

Ameisenalgorithmen und ihre Anwendung auf das RCPSP

Sven Albrecht

Hintergrund und Layout von <http://www.learnopenoffice.org>

S. Albrecht: "Ameisenalgorithmen", Seminar *Complex Scheduling* WS 2007/2008

1

Outline

- Einleitung und Motivation
- ACO Metaheuristik
 - Formalisierung
 - Charakteristiken
 - Ameisenattribute
- ACO für das RCPSP
 - Formalisierung
 - Heuristiken
 - Zusätzliche Erweiterungen
 - Ergebnisse

S. Albrecht: "Ameisenalgorithmen", Seminar *Complex Scheduling* WS 2007/2008

2

Einleitung

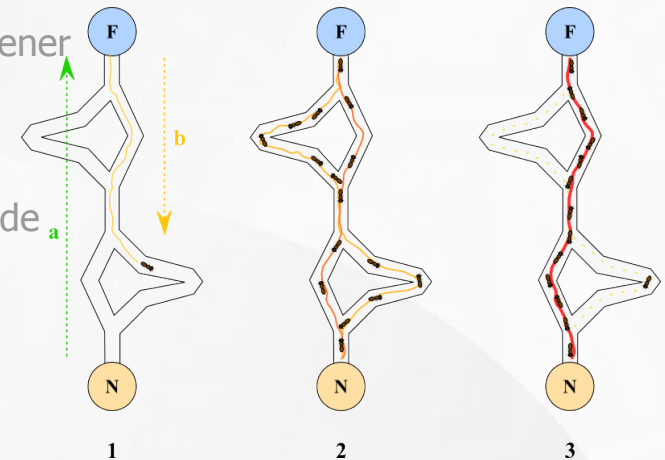
- Ameisenalgorithmen
 - biologisch inspiriert
 - Metaheuristik
 - Stigmergie
 - lokale Änderungen in Umgebung
 - Informationsaustausch
 - dezentral
 - robust gegenüber Änderungen

S. Albrecht: "Ameisenalgorithmen", Seminar *Complex Scheduling* WS 2007/2008

3

Biologische Inspiration I

- Nest und Futterquelle
 - Pfade verschiedener Längen
- Beobachtung
 - anfangs alle Pfade benutzt
 - später fast nur kürzester Pfad

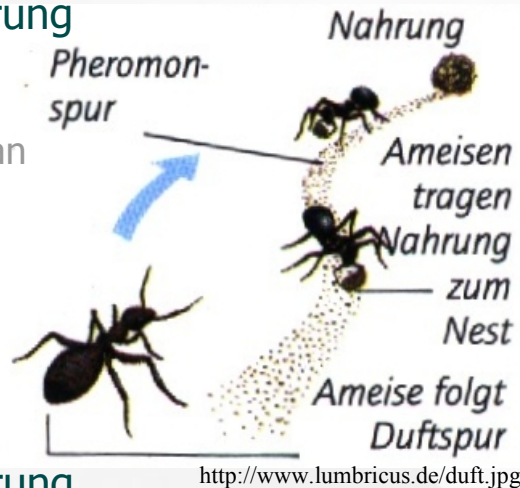


http://fr.wikipedia.org/wiki/Algorithme_de_colonies_de_fourmis

S. Albrecht: "Ameisenalgorithmen", Seminar *Complex Scheduling* WS 2007/2008

4

- lokale Umgebungsänderung
- Pheromonspuren
 - locken andere Ameisen an
 - evaporieren über Zeit
 - verstärken "günstige" Lösungen
 - reagieren flexibel auf Änderungen
- globale Verhaltensänderung



- ACO (ant colony optimization) Algorithmen werden formalisiert durch:
 - endliche Menge an Knoten (Components) $C = \{c_1, c_2, \dots, c_n\}$
 - endliche Menge an Kanten (connections/transitions) $L = \{l_{c_i, c_j} | (c_i, c_j) \in \tilde{C}\}, \tilde{C} \subseteq (C \times C)$
 - Kostenfunktion für jede Kante $l_{c_i, c_j} \in L$, parametrisiert mit Zeit t $J_{c_i, c_j} \equiv J(l_{c_i, c_j}, t)$

ACO Meta-Heuristik: Formalisierung II

- endliche Menge an Bedingungen $\Omega \equiv \Omega(C, L, t)$
- Zustände als Folge von Elementen der Knotenmenge $s = \langle c_i, c_j, \dots, c_k, \dots \rangle$
- Sei S die Menge aller Zustände, so ist $\tilde{S} \subseteq S$ die Menge aller gültigen Zustände unter Berücksichtigung von $\Omega(C, L, t)$
- Nachbarschaft zwischen Zuständen $s_1, s_2 \in S$ existiert, wenn gilt: $s_1 = \langle c_i, \dots, c_{s1} \rangle$, $c_{s2} \in C$, $l_{c_{s1}, c_{s2}} \in L$ und $s_2 = \langle s1, c_{s2} \rangle$

