

---

# Otimização da localização das bases de ambulâncias e do dimensionamento das suas regiões de cobertura em rodovias

ANA PAULA IANNONI

REINALDO MORABITO

UFSCar

---

## Resumo

Neste artigo combinamos extensões do modelo hipercubo de filas com algoritmos genéticos para otimizar a configuração e operação de sistemas médicos emergenciais em rodovias. Inicialmente apresentamos um método para localizar as bases de ambulâncias ao longo da rodovia, de forma a otimizar as principais medidas de desempenho do sistema. Em seguida estendemos a abordagem para apoiar duas decisões combinadas: a localização das bases de ambulâncias e o dimensionamento das regiões de cobertura de cada base na rodovia. Por exemplo, a abordagem permite determinar os locais para posicionar as bases de ambulâncias e dimensionar os tamanhos das suas áreas de atuação, tais que minimizem o tempo médio de resposta aos usuários e/ou o desbalanceamento das cargas de trabalho das ambulâncias do sistema. Para ilustrar a aplicação dos métodos propostos, analisamos os resultados de dois estudos de caso em rodovias brasileiras.

## Palavras-chave

Localização e despacho de ambulâncias, algoritmo genético, modelo hipercubo, filas espacialmente distribuídas, rodovias.

## *Optimizing the location of ambulance bases and the districting of their covering regions on highways*

### Abstract

*In this paper we combine extensions of the hypercube queueing model with genetic algorithms to optimize the configuration and operation of emergency medical systems on highways. Initially we present a method to locate the ambulance bases along the highway so that the main system performance measures are optimized. Then we extend the approach to support combined decisions: the location of ambulance bases and the districting of the covering regions of each base on the highway. For instance, the approach can search for locals to place the ambulance bases and determine the sizes of their operation areas, such that the mean user response time and/or the ambulance workload imbalance are minimized. To illustrate the application of the proposed methods, we analyze the results of two case studies of Brazilian highways.*

### Key words

*Location and dispatching of ambulances, genetic algorithm, hypercube model, spatially distributed queues, highways.*

## INTRODUÇÃO

Os sistemas de atendimento emergencial (SAEs) em rodovias têm a função de socorrer as vítimas de acidentes nas rodovias e, se necessário, realizar o transporte das mesmas aos hospitais das cidades mais próximas. Estes sistemas em geral não admitem filas de espera e operam com políticas particulares de despacho de ambulâncias, em que somente alguns veículos podem ser despachados para atender a um chamado emergencial em uma região da rodovia (*backup* parcial), devido às restrições de tempo e distância. Além disso, algumas políticas também envolvem múltiplos despachos, isto é, em alguns casos, dependendo do tipo de chamada, é necessário despachar mais de uma ambulância para atender a um acidente.

## A apresentamos uma abordagem para otimizar a configuração e operação de sistemas de atendimento médico emergencial em rodovias.

Nos estudos de Swersey (1994), Owen e Daskin (1998), Chiyoshi *et al.* (2000) e Brotcorne *et al.* (2003), são revisados os principais modelos de localização para analisar SAEs. Em particular, o modelo hipercubo proposto em Larson (1974), baseado em teoria de filas espacialmente distribuídas, tem se mostrado efetivo para analisar SAEs em rodovias. O modelo envolve a solução de um sistema linear de  $O(2^N)$  equações (em que  $N$  é o número de ambulâncias), cujas variáveis envolvidas correspondem às probabilidades de estado do sistema em equilíbrio. Por meio destas probabilidades, podem ser estimadas importantes medidas de desempenho para análise e gerenciamento do sistema. No Brasil, alguns exemplos de aplicação do modelo hipercubo em sistemas de atendimento médico aparecem em alguns SAEs urbanos (COSTA, 2004, TAKEDA *et al.*, 2007) e SAEs em rodovias dos Estados de São Paulo e Rio de Janeiro (MENDONÇA; MORABITO, 2001; IANNONI; MORABITO, 2006c). O modelo hipercubo também vem sendo estudado para aplicação em SAEs que atuam em casos de ataques terroristas e catástrofes naturais de grande escala (LARSON, 2004).

Em particular, em um estudo anterior (IANNONI; MORABITO, 2006c), o modelo hipercubo múltiplo despacho foi estendido para analisar um dos estudos de caso do presente artigo, considerando duas diferentes configurações do sistema: uma com servidores idênticos (configuração aqui analisada) e outra com servidores diferenciados. Os resultados obtidos com as extensões do modelo hipercubo foram comparados com os resultados da análise dos dados reais do estudo de caso. No entanto, diferentemente do presente tra-

balho, aquele estudo não utilizou um método de otimização para prescrever uma configuração ótima (ou perto da ótima) para o sistema.

Alguns estudos também têm sido desenvolvidos para integrar o modelo hipercubo aproximado em procedimentos de otimização; por exemplo, Batta *et al.* (1989), Saydam e Aytug (2003), Chiyoshi *et al.* (2003), Galvão *et al.* (2005) e Rajagopalan *et al.* (2008). Nestes estudos, abordagens de otimização foram apresentadas para determinar a localização das ambulâncias em SAEs, ou seja, resolver um problema de localização probabilístico (*location problem*), não tratando o problema de dimensionamento das áreas de cobertura das ambulâncias (*districting problem*). Em particular, em outro estudo anterior (IANNONI; MORABITO, 2006a), o modelo hipercubo com *backup* parcial foi integrado em um algoritmo genético (algoritmo AG/hipercubo de dimensionamento) para apoiar decisões de dimensionamento das áreas de cobertura de cada ambulância, de forma a determinar uma configuração ótima (ou perto da ótima) definida pelo tamanho

destas áreas, sem modificar a localização original das ambulâncias. O algoritmo foi aplicado para analisar um dos estudos de caso analisados no presente artigo (com único despacho). O estudo mostrou que as medidas de desempenho do sistema podem ser melhoradas somente modificando as áreas de atuação de cada ambulância, isto é, sem ter que repositonar as ambulâncias.

O presente estudo pode ser visto como uma extensão do estudo anterior em Iannoni e Morabito (2006a), em que inicialmente modificamos o algoritmo AG/hipercubo de dimensionamento para otimizar a localização das bases de ambulâncias ao longo da rodovia (*location problem*), aqui chamado algoritmo AG/hipercubo de localização. Depois estendemos este algoritmo para otimizar as decisões combinadas de localização das bases de ambulâncias (*location problem*) e dimensionamento das regiões de cobertura de cada base (*districting problem*), aqui chamado de algoritmo AG/hipercubo de localização e dimensionamento. Discute-se também como adaptar os algoritmos para gerar curvas de *trade-off* entre diferentes medidas de desempenho do sistema.

Para ilustrar o desempenho dos algoritmos, resultados computacionais são analisados aplicando-os a dois estudos de caso: o primeiro é um SAE que operava em trechos da rodovia Presidente Dutra (o antigo sistema *Anjos do Asfalto*), que foi inicialmente estudado em Mendonça e Morabito (2001), e o segundo é um SAE de uma concessionária de rodovias no Estado de São Paulo (sistema *Centrovias*), recentemente estudado em Iannoni e Morabito (2006c). Para

verificar a qualidade das soluções produzidas pelos algoritmos, utilizamos um simples procedimento enumerativo, que também incorpora o modelo hipercubo e que é capaz de obter a configuração ótima de problemas de tamanho pequeno. Para validar as medidas de desempenho produzidas pelo modelo hipercubo, elas são comparadas com as obtidas via simulação discreta.

Este artigo está organizado da seguinte forma: apresenta-se uma breve descrição dos estudos de caso dos SAEs em rodovias; na seção seguinte descreve-se resumidamente como o modelo hipercubo pode ser adaptado para analisar estes SAEs; continuando apresenta-se o algoritmo AG/hipercubo de localização para apoiar as decisões de localização das bases de ambulâncias (*location problem*), e este algoritmo é estendido para apoiar as decisões combinadas de localização e dimensionamento das áreas de cobertura das bases das ambulâncias (*location and districting problem*). A seguir, apresentam-se e analisam-se os resultados computacionais obtidos com a aplicação dos algoritmos nos estudos de caso e, finalmente, discutem-se as conclusões deste estudo e as perspectivas para pesquisas futuras.

