

Otimização por Colônia de Formigas (Ant Colony Optimization - ACO)

Eros Moreira de Carvalho
Gabriel Silva Ramos

CI209 - Inteligência Artificial
BCC - Bacharelado em Ciência da Computação
DInf - Departamento de Informática
UFPR - Universidade Federal do Paraná

11 de Junho de 2007

- 1 ACO
- 2 TSP
- 3 Algoritmos
- 4 Testes
- 5 Conclusões

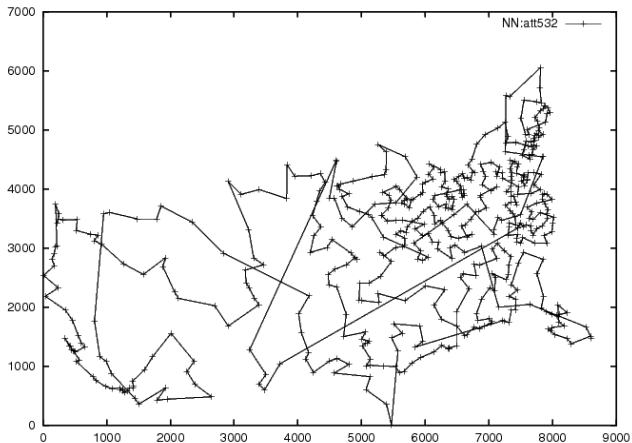
ACO - Otimização por Colônia de Formigas

- Metaheurística baseada em população
- Inspirada na natureza: cooperação de colônias de formigas
- Proposta inicialmente por Marco Dorigo, Vitorio Maniezzo e Alberto Colorni em 1992

Motivações

- Usar tanto informação heurística quanto acúmulo de experiência na construção de uma solução.
- Construção meramente gulosa gera poucas soluções e, em geral, elas são sub-ótimas.
- Uma construção gulosa e randômica, como em GRASP, aumenta a diversidade, mas ainda não incorpora o conhecimento que se obtém durante a busca.
- O uso dos feromônios na construção da solução tem tanto um efeito de diversificação quanto de intensificação.

Grafo de uma construção gulosa



Pseudo-código

Iniciar parâmetros, iniciar rastros de feromônio

Agendar Atividades

 Construção de Soluções

 Ações Globais [Opcional]

 Atualização dos Feromônios

Fim do Agendamento

Fase: Construção de Soluções

- Cada formiga inicia em um nó aleatório do *grafo de construção*;
- A cada iteração, cada formiga decide para qual nó irá em seguida, baseadas em informações *heurísticas* e nos *feromônios* das arestas;
- Repete-se o passo anterior até que todas as formigas tenham construído uma solução completa.

Fase: Ações Globais

- Nesta fase opcional, um agente com conhecimento global do sistema executa ações que não poderiam ser executadas por uma formiga individualmente;
- A ação mais comum nesta fase é aplicar uma *Busca Local* em cada uma das soluções construídas pelas formigas na fase de Construção de Soluções;
- Embora opcional, essa etapa tem sido usada para se obter os melhores resultados com o ACO. ACO com busca local pode ser considerado um algoritmo híbrido[4].

Fase: Atualização dos Feromônios

- Nesta fase, os rastros de feromônio de cada aresta pertencente às soluções de cada formiga são atualizados. Esta fase envolve duas etapas: a *evaporação* e o *incremento* do feromônio depositado em cada aresta;
- Na *evaporação*, o feromônio é diminuído por um fator constante e produz um efeito de diversificação no algoritmo;
- O *incremento* é baseado no uso da aresta no caminho das formigas. Quanto mais a aresta for usada pelas formigas, maior será o depósito de feromônio. O incremento produz um efeito de intensificação no algoritmo.

Aplicando o ACO ao TSP

Na aplicação da metaheurística ACO ao problema do Caixeiro Viajante Simétrico deve-se:

- Associar cada cidade a um nó do *grafo de construção* das formigas;
- Associar cada distância entre duas cidades ao custo de sua aresta correspondente no *grafo de construção*;
- Considerar o valor *heurístico* de cada aresta com sendo o inverso da distância entre as duas cidades conectadas.