ACO Meta-Heuristik: Formalisierung III

- Lösung $\Psi \in \tilde{S}$
 - erfüllt alle Bedingungen des Problems
 - mehrdimensional es existieren verschiedene Folgen von Elementen aus
- Kosten $J_\Psi(L, t)$ für jede Lösung Ψ , wobei $J_\Psi(L, t)$ eine Funktion über alle J_{c_i, c_j} aller zur Lösung gehörenden Kanten ist
- Anwendung auf Optimierungsprobleme
 - Repräsentiere Problem als Graph $G(C, L)$
 - ACO versucht Pfade Ψ mit möglichst geringen Kosten zu finden, unter Berücksichtigung von Ω

- **Generelle Charakteristiken von ACO:**
 - gute Lösungen nur durch Interaktion
 - jede Ameise nutzt nur lokale Informationen
 - indirekte Kommunikation über Pheromone τ
 - Ameisen selbst sind nicht lernfähig / adaptiv
- **Pheromone τ**
 - werden in einzelnen Knoten c_i gespeichert
 - τ_{ij} beeinflusst Wahl für Kante l_{ij} als nächsten Schritt
 - evaporieren mit der Zeit

- **Einzelne Ameisen k**
 - suchen nach bester Lösung $J_{\Psi} = \min_{\Psi} J_{\Psi}(L, t)$
 - besitzen Gedächtnis M^k
 - Zum Finden von zulässige Lösungen
 - Bewertung von Lösungen
 - Rückverfolgung des Pfades
 - besitzen in Zustand $s_r = \langle s_{r-1}, i \rangle$ eine zulässige Nachbarschaft $N_i^k = \{j | (j \in N_i) \wedge \langle s_r, j \rangle \in \tilde{S}\}$
 - generieren Lösung inkrementell
 - in c_i probabilistische Wahl eines Nachfolgers aus N_i^k

- probabilistische Entscheidung nach *ant-routing-table* $A_i = [a_{ij}]$ in jedem Knoten c_i
 - Pheromon – und Heuristikwerte
 - dem Gedächtnis M^k
 - den Bedingungen
- optional:
 - Startzustand s_s^k und Endbedingungen c^k
 - bei Bewegung von c_i nach c_j Pheromonupdate τ_{ij} für Kante l_{ij}
 - Pfad rückwärts ablaufen für Pheromonupdate
- stirbt nach Erstellung einer Lösung

- **Idee:**
 - Aktivitäten als Knoten
 - Kanten zu allen zulässigen Nachfolgern
 - Start stets bei Dummy-Knoten 0
 - Ende stets bei Dummy-Knoten $n+1$
 - Interpretiere generierten Pfad als Liste für SSGS oder PSGS
 - Benutze Heuristik und Pheromone für Entscheidung

• Entscheidung für nächsten Knoten in c_i

- direkte (lokale) Auswertung: τ_{ij} Pheromone
 η_{ij} Heuristik
 E wählbare Aktivitäten

$$p_{ij} = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{h \in E} [\tau_{ih}]^\alpha [\eta_{ih}]^\beta}$$

- summierte Auswertung:

$$p'_{ij} = \frac{\sum_{k=1}^i ([\gamma^{i-k} \tau_{kj}])^\alpha [\eta_{ij}]^\beta}{\sum_{h \in E} (\sum_{k=1}^i [\gamma^{i-k} \tau_{kh}])^\alpha [\eta_{ih}]^\beta}$$

α, β relativer Einfluß von Pheromon bzw. Heuristik
 γ relativer Einfluß der Historie

• Pheromone

- evaporieren nach einer Generation mit Evaporationsrate ρ $\tau_{ij} = (1 - \rho) \tau_{ij}$

- Kanten l_{ij} aus bisher beste Lösung mit Makespan T^* werden verstärkt $\tau_{ij} = \tau_{ij} + \rho \cdot \frac{1}{2T^*}$

ACO für RCPSP – Heuristiken I

• Adaption von Prioritätsregeln

- normalized LFT $\eta_{ij} = \max_{k \in E} LF_k - LF_j + 1$
- normalized LST $\eta_{ij} = \max_{k \in E} LS_k - LS_j + 1$
- normalized MSL $\eta_{ij} = \max_{k \in E} (LS_k - ES_k) - (LS_j - ES_j) + 1$

mit LF_i 'latest finish time' von Aktivität i
 LS_i 'latest start time' von Aktivität i
 ES_i 'earliest start time' von Aktivität i

