

UMA APLICAÇÃO DOS CONCEITOS DE AUTO-ORGANIZAÇÃO PARA DESENVOLVIMENTO DE ALGORITMOS: A META-HEURÍSTICA DE COLÔNIA ARTIFICIAL DE ABELHAS

Luiz Andrade

Resumo

Este trabalho detalha as origens do Algoritmo de Colônia Artificial de Abelhas, uma meta-heurística inspirada em sociedades desses insetos que é utilizado como base para o desenvolvimento de algoritmos de solução para problemas computacionalmente complexos. São apresentados os conceitos básicos do algoritmo e duas aplicações práticas, sendo uma delas aplicada a um sistema de ambulâncias no Município de São Paulo.

1. Sobre o desenvolvimento de meta-heurísticas

Muitos problemas encontrados em situações práticas do estudo da logística recaem em modelos matemáticos com complexidade não polinomial. Isso significa que conforme o tamanho desses problemas cresce, o custo computacional de tempo de processamento cresce de maneira exponencial, tornando a solução exata de instâncias reais inviável por meios clássicos de otimização. Os problemas com esse tipo de característica foram estudados por Garey e Johnson (1979), numa publicação que trata sobre a complexidade de algoritmos e sua inviabilidade computacional. Nessa publicação, os autores demonstram que para certas classes de modelos matemáticos, mesmo computadores imaginários com capacidade de processamento e memória inigualáveis, levariam anos para encontrar a solução. Exemplos de problemas listados pelos autores como sendo computacionalmente intratáveis são problemas de cobertura ou de particionamento. Estes possuem natureza altamente combinatória, ou seja, existem muitas combinações das variáveis de decisão que devem ser testadas.

Essa situação apontada por Garey e Johnson (1979) sugere a utilização de algoritmos numéricos que são capazes de encontrar boas soluções aproximadas em tempos de processamento viáveis. Esses algoritmos de aproximação, segundo Talbi (2009), podem ser separados em heurísticas e meta-heurísticas. As heurísticas são algoritmos desenvolvidos para problemas particulares e, assim sendo, têm sua aplicabilidade restrita a esses problemas específicos que originaram o seu desenvolvimento. As meta-heurísticas, por sua vez, representam uma classe mais abrangente e

genérica de algoritmos de aproximação e são adaptáveis a uma diversidade muito grande de problemas.

O desenvolvimento dessas meta-heurísticas, nos últimos anos, acompanhou uma crescente e rápida evolução da capacidade de processamento dos computadores, e veio acompanhada também de um aumento considerável da complexidade dos modelos matemáticos formulados para diferentes tipos de problemas. Pode-se afirmar que essa classe particular de algoritmos compõe um ramo interdisciplinar da matemática aplicada e da ciência da computação. Existem inúmeros algoritmos que se enquadram na categoria de meta-heurísticas. Uma referência bastante completa pode ser encontrada em Luke (2010), em que uma lista extensa desses algoritmos é apresentada.

As meta-heurísticas podem ser utilizadas para a solução de praticamente qualquer problema matemático considerado complexo o bastante para ser intratável do ponto de vista de busca de soluções exatas. Esses algoritmos de aproximação possuem a característica de explorar o espaço de busca das soluções dos problemas de maneira eficiente, restringindo a busca a regiões promissoras desse espaço; isso é o que torna essa abordagem aproximativa atraente para problemas complexos. De acordo com Talbi (2009), as meta-heurísticas são utilizadas com três propósitos: resolver problemas de maneira mais rápida; resolver problemas grandes e complexos; e obter algoritmos robustos para a solução de problemas.

Dentre os algoritmos classificados como meta-heurísticas, existe uma subclasse composta por algoritmos de *Swarm Optimization* ou *Swarm Intelligence*, que podem ser definidos como algoritmos que buscam, de alguma forma, mimetizar o comportamento de sociedades ou conjuntos de organismos, agrupados com o objetivo de vencer tarefas complexas, tarefas praticamente impossíveis de serem vencidas por indivíduos dessas sociedades. Alguns exemplos dessas sociedades que inspiram esses algoritmos são: colônias de insetos, conjuntos de pássaros, colônias de bactérias, etc. Esses algoritmos se inspiram na inteligência coletiva dessas sociedades que realizam tarefas complexas no mundo natural, adaptando esses comportamentos para a ciência de desenvolvimento de algoritmos.

Essas comunidades possuem algumas características interessantes do ponto de vista computacional. A principal dessas características é o fato de essas organizações realizarem diariamente diversas tarefas complexas sem a existência de uma inteligência centralizada, e com

um comportamento auto-organizado. Essa característica de auto-organização e controle descentralizado adere bem aos mecanismos estocásticos de algoritmos de programação evolucionária, que são algoritmos populacionais de otimização que se utilizam de processos randômicos para busca do espaço de solução, se assemelhando a um processo de seleção natural.

Segundo Talbi (2009), outro ponto de interesse dessas sociedades que inspiram o desenvolvimento de algoritmos, é o fato de as partículas mais simples dos sistemas, os agentes, serem compostos de organismos muito simples que cooperam de alguma forma por algum mecanismo direto ou indireto de comunicação.

Todos os algoritmos de *Swarm Optimization*, pela sua própria definição, são algoritmos populacionais, no sentido em que trabalham com uma população de soluções para resolver um problema particular. Além disso, existe sempre a consideração de uma parcela de aleatoriedade no desenvolvimento desses algoritmos, que adere aos princípios da auto-organização e controle descentralizado. Existem muitos algoritmos nessa categoria sendo que alguns dos principais são: *Particle Swarm Optimization*, *Ant Colony Optimization*, *Bee colony Optimization*, etc. Para maiores referências, pode-se consultar Panigrahi *et al.* (2011), Luke (2010), Talbi (2009), e Kennedy e Eberhart (2001).

As colônias de insetos são temas recorrentes no estudo de algoritmos de Swarm Intelligence, e especificamente as colônias de abelhas, que são organismos extremamente bem adaptados a diversos tipos de ambientes e que trabalham de maneira eficiente, apesar da complexidade de sua colônia. Um algoritmo que foi desenvolvido inspirado no comportamento das abelhas na procura de alimentos é o *Artificial Bee Colony Algorithm* (ABC) apresentado pela primeira vez por Karaboga (2005).

2. A meta-heurística de Colônia Artificial de Abelhas

O algoritmo ABC em sua forma clássica, desenvolvido por Karaboga (2005), é uma meta-heurística para a otimização de funções contínuas, multidimensionais e multimodais (GAO; LIU, 2011). Trata-se de uma analogia com o comportamento de busca de alimentos de abelhas melíferas.