Algoritmos

Principais Algoritmos de ACO:

- Ant System - AS
- Ant Colony System - ACS
- MAX-MIN Ant System - MMAS

Outras variações:

- Elitist Ant System
- Rank-based Ant System
- Best-Worst Ant System

Notação

- m - parâmetro que indica o número de formigas na colônia.
- α (alpha) - parâmetro que indica a importância do feromônio.
- β (beta) - parâmetro que indica a importância da informação heurística, no TSP, o custo entre uma cidade e outra.
- ρ (rho) - parâmetro que regula a taxa de evaporação do feromônio.
- τ (tau) - τ_{ij} indica a quantidade de feromônio na aresta ij , e τ_0 é a quantidade de feromônio inicial das arestas.

Notação

- η (eta) - $\eta_{ij} = \frac{1}{d_{ij}}$, onde d_{ij} é a distância entre as cidades i e j .
- N_i^k - Conjunto das cidades que uma formiga k ainda não visitou.
- $\Delta\tau_{ij}^k$ - é definida por

$$\begin{cases} \frac{1}{L^k}, & \text{se aresta } ij \text{ faz parte do caminho da formiga } k \\ 0, & \text{caso contrário} \end{cases}$$

, onde L^k é o custo do caminho percorrido pela formiga k .

- L_{nn} - Custo de um caminho guloso (*nearest neighbor heuristic*)

Ant System

- Desenvolvido por Marco Dorigo, Vitorio Maniezzo e Alberto Coloni em 1996.
- *Fase de Construção*: as formigas que atravessam o grafo em construção fazem uma decisão probabilística em cada nodo. Essa decisão é definida pela fórmula:

$$p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta}$$

- *Atualização do Feromônio*: o feromônio é atualizado por *todas* as formigas, ao completar o caminho:

$$\tau_{ij} = (1 - \rho) \cdot \tau_{ij} + \sum_{k=1}^m \Delta \tau_{ij}^k$$

- *Melhores parâmetros*: $m = n$ $\alpha = 1$ $\beta = 5$ $\rho = 0.5$
 $\tau_0 = 1/\rho \cdot L_{nn}[2]$.

Ant Colony System - ACS

- Desenvolvido por Marco Dorigo e Luca Gambardella em 1997.
- *Fase de Construção*: Com probabilidade q_0 , escolhe a aresta que maximiza a expressão $\tau_{ij}\eta_{ij}^\beta$, caso contrário utiliza a mesma regra de decisão do AS
- *Atualização Local de Feromônio*: a cada iteração da construção, cada formiga atualiza o feromônio da última aresta pela qual passou: $\tau_{ij} = (1 - \varphi) \cdot \tau_{ij} + \varphi \cdot \tau_0$
- *Atualização Global de Feromônio*: feito somente pela melhor formiga, ou seja, somente as arestas visitadas pela melhor formiga são atualizadas: $\tau_{ij} = (1 - \rho) \cdot \tau_{ij} + \rho \cdot \Delta\tau_{ij}^{melhor}$
- *Melhores parâmetros*: $m = 10$ $\alpha = 0.1$ $\beta = 2$ $\rho = 0.1$ $q_0 = 0.9$ $\varphi = 0.1$ $\tau_0 = 1/n \cdot L_{nn}[1]$.

MAX-MIN Ant System - MMAS

- Desenvolvido por Thomas Stützle e Holger Hoos em 1999.
- *Atualização do Feromônio*: feito somente pela melhor formiga, ou seja, somente as arestas visitadas pela melhor formiga são atualizadas: $\tau_{ij} = (1 - \rho) \cdot \tau_{ij} + \Delta\tau_{ij}^{melhor}$
- *Melhor formiga*: pode ser tanto a melhor formiga até o momento ou a melhor formiga de uma iteração. ACS foca na melhor formiga até o momento, enquanto MAX-MIN foca na melhor formiga de uma iteração.
- *Estratégia dinâmica*: usar ora a melhor formiga até o momento, ora a melhor formiga de uma iteração. Algumas implementações do MAX-MIN usam essa estratégia [4].

MAX-MIN Ant System - MMAS

- *Limites para a taxa de Feromônio*: para evitar a estagnação precoce, os rastros de feromônio são limitados a um intervalo $[\tau_{min}, \tau_{max}]$.
- *Inicialização*: $\tau_0 = \tau_{max}$.
- τ_{max} : O limite máximo é definido por $\frac{1}{\rho \cdot L^*}$, onde L^* seria o tamanho do caminho ótimo. Quando não se tem o conhecimento deste valor, pode-se aproximar dele por L_{bs} , isto é, pelo tamanho do melhor caminho encontrado até o momento (Na inicialização utiliza-se L_{nn}).

