

Introdução a Inteligência de Enxames

Computação Natural
Gisele L. Pappa

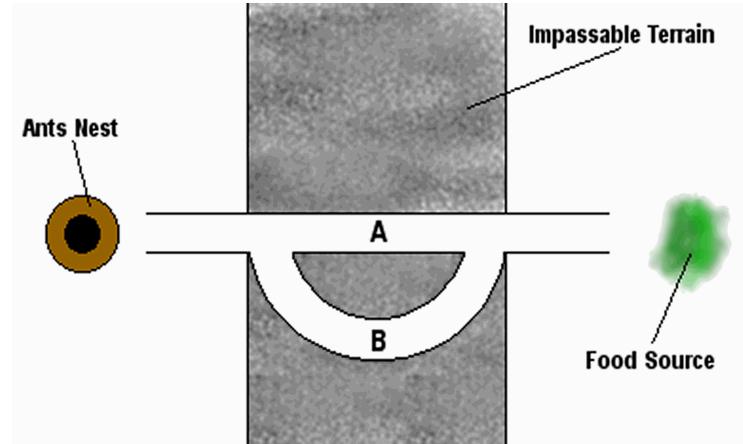
Pense nisso

- Cupins
- Constrõem estruturas extremamente complexas
- Como eles fazem isso ?
 - Caminham na direção do feromônio mais forte
 - Depositam o que carregam onde o cheiro é mais forte
- Quem controla essa construção ?
- Quem planeja ? (existe um plano ?)



Pense nisso ...

- Formigas ...
 - Buscam comida
 - Constroem formigueiros
 - Nenhuma formiga sabe o que está acontecendo
 - Como elas encontram comida?
 - Pense em como nós encontramos comida (esqueçam os supermercados)
 - Por comida, as formigas:
 - Seguem o cheiro mais forte de feromônio



Pense nisso ...

- Rebanhos e bandos de animais, peixes ...
 - Gostam de estar em grupos
 - São atraídos a grupos (pense em você mesmo)
 - Existe uma coordenação impressionante do grupo
 - Fique onde você está!
 - Não me empurre!
 - Não entre no meio!



Introdução

- Todos esses são exemplos de **criaturas sociais**
- Através de interações de todos os tipos elas exibem um tipo de **comportamento emergente**
- O grupo se torna “inteligente”, embora cada um de seus componentes seja desprovido de inteligência.

Comunicação

- Direta
 - Contato, troca de comida, etc
- Estigmergia (indireta)
 - Modificação do ambiente por um agente afeta o comportamento de um outro agente
 - Através de depósitos de feromônio



Algumas definições

- Um enxame (*swarm*) é:
 - Uma coleção de “agentes” que cooperam
- Inteligência é:
 - A habilidade do sistema de se adaptar (Fogel, 1995)
- Assim, inteligência de enxames é:
 - Coleção de indivíduos que cooperam e se adaptam a novas situações

O que é ‘*Swarm Intelligence*’?

Existem dois tipos principais:

1. Inspirados na biologia (**Formigas**, abelhas)

Inteligência de enxames é a propriedade de um sistema de agentes não-inteligentes de capacidades limitadas exibirem inteligência coletivamente

[Bonabeau, Dorigo and Theraulaz]

O que é ‘*Swarm Intelligence*’?

2. Inspirados em comportamentos sociais mais gerais

Inteligência de enxames é uma tentativa de criar algoritmos ou dispositivos para solução de problemas distribuídos inspirado no comportamento coletivo de insetos sociais e outras sociedades de animais

[Kennedy and Eberhart]

Formigas – Um exemplo

- Uma formiga é *burrinha*.. Como o formigueiro se torna inteligente ?
- O segredo de tudo está no **formigueiro**
 - Temos que pensar nele como um organismo
- Nós somos uma enorme coleção de células
 - Uma comunidade
 - Cooperam dentro do corpo humano

Formigueiros

- Um formigueiro são milhares de “tarefas” especializadas
 - Comida, defesa, alimentação, remover os mortos
- Distribuído
- Formigas podem mudar de tarefa
- Armazenamento de comida
- Como elas se tornam inteligentes quando olhamos pro formigueiro ?
 - Auto-organização