Organizados em colônias, estes insetos possuem três tipos de comportamento na busca de alimentos (KARABOGA, 2005): trabalhadoras, exploradoras e oportunistas. As abelhas

trabalhadoras são aquelas que efetivamente estão alocadas em alguma fonte de néctar próxima da colmeia; essas abelhas realizam viagens à colônia levando néctar colhido e informações acerca da quantidade de néctar da fonte onde ela está alocada. Essa informação é passada para as outras abelhas em um local chamado área de dança, onde as abelhas trabalhadoras realizam movimentos que transmitem a informação sobre a proximidade e quantidade de néctar da fonte a que estão alocadas.

Na área de dança, as abelhas oportunistas assistem à dança das abelhas trabalhadoras e tomam uma decisão sobre qual fonte desejam visitar naquele instante. A decisão das abelhas oportunistas é tomada em função da proximidade e da quantidade de néctar das fontes alocadas às abelhas trabalhadoras. Cada abelha oportunista toma uma decisão e uma vez escolhida uma fonte, elas a visitam e retornam à colmeia com mais néctar, onde aguardam a volta de outras abelhas trabalhadoras para repetirem o processo de escolha.

As abelhas exploradoras são aquelas que realizam buscas randômicas nos arredores da colmeia para encontrarem novas fontes de néctar. Quando uma abelha trabalhadora esgota a sua fonte de néctar ela se torna uma abelha exploradora, realiza uma busca randômica ao redor da colmeia e se aloca a uma nova fonte.

Dessa forma, segundo Karaboga e Akay (2009), as colônias de abelhas, através da interação entre oportunistas e trabalhadoras desenvolvem uma inteligência coletiva que otimiza a sua busca de alimentos, visto que mais abelhas oportunistas irão para as fontes de alimento com mais néctar, intensificando o processo de busca de fontes promissoras; ao mesmo tempo em que as abelhas exploradoras garantem uma diversidade de locais para a busca de alimento. A Figura 1 ilustra esse comportamento.

Do ponto de vista matemático, é feita uma analogia entre o espaço ao redor da colmeia e o espaço de busca relacionado com o problema que o algoritmo resolve, além disso, trabalha-se com uma população $P=\{1,2,3,...,2N\}$ de abelhas, sendo que cada abelha possui uma solução alocada a ela em cada iteração do algoritmo. Estas soluções são análogas às fontes de néctar do caso natural.

Inicialmente são consideradas N abelhas exploradoras e N abelhas trabalhadoras, contudo essa distribuição pode variar. Cada abelha j é representada por um vetor $X_j=\{x_{1j}, x_{2j},...,x_{nj}\}$ em que cada componente x_{ij} representa o valor da variável de decisão ou parâmetro de otimização x_i para

a solução alocada à abelha j . Assim como em outros algoritmos populacionais, a aptidão (*fitness*) de cada solução alocada a um indivíduo é mensurada de acordo com a função objetivo e com as restrições do problema que será resolvido pelo ABC.

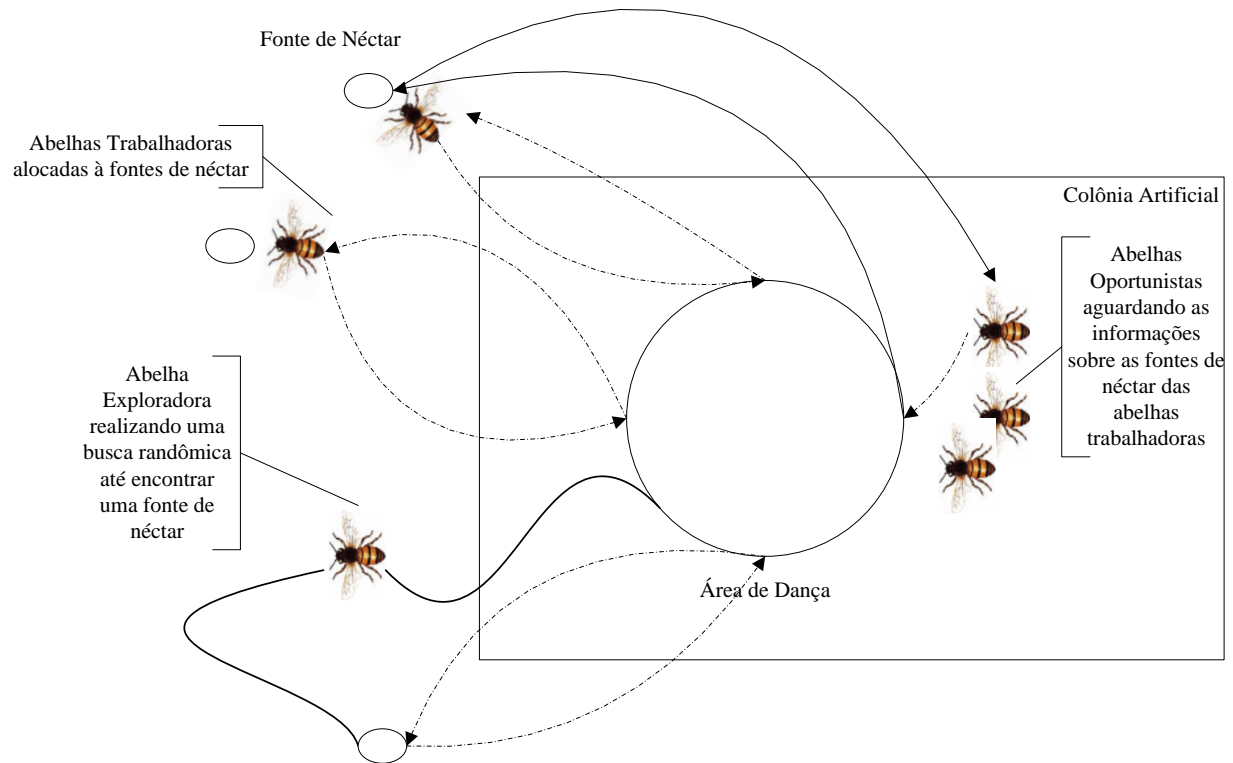


Figura 1 - Ilustração do método de busca de alimentos de colônias de abelhas melíferas

Fonte: Próprio autor

O comportamento algorítmico dos três agentes da meta-heurística, as abelhas trabalhadoras, as exploradoras e as oportunistas, é descrito da seguinte forma:

- As abelhas exploradoras realizam buscas randômicas pelo espaço de busca, sendo que ao final dessa busca, uma solução com certo grau de aleatoriedade criada estará alocada a essa abelha e o seu status passará ao de abelha trabalhadora, pois ela estará explorando uma solução específica.
- As abelhas trabalhadoras realizam buscas locais nas soluções alocadas a elas e realizam viagens periódicas à colmeia, levando informações sobre a sua fonte para a área de dança. As abelhas utilizam um mecanismo guloso de seleção, isto é, se a aptidão da solução resultante da busca local for melhor que a solução atualmente alocada, a abelha se aloca a essa

nova solução, caso contrário, a abelha continua com a solução antiga. Esse processo é realizado até que se tenham atingido um número pré-definido *limit* de buscas sem melhoria de solução, neste caso, a abelha trabalhadora muda de status e se transforma numa abelha exploradora que irá realizar uma busca randômica procurando outra fonte de néctar para se alocar. Na área de dança, ocorre uma interação entre as abelhas trabalhadoras que transmitem informações sobre a quantidade de néctar contido em suas fontes, e entre as abelhas oportunistas que utilizam essas informações para tomarem decisões.

