

PAWEŁ REMBELSKI (Warszawa)

Markowski model dyskretnego algorytmu mrówkowego

Streszczenie. Algorytm inspirowany naturą zaproponowany przez M. Dorigo został w pracy przedefiniowany jako łańcuch Markowa. Istotą rozwiniętego modelu jest wyznaczenie wszystkich podstawowych obiektów jego działania, wskazanie na skończoność przestrzeni stanów oraz wyprowadzenie wyrażeń na składowe podstawowego operatora, macierzy przekształcenia w pojedynczym kroku. Jednocześnie sformułowano warunki zachowania się asymptotycznego, by uzyskać ważną własność punktowej asymptotycznej zbieżności.

Słowa kluczowe: algorytmy optymalizacyjne, algorytmy inspirowane naturą, dyskretny algorytm mrówkowy, feromon, łańcuch Markowa, punktowa zbieżność.

Wprowadzenie. Problemy optymalizacyjne, polegające na znajdowaniu minimum czy maksimum danej funkcji rzeczywistej, pełnią ważną rolę we współczesnych zastosowaniach matematyki do zagadnień modelowych w fizyce i rozwiązywania problemów praktycznych stawianych przez współczesną technikę. Wiele rozwiązań konkretnych problemów nie byłoby możliwe do uzyskania, gdyby nie zostały odpowiednio rozwinięte metody optymalizacji funkcji rzeczywistych. Wszystkie **zadania optymalizacji** mają w całej swej różnorodności kilka wspólnych cech. Każde z nich jest scharakteryzowane przez pewien zbiór $\mathcal{R} \subset \mathcal{Y}$, nazywany przestrzenią rozwiązań lub przestrzenią poszukiwań, o której zakładamy, że jest podzbiorem zwartym i ograniczonym pewnej przestrzeni polskiej (metrycznej i ośrodkowej) \mathcal{Y} takiej, że każdy punkt jest punktem skupienia. Jeśli przestrzeń \mathcal{Y} ma strukturę liniową, to wtedy zakłada się, że jest skończenie wymiarowa. Wśród elementów tego zbioru szukamy rozwiązań zadania. Ponadto na przestrzeni \mathcal{R} jest określona **funkcja jakości** (funkcja oceny, funkcja celu) $\|\cdot\| : \mathcal{R} \rightarrow \mathbb{R}$, przyporządkowująca elementowi $\omega \in \mathcal{R}$ jego ocenę ze zbioru liczb rzeczywistych \mathbb{R} . W funkcji jakości $\|\cdot\|$ zawieramy różne kryteria, których spełnienia wymagamy od pożądanego rozwiązania. Funkcja celu nie musi być gładka. Dzięki niej możemy porównywać otrzymane rozwiązania. Brak różniczkowalności, a często nawet ciągłości funkcji celu sprawia, że klasyczne narzędzia analizy

matematycznej nie mogą mieć tutaj zastosowania. Z drugiej strony, jeśli zbiór \mathcal{R} jest mocy mniejszej niż continuum ale o ogromnej liczbie elementów, powiedzmy rzędu 2^{64} , jego sekwencyjne przeszukiwanie w skończonym czasie jest nieefektywne.

Znalezienie zadowalającego rozwiązania dla jednego z takich problemów wymaga dużej wiedzy o zadaniu i znajomości algorytmów optymalizacyjnych. Należy zdawać sobie sprawę z tego, że rozwiązywanie praktycznych problemów jest w dużej mierze sztuką, w której pełni znaczącą rolę nie tyle praktyka, co konieczność korzystania z metod heurystycznych.

Zazwyczaj sam fakt wykorzystywania w obliczeniach komputerów powoduje, że w efekcie obliczeń znajdujemy jedynie pewne przybliżone rozwiązanie. A to dlatego, że liczby w komputerze są reprezentowane przez skończony ciąg bitów, co powoduje kumulowanie się błędów zaokrągleń. Ponadto, w obliczeniach komputerowych nie realizujemy zazwyczaj procesów ciągłych, a jedynie ich dyskretne przybliżenia i to przez odpowiednią skończoną liczbę dyskretnych kroków.

Obok powyższego pojawiają się inne problemy: niedoskonałość metod numerycznych czy duża złożoność obliczeniowa. W takich przypadkach zastosowanie znanego dokładnego algorytmu nie jest możliwe i jesteśmy zmuszeni poszukiwać innych podejść do rozwiązywanego problemu optymalizacyjnego. Pewną klasą algorytmów, które dowiodły swojej przydatności w najtrudniejszych zadaniach praktycznych, to **algorytmy probabilistyczne**. Algorytmy te znajdują rozwiązanie problemu (zwykle suboptymalne) z prawdopodobieństwem mniejszym od 1. Natomiast ich przebieg działania zależy nie tylko od właściwości zadania, ale także od czynników losowych.

Wśród algorytmów probabilistycznych dużą klasę zajmują w informatyce **algorytmy inspirowane naturą** – ogólnie biologią. Jest wiele algorytmów, których pochodzenie wiązane jest z inspiracją naturą, choć czasami trudno bezwzględnie stwierdzić, czy dany algorytm powstał jako próba naśladownictwa natury, czy też powstał jako pewna heurystyka (czyli w sposób zdroworozsądkowy), a później skojarzył się z istniejącym procesem w biologii i został analogicznie nazwany. Algorytmy inspirowane naturą zajmują we współczesnej informatyce miejsce szczególne, a ich rozwój w obecnym tysiącleciu jest szczególnie burzliwy. Są one często odporne na zakłócenia, tzn. dają dobre wyniki w sytuacjach, gdy mamy do czynienia z niedoskonałymi czy niekompletnymi danymi, gdy obliczeń należy dokonać przy zmiennych warunkach i pewnej niepewności odnoszącej się do samych źródeł danych czy informacji, jakie z tych danych wyciągamy.

Jednym z najczęściej używanych sposobów poszukiwania rozwiązań nieładkich problemów optymalizacyjnych i powiązanych z naturą jest **algorytm genetyczny**. Jest usytuowany w nurcie **obliczeń ewolucyjnych**

i pojawił się jako pewna implementacja procesu ewolucji (por. Hollstien [12] i Holland [11]) doboru organizmów żywych zauważona w przyrodzie, gdy osobniki danego gatunku, mając lepsze przystosowania do warunków, w których żyją, mają większe prawdopodobieństwo przeżycia, niż osobniki gorzej przystosowane. Ponadto potomkowie osobników lepiej dostosowanych otrzymują lepszy materiał genetyczny. Zjawiska znane z natury, jak kojarzenie się osobników czy występowanie mutantów, są w algorytmach genetycznych odtwarzane poprzez operatory genetyczne, działające na osobnikach, tj. parach czy pojedynczych osobnikach [9, 3, 18, 13].

Obliczenia ewolucyjne należą do rozwijającej się dziedziny wiedzy zwanej **inteligencją obliczeniową** (ang. *computational intelligence*) wyrosłej z dziedziny zwanej sztuczną inteligencją [2]. Rozwijane w tej dziedzinie metody obliczeniowe czasami noszą nazwę **metod miękkich**, jako przeniesienie z angielskiego *soft computing*, gdyż stosowane tutaj podstawy algorytmów nie są dobrze umotywowane, a inspirację dla nich zaczerpnięto z przyrody. Często brak jest jednak pełnych dowodów ich poprawności czy też rezultatów dotyczących ich zbieżności, a dalej także oczekiwanej złożoności obliczeniowej.

Algorytmy probabilistyczne, a wśród nich algorytmy genetyczne, na ogół nie przeszukują przestrzeni poszukiwań (odpowiednik przestrzeni potencjalnych rozwiązań problemu optymalizacyjnego) w sposób wyłącznie losowy. Zwykle bazują one na pewnych heurystykach, np. na założeniach dotyczących kształtu i regularności funkcji jakości zadania. Heurystyczne są tutaj nie tylko założenia o funkcji jakości (inaczej celu), ale też same metody tworzenia algorytmu obliczeniowego.

Na początku nowego tysiąclecia algorytmy genetyczne mają opinię narzędzi do rozwiązywania trudnych zadań optymalizacyjnych. Jednakże, mimo iż narasta lawinowo liczba ich zastosowań, rozwój teorii algorytmów ewolucyjnych jest powolny. Znacząco narasta potrzeba zrozumienia natury tych metod, mimo że wiedza na ich temat jest nadal niedostateczna i fragmentaryczna [9, 3, 18, 2].

W szerokim spektrum metod ewolucyjnych można wyróżnić podklasę algorytmów inspirowanych naturą, które odtwarzają znane z natury działania stadne, w szczególności wśród mrówek czy pszczół. Tutaj poważną rolę w rozwiązywaniu trudnych zadań optymalizacyjnych, zaczynając od klasycznego problemu komiwojażera, pełnią **algorytmy mrówkowe**. Ich powstanie łączy się z nazwiskiem Dorigo [6, 4, 8]. Współcześnie, te zainspirowane biologią algorytmy są kojarzone z koloniami mrówek [1] i stanowią jedną z intensywniej analizowanych technik optymalizacyjnych. Ich popularność rośnie zarówno za przyczyną ich elastyczności z jednej strony, jak i efektywności z drugiej. Podąża za tym również prostota, jaką odkrywamy dzięki analogiom pomiędzy wirtualnymi i rzeczywistymi koloniami mrówek,

na podstawie obserwacji świata owadów społecznych i mechanizmów zaobserwowanych wśród kolonii mrówek. Jego kluczowym elementem jest wydzielanie śladu feromonowego, który jest źródłem informacji dla innych mrówek, pełniąc podstawową rolę przy przemieszczaniu się mrówek i budowaniu przez nie swoich ścieżek, a dla użytkownika algorytmu na tym bazującego, znalezienia rozwiązania suboptymalnego problemu optymalizacyjnego.