## ESTUDOS DE CASO DE SAES EM RODOVIAS

### SAE Anjos do Asfalto

O sistema *Anjos do Asfalto*, inicialmente estudado por Mendonça e Morabito (2001), fornecia atendimento médico emergencial em parte da rodovia Presidente Dutra entre as cidades de São Paulo e Rio de Janeiro. Este SAE possuía seis bases fixas ao longo de um trecho linear da rodovia, sendo que em cada base ficava posicionada uma ambulância. A política de despacho consistia em enviar a ambulância da base mais próxima do local da chamada. Se a ambulância estivesse ocupada, a segunda ambulância mais próxima (chamada *backup*) era despachada. Se ela também estivesse ocupada, a chamada era transferida para outro SAE e era considerada perdida para o sistema. Além disso, neste sistema apenas

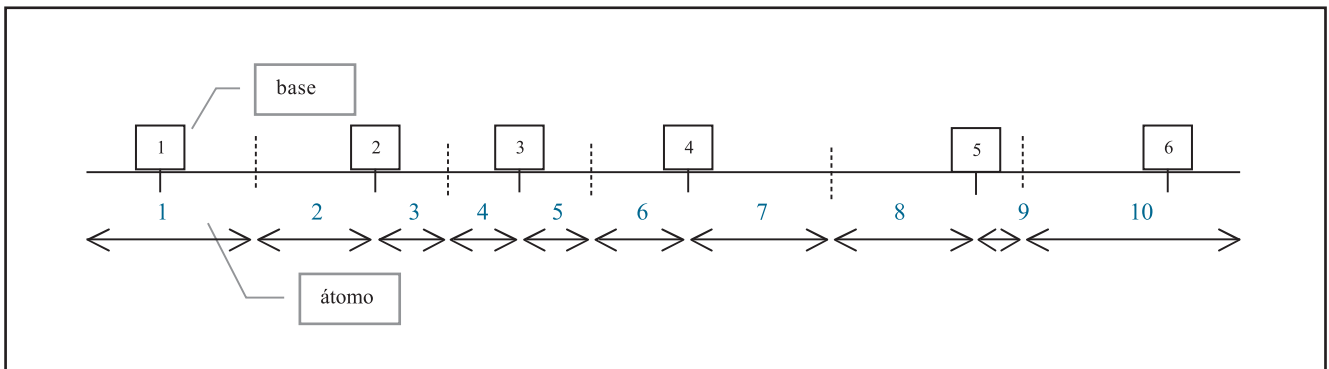
um servidor podia ser enviado a cada chamada. Esta política particular de despacho é chamada único despacho com *backup* parcial.

A Figura 1 ilustra a distribuição das seis bases de ambulâncias do SAE *Anjos do Asfalto*. A distância entre duas bases adjacentes é dividida em duas regiões (chamadas átomos), cada uma com uma lista de preferência de despacho. De acordo com esta lista e exceto para as ambulâncias 1 e 6, todas as ambulâncias são despachadas como preferenciais para dois átomos (à esquerda e à direita de sua base) e como *backup* para outros dois átomos (direita e esquerda das ambulâncias adjacentes à esquerda e à direita, respectivamente). Para mais detalhes deste sistema, o leitor pode consultar Mendonça e Morabito (2001) e Iannoni e Morabito (2006a).

### SAE Centrovias

Neste sistema, localizado no interior do Estado de São Paulo e analisado em Iannoni e Morabito (2006c), há cinco bases de ambulâncias (chamadas SAU – Serviço de Atendimento ao Usuário) ao longo dos trechos de concessão das rodovias, sendo que cada base possui uma ambulância. As ambulâncias são todas idênticas, e a central de operações está localizada em uma das bases. A política de despacho consiste em enviar a ambulância da base mais próxima do local da chamada, se a chamada requer apenas uma ambulância (único despacho). Se a ambulância estiver ocupada, a segunda ambulância mais próxima (*backup*) é despachada. Quando a chamada requer duplo despacho, as duas ambulâncias mais próximas são acionadas. Se alguma delas estiver ocupada, a ambulância disponível é despachada. Caso as duas ambulâncias mais próximas estiverem ocupadas, a chamada (seja único ou duplo despacho) é transferida para outro SAE e é considerada perdida (*backup* parcial). A Figura 2 ilustra a área de atuação do SAE analisado, envolvendo trechos de rodovias do interior do Estado de São Paulo.

Figura 1: Bases de ambulâncias e átomos do sistema.



**MODELO HIPERCUBO PARA SAES EM RODOVIAS**

Como os dois estudos de caso possuem algumas características diferentes, descrevemos brevemente a seguir como o modelo hipercubo pode ser adaptado para análise do SAES com único despacho (modelo 1 – SAE *Anjos do Asfalto*) e único e duplo despacho (modelo 2 – SAE *Centrovias*). O modelo 2, por sua vez, é uma extensão do modelo hipercubo múltiplo despacho de Chelst e Barlach (1981), desenvolvido para aplicação em sistemas de patrulhamento policial.

**Hipóteses dos modelos 1 e 2**

Algumas hipóteses específicas dos modelos 1 e 2 são:

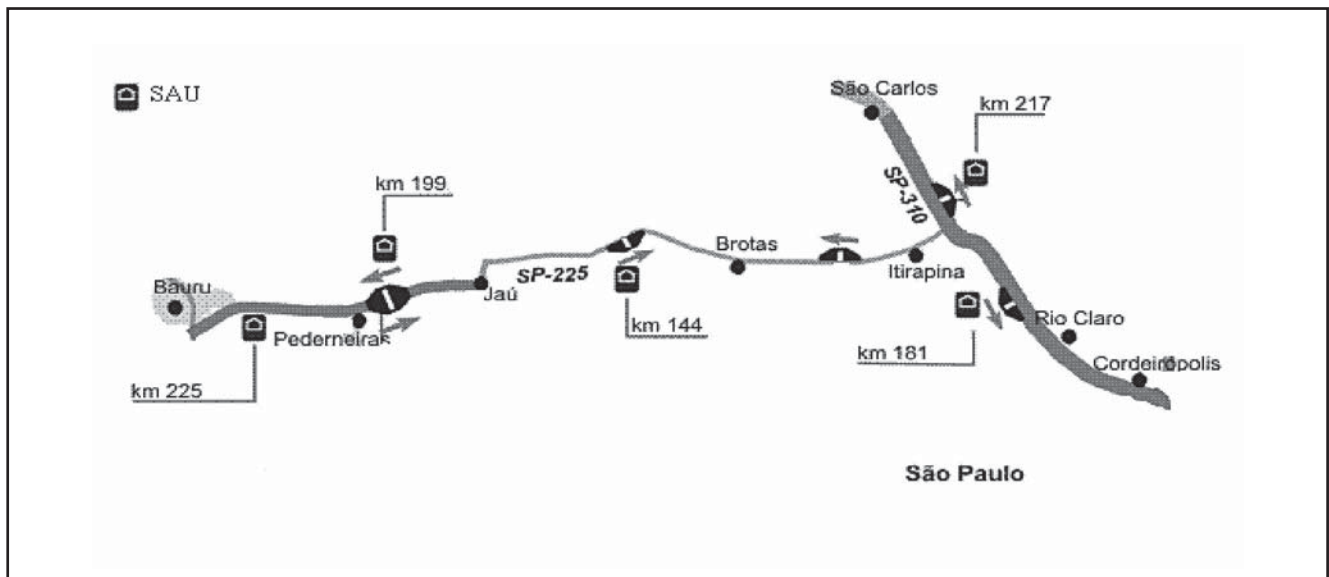
- No modelo 1, as chamadas chegam em cada átomo  $j$  com taxa média de chegada  $\lambda_j$ , de acordo com um processo de Poisson, independente dos demais átomos, e todas as chamadas são consideradas de um mesmo tipo (único despacho). No modelo 2, há dois tipos de chamadas: chamadas tipo 1 (único despacho), com taxa média de chegada  $\lambda_j^{[1]}$ , e chamadas tipo 2 (duplo despacho), com taxa média de chegada  $\lambda_j^{[2]}$ ;
- O despacho dos servidores é realizado de acordo com uma lista de preferência para cada átomo. De acordo com esta lista, no caso de uma chamada tipo 1, a primeira ambulância da lista (a mais próxima) é despachada e, se esta estiver ocupada, a segunda da lista é enviada (*backup*). Se a ambulância *backup* estiver ocupada, a chamada é perdida para o sistema, mesmo se houver outras ambulâncias disponíveis (*backup* parcial). No modelo 2, no caso de uma

chamada tipo 2, as duas primeiras ambulâncias da lista são despachadas e, se apenas uma delas estiver disponível, esta deve atender à chamada sozinha (possivelmente, com a ajuda de outro SAE). Se ambas as ambulâncias estiverem ocupadas, tanto chamadas tipo 1 quanto tipo 2 são perdidas, dado que uma terceira ambulância nunca é despachada (*backup* parcial).

- Os modelos admitem que os tempos de atendimento têm distribuição exponencial negativa. Como discutido em Chelst e Barlach (1981), no modelo 2, chamadas tipo 1 são atendidas por uma única ambulância  $i$  com taxa média de atendimento  $\mu_i$ , e chamadas tipo 2 são atendidas por duas ambulâncias  $i$  e  $k$ , que operam de forma independente entre si, com taxas médias de atendimento  $\mu_i$  e  $\mu_k$ , respectivamente.