MAX-MIN Ant System - MMAS

- τ_{min} : Geralmente determinado empiricamente ou analiticamente, como proposto por Stützle[4], por:
$$\tau_{min} = \frac{\tau_{max} \cdot (1 - P_{dec})}{k \cdot P_{dec}}$$
onde k é o número de escolhas que a formiga ainda pode fazer e $P_{dec} = \frac{1}{n} \sqrt[n]{P_{best}}$, onde P_{best} é a probabilidade de uma formiga construir o melhor caminho até agora e n o número de passos no caminho.
- *Melhores parâmetros*: $m = n$ $\alpha = 1$ $\beta = 2$ $\rho = 0.02$ [4].

Comparativo entre as atualizações de Feromônio

Alg.	Expressão	Realizado
AS	$\tau_{ij} = (1-\rho).\tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k$	por todas as formigas ao completar o caminho
ACS	$\tau_{ij} = (1 - \varphi).\tau_{ij} + \varphi.\tau_0$	por todas as formigas em cada passo da construção
ACS	$\tau_{ij} = (1 - \rho).\tau_{ij} + \rho.\Delta\tau_{ij}^{melhor}$	apenas pela melhor formiga até agora ao completar o caminho
MMAS	$\tau_{ij} = (1 - \rho).\tau_{ij} + \Delta\tau_{ij}^{melhor}$	apenas pela melhor formiga da iteração (ou melhor até o momento) ao completar o caminho

Comparações entre Algoritmos

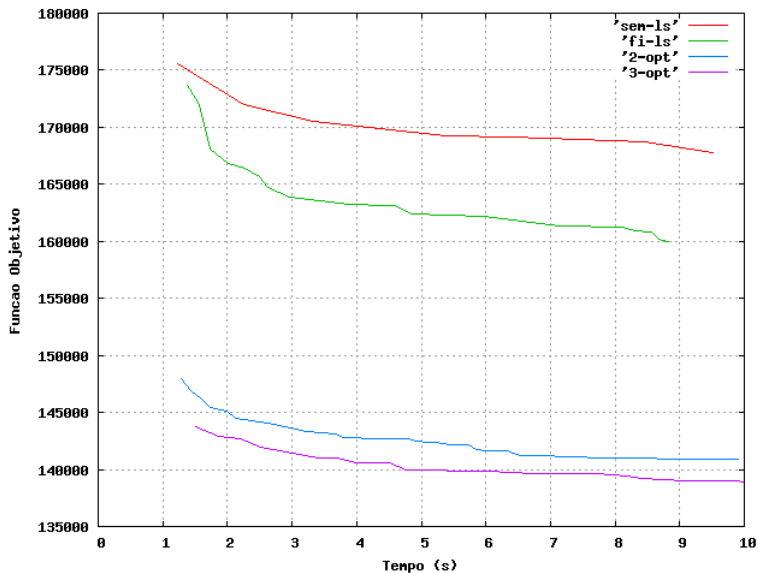
- Foram utilizados os melhores parâmetros da literatura e alguns escolhidos pela equipe.
- Buscas locais: *2-opt*, *3-opt*, *First Improvement* e nenhuma busca.
- Critério de Parada: tempo limite de 10 segundos por teste

Instância: **pcb3038** (Ótimo: 137694)

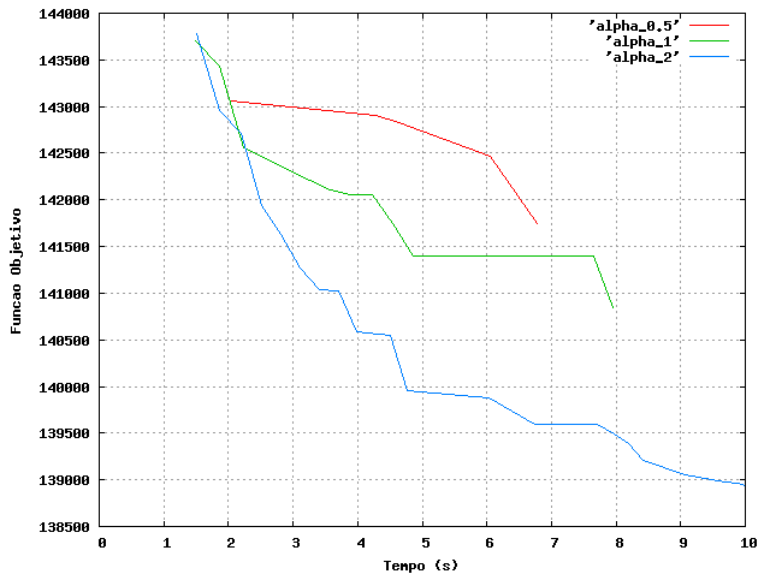
3 Algs x 108 Comb. de parâmetros x 10 testes cada = 3240 testes