- As abelhas oportunistas aguardam as informações sobre as fontes exploradas pelas abelhas trabalhadoras e por meio de um sorteio, selecionam uma fonte para ser explorada. A abelha então se desloca para essa fonte sorteada e realiza uma busca local na fonte de néctar. Essas abelhas utilizam o mesmo mecanismo guloso das abelhas trabalhadoras.

O algoritmo, na forma como foi apresentado por Karaboga e Akay (2009) utiliza mecanismos simples de busca local, sendo essa busca idêntica para os diferentes tipos de abelha. Considerando a solução X_j armazenada na abelha j , na busca local para gerar uma solução alternativa V_j , aplica-se a expressão (1).

$$v_{ij} = x_{ij} + \delta_{ij} \cdot (x_{ij} - x_{ik}) \quad (1)$$

Tal que $k=\{1,2,...,2N\}$ e $i=\{1,2,...,D\}$, são índices randomicamente selecionados, D é a dimensão do problema, e δ_{ij} é um número aleatório entre -1 e 1. Essa expressão só é válida para problemas com espaço de solução contínuo. Para a geração de soluções iniciais a partir da atuação de abelhas exploradoras, o ABC utiliza a expressão (2) para cada dimensão do problema.

$$x_{ij} = x_{i,min} + \varphi_{ij} \cdot (x_{i,max} - x_{i,min}) \quad (2)$$

Sendo $x_{i,min}$ o valor mínimo da variável x_i , $x_{i,max}$ o valor máximo da variável x_i , e φ_{ij} um número aleatório entre zero e um. Essa expressão só é válida em espaços de busca contínuos. As expressões (2.244) e (2.245) fazem parte do ABC clássico, sendo que outras estratégias de busca local e geração de soluções iniciais foram propostas e implementadas por outros trabalhos, especialmente para a otimização de problemas combinatórios com restrições.

Os autores citam que um dos pontos positivos do algoritmo é a necessidade de pouca parametrização do próprio algoritmo, sendo que os parâmetros necessários são basicamente o tamanho ($2N$) da população de soluções e sua distribuição, o número (MCN) de ciclos do algoritmo e o valor do parâmetro (*limit*).

O mecanismo de sorteio das abelhas oportunistas utilizado pelos autores considera para cada solução alocada a uma abelha trabalhadora, uma probabilidade de ser sorteada igual a sua aptidão dividida pela soma das aptidões das outras soluções de abelhas trabalhadoras, de acordo com a expressão (3).

$$P(X_j) = \frac{fitness(X_j)}{\sum_{i \in T} fitness(X_i)} \quad (3)$$

Onde $P(X_j)$ é a probabilidade da solução j ser selecionada no sorteio, $fitness(X_i)$ é a aptidão da solução i , e T é o conjunto de abelhas trabalhadoras. A probabilidade de uma solução ser sorteada pelas abelhas oportunistas é tão maior quanto for a sua aptidão; isso promove uma intensificação da busca de regiões promissoras do espaço de solução. Ao mesmo tempo, a atuação de abelhas exploradoras garante uma diversificação das soluções exploradas. Esse balanço entre diversificação e intensificação é o que resulta numa busca eficiente do espaço de soluções, característica útil das meta-heurísticas.

De uma maneira geral, considerando T como o conjunto de abelhas trabalhadoras, E o conjunto de abelhas exploradoras, O o conjunto de abelhas oportunistas; $cont_j$ um contador interno de cada abelha trabalhadora e it como a iteração corrente do algoritmo, o pseudocódigo que descreve o funcionamento do algoritmo é apresentado na Figura 2. Na Figura 3 um fluxograma ilustra o funcionamento do algoritmo ABC.

3. Desenvolvimento de Algoritmo para o *Capacitated Clustering Problem*

Andrade e Cunha (2011) desenvolveram uma heurística baseada no algoritmo ABC para um problema de clusterização (ou agrupamento) capacitado (CCP), também conhecido como o problema “p-medianas capacitado” (SCHEUERER; WENDOLSKY, 2006). Basicamente o problema pode ser definido da seguinte forma: dado um conjunto de n clientes e suas respectivas

demandas, particionar esses pontos em p conjuntos ou agrupamentos, chamados clusters, de tal modo que a capacidade de cada um dos p conjuntos não seja ultrapassada e a soma das distâncias

```

0  Algoritmo ABC (T conjunto de abelhas trabalhadoras; E
   conjunto de abelhas exploradoras; O conjunto de abelhas
   oportunistas)
1      Inicia População:  $T = \emptyset$ ;  $E = \{1, 2, \dots, N\}$ ;  $O =$ 
    $\{1, 2, \dots, N\}$ 
2      enquanto  $it \leq MCN$  faça
3          para cada  $X_j \in E$  faça
4              Busca randômica no espaço de solução
5               $E = E - \{X_j\}$ 
6               $T = T + \{X_j\}$ 
7               $cont_j = 0$ 
8          fim
9          para cada  $X_j \in T$  faça
10             Busca local de abelha trabalhadora (solução
alternativa  $X'_j$ )
11             se  $fitnessX_j < fitnessX'_j$  então
12                  $X_j \leftarrow X'_j$ 
13                  $cont_j = 0$ 
14             caso contrário
15                  $cont_j = cont_j + 1$ 
16             fim
17             se  $cont_j \geq limit$  então
18                  $T = T - \{X_j\}$ 
19                  $E = E + \{X_j\}$ 
20             fim
21         fim
22         Avalia aptidão das abelhas trabalhadoras
23         para cada  $X_j \in O$  faça
24             Sorteie uma solução de uma abelha não
oportunista
25             Busca local na solução sorteada (solução
alternativa  $X'_j$ )
26             se  $fitnessX_j < fitnessX'_j$  então
27                  $X_j \leftarrow X'_j$ 
28             fim
29         fim
30         Avalia aptidão de todas as abelhas
31         Guarda solução com maior aptidão:  $X_b$ 
32          $it = it + 1$ 
33     fim
34 fim

```

Figura 2 – Pseudocódigo genérico do algoritmo de colônia artificial de abelhas

Fonte: Próprio autor

entre os pontos e a mediana de seus respectivos clusters seja mínima. Para maiores referências sobre o problema sugere-se consultar, por exemplo, Mulvey e Beck (1984). . Dentre os problemas práticos que podem ser modelados como o CCP pode-se citar a alocação de clientes a veículos em problemas de roteirização (KOSKODIS; POWEL, 1992), ou ainda problemas de localização de armazéns.