ACO für RCPSP – Heuristiken II

- normalized MST $\eta_{ij} = |S_j^*| - \min_{k \in E} |S_k^*| + 1$
- normalized GRPWA $\eta_{ij} = p_j + \sum_{i \in S_j^*} p_i - \min_{k \in E} (p_k + \sum_{i \in S_k^*} p_i) + 1$
- normalized WRUP $\eta_{ij} = \omega |S_j| + (1 - \omega) \sum_{l \in Q} \frac{r_{jl}}{R_l}$

$$- \min_{k \in E} (\omega |S_k| + (1 - \omega) \sum_{l \in Q} \frac{r_{kl}}{R_l}) + 1$$

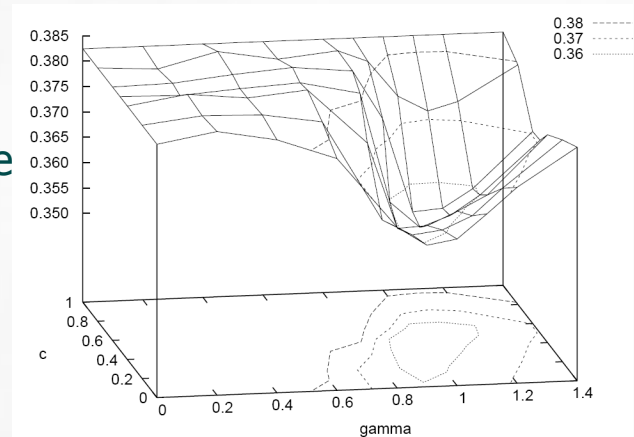
mit S^* Menge aller Nachfolger R_l Kapazität Resource l
 p_i Dauer von Aktivität i r_{il} Ressourcenanforderung
 $\omega \in [0, 1]$

- Kombination von direkter und lokaler Auswertung $\tau'_{ij} = c x_i \tau_{ij} + (1-c) y_i \sum_{k=1}^i \gamma^{i-k} \tau_{kj}$
mit $x_i = \sum_{h \in E} \sum_{k=1}^i \gamma^{i-k} \tau_{kh}$, $y_i = \sum_{h \in E} \tau_{ih}$, $c \in [0,1]$
- Änderung der Parameter α, β, ρ
 - Gewichtung Heuristik / Pheromone verschieden
 - ρ beeinflusst die Konvergenzgeschwindigkeit
- „Vergessen“ der besten Lösung
 - verhindert zu schnelle Konvergenz

- Lokale Suche um gefundene Lösungen
 - prüfe auf Rechtsverschiebungen
 - paarweise Vertauschen von Aktivitäten
- Bidirektionale Planung
 - zwei unabhängige Kolonien
 - Planerstellung von vorn und hinten
 - führe bessere Kolonie weiter

- PSPLIB j120.sm Instanzen als Benchmark
- ACO mit folgenden Modifikationen
 - Parameter $\alpha=1, c=0.5, \gamma=1$ konst.
 - $\beta=2$ linear abnehmend
 - zunächst $\rho=0.025$ gegen Ende $\rho=0.075$
 - "vergesse" beste Lösung nach 10 Generationen
 - 5 Ameisen pro Generation
 - bidirektionale Planung für 100 Generationen
 - nach max. 850 Generationen lokale Suche
 - insgesamt werden 5000 Pläne erstellt

- Einfluß der Parameter γ und c :
 - beste Abweichung von Kritischem Pfad bei $\gamma=1, c=0.5$
- Heuristiken
 - nLST & nLFT am besten
- Bidirektionale Suche
 - besser als nur eine Richtung



- **Vergleiche mit**
 - genetischen Algorithmen
 - Branch & Bound
 - Simulated Annealing
- **maximal 5000 Schedules**
- **ACO am Besten im Vergl.**
 - für 278 von 600 Problemen beste bekannte Lösung
 - ohne Beschränkung 130 neue beste Lösungen

Algorithm	Reference	deviation from LB in %
AS-RCPSP	this paper	35.43*
Self-adapting GA	[15]	35.60*
LS with LR-IP	[31]	(36.2)
GA 1	[14]	36.74*
TO B&B	[12]	(37.1)
SA	[3]	37.68*
GA 2	[14]	38.49
adaptive sampling 1	[33]	38.70
single pass/sampling 1	[22]	38.75
single pass/sampling 2	[21], [22]	38.77*
adaptive sampling 2	[23]	40.45*
GA 3	[27]	40.69*
single pass/sampling 3	[22]	41.84*
GA 4	[14]	42.25
random sampling 1	[20]	43.05*
random sampling 2	[20]	47.61