Alg.	Média	σ	Melhor	Pior	Tempo Méd.
AS	158469.9	19366.7	141109	239852	7.06
ACS	154165.0	13257.9	139512	178748	7.88
MMAS	162705.5	27080.5	138924	256076	8.27

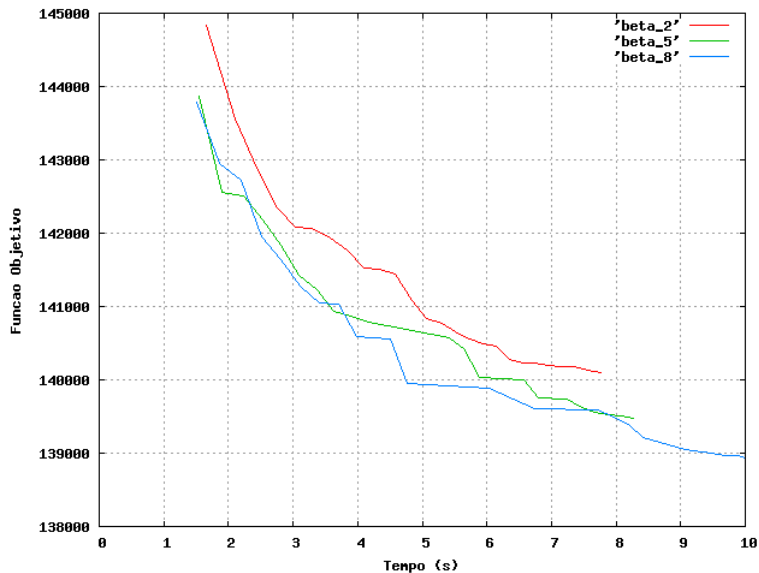
Variação da busca local



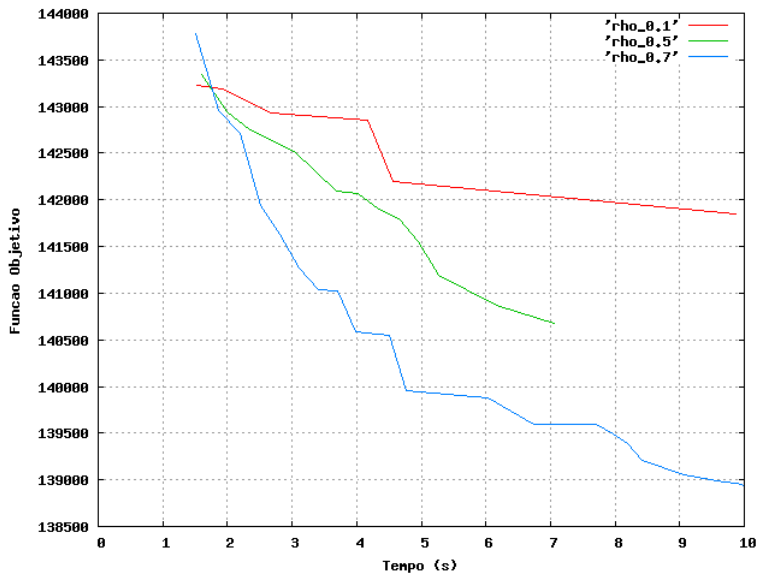
Variação do α



Variação do β



Variação do ρ

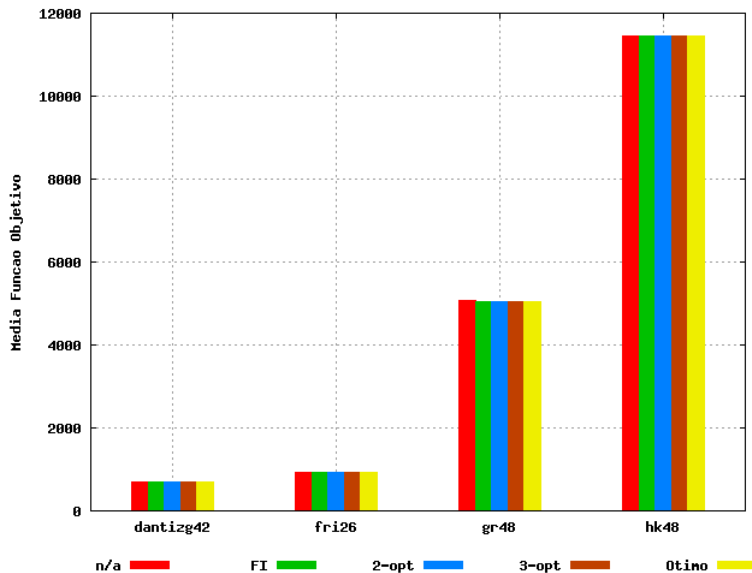


Testes com as bases propostas

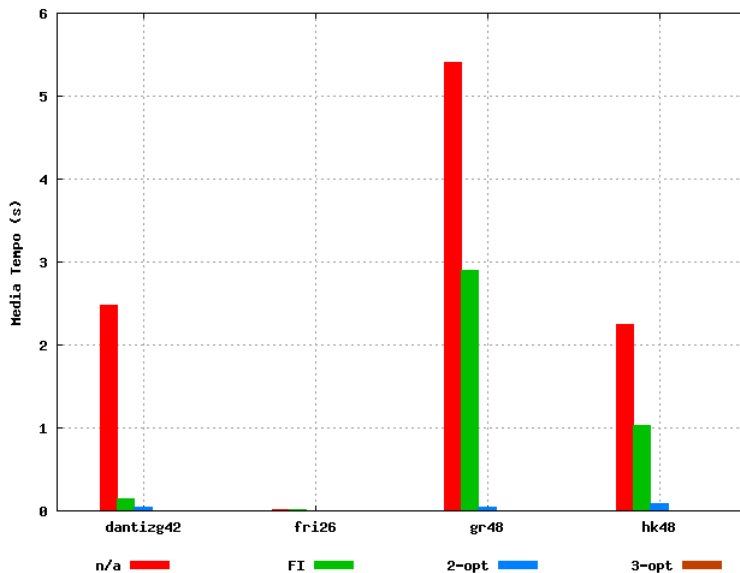
- Nos testes foi utilizado o algoritmo MAX-MIN Ant System com os seguintes parâmetros: $m = 25$, $\alpha = 2$, $\beta = 8$, $\rho = 0.7$
- Buscas locais: *2-opt*, *3-opt*, *First Improvement* e nenhuma busca.
- Para cada base: 10 testes $\times$ 4 Buscas locais = 40 testes

base	Média	σ	Melhor	Pior	Tempo Méd.
dantzig42	699	0	699	699	0.66
fri26	937	0	937	937	0.008
gr48	5051.1	7.4	5046	5069	2.09
hk48	11461	0	11461	11461	0.84

Médias de custo



Médias de tempo



Aplicações

- TSP
- Roteamento em redes de comunicação
- Roteamento de Veículos

Conclusões

- MAX-MIN Ant System se mostrou o melhor dentre os algoritmos analisados.
- A busca local é muito importante para a obtenção de bons resultados com ACO [3].
- Inicialização dos feromônios levando em conta o L_{nn} é importante para a construção de boas soluções.
- O uso de conhecimento adquirido durante a busca para gerar novas soluções é uma idéia promissora para algoritmos baseados na construção de soluções.

Referências



M. Dorigo and L.M. Gambardella.

Ant colony system: A cooperative learning approach to the traveling salesman problem.

IEEE Transactions on Evolutionary Computation, Vol.1, No.1,
pages 1–24, 1997.



M. Dorigo, V. Maniezzo, and Colorni A.

Ant system: Optimization by a colony of cooperatings agents.

IEEE Transactions on Systems, Man, and Cybernetics-Part B,
26(1):1–13, 1996.



M. Dorigo and T. Stützle.

Ant colony optimization metaheuristic: Algorithms,
applications, and advance, 2002.



T. Stützle.

Max-min ant system.

Preprint submitted to Elsevier Science, pages 1–26, 1999.