Auto Organização

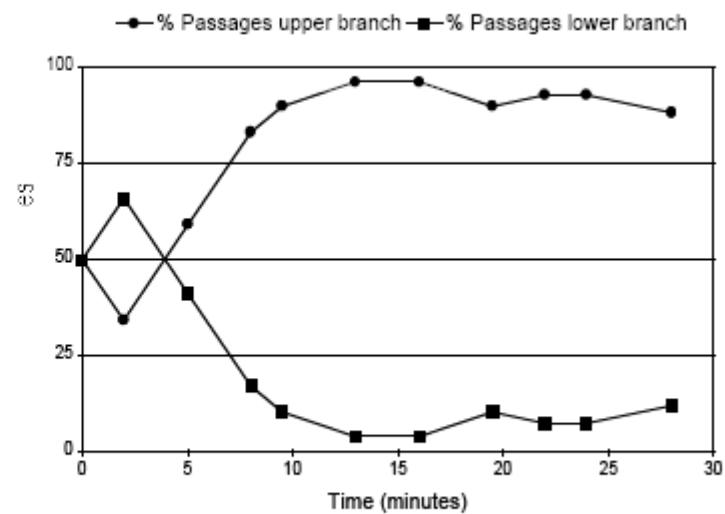
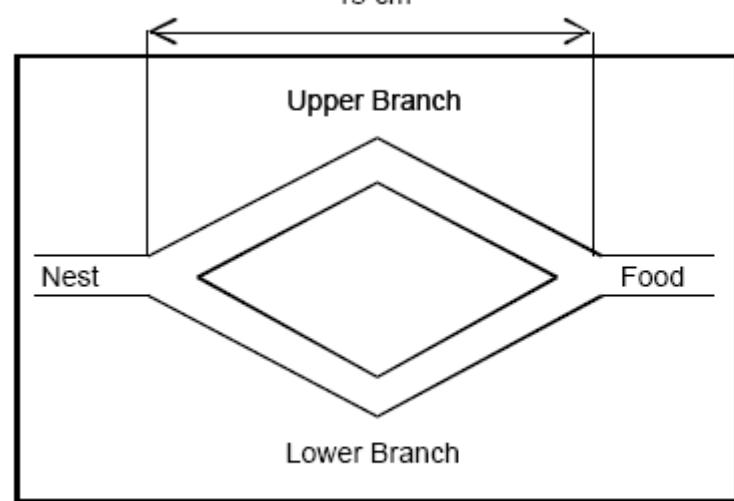
- Esses sistemas podem entrar em equilíbrio, ou tornar-se aleatórios
- Eles estão a beira do *caos*
- Como auto organização aparece em *swarms*?
 - Tomada de decisão coletiva

Auto Organização

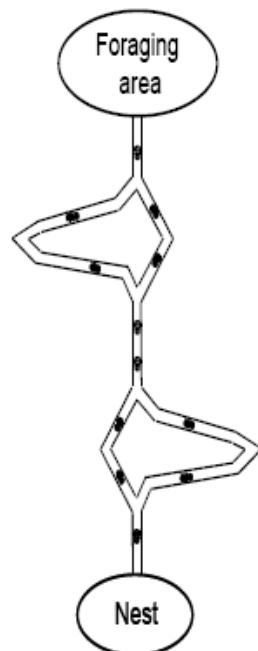
- Quando as formigas buscam comida.. O que acontece ?
- Formigas
 - Tem interações
 - Feedback positivo
 - Feedback negativo
 - Funções amplificadoras
 - No sentido de que um coisa pequena causa um efeito grande
- Como os quatro itens acima aparecem no formigueiro nessa tarefa ?