Wraz z istotnym rozwojem praktycznych aspektów zastosowań algorytmów mrówkowych, podano także częściowe wyniki teoretyczne, dotyczące ich zbieżności i oczekiwanego czasu działania, por. [7, 10]. Wspomniane rezultaty dotyczą określonej podklasy algorytmów mrówkowych, nakierowanych na rozwiązanie jednostkowych problemów optymalizacyjnych, i nie noszą charakteru wyników uniwersalnych. W niniejszym artykule staramy się przedstawić rezultaty badań zachowań asymptotycznych algorytmów mrówkowych w bardziej ogólnym ujęciu. W tym celu:

- wprowadzamy niezbędne formalne ujęcie problemu optymalizacyjnego, dla którego przewidujemy szerokie spektrum aplikacji praktycznych,
- przedstawiamy własny model (podklasę) algorytmu mrówkowego zwany **dyskretnym algorytmem mrówkowym** (ang. *Discrete Ant System*, w skrócie DAS),
- budujemy markowski model teoretyczny dla wskazanego narzędzia optymalizacyjnego,
- wykazujemy w zbudowanym modelu probabilistycznym własność punktowej asymptotycznej zbieżności dla algorytmu DAS.

W tym miejscu należy wspomnieć, że prezentowany sposób analizy zachowań ewolucyjnych algorytmów probabilistycznych ma swoją inspirację w wynikach moich starszych kolegów (por. [22, 23, 15, 14, 16]), którzy korzystając z podejścia znanego ze stochastycznych układów dynamicznych [20, 17], [25, 19, 21], sformułowali i udowodnili pewne własności prostego algorytmu genetycznego. W szczególności jego własności asymptotyczne. Jeden z ich wyników, dotyczący punktowej zbieżności asymptotycznej, której klasyczne algorytmy genetyczne nie posiadają, był dla mnie inspiracją do opracowania takiego modelu algorytmu mrówkowego (wspomniana heurystyka DAS), aby ta własność była przez ten algorytm osiągnięta.

1. Problem optymalizacyjny. Prezentację wyników naszej pracy badawczej rozpoczniemy od ustalenia formalnych kryteriów, jakie będą definiowały wspomniane we wstępie zadanie optymalizacyjne. Wpierw założymy, że skończenie wymiarowa przestrzeń \mathcal{Y} , w której rozważamy zadania optymalizacyjne, będzie nieskończonym zbiorem słów $\mathcal{Y} = \Sigma^*$ nad pewnym skończonym alfabetem Σ . Dalej przez **zadanie (problem) optymalizacyjne**

w przestrzeni Σ^* będziemy rozumieli piątkę:

$$(1) \quad (\Sigma, \mathcal{R}, \Delta, \|\cdot\|, [\searrow \mid \nearrow]),$$

gdzie kolejno:

- $\Sigma = \{x_1, x_2, \dots, x_n\}$ – zadany skończony zbiór n indeksowanych **symboli**,
- $\mathcal{R} \subset \Sigma^*$ – zadany skończony zbiór τ indeksowanych **słów-rozwiązań** $\{\omega_1, \omega_2, \dots, \omega_\tau\}$,
- $\Delta : \Sigma^* \rightarrow \{0, 1\}$ – **funkcja akceptacji** słowa $\omega \in \Sigma^*$ taka, że

$$(2) \quad \Delta(\omega) = \begin{cases} 1, & \text{gd}y \exists (\omega' \in \Sigma^*, \omega'' \in \mathcal{R}) (\omega \circ \omega' = \omega'') \\ 0, & \text{wp.p.} \end{cases}$$

- $\|\cdot\| : \mathcal{R} \rightarrow \mathbb{R}_+$ – **funkcja oceny** słowa-rozwiązania $\omega \in \mathcal{R}$,
- $[\searrow \mid \nearrow]$ – kierunek optymalizacji, \searrow dla minimalizacji oraz \nearrow dla maksymalizacji funkcji oceny.

Warto w tym miejscu zauważyć, że postać zbioru indeksowanych rozwiązań \mathcal{R} i sposób interpretacji jego elementów jest zdeterminowany przez charakterystykę zadanego problemu optymalizacyjnego. Na przykład dla problemu komiwojażera \mathcal{R} będzie zbiorem wszystkich n -elementowych permutacji elementów zbioru symboli (zakładamy, że symbole reprezentują etykiety wierzchołków grafu wejściowego). Natomiast dla problemu plecakowego zbiór ten będzie zawierał słowa-rozwiązania dowolnej długości, z dokładnością do kombinacji z powtórzeniami i ograniczeń na licznosc obiektów danego rodzaju. Funkcja akceptacji słowa Δ ma charakter techniczny i określa, czy kierunek konstrukcji rozwiązania wewnątrz algorytmu przebiega poprawnie, tj. czy aktualnie zbudowany fragment słowa jest prefiksem dowolnego elementu zbioru \mathcal{R} , wtedy $\Delta(\omega) = 1$, czy też nie, wtedy $\Delta(\omega) = 0$. Podobnie jak w przypadku interpretacji elementów zbioru słów-rozwiązań, tak i dla funkcji Δ jej postać jest uwarunkowana bezpośrednio charakterystyką rozwiązywanego problemu optymalizacyjnego. Przedostatni element rozważanej piątki to wspomniana we wstępie funkcja jakości rozwiązania $\|\cdot\|$. W dalszej części pracy będziemy zakładali, bez straty ogólnosci wyników, **minimalizację funkcji oceny** w przeciwdziedzinie liczb rzeczywistych dodatnich, tym samym wszystkie rozważane przez nas piątki definiujące instancję problemu optymalizacyjnego będą miały postać

$$(3) \quad (\Sigma, \mathcal{R}, \Delta, \|\cdot\|, \searrow).$$

Teraz niech $\omega^* \in \mathcal{R}$ oznacza słowo-rozwiązanie **optymalne**, tj. takie rozwiązanie, które spełnia poniższy warunek

$$(4) \quad \forall (1 \leq i \leq \tau) (\|\omega_i\| \geq \|\omega^*\|).$$

W zbiorze \mathcal{R} wyznaczamy podzbiór słów-rozwiązań optymalnych $\mathcal{R}^* \subseteq \mathcal{R}$

zawierający wszystkie rozwiązania spełniające ograniczenie zapisane w (4). Algorytmicznym celem naszego zadania optymalizacyjnego, czyli celem działania algorytmu probabilistycznego, w skończonej wymiarowej przestrzeni $\mathcal{Y} = \Sigma^*$ jest znalezienie **dowolnego** słowa-rozwiązania $\omega^* \in \mathcal{R}^*$.

Przedstawione powyżej sformułowanie problemu optymalizacyjnego jest dostatecznie uniwersalne. Pozwala na wyrażenie wielu praktycznych zagadnień kombinatorycznych z dziedziny teorii algorytmów, szczególnie tych z klasy problemów NP-zupełnych, jak np. problem komiwojażera, problem marszrutyzacji, problem plecakowy, problem kolorowania wierzchołkowego grafów. Z punktu widzenia tematyki prezentowanej pracy ustalenie konkretnego problemu optymalizacyjnego nie jest istotne. Stąd, w dalszej części artykułu, przedstawimy wyniki teoretyczne bez bezpośredniego odniesienia do instancji problemu optymalizacyjnego.

2. Dyskretny algorytm mrówkowy. Dyskretny algorytm mrówkowy jest rozwinięciem systemu mrówkowego (ang. *Ant System*, w skrócie AS) wprowadzonego przez Doriego w [5]. Heurystyka zakłada kolektywną pracę i gromadzenie informacji przez jednostki (osobniki) nazywane **mrówkami** oraz konstruowanie indywidualnych rozwiązań w oparciu o globalną wiedzę zbioru m indeksowanych mrówek $A = \{a_1, a_2, \dots, a_m\}$, zwanego dalej **mrowiskiem**. Ogólny schemat algorytmu DAS ma charakter iteracyjny:

ω^{-1} – słowo-rozwiązanie skonstruowane w poprzedniej iteracji
 ω – słowo-rozwiązanie konstruowane w aktualnej iteracji
 ω^* – najlepsze słowo skonstruowane do aktualnej iteracji włącznie

- (1) dopóki nie jest prawdziwy zewnętrzny warunek stopu
 - (2) kolejno dla każdej mrówki a_j , gdzie $j = 1, 2, \dots, m$
 - (3) $\omega \leftarrow x_{l_1}$ // wyznacz symbol początkowy x_{l_1} słowa ω
 - (4) dopóki nie jest prawdziwy wewnętrzny warunek stopu
 - (5) $\omega \leftarrow \omega x_{l_{i+1}}$ // wyznacz i dodaj kolejny symbol $x_{l_{i+1}}$ do słowa ω
 - (6) w miejsce ω^* wybierz rozwiązanie lepsze z ω^* oraz ω
 - (7) wykonaj aktualizację globalnej informacji feromonowej względem ω^{-1} oraz ω
 - (8) $\omega^{-1} \leftarrow \omega$
- (9) poszukiwane słowo-rozwiązanie to ω^*
-

Rozważany algorytm bazuje na pojęciu **współczynnika nasycenia śladu feromonowego**

$$(5) \quad \tau \in \mathbb{N}_+,$$

który jest nośnikiem wspomnianej globalnej wiedzy zbioru mrówek o specyficie przeszukiwanej przestrzeni rozwiązań. Wartość współczynnika τ cha-

rakteryzuje jakość rozwiązań, jakie budują mrówki korzystając z użycia w danym rozwiązaniu symboli w określonej kolejności. Niech dalej τ_{max} będzie zadaną, **maksymalną** wartością współczynnika nasycenia śladu feromonowego oraz $\mathbb{H} = \{1, 2, \dots, \tau_{max}\}$ będzie zbiorem potencjalnych wartości współczynnika nasycenia śladu feromonowego. Konstruujemy **wektor stopni nasycenia** śladu feromonowego

$$(6) \quad F \in \mathbb{H}^n \quad \text{i} \quad F = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{bmatrix}$$

wymiaru n , gdzie $F[i]$ określa ilość śladu feromonowego związanego z dopisaniem przez mrówkę do budowanego słowa-rozwiązania symbolu x_i . Przez $\mathcal{F} = \{F_1, F_2, \dots, F_f\}$ oznaczymy zbiór wszystkich możliwych indeksowanych wektorów stopni nasycenia śladu feromonowego.

WNIOSEK I. Zbiór $\mathcal{F} = \{F_1, F_2, \dots, F_f\}$ jest zbiorem skończonym takim, że $f = (\tau_{max})^n$.