Estas políticas de despacho específicas dos SAES em rodovias também requerem modificações nas equações de equilíbrio do sistema, com relação ao modelo hipercubo original (a solução destas equações resulta nas probabilidades de equilíbrio de cada estado do sistema). Neste modelo, uma transição pode ocorrer somente com a mudança de estado de apenas um servidor do sistema, e uma chamada é perdida apenas quando todos os servidores estão ocupados. Por outro lado, no modelo 2, uma transição também ocorre quando dois servidores se tornam ocupados com a chegada de uma chamada tipo 2 no sistema. Além disso, nos modelos 1 e 2, chamadas podem ser perdidas se os servidores de sua lista de despacho estão ocupados, mesmo que haja outros servidores disponíveis no sistema. Mais detalhes sobre as equações de

**Figura 2: SAE em trechos de rodovias do interior do Estado de São Paulo.**



equilíbrio e a aplicação dos modelos 1 e 2 para análise dos dois estudos de caso podem ser encontrados em Mendonça e Morabito (2001) e Iannoni e Morabito (2006c).

### Medidas de desempenho dos modelos 1 e 2

A partir das probabilidades de equilíbrio dos modelos 1 e 2, é possível obter interessantes medidas de desempenho para avaliar o SAE, tais como: tempos médios de resposta ao usuário, probabilidades de perda de chamadas, cargas de trabalho das ambulâncias, frações de despachos de cada ambulância para cada átomo, fração de despachos em que o tempo de viagem excede um tempo limite, etc.

Para o modelo 1, as deduções das fórmulas para calcular as medidas de desempenho são mais simples que para o modelo 2, e podem ser encontradas em Larson e Odoni (1981) e Mendonça e Morabito (2001). Para o modelo 2, as fórmulas são apresentadas com mais detalhes em Iannoni e Morabito (2006c), onde foram adaptadas das equações e formulações propostas em Larson e Odoni (1981) e Chelst e Barlach (1981).

Por exemplo, no modelo 1, o tempo médio de viagem é definido por:  $\bar{T} = \sum_{i=1}^N \sum_{j=1}^{N_A} f_{ij} t_{ij}$ , onde  $f_{ij}$  é a fração de despachos do servidor  $i$  ao átomo  $j$  (obtido a partir das probabilidades de equilíbrio),  $f_{ij}$  é o tempo de viagem do servidor  $i$  ao átomo  $j$  (calculado utilizando os dados de tempo de viagem entre os átomos),  $N$  é o número de servidores e  $N_A$  é o número de átomos.

No modelo 2, o tempo médio de viagem no sistema é definido por:

$$\bar{T} = \sum_{j=1}^{N_A} \left[ \sum_{i=1}^N (f_{ij}^{(1)} + f_{ij}^{(2)}) t_{ij} + \sum_{i=1}^{N-1} \sum_{k=i+1}^N f_{(i,k)j}^{(2)} \min(t_{ij}, t_{kj}) \right],$$

onde  $f_{ij}^{(1)}$  corresponde à fração de despachos em que a ambulância  $i$  é despachada para atender chamadas tipo 1 no átomo  $j$  (único despacho),  $f_{(i,k)j}^{(2)}$  é a fração de despachos em que ambulâncias  $i$  e  $k$  são despachadas simultaneamente para o átomo  $j$  para atender chamadas tipo 2,  $f_{ij}^{(2)}$  é a fração de despachos em que somente a ambulância  $i$  é despachada para atender chamadas tipo 2 no átomo  $j$  (i.e., quando a ambulância  $k$  está ocupada).

O desbalanceamento das cargas de trabalho dos servidores pode ser estimado pelo desvio padrão das cargas de trabalho. Outra medida interessante para análise do sistema é a fração de chamadas atendidas em mais de 10 minutos. Por exemplo, no modelo 2, esta medida é calculada por:

$$P_{tv>10} = \sum_{i=1}^N \sum_{j=1}^{N_A} (f_{ij}^{(1)} + f_{ij}^{(2)} + f_{(i,k)j}^{(2)}) p(t_{ij} > 10), \text{ onde } o$$

termo  $(f_{ij}^{(1)} + f_{ij}^{(2)} + f_{(i,k)j}^{(2)}) p(t_{ij} > 10)$  é a fração de todos os despachos do servidor  $i$  ao átomo  $j$  para os quais o tempo de viagem excede 10 minutos. Neste estudo, calculamos  $p(t_{ij} >$

10) determinando para qual fração do átomo  $j$  uma chamada de emergência é atendida pela ambulância  $i$  em tempo de viagem maior que 10 minutos. Mais detalhes sobre o cálculo destas e outras medidas dos modelos 1 e 2 podem ser encontrados em Chelst e Barlach (1981) e Iannoni e Morabito (2006a, 2006c).

**M**odificamos o algoritmo AG/hipercubo de dimensionamento para localizar ambulâncias ao longo da rodovia.

### ALGORITMO AG/HIPERCUBO DE LOCALIZAÇÃO

Conforme mencionado, em Iannoni e Morabito (2006a) foi proposta uma abordagem integrando o modelo hipercubo com um algoritmo genético, para otimizar o dimensionamento dos átomos do sistema (*districting problem*) em termos das principais medidas de desempenho, preservando a localização original das ambulâncias. A seguir modificamos este algoritmo AG/hipercubo de dimensionamento para localizar ambulâncias ao longo da rodovia (*location problem*), aqui chamado de algoritmo AG/hipercubo de localização. Ao invés de considerar apenas um conjunto discreto de possíveis locais para posicionar as ambulâncias, supomos que a localização delas pode ocorrer em qualquer ponto da rodovia.

Trata-se de um algoritmo simples e de fácil implementação, que segue a estrutura básica descrita em textos clássicos de algoritmos genéticos e populacionais, como Holland (1975), Goldberg (1989) e Michalewicz (1996). Ao relocar as bases de ambulâncias, o algoritmo redistribui a rodovia em átomos (ou regiões) e recalcula as taxas de chegada em cada átomo de forma a preservar a distribuição original da demanda ao longo da rodovia. Por simplicidade, o algoritmo divide a distância entre duas bases em dois átomos iguais (com exceção das extremidades).

Em casos como o SAE *Centrovias* envolvendo mais de uma rodovia, algumas extensões são necessárias no algoritmo. Por exemplo, o algoritmo considera que há dois trechos lineares (ou um conjunto de trechos lineares): um trecho com  $n_1$  ambulâncias e outro com  $n_2$  ambulâncias. Ele também considera que as ambulâncias de um trecho podem atender chamadas de outro trecho, como ocorre na prática. Como o SAE *Centrovias* considera múltiplo despacho de ambulâncias e há dois tipos de chamadas no sistema (chamadas tipo 1 e tipo 2), ao recalculer as taxas de chegadas em cada átomo, o algoritmo considera as duas taxas de chamadas em cada átomo. Além disso, ao recalculer as distâncias entre as bases de ambulâncias ao centróide de cada átomo e as listas



de preferência de despacho de cada átomo, o algoritmo considera que as ambulâncias de um trecho podem viajar a outro trecho. A seguir, descrevemos brevemente os principais componentes considerados na implementação do algoritmo AG/hipercubo de localização.

**Geração e representação dos cromossomos**

O algoritmo utiliza uma representação decimal dos cromossomos. Por meio desta representação, cada cromossomo é representado pelo vetor  $y = (y_1, y_2, \dots, y_N)$ , onde  $y_1$  é referente à proporção do trecho analisado, sendo que  $y_1 < y_2, \dots < y_N$ . Ele utiliza um procedimento para gerar aleatoriamente a população inicial, admitindo que  $0 \leq y_i \leq 1$ . Para gerar possíveis configurações do sistema, este procedimento simplesmente adiciona incrementos  $k\Delta$ , para cada localização, onde  $\Delta$  é fixo e  $k$  é um inteiro sorteado aleatoriamente dentro do intervalo  $[0, M = 1/\Delta]$ . Portanto, há  $M+1$  valores possíveis para cada  $y_i$ . O mesmo vetor discreto foi utilizado no processo de substituição dos genes no operador de mutação. Por exemplo, considerando o primeiro estudo de caso da Figura 1, se o trecho total da rodovia é  $D = 100$  km e o cromossomo  $y$  for gerado da forma descrita acima (i.e., sendo  $\Delta = 0,01$  e se os valores sorteados aleatoriamente para  $k$  no intervalo  $[0, M = 100]$  são (10; 25; 40; 60;75;90), então  $y = (y_1, y_2, y_3, y_4, y_5, y_6) = (0,10; 0,25; 0,40; 0,60; 0,75; 0,90)$ , e a configuração do sistema pode ser representada pela Figura 3, abaixo:

Introduzimos também uma restrição na geração de novos cromossomos relacionada à distância mínima entre as bases. Assim, o algoritmo leva em consideração a condição adicional de que a distância mínima entre as bases deve ser  $d_{min}$ , dependendo das características de operação do sistema analisado. Sendo o tamanho total do trecho analisado  $D$ , então:  $y_i - y_{i-1} \leq d_{min}/D$ . Por exemplo, se  $d_{min} = 20$  km, cada configuração gerada deve satisfazer a condição de que a distância mínima entre as bases adjacentes  $i - 1$  e  $i$  seja de pelo menos 20 km. No caso do SAE *Centrovias*, é necessário considerar o parâmetro  $d_{min}$  para cada trecho e o cromosso-