Otimização com Colônias de Formigas

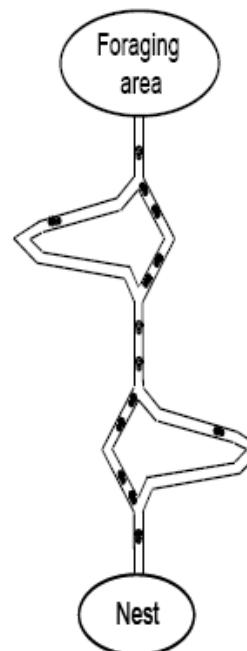
Experimentos Reais – Formigas em Busca de Comida



(a)



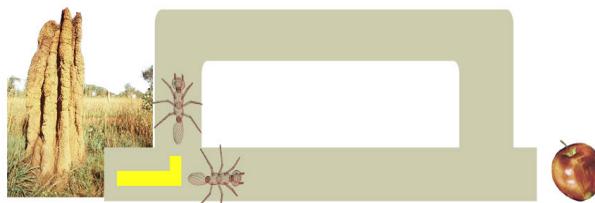
(a)



(b)

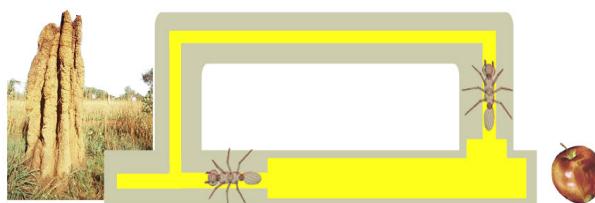
Colônias de Formigas

Inicialmente, cada formiga escolhe
um caminho aleatoriamente



Formigas depositam feromônio enquanto
andam (inicialmente, os 2 caminhos tem
+- a mesma quantidade de feromônio)

- Depois de um período de tempo fixo, a formiga seguindo o caminho mais curto vai depositar mais feromônio no seu caminho (formigas se movem com a mesma velocidade).
- Assim, caminhos mais curtos terão uma maior concentração de feromônio



Com o tempo... toda a
colônia de formigas vai
convergir para o caminho
mais curto

Quanto mais feromônio no caminho, maior a
probabilidade das próximas formigas
escolherem aquele caminho (**feedback positivo**)

Analogia com Computação

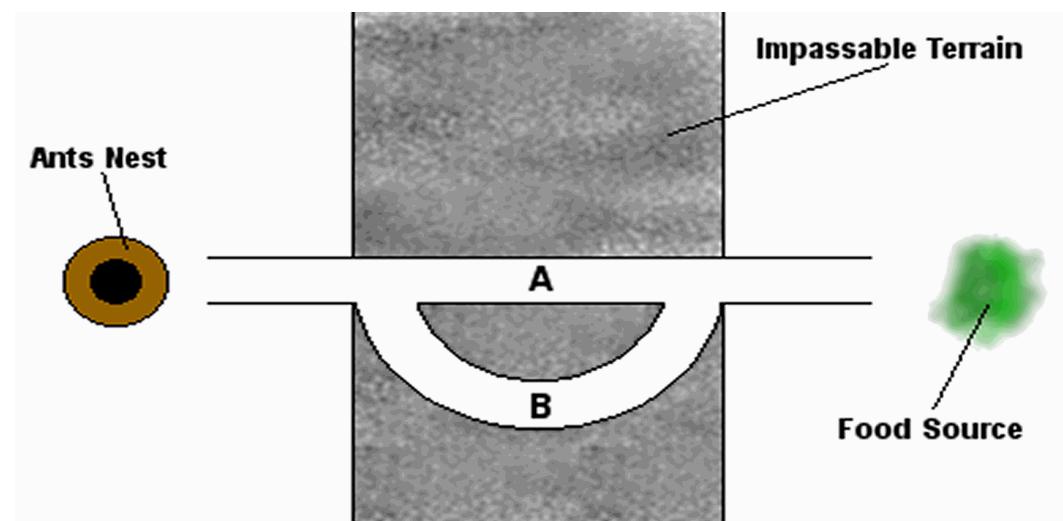
- Cada **caminho** representa uma possível solução
- Quantidade de feromônio depositada é proporcional a **qualidade** da solução
- Se formigas têm que escolher entre dois caminhos, a probabilidade delas seguirem o caminho melhor é maior

Como o sistema pode se tornar caótico?