Analogicznie wprowadzamy **macierz stopni nasycenia** śladu feromonowego

$$(7) \quad H \in \mathbb{H}^{n \times n} \quad \text{i} \quad H = \begin{bmatrix} h_{1,1} & h_{2,1} & \cdots & h_{n,1} \\ h_{1,2} & h_{2,2} & \cdots & h_{n,2} \\ \vdots & \vdots & \ddots & \vdots \\ h_{1,n} & h_{2,n} & \cdots & h_{n,n} \end{bmatrix}$$

wymiaru $n \times n$, gdzie $H[i, j]$ determinuje ilość feromonu powiązanego z dołączeniem do słowa symbolu x_j tuż po symbolu x_i . Przez $\mathcal{H} = \{H_1, H_2, \dots, H_h\}$ oznaczymy zbiór wszystkich możliwych indeksowanych macierzy stopni nasycenia śladu feromonowego.

WNIOSEK II. Zbiór $\mathcal{H} = \{H_1, H_2, \dots, H_h\}$ jest zbiorem skończonym takim, że $h = (\tau_{max})^{n^2}$.

Zgodnie z przedstawionym na początku rozdziału szkicem dyskretnego algorytmu mrówkowego, metoda ta jest procesem iteracyjnym. W celu realizacji jego poszczególnych elementów wyróżniamy następujące reguły ewolucji:

- reguła wyboru sąsiedztwa (ang. *Neighbor Choosing Rule*, w skrócie NCR),
- reguła konstrukcji rozwiązania (ang. *Solution Construction Rule*, w skrócie SCR),
- reguła akceptacji rozwiązania (ang. *Solution Acceptance Rule*, w skrócie SAR),

- reguła aktualizacji feromonu (ang. *Pheromone Update Rule*, w skrócie PUR).

Każdej z tych reguł poświęcimy chwilę uwagi, gdyż są one kluczowe z punktu widzenia konstrukcji modelu probabilistycznego algorytmu DAS, który to model zaprezentujemy w następnym rozdziale. Pierwsza z reguł, tj. **reguła wyboru sąsiedztwa** NCR, realizuje probabilistykę wspomnianej wcześniej kolejności wyboru symboli w trakcie konstrukcji słowa-rozwiązania ω przez mrówkę. Przyjmujemy, że

$$(8) \quad \text{NCR} : \mathcal{F} \times \Sigma^* \rightarrow \Sigma$$

jest **funkcją** niedeterministyczną dla argumentów kolejno wektora kolumnowego stopni nasycenia $F \in \mathcal{F}$ i aktualnie skonstruowanego słowa $\omega \in \Sigma^*$. Dalej $\text{NCR}(F, \omega) = x_i$ z prawdopodobieństwem zadanym poniższym wzorem

$$(9) \quad \text{Pr}(\text{NCR}(F, \omega) = x_i) = \begin{cases} \frac{F[i]}{\sum_{\{j: \Delta(\omega x_j)=1\}} F[j]} & \text{gdy } \Delta(\omega x_i) = 1 \\ 0 & \text{w p.p.} \end{cases}$$

Korzystając z tego, że $\mathbb{H} = \{1, 2, \dots, \tau_{max}\}$ jest zbiorem liczb naturalnych dodatnich, wyprowadzamy wniosek:

WNIOSEK III. *Dla dowolnego wektora kolumnowego $F \in \mathcal{F}$, dowolnego słowa $\omega \in \Sigma^*$ oraz dowolnego symbolu $x \in \Sigma$ zachodzi*

$$\text{Pr}(\text{NCR}(F, \omega) = x) \in (0, 1]$$

jeżeli $\Delta(\omega x) = 1$ oraz

$$\text{Pr}(\text{NCR}(F, \omega) = x) = 0$$

w przeciwnym przypadku.

Omówiona reguła wyboru sąsiedztwa NCR stanowi podstawę kolejnego, w konsekwencji także niedeterministycznego mechanizmu ewolucji, czyli **reguły konstrukcji rozwiązania** SCR, gdzie

$$(10) \quad \text{SCR} : \mathcal{F} \times \mathcal{H} \rightarrow \mathcal{R}.$$

Proces konstrukcji słowa-rozwiązania przez pojedynczą mrówkę można wyróżnić w przedstawionym powyżej schemacie ogólnym algorytmu DAS w wierszach:

-
- ...
 - (3) $\omega \leftarrow x_{l_1}$ // wyznacz symbol początkowy x_{l_1} słowa ω
 - (4) dopóki nie jest prawdziwy wewnętrzny warunek stopu
 - (5) $\omega \leftarrow \omega x_{l_{i+1}}$ // wyznacz i dodaj kolejny symbol $x_{l_{i+1}}$ do słowa ω
 - ...
-

Niech dalej $F \in \mathcal{F}$ oraz $H \in \mathcal{H}$ oznaczają odpowiednio wektor globalnego stopnia nasycenia śladu feromonowego odpowiadający za **wybór symbolu początkowego** oraz macierz globalnego stopnia nasycenia śladu feromonowego determinująca **wybór kolejnych symboli** w konstruowanym słowie-rozwiązaniu $\omega \in \mathcal{R}$. Wektor F oraz macierz H są wspólne dla całej populacji mrówek i stanowią nośnik globalnej informacji heurystycznej, dotyczącej eksploracji przestrzeni rozwiązań. Reguła SCR jest złożeniem ciągu zdarzeń niezależnych, generowanych przez wielokrotne zastosowanie wprowadzonego powyżej mechanizmu NCR. Kolejno na podstawie wektora F mrówka wyznacza pierwszy element słowa-rozwiązania ω

$$(11) \quad \omega \leftarrow \text{NCR}(F, \epsilon).$$

Dopóki nie jest prawdziwy **wewnętrzny warunek stopu** (warunek ten jest zadany i ma charakter indywidualny dla każdego zadania optymalizacyjnego, np. dla problemu komiwojażera jest to weryfikacja, czy słowo ω jest permutacją zbioru elementów Σ) mrówka powtarza następującą czynność: niech $\omega = x_{l_1}x_{l_2}\dots x_{l_i}$, na podstawie l_i -tego wektora kolumnowego macierzy H oznaczonego przez $H[l_i, \cdot]$ wyznacz kolejny element rozwiązania

$$(12) \quad x_{l_{i+1}} \leftarrow \text{NCR}(H[l_i, \cdot], \omega)$$

z prawdopodobieństwem zadanym regułą NCR, a następnie dodaj wyznaczony symbol na koniec słowa ω

$$(13) \quad \omega \leftarrow \omega x_{l_{i+1}}.$$

Bezpośrednio z konstrukcji reguł NCR i SCR wynika poniższy wniosek:

WNIOSEK IV. Dla dowolnego wektora kolumnowego $F \in \mathcal{F}$, dowolnej macierzy $H \in \mathcal{H}$ oraz dowolnego słowa-rozwiązania $\omega \in \mathcal{R}$ postaci $\omega = x_{l_1}x_{l_2}\dots x_{l_r}$ zachodzi

$$(14) \quad Pr(\text{SCR}(F, H) = \omega) = Pr(\text{NCR}(F, \epsilon) = x_{l_1}) \cdot$$

$$(15) \quad \prod_{i=1}^{r-1} Pr(\text{NCR}(H[l_i, \cdot], x_{l_1}x_{l_2}\dots x_{l_i}) = x_{l_{i+1}})$$

stąd na podstawie wniosku III (strona 88)

$$(16) \quad Pr(\text{SCR}(F, H) = \omega) \in (0, 1].$$

Wniosek IV ma istotne implikacje probabilistyczne. Zgodnie z jego treścią dowolne słowo-rozwiązanie $\omega \in \mathcal{R}$, a zatem także słowo-rozwiązanie optymalne należące do zbioru słów-rozwiązań optymalnych $\mathcal{R}^* \subseteq \mathcal{R}$ może, z dodatnim prawdopodobieństwem, być rezultatem zastosowania reguły SCR w dowolnym stanie struktur feromonowych F oraz H . Do własności tej powrócimy przy okazji omawiania punktowej zbieżności algorytmu DAS w ostatnim rozdziale artykułu.

Z regułą konstrukcji rozwiązania SCR związany jest **nowy mechanizm**, który nie znajduje swojej analogii w klasycznym systemie mrówkowym, podanym przez Dorigo (por. [5]) oraz jego kontynuacjach (por. [8]). Genezę wprowadzenia dodatkowego operatora stanowi problem wyznaczenia **stopnia różnorodności** rozwiązań o bliskiej sobie wartości funkcji oceny. W szczególności może się zdarzyć w trakcie ewolucji pojedynczej mrówki, że aktualnie wyznaczone rozwiązanie ω jest tak samo dobre ze względu na wartość funkcji kosztu, jak większość rozwiązań aktualnie konstruowanych przez inne mrówki w obrębie pojedynczego mrowiska. Powstaje zatem pytanie, czy mimo równej wartości funkcji $\|\cdot\|$ rozwiązanie ω niesie ze sobą istotnie nową wartość eksploracyjną. Brak umiejętności odpowiedzi na tak postawione pytanie, szczególnie w końcowej fazie wykonania algorytmu mrówkowego, prowadzi zazwyczaj do zbyt szybkiej zbieżności metody względem jednego z rozwiązań suboptymalnych. Warto wspomnieć, że problem ten znajduje swoje odzwierciedlenie w całej klasie algorytmów heurystycznych i nie jest jedynie domeną metod mrówkowych, np. w przypadku algorytmów genetycznych stosuje się operator mutacji w celu jego redukcji. Określenie stopnia różnorodności rozwiązania, w odniesieniu do aktualnej tendencji eksploracyjnej mrowiska, zrealizowane jest za pomocą dodatkowej niedeterministycznej **funkcji innowacyjności** rozwiązania

$$(17) \quad \text{INV} : \mathcal{F} \times \mathcal{H} \times \mathcal{R} \rightarrow \{0, 1\}$$

takiej, że dla rozwiązania $\omega \in \mathcal{R}$ postaci $\omega = x_{l_1} x_{l_2} \dots x_{l_r}$, zachodzi

$$(18) \quad Pr(\text{INV}(F, H, \omega) = 1) = \frac{1}{r} \cdot \left(\frac{1 + \max(F) - F[l_1]}{1 + \max(F)} + \right.$$

$$(19) \quad \left. \sum_{i=1}^{r-1} \frac{1 + \max(H[l_i, \cdot]) - H[l_i, l_{i+1}]}{1 + \max(H[l_i, \cdot])} \right),$$

gdzie $\max(F)$ oraz $\max(H[l_i, \cdot])$ oznaczają odpowiednio wartość elementu maksymalnego wektora F oraz wartość elementu maksymalnego l_j -tego wektora kolumnowego macierzy H . Funkcja INV wyznacza uśrednioną, względem liczby elementów rozwiązania, wartość różnicy śladu feromonowego aktualnie najbardziej popularnych rozwiązań (elementy maksymalne $\max(F)$ oraz $\max(H[l_i, \cdot])$) i śladu feromonowego związanego z rozważanym rozwiązaniem ω (tj. elementy $F[l_1]$ oraz $H[l_i, l_{i+1}]$). Jeżeli ślad feromonowy danego rozwiązania ω jest bliski śladowi feromonowemu aktualnie najbardziej popularnych rozwiązań, to $\text{INV}(F, H, \omega) = 0$ z prawdopodobieństwem bliskim jedności. W przeciwnym przypadku, tj. jeżeli skonstruowane rozwiązanie niesie ze sobą istotną dawkę nowych wartości eksploracyjnych, w odniesieniu do globalnego trendu mrowiska, wartość funkcji innowacyjności rozwiązania jest równa 1 z prawdopodobieństwem bliskim jedności.