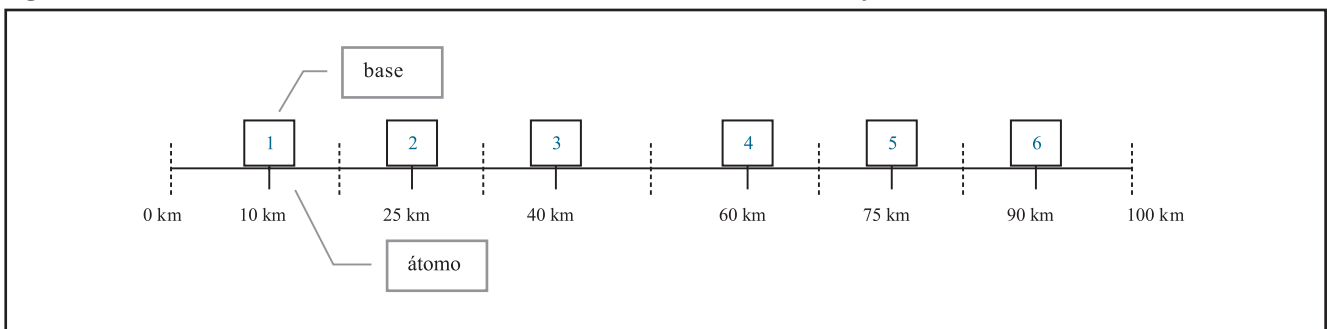
mo  $y$  é dividido em partes (cada parte representando um trecho). Por exemplo,  $y = (y_1, \dots, y_{n1}, y_{n1+1}, \dots, y_{n1+n2})$ , onde  $n_1$  é o número de ambulâncias no trecho 1 e  $n_2$  é o número de ambulâncias no trecho 2.

**Avaliação e função de aptidão (fitness) e seleção de cromossomos, crossover e mutação.**

O algoritmo utiliza os modelos hipercubos 1 e 2, apresentados em “Modelo hipercubo para SAEs em rodovias”, para avaliação das medidas de desempenho de cada configuração (representada por um cromossomo). Similarmente ao estudo em Iannoni e Morabito (2006a), analisamos separadamente três funções *fitness* representando três diferentes objetivos: o primeiro objetivo é minimizar o tempo médio de resposta no sistema:  $\min f(y) = \bar{T}(y)$ ; o segundo objetivo é minimizar o desbalanceamento das cargas de trabalho dos servidores:  $\min f(y) = \sigma_p(y)$ ; o terceiro objetivo é minimizar a fração de chamadas atendidas em mais de 10 minutos:  $f(y) = P_{t < 10}(y)$ . As expressões que calculam estas medidas foram apresentadas na seção “Medidas de desempenho dos modelos 1 e 2”.

De forma similar ao algoritmo AG/Hipercubo de dimensionamento, a seleção de cromossomos pais é baseada no método da roleta de probabilidades (GOLDBERG, 1989; MICHALEWICZ, 1996). Após a escolha de um par de cromossomos pais, aplicamos com probabilidade  $p_c$  o conhecido *crossover de um ponto* (e com probabilidade  $1 - p_c$ , o par selecionado é preservado na próxima população). O procedimento de mutação é aplicado a cada gene do cromossomo com uma probabilidade  $p_m$ . Para substituir aleatoriamente o gene, utilizou-se o mesmo procedimento discreto de geração da população inicial descrito acima. Por exemplo, para cada  $y_i$  (com probabilidade  $p_m$ ) este valor é alterado para  $y_i = k\Delta$ , onde  $k$  é uniformemente sorteado no intervalo  $[0, \dots, M]$ . Devido à restrição de distância mínima entre as bases, o *crossover* e mutação podem gerar cromossomos inválidos. Assim sendo, substituímos um cromossomo pai por um cromossomo filho  $y$  somente se este filho  $y$  for viável, isto é, se satisfizer  $y_i - y_{i-1} \leq d_{min}/D$  no mesmo trecho linear.

**Figura 3: Localização das ambulâncias de acordo com o vetor  $y$  do exemplo.**



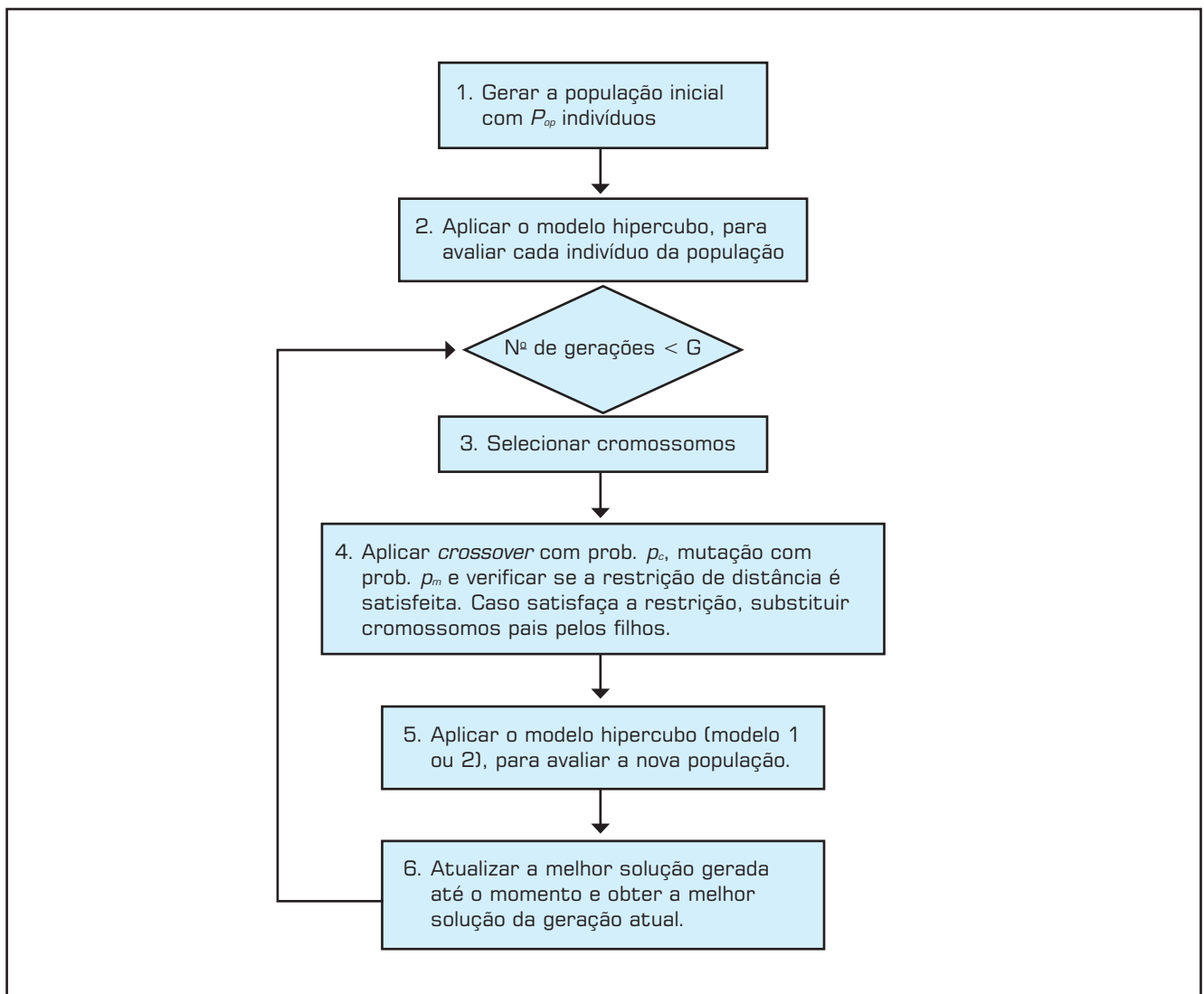
Para uma escolha adequada dos principais parâmetros do algoritmo, tais como probabilidades de *crossover* ( $p_c$ ) e mutação ( $p_m$ ), número de gerações ( $G$ ) e tamanho da população ( $P_{op}$ ), realizamos inicialmente testes computacionais extensivos com diferentes combinações de valores dentro de intervalos. Os melhores resultados foram obtidos com:  $p_c = 0,7$ ,  $p_m = 0,1$ ,  $G = 1000$  gerações e  $P_{op} = 200$  indivíduos. Os intervalos  $\Delta = 0,03$  e  $\Delta = 0,01$  (i.e.,  $M = 33$  e  $M = 100$ , respectivamente) foram testados para cada uma das três funções de aptidão. Um parâmetro adicional corresponde à distância mínima entre as bases  $d_{min}$ . Desta forma, foram realizados diferentes experimentos para  $d_{min} = 20$  km e  $d_{min} = 30$  km, respectivamente. Por exemplo, na configuração original do SAE *Anjos do Asfalto*, a menor distância entre as bases é de 21 km.