- Se formigas liberam feromônio aleatoriamente, a trilha não vai se formar
- Se o feromônio liberado é muito forte, um caminho sub-ótimo pode ser encontrado
- Formigas exploram ... A maioria das vezes elas escolhem o melhor caminho ou um outro muito próximo do melhor para a comida

Otimização de Colônias de Formigas

- O comportamento das formigas quando buscam comida é utilizado como inspiração
- Como funciona ?
 1. Inicialmente a formiga “passeia” aleatoriamente
 2. Enquanto passeiam elas liberam feromônio
 3. Formigas seguem feromônio



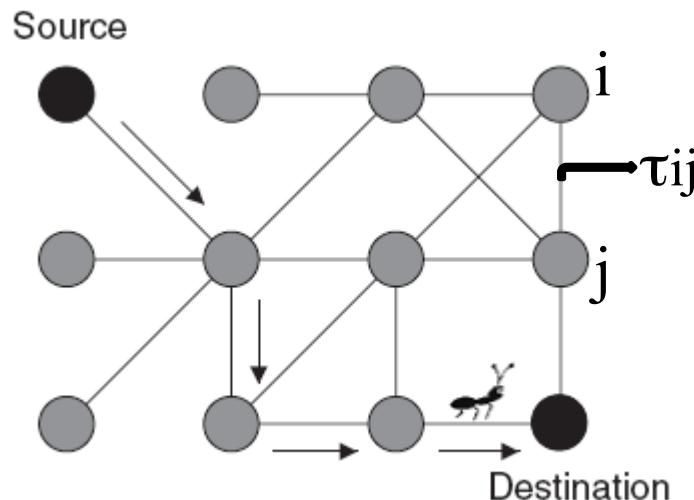
Idéias Básicas

- As soluções não são representadas pelas formigas, e sim pelos **caminhos** percorrido por elas
 - Uma formiga constrói o caminho (solução candidata) incrementalmente, normalmente adicionando um componente de cada vez a solução
- Conforme as formigas se movem no espaço, elas depositam uma quantidade de feromônio proporcional a qualidade da solução
- Quando as formigas têm que escolher entre dois caminhos, a probabilidade delas seguirem o melhor caminho é dada de acordo com a concentração de feromônio

Otimização de Colônia de Formigas

Em sua versão mais simples

- Assuma um grafo conectado, onde buscamos o menor caminho de um ponto A a um ponto B
- Associado a cada aresta do grafo está uma quantidade de feromônio τ
- Cada formiga é capaz de sentir (ler) ou deixar (escrever) feromônio



Otimização de Colônia de Formigas

- Uma formiga escolhe uma aresta para continuar seu caminho de acordo com uma probabilidade
 - Que depende da concentração de feromônio de cada aresta.
- O feromônio em cada aresta é atualizado conforme as formigas caminham
- Cada vez que a formiga caminha, o feromônio na aresta deve ser atualizado
 - Existe uma taxa de evaporação para evitar convergência

Otimização de Colônia de Formigas

- A versão mais geral do ACO inclui também uma **função** (η) (*desirability*) para medir a qualidade de cada componente que pode ser adicionado a uma solução candidata parcial
 - Usa informação **local**, e **dependente** do problema em questão
- Essa função é equivalente a *fitness* em algoritmos evolucionários.

ACO (maxIt , N , τ_0)

Inicializa τ_{ij} (igualmente para cada aresta)

Distribui cada uma das k formigas em um nó selecionado aleatoriamente

$t = 1$

while ($t < \text{maxIt}$) do //número de iterações

 for $i = 0$ to N do //para cada formiga

1. Constrói uma solução aplicando **uma regra de transição probabilística** ($e-1$) vezes // e é o número de arestas do grafo
2. **Avalia o custo** de cada solução construída
3. Atualiza melhor solução