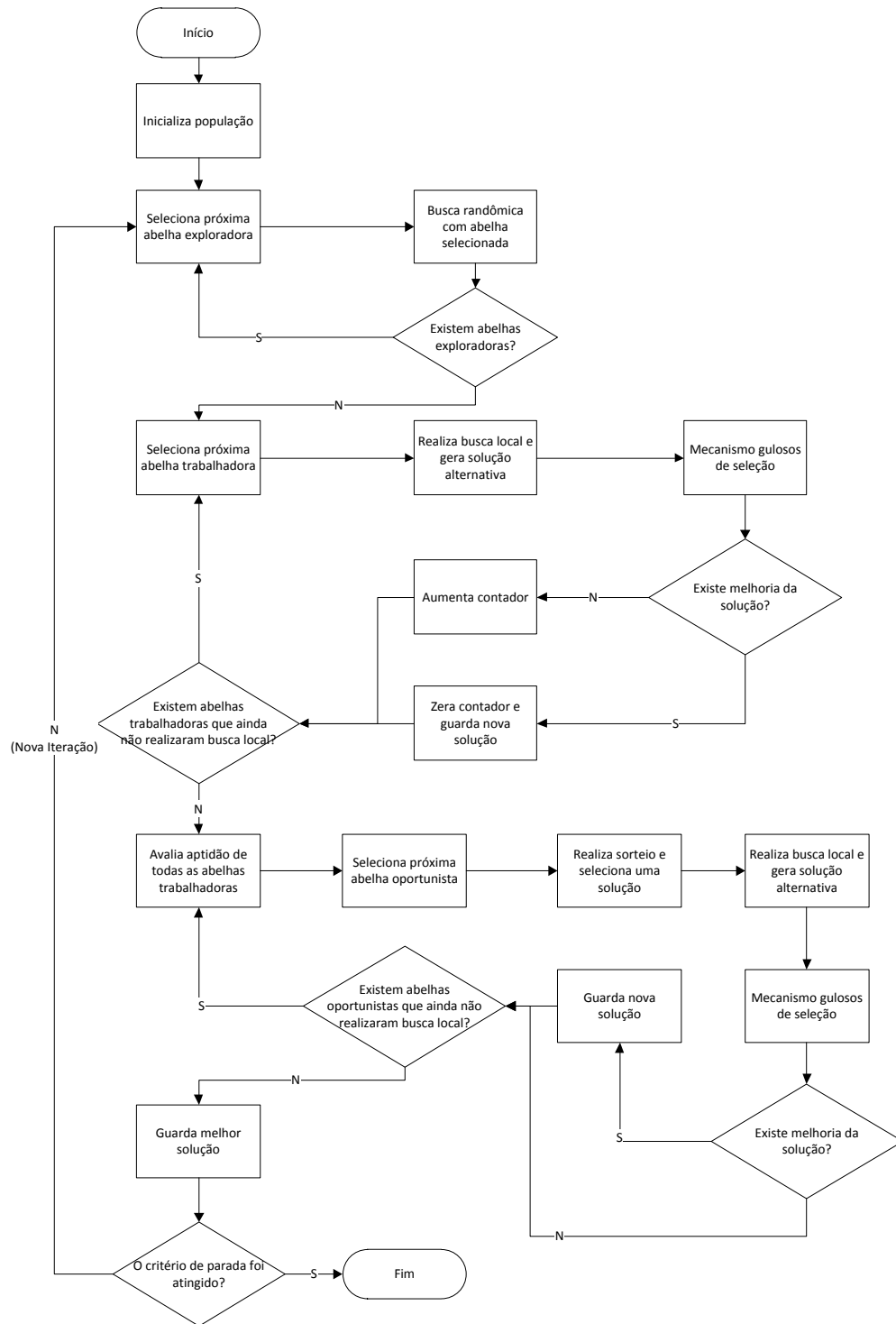


Figura 3 – Fluxograma de funcionamento do algoritmo de colônia artificial de abelhas

Fonte: Próprio autor

O algoritmo proposto por Andrade e Cunha (2011) para o CCP trabalha com uma população $P=\{1,2,3,...,N\}$ de abelhas, sendo que cada abelha possui uma solução alocada a ela em cada iteração do algoritmo. Assim, cada abelha j é representada por um vetor $X_j=\{x_{1j}, x_{2j},...,x_{nj}\}$, que cada componente x_{ij} corresponde para a abelha j , a semente a que o ponto i está alocado, se o ponto i for definido como semente x_{ij} recebe o valor (-1). Cada abelha possui uma lista de sementes independente das sementes das demais. Além disso, cada abelha possui um contador interno, $cont_j$, cujo propósito será explicado adiante, um marcador do tipo da abelha e um marcador do valor da função aptidão. A Figura 4 ilustra a estrutura de cada solução/abelha.