Vários estudos, como Jaszkievicz (2002) e Arroyo e Armentano (2005), têm ressaltado a superioridade dos algoritmos genéticos híbridos, os quais incluem um procedimento de busca local para melhorar as soluções geradas após o *crossover* e a mutação. Esta alternativa não foi considerada no presente algoritmo, dado que o processo de avaliação de soluções, baseado na solução de sistemas lineares de  $O(2^N)$  equações do modelo hipercubo, se tornaria computacionalmente custoso. A Figura 4 apresenta o esquema geral do algoritmo AG/hipercubo de localização.

#### Geração de curvas de *trade-off*

Algumas medidas de desempenho podem ser conflitantes em termos dos diferentes interesses das partes envolvidas na operação dos SAEs. Por exemplo, o tempo médio de

Figura 4: Passos do algoritmo AG/hipercubo de localização de ambulâncias.



resposta numa região é uma medida de desempenho externa do sistema, que interessa principalmente ao usuário do sistema. Por outro lado, o balanço das cargas de trabalho dos servidores é uma medida de desempenho interna do sistema, que interessa particularmente aos operadores do sistema. Similarmente ao estudo em Iannoni e Morabito (2006a), utilizamos o conceito de dominância de soluções (chamado dominância de Pareto) para analisar a relação entre estes objetivos conflitantes. Um simples método de gerar soluções ótimas de Pareto é o método  $\epsilon$ -restrito (COHON, 1978; ARROYO; ARMENTANO, 2005). Este consiste em otimizar um objetivo, enquanto os demais objetivos são limitados por um valor  $\epsilon$ . Por exemplo, considerando os objetivos:  $\min f_1(y) = \bar{T}(y)$  e  $\min f_2(y) = \sigma_p(y)$ , temos a função objetivo:  $\min Z = (f_1(y), f_2(y)) = (\bar{T}(y), \sigma_p(y))$ . Utilizando o método  $\epsilon$ -restrito para minimizar o tempo médio de viagem no sistema, e considerando o desvio padrão das cargas de trabalho como uma restrição, o problema pode ser formulado como:

$$\begin{aligned} \text{Min } Z &= f_1(y) = \bar{T}(y) \\ \text{s.a. } f_2(y) &= \sigma_p(y) \leq \epsilon \\ y &\in Y^* \text{ onde } \epsilon \text{ corresponde ao valor do limite superior de } \sigma_p. \end{aligned}$$

## O algoritmo utiliza modelos hipercubos para avaliação das medidas de desempenho de cada configuração.

Para resolver este problema, o algoritmo AG/hipercubo de localização foi modificado da seguinte forma: durante uma simples rodada, variamos os valores de  $\epsilon$ , e para cada valor o algoritmo procura as melhores soluções durante um conjunto de  $G = 1000$  gerações. A cada  $G$  gerações, o algoritmo carrega (guarda na memória) as melhores soluções encontradas durante a busca atual e as anteriores para as próximas  $G$  gerações. Para isso, o algoritmo verifica se na busca atual ocorreu melhoria nos melhores valores da função objetivo  $\bar{T}$  encontrados até o momento para cada valor de  $\sigma_p$ , e atualiza estes valores e os guarda para o próximo conjunto de  $G$  gerações em uma tabela virtual. Além disso, modificações adicionais são feitas durante o processo de geração da população inicial e no procedimento de geração de cromossomos por meio de *crossover* e mutação, para garantir que sejam gerados apenas cromossomos viáveis para o problema. Por exemplo, após o procedimento de *crossover* e mutação, substituímos um cromossomo pai por um cromossomo filho  $y$  somente se este filho for viável, isto é, se satisfizer  $\sigma_p(y) \leq \epsilon$ . Note que, para verificar isso, precisamos aplicar o modelo hipercubo para avaliar cada filho  $y$ . Após variar todos os diferentes valores de  $\epsilon$ , o algoritmo obtém uma curva aproximada de *trade-off* entre  $\bar{T}$  e  $\sigma_p$  (ou seja, as

funções não-dominadas encontradas durante toda a rodada considerando os possíveis valores de  $\epsilon$ ).

### ALGORITMO AG/HIPERCUBO DE LOCALIZAÇÃO E DIMENSIONAMENTO

A seguir estendemos o algoritmo AG/hipercubo de localização para considerar as duas decisões: (i) localizar as ambulâncias; (ii) determinar suas áreas de cobertura (tamanho dos átomos do sistema). O algoritmo AG/hipercubo de localização e dimensionamento possui basicamente dois passos. No passo 1, o algoritmo otimiza a localização das bases de ambulâncias aplicando o algoritmo AG/hipercubo de localização. Em seguida, partindo da solução do passo 1, o algoritmo otimiza o dimensionamento dos tamanhos dos átomos desta solução aplicando o algoritmo AG/hipercubo de dimensionamento (passo 2) descrito em Iannoni e Morabito (2006a).

No passo 2, consideramos a localização inicial da melhor solução encontrada no passo 1 (lembre-se que nas soluções do passo 1, a distância entre duas bases adjacentes é dividida em dois átomos iguais), e aplicamos um procedimento que modifica o tamanho dos átomos de forma a gerar diferentes configurações para o sistema. Cada cromossomo do sistema pode ser representado por um vetor  $x = (x_1, x_2, \dots, x_{N-1})$ , onde cada  $x_i$  é a proporção da distância entre duas ambulâncias adjacentes  $i$  e  $i+1$ , que determina o tamanho do átomo preferencial da ambulância  $i$ . Se a distância entre estas duas bases é  $d_i$  o tamanho do átomo preferencial para a ambulância  $i$  é  $x_i d_i$ . A distância residual torna-se o átomo preferencial da base  $i+1$  (isto é,  $d_i - x_i d_i$ ). Conforme sugerido pelos gerentes e operadores do sistema, admitimos que  $0,2 \leq x_i \leq 0,8$ . O cromossomo  $x$  é gerado por um procedimento similar ao utilizado no passo 1 para o cromossomo  $y$ . Neste procedimento, adicionamos 0,2 (o limite inferior do intervalo) a um incremento  $k\Delta$ , onde  $\Delta$  é fixo e  $k$  é um inteiro sorteado aleatoriamente dentro do intervalo  $[0, M = (0,8-0,2)/\Delta]$ . Após determinar o novo tamanho dos átomos do sistema, aplicamos um procedimento para recalculas as taxas de chegada em cada átomo de forma a preservar a distribuição de demanda ao longo da rodovia. Os demais passos do algoritmo AG/hipercubo de dimensionamento, tais como gerar possíveis configurações do sistema, avaliação da *fitness*, seleção de cromossomos, *crossover* e mutação, são similares aos descritos para o algoritmo AG/hipercubo de localização.

Note que dois algoritmos genéticos (localização e dimensionamento) são aplicados sequencialmente, um em cada passo da abordagem. Alternativamente, poder-se-ia também aplicar o algoritmo genético de dimensionamento (passo 2) para cada solução (indivíduo gerado) no algoritmo gené-



tico de localização (passo 1). No entanto, esta alternativa tornaria a abordagem muito custosa em termos dos tempos computacionais, como se pode verificar pelos resultados apresentados na próxima seção.

Para uma escolha adequada dos parâmetros dos dois algoritmos genéticos envolvidos, tais como probabilidades de *crossover* ( $pc_1$  e  $pc_2$ ) e mutação ( $pm_1$  e  $pm_2$ ), número de gerações ( $G_1$  e  $G_2$ ), tamanho da população ( $Pop_1$  e  $Pop_2$ ) e precisão  $\Delta_1$  e  $\Delta_2$ , testes computacionais extensivos foram realizados com diferentes combinações de valores dentro de intervalos. Os melhores resultados foram obtidos com:  $pc_1 = 0,7$  e  $pc_2 = 0,5$ ;  $pm_1 = 0,1$  e  $pm_2 = 0,06$ ;  $G_1 = 1000$  e  $G_2 = 1000$ .

Também investigamos uma pequena modificação do algoritmo acima, em que no passo 1, em vez de o algoritmo AG/hipercubo de localização guardar apenas a melhor solução (do ponto de vista de localização), ele guarda as  $nc$  melhores soluções, e no passo 2, o algoritmo AG/hipercubo de dimensionamento é aplicado para cada uma destas  $nc$  configurações do passo 1, retornando a melhor solução obtida (do ponto de vista de localização e dimensionamento). Note que, desta forma, a versão anterior do algoritmo pode ser vista como o caso particular desta em que  $nc = 1$ .

## RESULTADOS COMPUTACIONAIS

Os algoritmos AG/hipercubo de localização e AG/hipercubo de localização e dimensionamento foram implementados em linguagem Pascal e executados em um microcomputador Athlon 1.66 GHz Centrino Duo T2300.