 end for

 Atualiza as trilhas de feromônio

$t = t + 1$

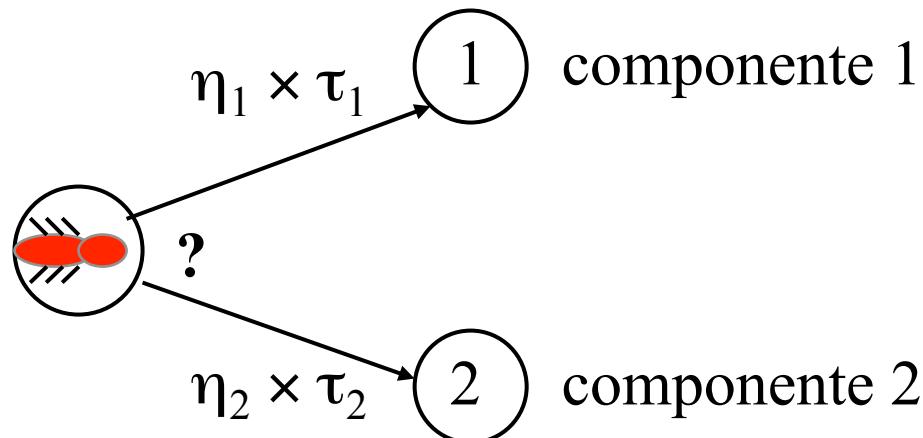
end while

Principais elementos de um algoritmo de ACO

- Representação do problema
 - Especificar os elementos que as formigas usarão para construir **incrementalmente** a solução para um problema
 - Garantir a construção de soluções válidas

Principais elementos de um algoritmo de ACO

- **Regra de transição probabilística:** baseada no valor da função η e na quantidade de feromônio τ associada a cada componente de uma solução candidata
 - Essa regra decide qual componente será inserido na solução parcial
 - Normalmente, a probabilidade de escolher um componente i é proporcional ao produto $\eta_i \times \tau_i$



Principais elementos de um algoritmo de ACO

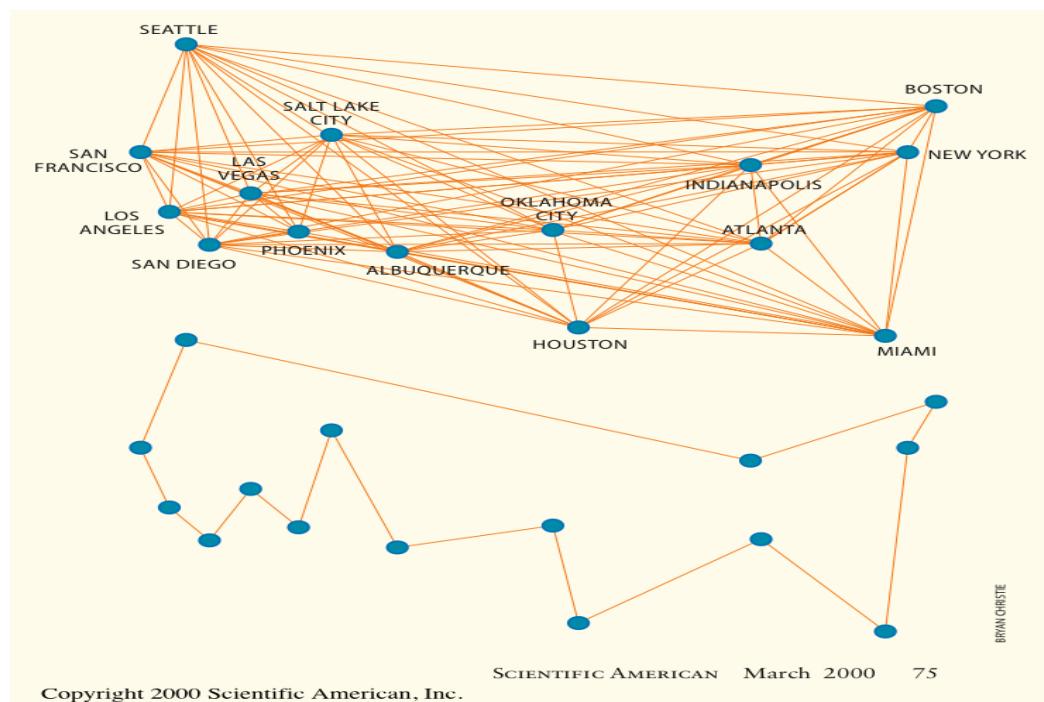
- **Regra para atualização do feromônio:** especifica como atualizar a quantidade de feromônio (τ) associada a cada componente inserido no caminho seguido por uma formiga
 - Feromônio aumenta proporcionalmente a qualidade do caminho (solução)
 - Usa um mecanismo **global**, e uma estratégia de adaptação **independente** do problema