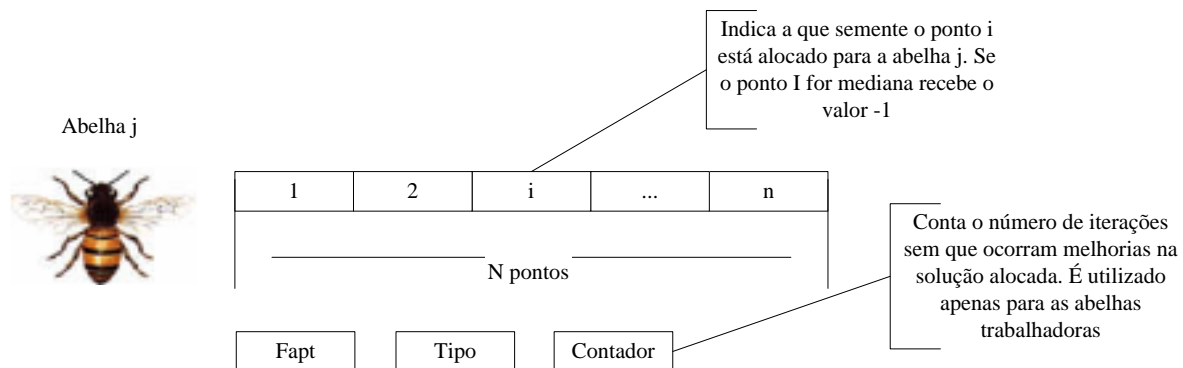


Figura 4 – Representação da estrutura de cada abelha para o CCP

Fonte: Próprio autor

O algoritmo foi testado em seis instâncias de testes: sjc1.dat (100 pontos, 10 medianas), sjc2.dat (200 pontos, 15 medianas), sjc3a.dat (300 pontos, 25 medianas), sjc3b.dat (300 pontos, 30 medianas), sjc4a.dat (402 pontos, 30 medianas) e sjc4b.dat (402 pontos, 40 medianas). Essas instâncias foram propostas por Lorena e Senne (2004) e encontram-se disponíveis em <http://www.lac.inpe.br/~lorena/instancias.html>. Visto que o algoritmo utiliza números aleatórios, para cada instância foram feitos 5 testes. Os resultados quanto aos valores mínimo e médio da função objetivo e quanto aos tempos médios de processamento são apresentadas na Tabela 1.

Na coluna de referência da Tabela 1, são encontrados os resultados obtidos para o mesmo problema por Boccia *et al.* (2008). No seu trabalho, os autores desenvolveram um algoritmo de planos de corte para o problema e constituem os melhores resultados até o momento do teste do algoritmo baseado em colônia de abelhas.

A Figura 5 apresenta uma representação pictórica das soluções encontradas pelo algoritmo ABC para as instâncias sjc2.dat e para a instância sjc4b.dat. O diâmetro dos círculos representa a demanda dos respectivos pontos.

Tabela 1 - Resultados dos testes computacionais

Instância	Solução Inicial		Solução Final		Tempo de Process. até a Melhor Solução (s)	Tempo Total de Process. (s)	Pior Resultado	Referência		
	Mín.	Média	Mín.	Média				Mín. de Referência	Desvio	Tempo de Process. de Referência (s)
sjc1.dat	20792	22270	17437	17450	55	103	17469,4	17289	0,85%	38
sjc2.dat	38644	40284	33271	33315	137	275	33352,2	33271	0,00%	128
sjc3a.dat	53806	54935	45623	45794	356	615	45959,9	45335	0,64%	459
sjc3b.dat	47502	48002	40842	40992	529	712	41054,5	40636	0,51%	72
sjc4a.dat	73402	74644	62653	62872	324	978	63043,1	61926	1,17%	1210
sjc4b.dat	61071	61967	52896	52967	1004	1279	53050,3	52458	0,84%	670

Fonte: Próprio autor

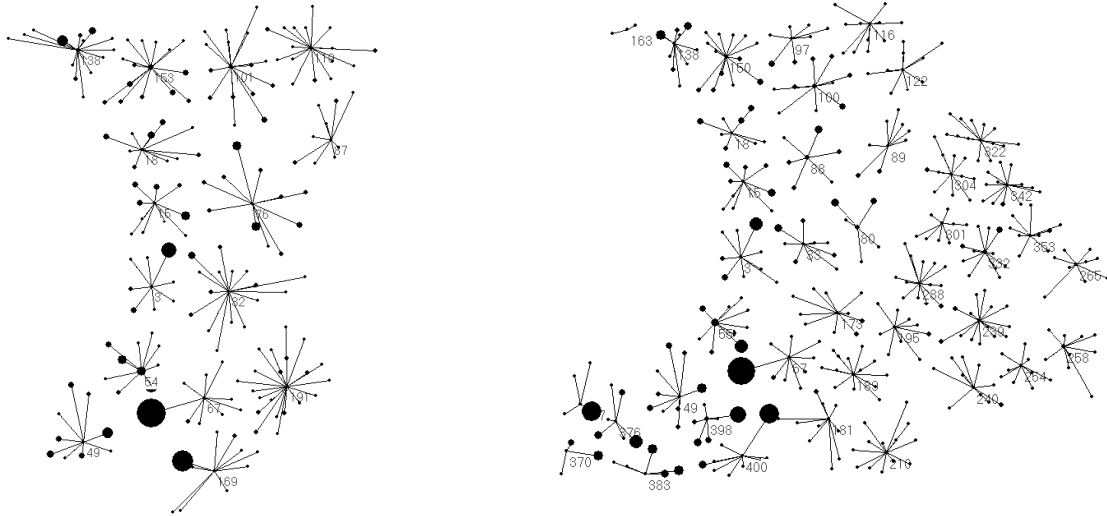


Figura 5 – Representação gráfica da solução encontrada para as instâncias sjc2 e sjc4b.dat

Fonte: Próprio autor

4. Desenvolvimento de Algoritmo para um Problema Prático de Localização de Ambulâncias

Esta seção descreve o desenvolvimento de uma heurística baseada no algoritmo ABC para um problema de distribuição de bases e ambulâncias em uma cidade ao longo do tempo (ANDRADE, 2012). Para a caracterização do problema, deve-se considerar também que: (i) existe uma quantidade finita de bases e ambulâncias de vários tipos; (ii) cada veículo possui um parâmetro de cobertura associado que define, em termos temporais, sua capacidade de cobertura; (iii) são conhecidas as distribuições espaço-temporais das demandas pelos serviços de atendimento de cada tipo de viatura em uma determinada região; (iv) são conhecidos também os

padrões de variação dos tempos de deslocamento nessa região; (v) uma vez que entre períodos subsequentes podem haver realocações de ambulâncias, deseja-se também minimizar o tempo de percurso dessas realocações de acordo com um fator de proporcionalidade; e (vi) a cobertura esperada do sistema é a probabilidade de que um chamado aleatoriamente gerado num ponto da região seja atendido em menos tempo que um parâmetro de cobertura desejado (por exemplo 10 minutos).