WNIOSEK V. Dla dowolnego wektora kolumnowego $F \in \mathcal{F}$, dowolnej macierzy $H \in \mathcal{H}$ oraz dowolnego słowa-rozwiązania $\omega \in \mathcal{R}$ zachodzi

$$(20) \quad Pr(INV(F, H, \omega) = 1) \in (0, 1),$$

a tym samym

$$(21) \quad Pr(INV(F, H, \omega) = 0) \in (0, 1).$$

Zgodnie z treścią powyższego wniosku dowolne słowo-rozwiązanie może być uznane za innowacyjne bądź nie z prawdopodobieństwem dodatnim, ale nie równym jedności. Tym samym charakterystyka operatora innowacyjności rozwiązania nie ogranicza w sensie probabilistycznym własności eksploracyjnych algorytmu mrówkowego, a jedynie wzmacnia jego działanie różnicujące w sytuacji nadmiernego skupienia rozwiązań w pojedynczym punkcie przeszukiwanej przestrzeni rozwiązań $\mathcal{R} \subset \Sigma^*$.

Trzecia z reguł ewolucji dyskretnego algorytmu mrówkowego to **reguła akceptacji rozwiązania SAR**

$$(22) \quad SAR : \mathcal{R} \times \mathcal{R} \rightarrow \mathcal{R}.$$

Rolą tego mechanizmu jest zagwarantowanie monotoniczności procesu optymalizacji, przez zapamiętywanie przez każdą z mrówek rozwiązania do tej pory indywidualnie najlepszego ω^* względem wartości funkcji oceny $\|\cdot\|$. Za realizację tego zadania w rozważanym wcześniej schemacie dyskretnego algorytmu mrówkowego odpowiada poniższa instrukcja:

$$(6) \quad \begin{array}{l} \dots \\ \text{w miejsce } \omega^* \text{ wybierz rozwiązanie lepsze z } \omega^* \text{ oraz } \omega \\ \dots \end{array}$$

Reguła akceptacji rozwiązania jest mechanizmem deterministycznym i polega na wykonaniu pojedynczego warunku

$$\begin{array}{l} SAR(\omega, \omega^*) \\ \text{jeżeli } \|\omega\| \leq \|\omega^*\| \\ \omega^* \leftarrow \omega // \text{ rozwiązanie } \omega \text{ jest nie gorsze niż dotychczas najlepsze } \omega^* \\ \text{w p.p.} \\ \omega^* \leftarrow \omega^* // \text{ rozwiązanie } \omega \text{ jest gorsze niż dotychczas najlepsze } \omega^* \end{array}$$

Ostatnia **reguła aktualizacji feromonu PUR** postaci

$$(23) \quad PUR : \mathcal{F} \times \mathcal{H} \times \mathcal{R}^2 \rightarrow \mathcal{F} \times \mathcal{H},$$

jest kluczowym elementem rozważanego algorytmu heurystycznego. Przedstawione do tej pory mechanizmy NCR, SCR oraz SAR mają charakter **stacyczny** ze względu na wymianę informacji wewnątrz mrowiska, a ich działanie zależy jedynie od aktualnego stanu informacji heurystycznej, zapisanej w wektorze F , macierzy H oraz postaci słowa-rozwiązania ω^* . Wymienione reguły, odpowiedzialne za konstrukcję i akceptację pojedynczego rozwiązania przez mrówkę, nie realizują w żadnym stopniu procesu komunikacji między mrówkami. Natomiast reguła aktualizacji feromonu PUR ma charakter **dynamiczny**, odnosi się do wcześniejszego kierunku ewolucji mrówki i dodatkowo modyfikuje zawartość wspomnianego wektora F i macierzy H . Tym samym stanowi mechanizm wymiany informacji wewnątrz mrowiska. Od sposobu konstrukcji tej reguły zależy w istotnej mierze efektywność procesu kooperacji mrówek w celu optymalizacji rozwiązań, a w rezultacie jakość zbieżności całego algorytmu mrówkowego. W rozważanym wcześniej schemacie heurystyki DAS, reguła aktualizacji feromonu ma swoje miejsce na końcu operacji wykonywanych w danej iteracji przez każdą z mrówek:

...
 (7) wykonaj aktualizację globalnej informacji feromonowej względem ω^{-1} oraz ω
 ...

Mechanizm PUR można podzielić na dwa kluczowe etapy: kolejno aktualizację elementów wektora F oraz aktualizację elementów macierzy H . W obu przypadkach zastosowanie mechanizmu wymaga znajomości dwóch wcześniej wprowadzonych słów-rozwiązań odpowiednio ω^{-1} – słowo zbudowane przez mrówkę w iteracji bezpośrednio poprzedzającej aktualnie rozważaną, oraz ω – słowo zbudowane przez mrówkę w aktualnej iteracji. Zestawienie wartości funkcji oceny dla tych elementów przeszukiwanego zbioru \mathcal{R} ma charakter lokalny (krótkoterminowy). Na potrzeby artykułu przyjmujemy, że badanie zależności tych dwóch wartości jest w pełni wystarczające. Należy jednak podkreślić, że w praktyce rozważamy zwykle dodatkowo relację wartości funkcji jakości dla słowa-rozwiązania aktualnie skonstruowanego ω , jak i do tej pory najlepszego ω^* . Tym samym analizujemy także globalną (długoterminową) własność ewolucji wartości funkcji oceny. Takie zestawienie krótko i długoterminowe daje pełniejszy obraz zachowania mrówki, a zatem pozwala na dokładniejszą i bardziej przemyślaną modyfikację globalnego śladu feromonowego wewnątrz reguły PUR.

Wprowadzimy teraz dwa pomocnicze operatory inkrementacji i dekrementacji elementów zbioru \mathcal{F} wszystkich możliwych wektorów stopni nasycenia śladu feromonowego, postaci stosownie $\text{inc}_{\mathcal{F}} : \mathcal{F} \times \mathcal{R} \rightarrow \mathcal{F}$ oraz $\text{dec}_{\mathcal{F}} : \mathcal{F} \times \mathcal{R} \rightarrow \mathcal{F}$. Rezultatem zastosowania operatora $\text{inc}_{\mathcal{F}}(F, \omega)$ bądź

$\text{dec}_{\mathcal{F}}(F, \omega)$ jest wektor powstały z wektora F przez wykonanie odpowiednio operacji inkrementacji

$$(24) \quad F[l_1] \leftarrow \min(\tau_{max}, F[l_1] + 1)$$

albo dekrementacji

$$(25) \quad F[l_1] \leftarrow \max(1, F[l_1] - 1)$$

na elemencie wektora F o indeksie l_1 , gdzie l_1 jest indeksem pierwszego symbolu rozwiązania $\omega = x_{l_1}x_{l_2} \dots x_{l_r}$. Podobnie przez $\text{inc}_{\mathcal{H}} : \mathcal{H} \times \mathcal{R} \rightarrow \mathcal{H}$ oraz $\text{dec}_{\mathcal{H}} : \mathcal{H} \times \mathcal{R} \rightarrow \mathcal{H}$ oznaczymy funkcje, które zadaną macierz stopni nasycenia śladu feromonowego $H \in \mathcal{H}$ przeprowadzają w macierz $\text{inc}_{\mathcal{H}}(H, \omega)$ albo $\text{dec}_{\mathcal{H}}(H, \omega)$ taką, że dla operatora inkrementacji zachodzi

$$(26) \quad H[l_i, l_{i+1}] \leftarrow \min(\tau_{max}, H[l_i, l_{i+1}] + 1),$$

a w przypadku dekrementacji

$$(27) \quad H[l_i, l_{i+1}] \leftarrow \max(1, H[l_i, l_{i+1}] - 1),$$

dla wszystkich elementów macierzy H o indeksach $[l_i, l_{i+1}]$, gdzie l_i dla $i = 1, 2, \dots, r$, jest indeksem kolejnych symboli rozwiązania $\omega = x_{l_1}x_{l_2} \dots x_{l_r}$.