$$\tau_{ij}(t+1) \leftarrow (1-\rho)\tau_{ij}(t) + \Delta\tau,$$

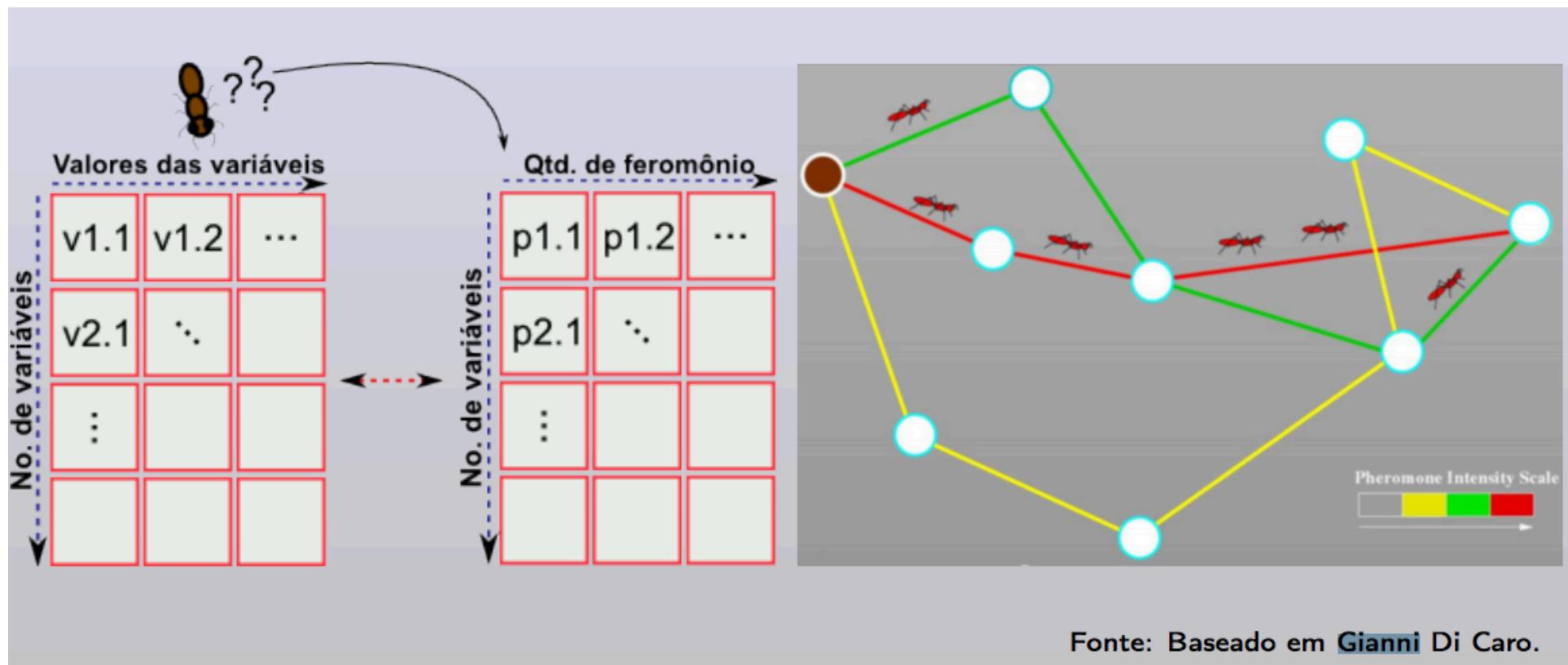
onde $\rho \in (0,1]$ é a taxa de evaporação de feromônio.

ACO para o problema do Caixeiro Viajante

- Assumimos um grafo totalmente conectado (existe uma aresta entre cada par de cidade (i,j))
- Feromônio é associado com arestas
 τ_{ij} corresponde ao feromônneo deixado quando a formiga caminha da cidade i para a cidade j



ACO para o problema do Caxeiro Viajante



ACO para o problema do Caxeiro Viajante

- Para cada formiga:
 - Escolhemos uma cidade s de início
 - Utilizamos valores de feromônio e da função para construir incrementalmente um caminho (adicionando uma cidade de cada vez ao caminho atual), considerando que:
 - A probabilidade da formiga ir da cidade i para a cidade j é proporcional a quantidade de feromônio τ_{ij} na aresta (i,j) e ao valor de η_{ij}
 - A aresta (i,j) só pode ser adicionada ao caminho se a cidade j ainda não foi visitada
 - O caminho é construído até que todas as cidades tenham sido visitadas, e no último passo uma aresta da cidade final para a inicial é inserida