Dessa forma, resolver o problema significa encontrar um plano de operação capaz de maximizar, em múltiplos períodos de um horizonte de planejamento, a cobertura esperada do sistema, e ao mesmo tempo capaz de minimizar as realocações de viaturas necessárias entre períodos subsequentes de acordo com um fator de proporcionalidade. Esse plano deve respeitar as restrições: (i) em todos os períodos, todos os pontos de demanda devem ser cobertos por uma viatura de cada tipo; (ii) em todos os períodos, a quantidade de bases e ambulâncias é constante; (iii) em todos os períodos, a quantidade de veículos posicionados em uma base não deve ultrapassar a capacidade de acomodação de viaturas dessa base; (iv) o plano de operação deve ser conexo, ou seja, as realocações resultantes no último período do horizonte de planejamento devem resultar a alocação de viaturas do primeiro período, sendo cíclico o plano completo. Este problema recebe o nome de “Problema de Localização de Bases, Alocação de Ambulâncias em múltiplos períodos e realocação entre períodos subsequentes”, (PLBAA^t).

O PLBAA^t tratado é definido num grafo G não direcionado, com um conjunto de pontos de demanda $i \in V$ e um conjunto de pontos candidatos $j \in W$ a receberem bases e veículos; assume-se que $W \subset V$, o que é verdadeiro na maioria dos casos práticos. Esses pontos constituem uma simplificação da realidade uma vez que representam uma determinada localização geográfica concentrada em um único ponto. A determinação do nível de agregação da demanda que resulta nos pontos i depende da precisão desejada na localização de bases. São considerados também períodos de tempo $t \in \tau = \{0, 1, 2, \dots, t, \dots, T\}$, sendo a soma dos períodos t equivalente ao horizonte de planejamento para o qual serão definidas as localizações de bases e alocações de ambulâncias.

No trabalho desenvolvido por Andrade (2012), o algoritmo desenvolvido, baseado em colônias de abelhas, utiliza como representação de uma solução para o PLBAA^t a estrutura representada pela Figura 6. A estrutura apresentada é composta por três elementos chave: um vetor que

indica o posicionamento das bases e duas matrizes que indicam a quantidade de ambulâncias em cada ponto ao longo dos períodos do horizonte de planejamento.

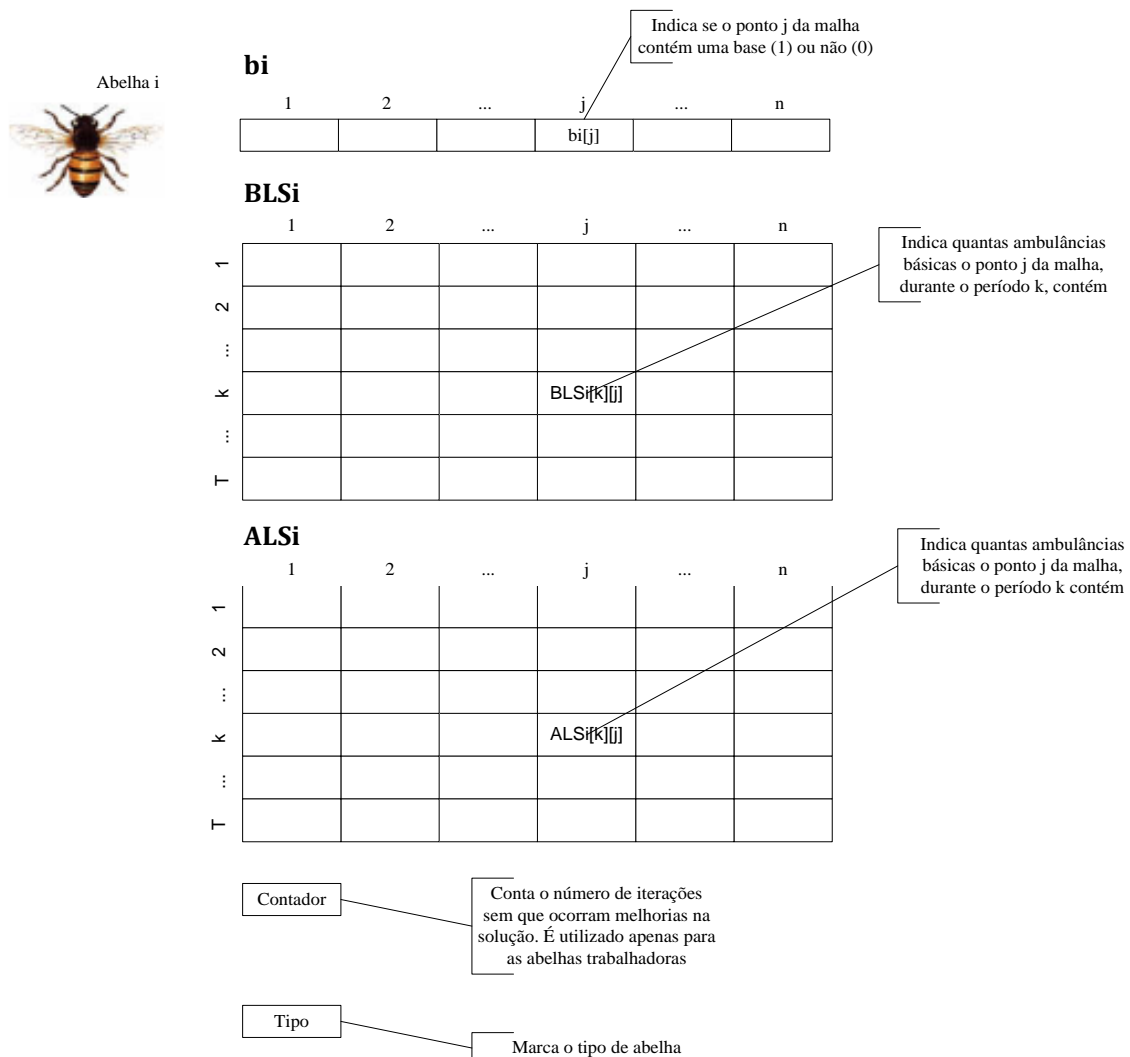


Figura 6 – Representação da estrutura de cada abelha para o PLBAA^t

Fonte: Próprio autor

Foram realizados testes de parametrização e testes de desempenho com o algoritmo desenvolvido por Andrade (2012) em uma série de instâncias de teste do problema. A Figura 7 apresenta o resultado de um dos testes para ilustrar a convergência do algoritmo em relação ao número de iterações (ou ciclos, analogamente). O eixo x do gráfico consiste no número de iterações e o eixo y , a cobertura esperada da abelha mais apta da população do algoritmo; as diferentes curvas representam rodadas diferentes do algoritmo para a mesma instância de teste, variando a semente de geração de números aleatórios.