Bazując na przedstawionych operatorach inkrementacji i dekrementacji możemy scharakteryzować, w kategoriach procedury algorytmicznej, zasadę działania mechanizmu aktualizacji feromonu PUR dla przyjętych struktur feromonowych F oraz H :

$$\begin{aligned} & \text{PUR}(F, H, \omega^{-1}, \omega) \\ & \underline{\text{jeżeli}} \|\omega\| < \|\omega^{-1}\| \\ & \quad F \leftarrow \text{inc}_{\mathcal{F}}(F, \omega), F \leftarrow \text{dec}_{\mathcal{F}}(F, \omega^{-1}) \\ & \quad H \leftarrow \text{inc}_{\mathcal{H}}(H, \omega), H \leftarrow \text{dec}_{\mathcal{H}}(H, \omega^{-1}) \\ & \underline{\text{w p.p.}} \\ & \underline{\text{jeżeli}} \|\omega\| = \|\omega^{-1}\| \\ & \quad \text{z prawdopodobieństwem } Pr(\text{INV}(F, H, \omega) = 1) \text{ wykonaj} \\ & \quad \quad F \leftarrow \text{inc}_{\mathcal{F}}(F, \omega) \\ & \quad \quad H \leftarrow \text{inc}_{\mathcal{H}}(H, \omega) \\ & \quad \text{albo} \\ & \quad \quad F \leftarrow \text{dec}_{\mathcal{F}}(F, \omega) \\ & \quad \quad H \leftarrow \text{dec}_{\mathcal{H}}(H, \omega) \\ & \underline{\text{w p.p.}} \\ & \quad F \leftarrow \text{inc}_{\mathcal{F}}(F, \omega^{-1}), F \leftarrow \text{dec}_{\mathcal{F}}(F, \omega) \\ & \quad H \leftarrow \text{inc}_{\mathcal{H}}(H, \omega^{-1}), H \leftarrow \text{dec}_{\mathcal{H}}(H, \omega) \end{aligned}$$

Podsumowując rozdział poświęcony heurystyce DAS podamy uproszczony schemat działania rozważanego algorytmu. Korzystając z wprowadzonych czterech reguł sterujących ewolucją pojedynczej mrówki, a tym samym

całego kolektywu mrówek, czyli mrowiska, dyskretny algorytm mrówkowy zapiszemy teraz jako iteracyjne złożenie poniższych kroków obliczeniowych:

-
- (1) dopóki nie jest prawdziwy zewnętrzny warunek stopu
 - (2) kolejno dla każdej mrówki a_j , gdzie $j = 1, 2, \dots, m$
 - (3) zastosuj regułę SCR z uwzględnieniem reguły NCR
 - (4) zastosuj regułę SAR
 - (5) zastosuj regułę PUR
-

3. Model teoretyczny algorytmu DAS. Dla uproszczenia rozważań przedstawionych w dalszej części artykułu omówimy teraz **model teoretyczny** algorytmu DAS dla **pojedynczej mrówki**. Rozszerzenie modelu algorytmu dla przypadku mrowiska składającego się z $m > 1$ mrówek, ze względu na ich sekwencyjną pracę w wewnętrznej pętli iteracyjnej

-
- ...
 - (2) kolejno dla każdej mrówki a_i , gdzie $i = 1, 2, \dots, m$
 - ...
-

jest zadaniem czysto technicznym. Proponowana redukcja sprowadza się do rezygnacji z wymienionej instrukcji składniowej. W rezultacie otrzymujemy następujący szkic działania rozważanego narzędzia optymalizacyjnego:

-
- (1) dopóki nie jest prawdziwy zewnętrzny warunek stopu
 - (2) zastosuj regułę SCR z uwzględnieniem reguły NCR
 - (3) zastosuj regułę SAR
 - (4) zastosuj regułę PUR
-

Dalej przez **stan mrówki**

$$(28) \quad s_{(t)} \in \mathcal{F} \times \mathcal{H} \times \mathcal{R}^2$$

w chwili t będziemy rozumieli czwórkę postaci:

$$(29) \quad s_{(t)} = \left(F_{(t)}, H_{(t)}, \omega_{(t)}, \omega_{(t)}^* \right),$$

gdzie:

- $F_{(t)}$ – wektor globalnego stopnia nasycenia śladu feromonowego, odpowiadający za wybór symbolu początkowego dla rozwiązania, które będzie konstruowane w trakcie przejścia ze stanu $s_{(t)}$ do stanu $s_{(t+1)}$,

- $H_{(t)}$ – macierz globalnego stopnia nasycenia śladu feromonowego, odpowiadająca za wybór kolejnych symboli w rozwiązaniu, które będzie konstruowane w trakcie przejścia ze stanu $s_{(t)}$ do stanu $s_{(t+1)}$,
- $\omega_{(t)}$ – słowo-rozwiązanie skonstruowane w chwili t ,
- $\omega_{(t)}^*$ – najlepsze słowo-rozwiązanie skonstruowane do chwili t włącznie.

Bezpośrednio ze sposobu działania dyskretnego algorytmu mrówkowego (sekwencyjne złożenie reguł DAS-SCR, DAS-SAR, DAS-PUR w pojedynczej iteracji względem stanów struktur feromonowych będących rezultatem wykonania poprzedniej iteracji), jak i wprowadzonej definicji stanu mrówki, możemy podać następujący wniosek, kluczowy z perspektywy dalszych wyników:

WNIOSEK VI. *Stan mrówki w chwili $t+1$ zależy jedynie od stanu mrówki w chwili t .*

Dalej przez $\mathcal{S} = \{s_1, s_2, \dots, s_s\}$ oznaczymy zbiór wszystkich indeksowanych stanów mrówki. Korzystając z wprowadzonych w poprzednim rozdziale wniosków I (strona 87) i wniosku II (strona 87) możemy wyznaczyć licznosc zbioru \mathcal{S} względem liczby symboli n zbioru Σ generującego przeszukiwaną przestrzeń rozwiązań \mathcal{R} oraz zadanej maksymalnej wartości współczynnika nasycenia śladu feromonowego τ_{max} .

WNIOSEK VII. *Zbiór $\mathcal{S} = \mathcal{F} \times \mathcal{H} \times \mathcal{R}^2$ jest zbiorem skończonym takim, że $|\mathcal{S}| = \mathfrak{s}$ oraz*

$$(30) \quad \mathfrak{s} = |\mathcal{F}| \cdot |\mathcal{H}| \cdot |\mathcal{R}|^2$$

$$(31) \quad = (\tau_{max})^n \cdot (\tau_{max})^{n^2} \cdot \mathfrak{r}^2 = (\tau_{max})^{n^2+n} \cdot \mathfrak{r}^2,$$

gdzie \mathfrak{r} jest liczbą elementów skończonego zbioru słów-rozwiązań $\mathcal{R} \subset \Sigma^*$.

Odnosząc się do pojęcia stanu pojedynczej mrówki, przejdziemy teraz do omówienia sposobu teoretycznego modelowania procesu ewolucji algorytmu DAS. Jak wspomnieliśmy we wstępie, prezentowane w artykule wyniki korzystają z podejścia probabilistycznego, zastosowanego przez moich starszych kolegów (por. [22, 23, 15, 14, 16]) do analizy zbieżności algorytmów genetycznych w oparciu rezultaty z dziedziny stochastycznych układów dynamicznych. Wprowadzimy zatem **algebraiczny opis** stanu dyskretnego algorytmu mrówkowego, tożsamy ze stanem pojedynczej mrówki.

Niech dalej

$$(32) \quad \widehat{U}_{(t)} \in [0, 1]^{\mathfrak{s}} \quad \text{i} \quad \widehat{U}_{(t)} = \begin{bmatrix} \widehat{u}_1 \\ \widehat{u}_2 \\ \vdots \\ \widehat{u}_{\mathfrak{s}} \end{bmatrix}$$

będzie stochastycznym wektorem kolumnowym wymiaru \mathfrak{s} reprezentującym **rozkład prawdopodobieństwa** stanu mrówki w chwili t , gdzie $\widehat{U}_{(t)}[i]$ określa prawdopodobieństwo tego, że mrówka w chwili t znajduje się w stanie s_i . Następnie definiujemy kolumnowo stochastyczną **macierz przejścia** pojedynczej mrówki między stanami wymiaru $\mathfrak{s} \times \mathfrak{s}$ postaci

$$(33) \quad \widehat{T} \in [0, 1]^{\mathfrak{s} \times \mathfrak{s}} \quad \text{i} \quad \widehat{T} = \begin{bmatrix} \widehat{t}_{1,1} & \widehat{t}_{2,1} & \cdots & \widehat{t}_{\mathfrak{s},1} \\ \widehat{t}_{1,2} & \widehat{t}_{2,2} & \cdots & \widehat{t}_{\mathfrak{s},2} \\ \vdots & \vdots & \ddots & \vdots \\ \widehat{t}_{1,\mathfrak{s}} & \widehat{t}_{2,\mathfrak{s}} & \cdots & \widehat{t}_{\mathfrak{s},\mathfrak{s}} \end{bmatrix}$$

tak, że $\widehat{T}[i, j]$ określa prawdopodobieństwo przejścia mrówki ze stanu s_i do stanu s_j . Na tej podstawie pojedyncza iteracja (przejście) algorytmu DAS w chwili t do chwili $t + 1$ sprowadza się do wykonania przekształcenia algebraicznego postaci

$$(34) \quad \widehat{U}_{(t)}\widehat{T} = \widehat{U}_{(t+1)}.$$

Uwzględnienie w tym miejscu wniosków VI (strona 95) i VII (strona 95) pozwala nam na wyrażenie procesu ewolucji dyskretnego algorytmu mrówkowego w języku teorii **łańcuchów Markowa** nad skończoną i dyskretną przestrzenią stanów mrówki \mathcal{S} . Prowadzi to do kluczowego z punktu widzenia wyników zawartych w artykule stwierdzenia:

WNIOSEK VIII. *Proces ewolucji stanów pojedynczej mrówki w przyjętym modelu teoretycznym algorytmu DAS jest procesem Markowa.*

Kontynuując, jeżeli $\widehat{U}_{(0)}$ jest wektorem rozkładu prawdopodobieństwa dla stanu początkowego pojedynczej mrówki (tj. tuż przed pierwszą iteracją heurystyki), to

$$(35) \quad \widehat{U}_{(0)}\widehat{T}^i = \widehat{U}_{(i)}.$$

jest wektorem prawdopodobieństwa dla stanów mrówki w chwili i , gdzie $i = 1, 2, 3, \dots$

