for job distribution in a set of parallel computers. *Proc. of the 14th International Conference on Systems Engineering, Coventry*, **1**, 78-83.

[5] Bubnicki Z. (2001) Uncertain variables and their application to decision making. *IEEE Trans. on SMC, Part A: Systems and Humans*, 31, 587-596.

[6] Bubnicki Z. (2001) Uncertain variables and their applications for a class of uncertain systems. *Int. J. Systems Science*, 32, 651-659.

[7] Bubnicki Z. (2002) Uncertain variables and their applications for control systems. *Kybernetes*, 31, 1260-1273.

[8] Bubnicki Z. (2002) *Uncertain Logics, Variables and Systems*. Springer, Berlin, London, New York.

[9] Bubnicki Z. (2002) Application of uncertain variables to decision making in a class of distributed computer systems. *Proc. of the 17th IFIP World Computer Congress, Montreal*, vol. "Intelligent Information Processing", Kluwer Academic Publishers, Norwell, MA, 261-264.

[10] Bubnicki Z. (2002) Learning process in a class of computer integrated manufacturing systems with parametric uncertainties. *J. Intelligent Manufacturing*, 13, 409-415.

[11] Bubnicki Z. (2003) Stability and stabilization of discrete systems with random and uncertain parameters. *Proc. of the 15th IFAC World Congress, Barcelona,* E, Pergamon, Oxford, 193-198.

[12] Bubnicki Z. (2003) Application of uncertain variables to task and resource distribution in complex computer systems. *Computer Aided Systems Theory - EUROCAST 2003. Lecture Notes in Computer Science*, **2809**, Springer, Berlin, Heidelberg, 38-49.

[13] Bubnicki Z. (2003) Application of uncertain variables to control of task allocation in a multiprocessor system. *Modelling, Identification and Control*, Acta Press, Zurych, 489-494.

[14] Bubnicki Z. (2003) Application of uncertain variables in a class of control systems with uncertain and random parameters. *Proc. of the European Control Conference, Cambridge.*

[15] Bubnicki Z. (2003) Application of uncertain variables and learning algorithm to task allocation in multiprocessor systems. *Proc. of the 8th International Symposium on Artificial Life and Robotics, Oita*, **1**, I19-I22.

[16] Bubnicki Z., Orski D. (2004) Application of uncertain variables to knowledge-based resource allocation in a group of research units. *Systems Science*, 30, 75-87.

[17] Bubnicki Z. (2004) *Analysis and Decision Making in Uncertain Systems.* Springer, Berlin, London, New York.

[18] Bubnicki Z. (2004) Application of uncertain variables and learning algorithms in a class of distributed knowledge systems. *Proc. of the 18th IFIP World Computer Congress*, vol. "The Symposium on Professional Practice in AI", Laboratoire d'Automatique et d'Analyse des Systemes, *Toulouse*, 111-120.

[19] Bubnicki Z. (2004) Uncertain variables and systems – new problems and results. *Lecture Notes in Artificial Intelligence*, **3070**, Springer, Berlin, London, New York, 17-29.

[20] Bubnicki Z. (2005) Uncertain variables and their applications in knowledge-based decision systems. New Results. *Int. J. Intelligent Systems* (w druku).

[21] Bubnicki Z. (2005) *Teoria i algorytmy sterowania.* PWN, Warszawa, II wyd.

[22] Bubnicki Z. (2005) *Modern Control Theory.* Springer, Berlin, London, New York.

[23] Bubnicki Z. (2005) Application of uncertain variables to stabilization and parametric optimization of uncertain dynamic systems. *Proc. of the 16th IFAC World Congress, Prague.*

[24] Cavendish D., Gerla M., Mascolo S. (2004) A control theoretical approach to congestion control in packet networks. *IEEE/ACM Trans. on Networking*, 12, 893-906.

[25] Deb S., Srikant R. (2004) Congestion control for fair resource allocation in networks with multicast flows. *IEEE/ACM Trans. on Networking*, 12, 274-285.

[26] del Angel G., Fine T.L. (2004) Optimal power and retransmission control policies for random access systems. *IEEE/ACM Trans. on Networking*, 12, 1156-1166.

[27] Dębicki T. (2003) Zastosowanie zmiennych niepewnych i losowych w problemie sterowania alokacją zadań w grupie równolegle pracujących procesorów. *Mat. Konf. Inżynieria wiedzy i systemy ekspertowe*, **2**, Wyd. PWr, Wrocław, 217-224.

[28] Dębicki T. (2005) Zastosowanie zmiennych niepewnych i losowych w problemie sterowania alokacją zadań w systemie wieloprocesorowym. *Materiały XV Krajowej Konferencji Automatyki, Warszawa.*

[29] Grzech A. (2002) *Sterowanie ruchem w sieciach teleinformatycznych.* Wyd. PWr, Wrocław.

[30] Lagoa C.M., Hao C., Movsichoff B.A. (2004) Adaptive control algorithms for decentralized optimal traffic engineering in the Internet. *IEEE/ACM Trans. on Networking*, 12, 415-428.

[31] Orski D., Sugisaka M. (2004) Application of uncertain variables in an adaptive decision system for a complex of operations. *Proc. of the 15th International Conference on Systems Science, Wrocław*, **2**, 307-314.

[32] Pedrycz W. (2004) *Knowledge-Based Clustering.* John Wiley & Sons, Hoboken, New Jersey.

[33] Rutkowski L. (2004) *New Soft Computing Techniques for System Modelling, Pattern Classification and Image Processing.* Springer, Berlin, London, New York.

[34] Siwek L. (2000) Control system of calls admission in computer network using fuzzy approach. *Proc. of the 17th IMACS World Congress "Scientific Computation, Applied Mathematics and Simulation", Lausanne*, IMACS, New Brunswick, 509-511.

[35] Toledo R. (2004) A study of a simple transportation problem applying uncertain variables. *Proc. of the 15th International Conference on Systems Science, Wrocław*, **2**, 339-345.

[36] Turowska M. (2004) Application of uncertain variables to stability analysis and stabilization for ATM ABR congestion control systems. *Proc. of the International Conference on Enterprise Information Systems, Porto*, **2**, INSTICC Press, Porto, 523-526.

[37] Turowska M. (2004) Application of uncertain variables to stability analysis of nonlinear congestion control system in ATM networks. *Proc. of the 15th International Conference on Systems Science, Wrocław*, **3**, 157-164.

[38] Turowska M. (2005) Sterowanie przeciążeniem w sieci ATM w warunkach niepewności. *Materiały XV Krajowej Konferencji Automatyki, Warszawa.*

Instytut Badań Systemowych
Polskiej Akademii Nauk

ISBN 83-89475-02-2