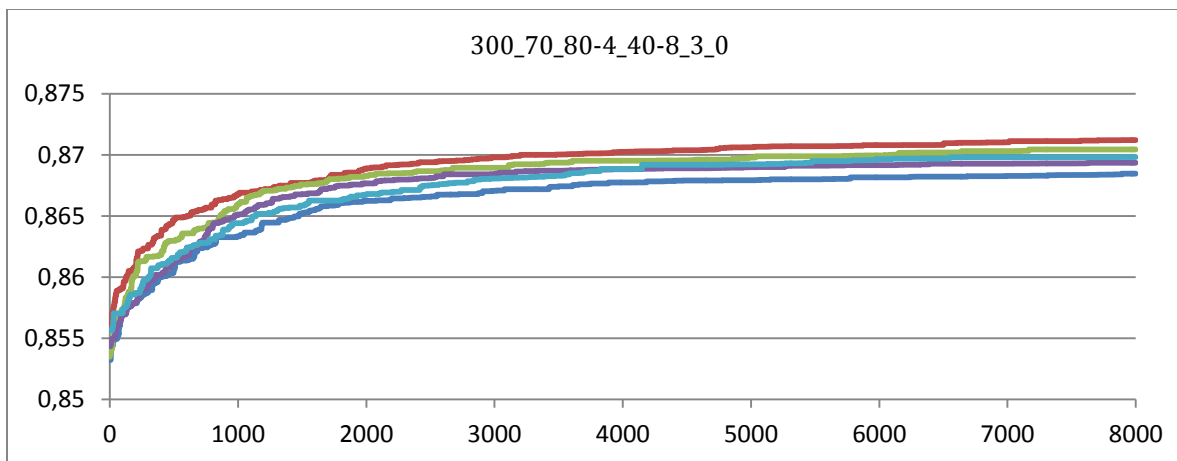


Figura 7 – Representação do resultado de um dos testes computacionais do algoritmo com colônia de abelhas

Fonte: Próprio autor

Para compreender as vantagens computacionais da meta-heurística de colônia de abelhas pode-se utilizar um teste computacional que mede o tempo de processamento para obtenção da solução versus o “ tamanho do problema”. Este tamanho pode assumir muitas definições, no caso do trabalho de Andrade (2012) uma das medidas utilizadas é a quantidade de períodos T considerada no horizonte de planejamento para o qual deseja-se o plano operacional ótimo. A Figura 8 apresenta uma comparação do crescimento do tempo computacional da meta-heurística com o tempo computacional do algoritmo SIMPLEX aplicado com técnicas de planos de corte do espaço de busca, uma técnica tradicional de otimização.

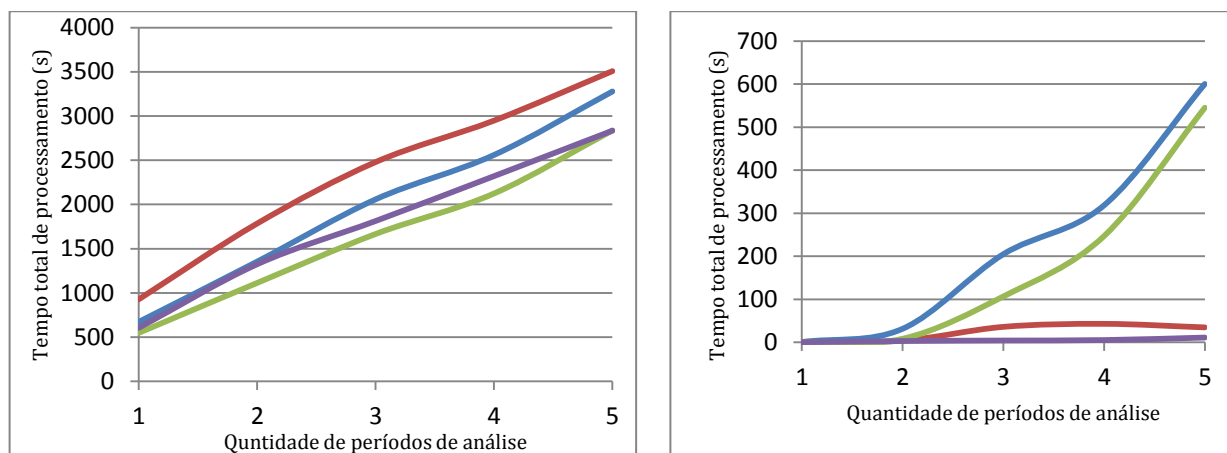


Figura 8 – Comparação entre o desempenho computacional da meta-heurística ABC (à esquerda) e o algoritmo SIMPLEX (à direita)

Fonte: Próprio autor

Pode-se observar que o tempo computacional do algoritmo baseado em colônias de abelhas cresce com uma tendência aproximadamente linear, enquanto o crescimento do tempo computacional apresenta um crescimento aproximadamente exponencial para algumas das instâncias de teste. Isso mostra que, para um determinado “tamanho do problema” métodos tradicionais não resolvem o problema.

Teste de aleatoriedade e convergência foram realizados e foi constatado que o Algoritmo com Colônia de Abelhas é um método robusto para a solução de problemas reais de localização de bases, alocação de ambulâncias e realocação entre períodos subsequentes, no sentido em que resolve problemas de diferentes tamanhos e características, sendo os resultados pouco influenciados pela semente de sorteio de números aleatórios.

O algoritmo foi empregado na análise e melhoria do Sistema de Atendimento Móvel Pré-Hospitalar de Urgência do Município de São Paulo (SAMU SP). O SAMU SP possui 100 bases e 140 viaturas (de dois tipos, básicas e avançadas). O modelo foi aplicado considerando o município de São Paulo representado por uma malha de 96 pontos. Os tempos de transporte entre os pontos e a demanda pelos serviços nos pontos foram definidas para 21 períodos que correspondem a uma semana sendo cada dia dividido em três turnos. A Figura 9 ilustra um dos resultados da aplicação. O gráfico mostra uma relação entre a quantidade de viaturas e a cobertura esperada. Cada curva corresponde a uma quantidade de bases e a um tempo de atendimento. Cada ponto de cada curva corresponde a uma plano operacional de máximo desempenho.

5. Considerações finais

Este trabalho apresenta os conceitos básicos e duas aplicações da meta-heurística de colônia artificial de abelhas. A meta-heurística em si, se mostra como uma ferramenta útil para construção de algoritmo de solução de problemas computacionalmente complexos.

A meta-heurística em si, é uma utilização dos conceitos de diversificação e exploração. Mecanismos randômicos orientam a interação de unidades de processamento que, com regras simples de decisão, constroem uma inteligência coletiva.