Znaczenie wniosku VIII można rozbudować o istotne aplikacje praktyczne, przez wskazanie procedury uzupełniania elementów macierzy przejścia \widehat{T} dla konkretnej instancji problemu optymalizacyjnego. Powstaje zatem pytanie, czy dla zadanej piątki

$$(36) \quad (\Sigma, \mathcal{R}, \Delta, \|\cdot\|, \searrow)$$

potrafimy wyznaczyć wartość dowolnego elementu $\widehat{T}[i, j]$ rozważanej macierzy. Odpowiedź jest pozytywna, wystarczy sięgnąć do przedstawionego wcześniej schematu dyskretnego algorytmu mrówkowego, własności reguł ewolucji jak i definicji stanu mrówki. Niech

$$(37) \quad s_i = (F_i, H_i, \omega_i, \omega_i^*)$$

$$(38) \quad s_j = (F_j, H_j, \omega_j, \omega_j^*)$$

będą odpowiednio stanem mrówki w chwili t oraz $t + 1$. Jak wspomnieliśmy wcześniej $\widehat{T}[i, j]$ określa prawdopodobieństwo przejścia mrówki w chwili t ze stanu s_i do stanu s_j w chwili $t + 1$. Korzystając ze schematu heurystyki DAS prawdopodobieństwo takiego przejścia jest dodatnie jeżeli spełniona jest sekwencja następujących zdarzeń:

$$(39) \quad \begin{aligned} \text{SCR}(F_i, H_i) &= \omega_j, & \text{SAR}(\omega_j, \omega_i^*) &= \omega_j^*, \\ \text{PUR}(F_i, H_i, \omega_i, \omega_j) &= (F_j, H_j), \end{aligned}$$

w przeciwnym przypadku $\widehat{T}[i, j] = 0$, tj. stany s_j jest nieosiągalny ze stanu s_i względem możliwości ewolucyjnych algorytmu DAS. Dalej zajmujemy się zatem jedynie tymi elementami macierzy przejścia, które spełniają powyższy warunek następstwa działania. W tym przypadku zachowanie mrówki zależy bezpośrednio od wzajemnych relacji wartości funkcji oceny słów-rozwiązań ω_i , ω_i^* oraz ω_j , które determinują działanie mechanizmów SAR oraz PUR. Wyróżniamy trzy rozłączne drogi ewolucji zachowania mrówki w chwili t :

- WARIANT I: $\|\omega_j\| < \|\omega_i^*\|$, stąd $\|\omega_j\| < \|\omega_i\|$ – nowe rozwiązanie ω_j jest lepsze od najlepszego rozwiązania wyznaczonego do tej pory ω_i^* , a tym samym jest lepsze od rozwiązania ω_i ,
- WARIANT II: $\|\omega_j\| \geq \|\omega_i^*\|$ oraz $\|\omega_j\| = \|\omega_i\|$ – nowe rozwiązanie ω_j jest nie lepsze od najlepszego rozwiązania wyznaczonego do tej pory ω_i^* i jest tak samo dobre jak rozwiązanie ω_i
- WARIANT III: $\|\omega_j\| \geq \|\omega_i^*\|$ oraz $\|\omega_j\| \neq \|\omega_i\|$ – nowe rozwiązanie ω_j jest nie lepsze od najlepszego rozwiązania wyznaczonego do tej pory ω_i^* i nie jest tak samo dobre jak rozwiązanie ω_i .

W wariantach I i III jedynym czynnikiem wprowadzającym niedeterminizm jest realizacja reguły konstrukcji rozwiązania SCR. Działanie reguł SAR oraz PUR w tym przypadku nie wnosi własnych zachowań probabilistycznych i jest jedynie konsekwencją wyniku reguły SCR. Stąd, jeżeli słowa-rozwiązania stanowiące składowe stanów mrówki s_i oraz s_j spełniają dowolny ze wskazanych dwóch wariantów wzajemnych zależności, to prawdopodobieństwo przejścia mrówki między rozważanymi stanami jest równe prawdopodobieństwu zajścia zdarzenia polegającego na konstrukcji słowa-rozwiązania ω_j w warunkach śladu feromonowego zapisanego w strukturach F_i oraz H_i , czyli

$$(40) \quad \widehat{T}[i, j] = Pr(\text{SCR}(F_i, H_i) = \omega_j).$$

Zgodnie z własnościami reguł wyboru sąsiedztwa NCR (wzór (9), strona 88) oraz konstrukcji rozwiązania SCR (wzór (15), strona 89) przedstawionymi w rozdziale trzecim artykułu, zależność (40) można rozwinąć do postaci

elementarnej (zakładamy, że $\omega_j = x_{l_1} x_{l_2} \dots x_{l_r}$)

$$(41) \quad \widehat{T}[i, j] = Pr(\text{NCR}(F_i, \epsilon) = x_{l_1}).$$

$$(42) \quad \prod_{p=1}^{r-1} Pr(\text{NCR}(H_i[l_p, \cdot], x_{l_1} x_{l_2} \dots x_{l_p}) = x_{l_{p+1}})$$

$$(43) \quad = \frac{F_i[l_1]}{\sum_{\{q:\Delta(\omega x_q)=1\}} F_i[q]} \cdot \prod_{p=1}^{r-1} \frac{H_i[l_p, l_{p+1}]}{\sum_{\{q:\Delta(\omega x_q)=1\}} H_i[l_p, q]}.$$

Dalej korzystając z wniosku IV (strona 89) dla wariantów ewolucji I i III przejścia mrówki między stanami s_i oraz s_j , zachodzi $\widehat{T}[i, j] \in (0, 1]$.

Wykonanie dyskretnego algorytmu mrówkowego w wariacie II, gdzie $\|\omega_j\| = \|\omega_i\|$, jest zależne dodatkowo od wyboru niedeterministycznego, ukrytego wewnątrz funkcji aktualizacji śladu feromonowego PUR. Przypomnijmy, zgodnie z schematem tego mechanizmu

...

jeżeli $\|\omega\| = \|\omega'\|$
z prawdopodobieństwem $Pr(\text{INV}(F, H, \omega) = 1)$ wykonaj
 $F \leftarrow \text{inc}_{\mathcal{F}}(F, \omega)$
 $H \leftarrow \text{inc}_{\mathcal{H}}(H, \omega)$
albo
 $F \leftarrow \text{dec}_{\mathcal{F}}(F, \omega)$
 $H \leftarrow \text{dec}_{\mathcal{H}}(H, \omega)$

...

w przypadku równej wartości funkcji jakości dla nowo skonstruowanego rozwiązania ω (teraz ω_j) jak i rozwiązania skonstruowanego w bezpośrednio poprzedzającej iteracji ω^{-1} (teraz ω_i) dokonujemy wyboru sposobu aktualizacji struktur feromonowych z prawdopodobieństwem określonym funkcją innowacyjności rozwiązania INV. Zatem przejście mrówki między stanami $s_i = (F_i, H_i, \omega_i, \omega_i^*)$ oraz $s_j = (F_j, H_j, \omega_j, \omega_j^*)$ jest równe w tym przypadku prawdopodobieństwu zajścia ciągu dwóch **zdarzeń niezależnych**, polegającego na (zdarzenie pierwsze) konstrukcji słowa-rozwiązania ω_j w warunkach śladu feromonowego zapisanego w strukturach F_i oraz H_i i (zdarzenie drugie) właściwej aktualizacji śladu feromonowego, z uwzględnieniem czynnika probabilistycznego zadanego działaniem funkcji innowacji rozwiązania. Stąd

$$(44) \quad \widehat{T}[i, j] = Pr(\text{SCR}(F_i, H_i) = \omega_j) \cdot Pr(\text{INV}(F_i, H_i, \omega_j) = 1)$$

dla $F_j = \text{inc}_{\mathcal{F}}(F_i, \omega_j)$, $H_j = \text{inc}_{\mathcal{H}}(H_i, \omega_j)$, albo

$$(45) \quad \widehat{T}[i, j] = Pr(\text{SCR}(F_i, H_i) = \omega_j) \cdot (1 - Pr(\text{INV}(F_i, H_i, \omega_j) = 1))$$

dla $F_j = \text{dec}_{\mathcal{F}}(F_i, \omega_j)$, $H_j = \text{dec}_{\mathcal{H}}(H_i, \omega_j)$. Ponownie wartość elementu $\widehat{T}[i, j]$ macierzy przejścia możemy podać w postaci elementarnej. Korzystając kolejno z własności funkcji INV (wzór (19), strona 99) dla wyniku (44) i podtrzymując założenie $\omega_j = x_{l_1} x_{l_2} \dots x_{l_r}$ mamy

$$(46) \quad \widehat{T}[i, j] = Pr(\text{SCR}(F_i, H_i) = \omega_j) \cdot Pr(\text{INV}(F_i, H_i, \omega_j) = 1)$$

$$(47) \quad \frac{F_i[l_1]}{\sum_{\{q:\Delta(\omega x_q)=1\}} F_i[q]} \cdot \prod_{p=1}^{r-1} \frac{H_i[l_p, l_{p+1}]}{\sum_{\{q:\Delta(\omega x_q)=1\}} H_i[l_p, q]}$$

$$(48) \quad \frac{1}{r} \cdot \left(\frac{1 + \max(F_i) - F_i[l_1]}{1 + \max(F_i)} + \right.$$

$$(49) \quad \left. \sum_{q=1}^{r-1} \frac{1 + \max(H_i[l_q, \cdot]) - H_i[l_q, l_{q+1}]}{1 + \max(H_i[l_q, \cdot])} \right).$$

Ponieważ $Pr(\text{SCR}(F_i, H_i) = \omega_j) \in (0, 1]$ i zgodnie z treścią wniosku V (strona 91) $Pr(\text{INV}(F_i, H_i, \omega_j) = 1) \in (0, 1)$, to ostatecznie wartość elementu rozważanej macierzy $\widehat{T}[i, j]$ dla wariantu II jest dobrze określoną liczbą rzeczywistą z przedziału $(0, 1)$. Przypadek zależności (45) można rozpisać analogicznie.

Powyższe rozważania mają istotny charakter jakościowy. Wykazują, że jeżeli znany jest nam wektor $\widehat{U}_{(0)}$ rozkładu prawdopodobieństwa mrówki w stanie początkowym dla zadanego problemu optymalizacyjnego, rozwiązywanego przy użyciu metody DAS, i zbudowaliśmy kompletną postać macierzy przejścia \widehat{T} zgodnie z przedstawionymi zasadami, to wyznaczenie t -tej potęgi wspomnianej macierzy i prawostronne przyłożenie otrzymanego rezultatu do wektora $\widehat{U}_{(0)}$ pozwoli nam określić w sposób dokładny rozkład prawdopodobieństwa wektora stanu w chwili t . Tym samym możemy czynić probabilistyczne próby przewidywania kierunku ewolucji dyskretnego algorytmu mrówkowego. Na tej podstawie w następnym rozdziale sformułujemy twierdzenie o punktowej zbieżności rozważanego narzędzia optymalizacyjnego.