### Resultados do algoritmo AG/hipercubo de localização

Resultados para o SAE Anjos do Asfalto: a configuração original do sistema *Anjos do Asfalto*, com base nos dados reportados por Mendonça e Morabito (2001), é representada pelo cromossomo  $y = (y_1, y_2, y_3, y_4, y_5, y_6) = (0,0; 0,2192; 0,3315; 0,4973; 0,7807; 1,0)$ . O trecho todo tem uma extensão de 187 km. As medidas de desempenho desta confi-

guração são:  $\bar{T} = 7,912$  minutos,  $\sigma_p = 0,0551$  e  $P_{r>0} = 0,299$  (conforme IANNONI; MORABITO, 2006a).

Para verificar a qualidade das soluções produzidas pelo algoritmo AG/hipercubo de localização, desenvolvemos um simples algoritmo enumerativo que incorpora o modelo hipercubo. Este procedimento exaustivo determina a configuração ótima em termos da localização das bases (sob uma precisão de  $\Delta$ ), testando todas as configurações possíveis para o sistema (considerando também a restrição da distância mínima  $d_{min}$ ). Note que este algoritmo enumerativo é tratável computacionalmente somente para problemas de tamanho pequeno, como o SAE *Anjos do Asfalto*, que tem apenas  $N = 6$  ambulâncias, e considerando uma precisão de  $\Delta = 0,01$ .

Inicialmente realizamos experimentos otimizando separadamente as seguintes funções de aptidão:  $\min f(y) = \bar{T}(y)$ ;  $\min f(y) = \sigma_p(y)$  ou  $\min f(y) = P_{r>10}(y)$ . Os correspondentes valores ótimos de  $\bar{T}(y)$ ,  $\sigma_p(y)$  e  $P_{r>10}(y)$  obtidos pelos algoritmos AG/hipercubo e enumerativo em cada experimento são apresentados na Tabela 1 (valores em negrito, onde AG é o algoritmo genético e AE é o algoritmo enumerativo). Estes resultados são também comparados com a configuração do sistema original. Os parâmetros adicionais utilizados para obter os resultados são:  $\Delta = 0,01$  e  $d_{min} = 20$ km.

Note na Tabela 1 que, nos três experimentos, as medidas  $\bar{T}$ ,  $\sigma_p$  e  $P_{r>10}$  são todas melhoradas quando comparadas com a configuração original do sistema. Estas medidas também são melhoradas quando comparadas com as soluções obtidas pelo algoritmo AG/hipercubo de dimensionamento apresentado em Iannoni e Morabito (2006a), que otimiza as áreas de cobertura das ambulâncias na localização original do sistema (i.e., apenas variando o tamanho dos átomos). Por exemplo, quando o objetivo é minimizar o tempo médio de viagem  $\bar{T}$ , o algoritmo AG/hipercubo de localização resulta em 6,2311 minutos (Tabela 1), enquanto que o AG/hipercubo de dimensionamento em Iannoni e Morabito (2006a) resulta em 7,778 minutos (similarmente para as demais medidas da Tabela 1).

O algoritmo AG/hipercubo de localização encontra a

Tabela 1: Resultados do algoritmo AG/hipercubo de localização para as três funções *fitness*.

MEDIDA	SISTEMA ORIGINAL		OBJETIVO min $\bar{T}(y)$	MELHORIA	OBJETIVO min $\sigma_p(y)$	MELHORIA	OBJETIVO min $P_{r>10}(y)$	MELHORIA
$\bar{T}(y)$	7,912	<b>AG</b> <b>AE</b>	<b>6,231</b> <b>6,231</b>	21,25%	7,703 7,344	2,64%	6,402 6,402	19,08%
$\sigma_p(y)$	0,0551	<b>AG</b> <b>AE</b>	0,0507 0,0507	7,98%	<b>0,0218</b> <b>0,0216</b>	60,44%	0,0480 0,0480	12,88%
$P_{r>10}(y)$	0,299	<b>AG</b> <b>AE</b>	0,166 0,166	44,48%	0,268 0,271	10,36%	<b>0,147</b> <b>0,147</b>	50,84%

mesma solução do algoritmo enumerativo para o primeiro e terceiro experimentos (Tabela 1). Note que no primeiro experimento,  $\bar{T}$  (sendo minimizado) é reduzido em 21,25%, e no terceiro experimento,  $P_{>10}(x)$  (sendo minimizado) é reduzido em 50,84%. No segundo experimento, o algoritmo AG/hipercubo encontra uma solução muito próxima da do algoritmo enumerativo (0,0218 e 0,0216, respectivamente). Com relação aos tempos computacionais, o algoritmo AG/hipercubo consome em média 650 segundos (10,8 minutos), enquanto que o algoritmo enumerativo consome mais de 11 horas para testar todas as possibilidades (ambos usando  $\Delta = 0,01$  e  $d_{min} = 20\text{km}$ ). A configuração (com 12 átomos) que minimiza o tempo médio de resposta no sistema ( $\min f(y) = \bar{T}(y)$ ) é representada pelo cromossomo  $y = (y_1, y_2, y_3, y_4, y_5, y_6) = (0,07; 0,23; 0,37; 0,56; 0,74; 0,88)$ .

**Simulação da configuração que minimiza o tempo médio de viagem:** para validar as medidas de desempenho desta configuração ótima, construímos um modelo de simulação discreta do sistema *Anjos do Asfalto* utilizando o *software Arena*. O procedimento utilizado para calcular a fase transiente e o tamanho da simulação é o mesmo descrito em Iannoni e Morabito (2006b). A Tabela 2 apresenta os resultados para o tempo de viagem  $\bar{TU}_i$  e a carga de trabalho  $\rho_i$  de cada ambulância do sistema. Estes resultados são validados por meio da análise do intervalo de confiança (nível de significância  $\alpha = 0,05$ ); note que todos os valores produzidos pelo modelo hipercubo estão dentro dos respectivos intervalos.

Em particular, o tempo médio de viagem no sistema calculado pela simulação é de  $\bar{T} = 6,3058$  min (Interv. conf. 6,1867 – 6,4248), enquanto que o resultado obtido pelo modelo hipercubo na Tabela 1 é  $\bar{T} = 6,2311$  min.

**Curva de trade-off:** a Figura 5 mostra o gráfico das soluções eficientes obtidas com a aplicação do algoritmo AG/hipercubo de localização modificado para solucionar o problema biobjetivo (objetivo  $\min f_i(y) = \bar{T}(y)$  e restrição  $\sigma_p(y)$ ), para diferentes valores de  $\varepsilon$  variando de 0,026 a 0,055 (usando  $\Delta = 0,01$ ). Variando os valores da restrição  $\sigma_p$  para atualizar os valores de  $\bar{T}$  em cada conjunto de gerações, obtemos os valores finais para as soluções eficientes mostrados na Figura 5.

**Resultados para o SAE *Centrovias*:** conforme Iannoni e Morabito (2006c), a configuração original do sistema é representada pelo cromossomo  $y = (y_1, y_2, y_3, y_4, y_5) = (0,135; 0,622; 0,368; 0,750; 0,924)$ , onde  $n_1 = 2$  e  $n_2 = 3$  (5 ambulâncias no total). O trecho 1 tem 74 km e o trecho 2 tem 144 km. Os operadores do sistema consideram duas possíveis configurações para o sistema em termos da divisão das áreas de influência de cada ambulância (tamanho dos átomos do sistema); veja Figuras 6 e 7. A primeira delas é analisada em Iannoni e Morabito (2006c), onde se aplicou o algoritmo hipercubo múltiplo despacho para analisar as medidas de desempenho. Note na Figura 6 que, nesta configuração, há 8 átomos no sistema (2 átomos no trecho 1 e 6 átomos

Tabela 2: Tempo de viagem (minutos) e carga de trabalho de cada servidor.

SERVIDOR $i$	MODELO	$\bar{TU}_i$	$\rho_i$
1	hipercubo	4,680	0,148
	simulação	4,687	0,152
	Intervalo de confiança	[4,517; 4,857]	[0,145; 0,166]
2	hipercubo	6,177	0,206
	simulação	6,402	0,214
	Intervalo de confiança	[6,051; 6,753]	[0,198; 0,229]
3	hipercubo	7,064	0,164
	simulação	7,145	0,162
	Intervalo de confiança	[6,78; 7,501]	[0,148; 0,176]
4	hipercubo	5,902	0,295
	simulação	5,806	0,299
	Intervalo de confiança	[5,631; 5,981]	[0,280; 0,318]
5	hipercubo	7,413	0,147
	simulação	7,653	0,144
	Intervalo de confiança	[7,245; 8,060]	[0,134; 0,154]
6	hipercubo	5,796	0,191
	simulação	5,812	0,199
	Intervalo de confiança	[5,644; 5,980]	[0,183; 0,215]

no trecho 2, com 2 átomos entre o ponto de cruzamento e a ambulância 3).