ACO para o problema do Caxeiro Viajante

- A função η_{ij} incorpora informações específicas do problema
 - Calculada de maneira local, baseada apenas no **custo da aresta (i,j)**
 - No problema do caxeiro, utilizamos: $\eta_{ij} = 1 / d_{ij}$
- A quantidade de feromônio τ_{ij} incorpora o resultado da aprendizagem (adaptação) da colônia como um todo, tentando vários caminhos
 - O valor é atualizado a cada iteração, baseado no custo (global) do caminho

ACO para o problema do Caxeiro Viajante

- Uma formiga na cidade i se move para a cidade ainda não visitada j com probabilidade:

$$p_{ij} = \frac{(\tau_{ij})^\alpha (\eta_{ij})^\beta}{\sum_{k \in N(i)} (\tau_{ik})^\alpha (\eta_{ik})^\beta}$$

α, β são pesos, e.g. $\alpha = 1, \beta = 2$
 $N(i)$ é o conjunto de vizinhos elegíveis de i
(i.e., nós que ainda não foram visitados)

O denominador é um fator de normalização, para que: $0 \leq p_{ij} \leq 1$

- Essa fórmula representa a **regra de transição proporcional aleatória**

Atualização de Feromônio

- Feromônio é atualizado depois que todas as formigas constrõem um caminho
- 1. Aplica-se uma regra de evaporação de feromônio, diminuindo seu valor em cada nó por uma constante:

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij}, \forall \text{aresta}(i,j) \in E,$$

onde ρ é a taxa de evaporação de feromônio, $0 < \rho \leq 1$

- 2. Todas as formigas depositam uma quantidade $\Delta\tau_{ij}$ de feromônio nas arestas (i,j) pelas quais elas passaram durante seu caminho, ou seja, para cada formiga

$$\tau_{ij} = \tau_{ij} + \Delta\tau_{ij}, \text{ onde } \Delta\tau_{ij} = \begin{cases} 1/C & \text{se a formiga passou pela aresta } (i,j) \\ 0 & \text{cc} \end{cases}$$

onde C é o custo total do caminho

Variações do método de atualização do Feromônio

- Inicialmente, 2 versões eram muito utilizadas:
 1. Atualizar o feromônio logo após a chegada a uma nova cidade
 - A quantidade de feromônio depositada por uma formiga é inversamente proporcional ao tamanho da aresta que acabou de ser visitada
 2. Atualizar o feromônio apenas depois que uma formiga construiu o caminho completo
 - A quantidade de feromônio depositada por uma formiga é proporcional a qualidade do tour

Variações do método de atualização do feromônio

- *Elitista* – adiciona mais feromônio às arestas do melhor caminho encontrado até o momento (desde a primeira iteração)
- *Baseado em Rank*– a quantidade de feromônio depositada por cada formiga diminui de acordo com sua posição no rank;
- ***Max-Min (abordagem mais popular hoje) :***
 - Apenas uma formiga atualiza o feromônio a cada iteração: ou a melhor formiga da iteração ou a melhor desde o princípio
 - Porém, nesse caso os valores de feromônio em cada aresta são limitados entre $[\tau_{\min}, \tau_{\max}]$, evitando que uma aresta se torne muito forte ou muita fraca

Parâmetros

- Número máximo de iterações
- Número de formigas
 - Normalmente igual ao número de arestas do grafo
- Taxa de feromônio inicial τ_0
- Pesos para concentração de feromônio (α) e qualidade da função (β) quando calculando a probabilidade de uma formiga escolher um caminho ou outro
- Taxa de evaporação

Leitura Recomendada

- E. Bonabeau and G. Theraulaz. Swarm Ants. Scientific American. 2000.
- **Ant Algorithms for Discrete Optimization**
Marco Dorigo, Gianni Di Caro and Luca M. Gambardella, Artificial Life, 5, 1999, 137-172
- **The Ant System: Optimization by a colony of cooperating agents**, IEEE Transactions on Systems, Man, and Cybernetics–Part B, Vol. 26, No.1, 1996, pp.1-13, Marco Dorigo, Vittorio Maniezzo and Alberto Colorni