4. Zbieżność punktowa algorytmu DAS. Rozważania dotyczące zbieżności punktowej rozpoczniemy od przypomnienia wybranych pojęć wprowadzonych w poprzednim rozdziale. Przyjeliśmy, że \mathfrak{s} jest liczbnością zbioru \mathcal{S} wszystkich indeksowanych stanów mrówki. Następnie $\widehat{U}_{(t)} \in [0, 1]^{\mathfrak{s}}$ jest stochastycznym wektorem kolumnowym, determinującym rozkład prawdopodobieństwa stanu mrówki w chwili t , a $\widehat{T} \in [0, 1]^{\mathfrak{s} \times \mathfrak{s}}$ jest macierzą przej-

ścia między stanami. Przez **zbieżność punktową** algorytmu heurystycznego w przyjętym modelu probabilistycznym będziemy rozumieli stwierdzenie istnienia takiego wyróżnionego wektora $\widehat{U} \in [0, 1]^5$, że

$$(50) \quad \forall \widehat{U}_{(0)} \in [0, 1]^5 : \lim_{t \rightarrow \infty} \widehat{T}^t \widehat{U}_{(0)} = \widehat{U},$$

gdzie \widehat{U} jest wektorem rozkładu prawdopodobieństwa stanu mrówki z wszystkimi elementami równymi 0, z wyjątkiem elementu określającego prawdopodobieństwo wyróżnionego stanu s , dla którego wartość odpowiedniego elementu wektora \widehat{U} jest równa 1. Wektor \widehat{U} będziemy nazywali dalej **punktem zbieżności**, a stan s **stanem zbieżności**. Warto zauważyć, że tak przyjęta definicja zbieżności punktowej implikuje takie działanie algorytmu mrówkowego, w którym dla dowolnego stanu początkowego i odpowiednio długiego iterowania reguł ewolucji, wynik działania winien zawsze prowadzić do tego samego wyróżnionego stanu końcowego s , z ustalonymi postaciami struktur feromonowych F i H oraz rozwiązaniami ω i ω^* .

Ze względów praktycznych powyższe kryterium działania algorytmu mrówkowego jest zbyt rygorystyczne. Z punktu widzenia zadania optymalizacyjnego, wszystkie stany $s \in \mathcal{S}$, dla $s = (F, H, \omega, \omega^*)$, gdzie ω^* jest rozwiązaniem optymalnym, tj. $\omega^* \in \mathcal{R}^*$, są **nierozróżnialne** względem porządnego celu działania dyskretnego algorytmu mrówkowego (tak jak przyjęliśmy w rozdziale pierwszym, szukamy dowolnego rozwiązania optymalnego, a nie wszystkich rozwiązań optymalnych). Istotne jest natomiast, by stan zbieżności s charakteryzował się tym, że element ω^* tego stanu, czyli rozwiązanie dotychczas najlepsze, był w rzeczywistości rozwiązaniem możliwie bliskim optymalnemu. Postać pozostałych elementów stanu, tj. wektora stopni nasycenia F , macierzy stopni nasycenia H jak i właśnie wyznaczonego słowa-rozwiązania ω jest drugorzędna. Na tej podstawie wszystkie stany $s \in \mathcal{S}$, dla których rozwiązanie dotychczas najlepsze ω^* jest rozwiązaniem optymalnym (jednym z rozwiązań optymalnych) rozważanego zadania optymalizacyjnego, możemy zgrupować w jeden **super-stan** s^* taki, że

$$(51) \quad s^* = \{s \in \mathcal{S} : \omega^* \in \mathcal{R}^*\}$$

i dalej

$$(52) \quad \mathcal{S}^* = (\mathcal{S} \setminus \{s \in \mathcal{S} : \omega^* \in \mathcal{R}^*\}) \cup \{s^*\} = \{s_1, s_2, \dots, s_{s^*}\}$$

będzie zbiorem indeksowanych stanów mrówki z super-stanem, gdzie $\mathcal{S}^* \subseteq \mathcal{S}$. Wszystkie dotychczasowe wyniki dla modelu teoretycznego heurystyki DAS i przestrzeni stanów mrówki \mathcal{S} zachodzą także w zredukowanej przestrzeni stanów \mathcal{S}^* . Wprowadzenie tej ostatniej pozwoli nam na zbadanie punktowej zbieżności dyskretnego algorytmu mrówkowego względem porządnego stanu zbieżności s^* . Postawimy zatem następującą tezę

TEZA . Niech $\widehat{U}^* \in [0, 1]^5$ będzie wyróżnionym wektorem rozkładu praw-

dopodobieństwa stanów mrówki, gdzie $\widehat{U}^*[v] = 1$ wttw, gdy v jest indeksem super-stanu s^* w zbiorze \mathcal{S}^* , wtedy

$$(53) \quad \forall \widehat{U}_{(0)}^* \in [0, 1]^{\mathcal{S}^*} : \lim_{t \rightarrow \infty} \widehat{T}^{*t} \widehat{U}_{(0)}^* = \widehat{U}^*.$$

Uzasadnienie tezy wymaga analizy zachowania ewolucyjnego mrówki, czyli probabilistycznej wędrówki po przestrzeni stanów \mathcal{S}^* . Na wstępie zauważamy, że zbiór \mathcal{S}^* można reprezentować graficznie w postaci skończonego **grafu skierowanego** $G = (V, E)$, gdzie $V = \mathcal{S}^*$ oraz $E \subseteq \mathcal{S}^* \times \mathcal{S}^*$. Wierzchołki (stany) grafu G są ułożone horyzontalnie warstwami (poziomami), ze względu na wartość funkcji oceny przynależnej do elementu ω^* danego stanu. Wszystkie stany o tej samej wartości wspomnianej cechy znajdują się w tej samej warstwie grafu. Zgodnie z **nierosnącą** charakterystyką reguły akceptacji rozwiązania SAR, krawędzie łączące wierzchołki prowadzą między stanami albo z tej samej warstwy do tej samej warstwy (identyczna wartość funkcji oceny), albo z warstwy niższej (większa wartość funkcji oceny) do warstwy wyższej (mniejsza wartość funkcji oceny). Najwyższa warstwa grafu składa się z pojedynczego wierzchołka reprezentującego super-stan s^* , z którego prowadzi tylko jedna krawędź cykliczna (s^*, s^*) .

Niech dalej $\widehat{U}_{(0)}^*$ będzie dowolnym wektorem rozkładu prawdopodobieństwa stanów mrówki w chwili początkowej $t = 0$. Zakładamy, że działanie algorytmu DAS prowadzi do następującego ciągu przekształceń wektora $\widehat{U}_{(0)}^*$ w czasie:

$$(54) \quad \widehat{U}_{(0)}^*, \widehat{U}_{(1)}^*, \widehat{U}_{(2)}^*, \dots$$

Wykażemy, że dla dowolnego elementu wektora $\widehat{U}_{(0)}^*[i]$, gdzie i **nie jest** indeksem super-stanu s^* w przestrzeni \mathcal{S}^* , zachodzi

$$(55) \quad \lim_{t \rightarrow \infty} \widehat{U}_{(t)}^*[i] = 0.$$

Innymi słowy, niezależnie od wyboru stanu początkowego $s_i \neq s^*$, prawdopodobieństwo tego, że w chwili $t = 0, 1, 2, \dots$ mrówka znajduje się w stanie s_i jest graniczenie równe 0, dla $t \rightarrow \infty$. Przyjmujemy, że początkowa wartość elementu $\widehat{U}_{(0)}^*[i]$ wektora rozkładu prawdopodobieństwa jest dodatnia (w przeciwnym przypadku rozważana zależność jest spełniona w sposób oczywisty). Stąd odnosząc się do opisanego grafu skierowanego, istnieje dodatnie prawdopodobieństwo tego, że mrówka w chwili $t = 0$ znajduje się w jednym ze stanów warstwy innej, niż warstwa najwyższa. W tym przypadku wykonanie pojedynczej iteracji algorytmu DAS prowadzi do jednego z trzech **rozłącznych** zdarzeń:

- zdarzenie $A_{(0)}$ – mrówka pozostanie w tej samej warstwie wierzchołków grafu G , stan s_i będzie dalej osiągalny w kolejnej iteracji algorytmu,

- zdarzenie $B_{(0)}$ – mrówka pozostanie w tej samej warstwie wierzchołków grafu G , stan s_i nie będzie dalej osiągalny w kolejnej iteracji algorytmu,
- zdarzenie $C_{(0)}$ – mrówka przejdzie do wyższej warstwy wierzchołków grafu G , stan s_i nie będzie dalej osiągalny w kolejnej iteracji algorytmu.

Bazując na schemacie dyskretnego algorytmu mrówkowego, probabیلیtyka zdarzenia $C_{(0)}$ określona jest jedynie przez niedeterminizm reguły konstrukcji rozwiązania NCR. Pozostałe reguły zachowują się w tym przypadku w sposób deterministyczny. Teraz, zgodnie z treścią wniosku IV (strona 89), dowolne słowo-rozwiązanie $\omega \in \mathcal{R}$, w szczególności także słowo-rozwiązanie optymalne zawarte w super-stanie s^* może, z dodatnim prawdopodobieństwem, być rezultatem zastosowania reguły NCR w dowolnym stanie struktur feromonowych F oraz H . Zatem prawdopodobieństwo zajścia zdarzenia $C_{(0)}$, dla dowolnego $s_i \neq s^*$ jest dodatnie. Na tej podstawie

$$(56) \quad Pr(A_{(0)}) = 1 - Pr(B_{(0)}) - Pr(C_{(0)}),$$

czyli

$$(57) \quad Pr(A_{(0)}) \in [0, 1).$$

Analogicznie można wykazać, że własność (57) zachodzi dla każdej kolejnej iteracji algorytmu DAS. Stąd

$$(58) \quad Pr(A_{(t)}) \in [0, 1),$$

gdzie $t = 1, 2, 3, \dots$

Następnie wystarczy zauważyć, że po wykonaniu pierwszego przejścia w grafie G , stan s_i jest nadal **osiągalny** z prawdopodobieństwem równym iloczynowi prawdopodobieństwa pierwotnego dla rozważanego stanu (równego $\widehat{U}_{(0)}^*[i]$) i prawdopodobieństwa zajścia zdarzenia $A_{(0)}$. Osiągalność nie oznacza faktu ponownej lokalizacji mrówki w wybranym stanie s_i , mówi jedynie o dostępności tego stanu (istnieje wiele stanów w tej samej warstwie grafu, do których przejście ze stanu s_i spełnia warunki zdarzenia $A_{(0)}$). Na tej podstawie

$$(59) \quad \widehat{U}_{(1)}^*[i] \leq \widehat{U}_{(0)}^*[i] \cdot Pr(A_{(0)})$$

oraz ogólnie, dla $t = 1, 2, 3, \dots$ prawdopodobieństwo tego, że w chwili t mrówka znajduje się w stanie s_i jest ograniczone przez

$$(60) \quad \widehat{U}_{(t)}^*[i] \leq \widehat{U}_{(0)}^*[i] \cdot \prod_{i=0}^{t-1} Pr(A_{(i)}).$$