Na segunda configuração há 10 átomos no sistema (a distância entre duas bases adjacentes é dividida em dois átomos iguais, com exceção das extremidades). Neste caso há apenas 1 átomo entre o ponto de cruzamento e a ambulância 3, e o trecho 1 possui 4 átomos e o trecho 2 possui 6 átomos (veja Figura 7). Considerando que a primeira configuração é muito específica para este estudo de caso, e que esta segunda configuração é mais simples e similar ao primeiro estudo de caso, no presente estudo aplicamos os algoritmos apenas para esta segunda configuração. Ao aplicar o modelo hiper-cubo múltiplo despacho nesta segunda configuração inicial, obtemos as seguintes medidas de desempenho:  $\bar{T} = 7,1530$  minutos,  $\sigma_p = 0,0151$  e  $P_{>10} = 0,228$ .

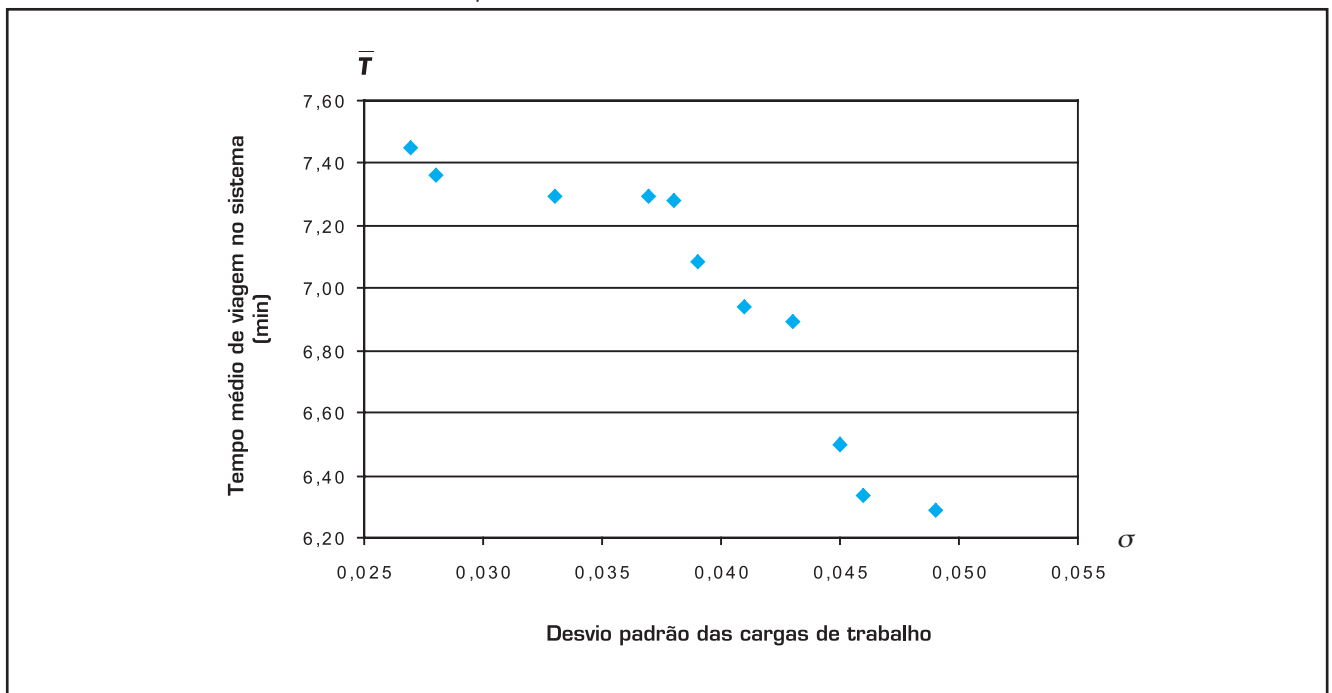
Para verificar a qualidade das soluções produzidas pelo algoritmo AG/hipercubo de localização, também aplicamos o algoritmo enumerativo, no entanto, este não foi capaz de testar todas as possibilidades em menos de 24 horas de tempos de execução (com os mesmos parâmetros anteriores  $\Delta = 0,01$  e  $d_{min} = 20$  km). Isso se deve ao fato de que, para o SAE *Centrovias*, o número de combinações possíveis é muito maior que para o SAE *Anjos do Asfalto*, pois o trecho é bem maior. Assim, apenas os valores de  $\bar{T}(y)$ ,  $\sigma_p(y)$  e  $P_{>10}(y)$  obtidos pelo algoritmo AG/hipercubo em cada experimento são apresentados na Tabela 3. O algoritmo consumiu em média 183 segundos para encontrar estes resultados. Aplicando-se

o algoritmo enumerativo para uma precisão menor  $\Delta = 0,03$ , obtém-se em cada experimento  $\bar{T}(y) = 4,5182$  minutos,  $\sigma_p(y) = 0,01019$  e  $P_{>10}(y) = 0,1492$ , respectivamente, dois deles piores e o outro muito próximo dos valores (em negrito) das soluções da Tabela 3, atestando a qualidade das soluções do algoritmo AG/hipercubo.

Note na Tabela 3 que, nos três experimentos,  $\bar{T}$ ,  $\sigma_p$  e  $P_{>10}$  são todos melhorados quando comparados com a configuração original do sistema. No primeiro experimento,  $\bar{T}$  (sendo minimizado) é reduzido em 36,92%, enquanto que  $\sigma_p$  aumenta significativamente em 108,6%. Similarmente, no segundo experimento,  $\sigma_p$  (sendo minimizado) é reduzido em 32,38%, enquanto que  $\bar{T}$  e  $P_{>10}$  aumentam em 33,32% e 101,7%, respectivamente, e no terceiro experimento,  $P_{>10}$  (sendo minimizado) é reduzido em 37,28%, enquanto que  $\sigma_p$  aumenta em 17,21% e  $\bar{T}$  diminui em 9,17%. Note como estas diferentes medidas de desempenho são conflitantes entre si, pois ao melhorar uma medida, pioramos outra. Por exemplo, ao minimizar  $\bar{T}$  pioramos significativamente o valor de  $\sigma_p$ .

Por exemplo, a configuração que minimiza o tempo médio de resposta no sistema ( $\min f(y) = \bar{T}(y)$ ) da Tabela 3 é representada pelo cromossomo  $y = (y_1, y_2, y_3, y_4, y_5) = (0,59; 0,87; 0,0; 0,55; 0,87)$ . Note que nesta configuração  $y_3 = 0,0$  (a ambulância 3 está localizada no início do trecho 2), porém não há divisão de átomos entre as ambulâncias 1 e 3, e a divisão ocorre entre as ambulâncias 1 (km 43,7) e 2 (km 64,4).

Figura 5: Curva de *trade-off* entre  $\bar{T}$  e  $\sigma_p$  obtida pelo algoritmo AG/hipercubo de localização.



**Simulação da configuração que minimiza o tempo médio de viagem:** similarmente ao estudo de caso anterior, avaliamos a solução ótima desta configuração por meio de simulação. Como discutido em Iannoni e Morabito (2006c), a hipótese de que a distribuição do tempo médio de serviço é exponencial é rejeitada para todos os servidores do sistema. Desta forma, realizamos análises estatísticas para determinar qual a melhor distribuição que se ajusta aos dados por meio do *software Best-Fit* para o modelo de simulação. Foram encontradas as seguintes distribuições para o tempo de serviço em cada servidor (que na simulação não inclui o tempo de viagem ao local da chamada): ambulância 1 – Lognormal (41,10; 57,24); ambulância 2 – Erlang (16,51; 3); ambulância 3 – Lognormal (69,46; 66,59); ambulância

4 – Erlang (9,59; 5) e ambulância 5 – Lognormal (49,51; 39,13). Os resultados para o tempo médio de viagem para chamadas tipo 1 e a carga de trabalho para cada servidor obtidos via simulação e o intervalo com 95% de confiança de cada resultado são apresentados na Tabela 4. Note que todos os valores produzidos pelo modelo hipercubo estão dentro dos respectivos intervalos. Em particular, o tempo médio de viagem no sistema calculado pela simulação é de  $\bar{T} = 4,5604$  min [Interv. conf. (4,4663; 4,6545)], enquanto que o resultado obtido pelo modelo hipercubo na Tabela 3 é  $\bar{T} = 4,5117$  min.

**Curva de trade-off:** similarmente à análise multiobjetivo para o sistema *Anjos do Asfalto*, a Figura 8 mostra a curva de

Figura 6: Configuração inicial 1 do SAE *Centrovias*.