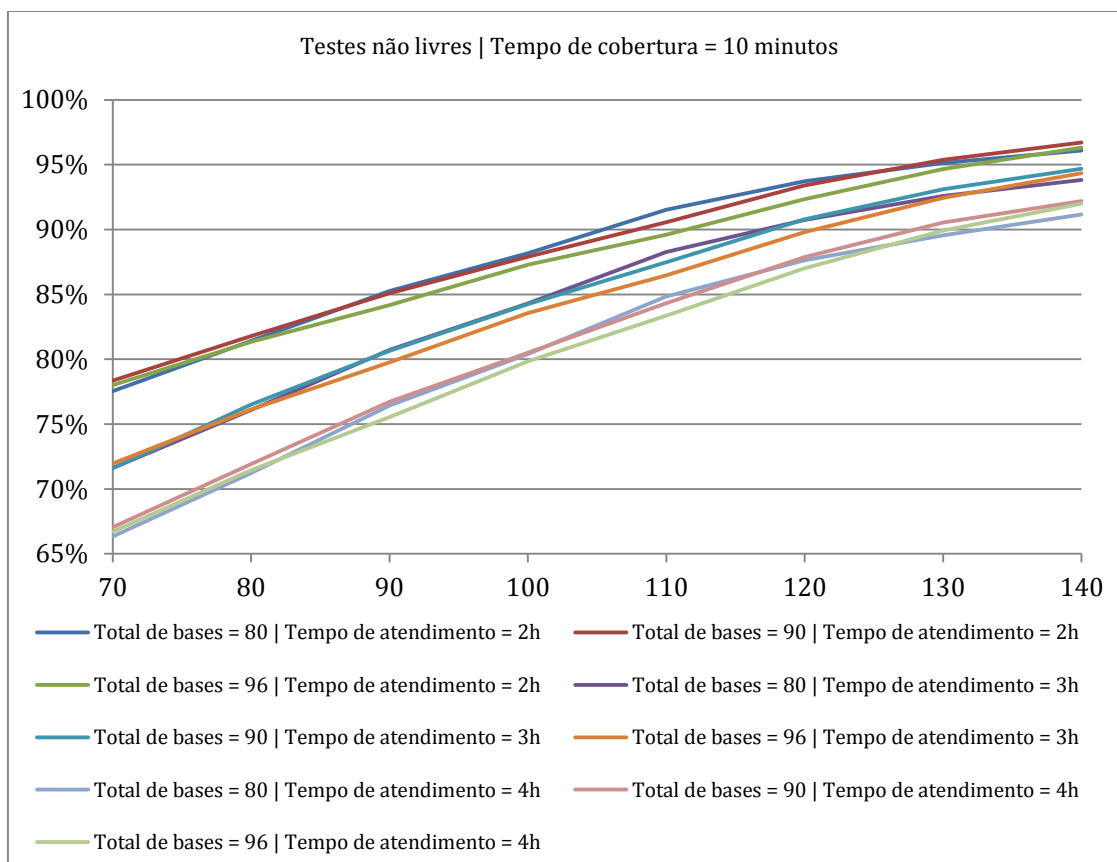


Figura 9 – Exemplo de aplicação do algoritmo baseado em colônia de abelhas para o SAMU SP

Fonte: Próprio autor

O algoritmo apresenta uma vantagens interessantes quanto ao seu desenvolvimento, necessita de poucos parâmetros, ao contrário de outras heurísticas como *Variable Neighborhood Search*, *Busta Tabu* e outras. De uma maneira geral, os resultados se mostraram muito próximos dos melhores resultados encontrados na literatura, o que comprova a eficácia do algoritmo. Os resultados deste trabalho apontam os algoritmos da classe *Swarm Optimization* como um campo promissor no desenvolvimento de algoritmos e heurísticas.

As aplicações práticas apresentadas mostram que esse tipo de abordagem tem seu papel na sociedade como um mecanismo de melhoria e aprimoramento de sistemas reais. Uma das aplicações tratou de uma melhoria do SAMU SP. Cabe ressaltar que após a implementação do algoritmo no processo de planejamento, o tempo de resposta, principal indicador de desempenho do sistema, reduziu de 18 minutos para 10 minutos, sem o acréscimo na quantidade de bases e ambulâncias.

Referências Bibliográficas

- Garey, M.R., Johnson, D.S. Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, San Francisco, 1979.
- Talbi, E.-G. (2009) Frontmatter, in Metaheuristics: From Design to Implementation, John Wiley & Sons, Inc., Hoboken, NJ, USA. doi: 10.1002/9780470496916.fmatter
- Luke, S. Essentials of Metaheuristics. Department of Computer Science, George Mason University, USA. 2010.
- Kennedy, J., Eberhart, R. C. Swarm Intelligence. 1st Edition. Morgan Kaufmann, 2001.
- Panigrahi, B.K., Shi, Y., Lim, M-H. Handbook of Swarm Intelligence (Adaptation, Learning, and Optimization), Vol. 8, Springer-Verlag 2011.
- Karaboga, D., (2005). An idea based on honey bee swarm for numerical optimization. Technical report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department.
- Gao W., Liu S. (2011). Improved artificial bee colony algorithm for global optimization. Information Processing Letters 111 871-882.
- Karaboga, D., Akay, B., (2009). A comparative study of Artificial Bee Colony algorithm. Applied Mathematics and Computation 214, 108-132.
- Scheuerer, S., Wendolsky, R. (2006). A scatter search heuristic for the capacitated clustering problem. European Journal of Operational Research 169(2), 533–547.
- Andrade L, Cunha C. Algoritmo de colônia artificial de abelhas para um problema de clusterização capacitado. Escola Politécnica da USP, Brasil (2011).
- Andrade L, Cunha C. Algoritmo de colônia artificial de abelhas para um problema de clusterização capacitado. In: Simpósio Brasileiro de Pesquisa Operacional, XLIII, 2011, Ubatuba. Anais do XLIII Simpósio Brasileiro de Pesquisa Operacional.
- Mulvey, J.M., Beck, M.P. (1984). Solving capacitated clustering problems. European Journal of Operational Research 18(13), 339-348.
- Lorena, L.A.N., Senne, E.L.F. (2004). A column generation approach to capacitated p-median problem. Computers and Operations Research 31, 863–876.
- Boccia M., Sforza A., Sterle C., Vasilyev I. (2008) A Cut and Branch Approach for the p-Median Problem Based on Fenchel Cutting Planes. Journal of Mathematical Modeling and Algorithms 7:43-58.
- Andrade, L. A. C. G. (2012) Heurística baseada em colônia artificial de abelhas para o problema de localização de bases, alocação e realocação de ambulâncias. 2012. 250p. Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Sistemas Logísticos, São Paulo.
- Koskosidis, Y.A., Powell, W.B. (1992). Clustering algorithms for consolidation of customer orders into vehicle shipments. Transportation Research B 26, 365-379.