W rezultacie, pamiętając o własności (58), ustalamy

$$(61) \quad \lim_{t \rightarrow \infty} \widehat{U}_{(t)}^*[i] \leq \lim_{t \rightarrow \infty} \left(\widehat{U}_{(0)}^*[i] \cdot \prod_{i=0}^{t-1} Pr(A_{(i)}) \right)$$

$$(62) \quad \leq \lim_{t \rightarrow \infty} \left(\widehat{U}_{(0)}^* [i] \cdot (1 - \epsilon)^t \right)$$

i pewnego dowolnie małego $\epsilon \in (0, 1]$. Stąd

$$(63) \quad \lim_{t \rightarrow \infty} \left(\widehat{U}_{(0)}^* [i] \cdot (1 - \epsilon)^t \right) = 0.$$

Wykazaliśmy zatem, że dla dowolnego elementu $\widehat{U}_{(0)}^* [i]$ dowolnego początkowego wektora rozkładu prawdopodobieństwa, gdzie i **nie jest** indeksem super-stanu s^* w przestrzenie \mathcal{S}^* , zachodzi

$$(64) \quad \lim_{t \rightarrow \infty} \widehat{U}_{(t)}^* [i] = 0.$$

Ponieważ każdy wektor $\widehat{U}_{(0)}^*$ jest w przyjętym modelu teoretycznym wektorem stochastycznym, to

$$(65) \quad \lim_{t \rightarrow \infty} \widehat{U}_{(t)}^* [v] = 1,$$

wttw, gdy v jest indeksem super-stanu s^* w zbiorze \mathcal{S}^* . Uzasadnia to poprawność postawionej tezy.

Ostateczny rezultat zapiszemy więc w postaci twierdzenia kończącego prezentację wyników pracy badawczej w tym rozdziale oraz całym artykule.

TWIERDZENIE (o zbieżności punktowej algorytmu DAS.) *Dyskretny algorytm mrówkowy jest punktowo zbieżny.*

Zakończenie. Zaprezentowane w poprzednim rozdziale twierdzenie o zbieżności punktowej dyskretnego algorytmu mrówkowego jest wynikiem bazowym dla dalszej analizy teoretycznej własności algorytmu DAS. Wynik ten istotnie odróżnia omawianą heurystykę od np. klasycznych algorytmów genetycznych, które tej własności nie posiadają (por. [22, 23, 15, 14, 16]). Własność punktowej zbieżności umożliwia rozpoczęcie prac badawczych w zakresie asymptotyki i złożoności rozważanej metody probabilistycznej. Niezależnie od rezultatów teoretycznych, autor prowadzi także prace eksperymentalne, ukierunkowane na implementację algorytmu DAS w silnie współbieżnym środowisku obliczeniowym. Dodatkowo rozważane jest także rozszerzenie dyskretnego algorytmu mrówkowego o mechanizm samoadaptacji, w zakresie parametrów sterujących (rozmiar populacji mrówiska m , maksymalna wartość współczynnika śladu feromonowego τ_{max}) jak i modyfikacji funkcji oceny słów-rozwiązań $\|\cdot\|$. Wszystkie prace rozwojowe prowadzone są przy równoczesnym założeniu dalszej możliwości analizy procesu ewolucji dyskretnego algorytmu mrówkowego, z użyciem przedstawionego modelu stochastycznego.

Artykuł powstał w ramach badań wspieranych projektem MNiSzW Nr N N519 5788038. Autor pracy jest wdzięczny Profesorom: Witoldowi Kosiń-

skiemu i Zbigniewowi Michalewiczowi oraz Dr. Stefanowi Kotowskiemu za cenne uwagi i dyskusje.

Literatura

- [1] U. Boryczka, *Algorytmy optymalizacji mrowiskowej*, Wydawnictwo Uniwersytetu śląskiego, Katowice, 2006.
- [2] P. Cichosz, *Systemy uczące się*, WNT, Warszawa 2000.
- [3] J. Cytowski, *Algorytmy genetyczne: podstawy i zastosowania*, Seria: Problemy Współczesnej Nauki – Teoria i Zastosowania, **18**, Akademicka Oficyna Wydawnicza PLJ, Warszawa, 1996.
- [4] M. Dorigo, G. Di Caro, L. M. Gambardella, *Ant Algorithms for Discrete Optimization*, Technical Report IRIDIA/98-10, Universit Libre de Bruxelles, Belgium, 10, 1999.
- [5] M. Dorigo, V. Maniezzo, A. Colorni, *Ant System: optimization by colony of cooperating agents*, IEEE Transactions on Systems, Man and Cybernetics, **26**, 1996.
- [6] M. Dorigo, L. M. Gambardella, *A Study of Some Properties of Ant-Q*. Proceedings of Fourth International Conference on Parallel Problem Solving from Nature, PPSNIV, Lecture Notes in Computer Science, str. 656–665, Berlin, 1996. Springer-Verlag.
- [7] M. Dorigo, T. Stützle, *A short convergence proof for a class of ACO algorithms*, IEEE Transactions on Evolutionary Computation, vol 6, 2002.
- [8] M. Dorigo, T. Stützle, *Ant Colony Optimization*. The MIT Press, 2004.
- [9] D. E. Goldberg, *Algorytmy genetyczne i ich zastosowania*, WNT, Warszawa, 1995.
- [10] W. J. Gutjahr, *ACO algorithms with guaranteed convergence proof to the optimal solution*, Information Processing Letters, vol 82, 2002.
- [11] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, 1975.
- [12] R. B. Hollstien, *Artificial Genetic Adaptation in Computer Control Systems*, Ph.D. Thesis, University of Michigan, 1971.
- [13] P. Kieś, Z. Michalewicz, *Podstawy algorytmów genetycznych*, *Matematyka Stosowana. Matematyka dla Społeczeństwa*, **1** (44), 2000, 68–91.
- [14] W. Kosiński, S. Kotowski, *Limit Properties of Evolutionary Algorithms*, Chapter 1 in: *Advances in Evolutionary Algorithms*, W. Kosiński (Ed.), IN-TECH 2008, pp. 1–28, ISBN 978-953-7619-11-4.
- [15] S. Kotowski, *Analysis of Genetic Algorithms by Dynamical System Methods*, *Analiza algorytmów genetycznych jako układów dynamicznych*, in Polish, Ph.D. Thesis, IPPT PAN, Warszawa 2008.
- [16] S. Kotowski, W. Kosiński, Z. Michalewicz, P. Synak, Ł. Brocki, *On Classification Tools for Genetic Algorithms*, *Fundamenta Informaticae*, **96**(4) 2009, 477–491, DOI 10.3233/FI-2009-189.
- [17] A. Lasota, *Asymptotyczne własności półgrup operatorów Markowa*, *Matematyka Stosowana. Matematyka dla Społeczeństwa*, **3** (45), 2002, 39–51.
- [18] Z. Michalewicz, *Algorytmy genetyczne + struktury danych = programy ewolucyjne*, WNT, Warszawa, 1996.
- [19] J. E. Rowe, *The dynamical system models of the simple genetic algorithm*, w *Theoretical Aspects of Evolutionary Computing*, Leila Kallel, Bart Naudts, Alex Rogers (Eds.), Springer, 2001, 31–57.
- [20] R. Rudnicki, *On asymptotic stability and sweeping for Markov operators*, *Bull. Polish Acad. Sci. Math.*, **43** (1995), 245–262.

- [21] R. Schaefer, *Podstawy genetycznej optymalizacji globalnej*, Wydawnictwo Uniwersytetu Jagiellońskiego, Kraków 2002.
- [22] J. Socała, *Asymptotic behaviour of the iterates of nonnegative operators on a Banach lattice*, Ann. Polon. Math., **68** (1), (1998), 1–16.
- [23] J. Socała, W. Kosiński, *Lower-bound function method in the convergence analysis of genetic algorithms*, (in Polish: Zastosowanie metody funkcji dolnej do badania zbieżności algorytmów genetycznych,) Matematyka Stosowana. Matematyka dla Społeczeństwa, PTM, Warszawa, **8** (49), 2007 , 33–44.
- [24] J. Socała, W. Kosiński, S. Kotowski, *O asymptotycznym zachowaniu prostego algorytmu genetycznego*, Matematyka Stosowana. Matematyka dla Społeczeństwa, PTM, Warszawa, **6** (47), 2005, 70–86.
- [25] M. D. Vose, *Modelling Simple Genetic Algorithms*, Evolutionary Computation, **3** (4) 453-472, 1996.
- [26] M. D. Vose, *The Simple Genetic Algorithm: Foundation and Theory*, MIT Press, Cambridge, MA, 1999.

Paweł Rembelski
Wydział Informatyki
Polsko-Japońska Wyższa Szkoła
Technik Komputerowych
ul. Koszykowa 86, 02-008 Warszawa
E-mail: rembelski@pjawstok.edu.pl

Abstract. Discrete Ant System based on M. Dorigo results on Ant System is introduced and defined as a Markov chain. This probabilistic model is presented in details with finite space characteristic and evolution operator description. Finally the pointwise convergence of Discrete Ant Algorithm is stated and justified.

Keywords: optimizing algorithms, nature based algorithms, discrete ant algorithm, pheromone, Markov chain, pointwise convergence.

(wpłynęło 27 lipca 2011 r.)