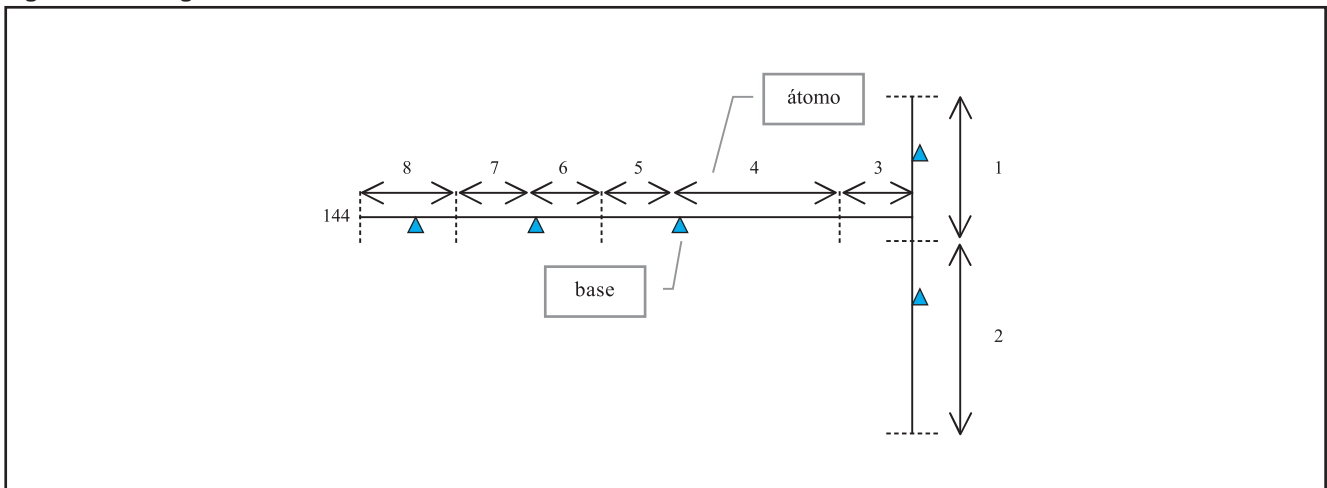
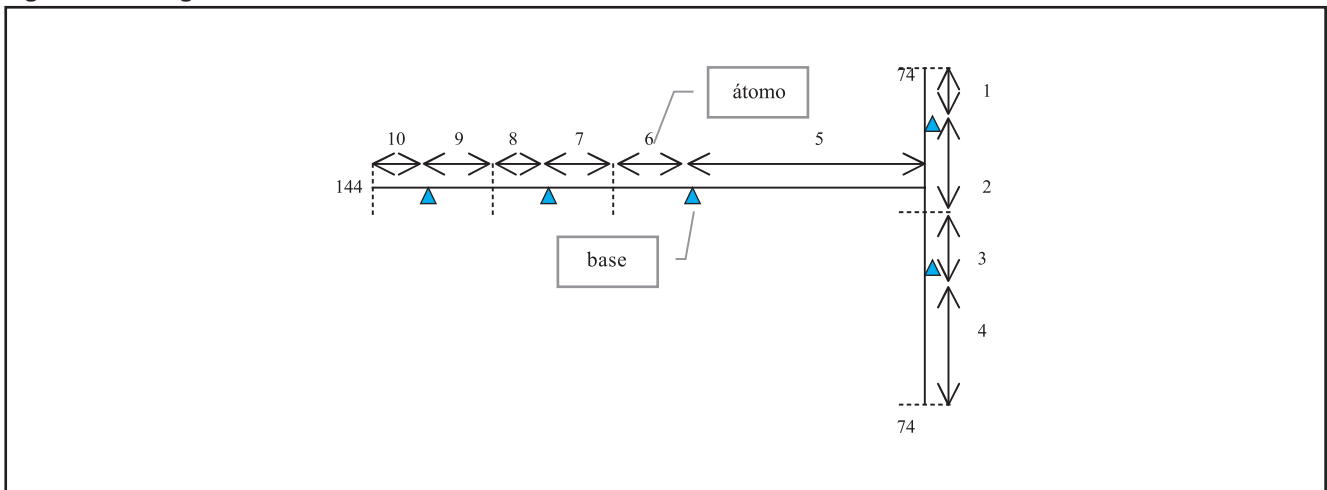


Figura 7: Configuração inicial 2 do SAE *Centrovias*.



trade-off entre  $\bar{T}$  e  $\sigma_p$ , obtida pelo algoritmo AG/hipercubo de localização. Neste caso, variamos os valores de  $\epsilon$  de 0,032 a 0,011 com base nos resultados obtidos nos experimentos anteriores na Tabela 3 (usando  $\Delta = 0,01$ ).

### Resultados do algoritmo AG/hipercubo de localização e dimensionamento

Resultados para o SAE *Anjos do Asfalto*: a configuração original do sistema é representada pelos cromossomos  $y = (y_1, y_2, y_3, y_4, y_5, y_6) = (0,0; 0,2192; 0,3315; 0,4973; 0,7807; 1,0)$  para localização das bases de ambulâncias e pelo cromossomo  $x = (x_1, x_2, x_3, x_4, x_5) = (0,50, 0,50, 0,50, 0,50, 0,22)$ . Como discutido anteriormente, os seguintes resultados são encontrados para esta configuração original:  $\bar{T} = 7,912$  minutos e  $\sigma_p = 0,0551$ . As Tabelas 5 e 6 apresentam os resultados para a primeira e segunda solução obtida pelo AG/hipercubo de localização e dimensionamento. Foram realizadas 20 rodadas, considerando minimizar  $\bar{T}$  e  $\sigma_p$ , respectivamente. Os parâmetros utilizados são:  $G_1 = 1000$ ;  $G_2$

$= 1000$ ;  $Pop_1 = 200$ ;  $Pop_2 = 100$ ;  $\Delta_1 = 0,01$  e  $\Delta_2 = 0,01$ . Os cromossomos que representam a solução da Tabela 5 são:  $y = (y_1, y_2, y_3, y_4, y_5, y_6) = (0,07; 0,23; 0,37; 0,56; 0,74; 0,88)$  para localização das bases de ambulâncias e  $x = (x_1, x_2, x_3, x_4, x_5) = (0,45, 0,42, 0,60, 0,42, 0,48)$ . Os cromossomos que representam a solução da Tabela 6 são:  $y = (y_1, y_2, y_3, y_4, y_5, y_6) = (0,12; 0,23; 0,35; 0,46; 0,68; 0,96)$  para localização das bases de ambulâncias e  $x = (x_1, x_2, x_3, x_4, x_5) = (0,78, 0,46, 0,80, 0,44, 0,48)$ .

Um experimento adicional foi realizado para analisar o desempenho da variação do algoritmo AG/hipercubo de localização e dimensionamento, que considera as melhores  $nc$  configurações no passo 1, ao invés de apenas uma (veja seção 5). Utilizamos arbitrariamente  $nc = 10$  configurações. A Tabela 7 apresenta as 10 melhores soluções obtidas no passo 1 (algoritmo AG/hipercubo de localização), ao lado das respectivas soluções obtidas no passo 2 (algoritmo AG/hipercubo de dimensionamento), com o objetivo de minimizar  $\bar{T}$ . Note que a melhor solução do passo 2 (solução

Tabela 3: Resultados do algoritmo AG/hipercubo de localização para as três funções *fitness*.

MEDIDA	SISTEMA ORIGINAL		OBJETIVO min $\bar{T}(x)$	MELHORIA	OBJETIVO min $\sigma_p(x)$	MELHORIA	OBJETIVO min $P_{>10}(x)$	MELHORIA
$\bar{T}(y)$	7,1530	<b>AG</b>	<b>4,5117</b>	36,92%	9,5365	-33,32%	6,4837	9,41%
$\sigma_p(y)$	0,0151	<b>AG</b>	0,0315	-108,6%	<b>0,01021</b>	32,38%	0,0176	-16,56%
$P_{>10}(y)$	0,228	<b>AG</b>	0,205	10,13%	0,460	-101,7%	<b>0,143</b>	37,28%

Tabela 4: Tempo de viagem (minutos) e carga de trabalho para cada servidor

SERVIDOR $i$	MODELO	$\bar{TU}_i^{(1)}$ (min)	$\rho_i$
1	hipercubo simulação	8,127	0,0243
	Intervalo de confiança	8,090 [7,767; 8,413]	0,0246 [0,0221; 0,0267]
2	hipercubo simulação	3,068	0,0223
	Intervalo de confiança	3,123 [3,065; 3,171]	0,0217 [0,0203; 0,0231]
3	hipercubo simulação	1,942	0,1024
	Intervalo de confiança	1,974 [1,868; 2,079]	0,1015 [0,0978; 0,1052]
4	hipercubo simulação	10,380	0,0188
	Intervalo de confiança	10,357 [10,022; 10,692]	0,0215 [0,0204; 0,0226]
5	hipercubo simulação	6,320	0,0321
	Intervalo de confiança	6,325 [6,275; 6,375]	0,0326 [0,0311; 0,0341]



8 - 6,1487 minutos) não é obtida a partir da melhor solução do passo 1 (solução 1 – 6,2311 minutos), o que mostra que a variação do algoritmo AG/hipercubo de localização e dimensionamento pode ser mais efetiva, apesar de demandar tempos computacionais bem maiores. Com relação aos tempos computacionais, o algoritmo com  $nc = 1$  consome em média 920 segundos (15,3 minutos) e com  $nc = 10$  consome em média 3.800 segundos (63,3 minutos).

Resultados para o SAE Centrovias: a configuração original do sistema é representada pelos cromossomos  $y = (y_1, y_2,$

$y_3, y_4, y_5) = (0,135; 0,622; 0,368; 0,750; 0,924)$  e  $x = (x_1, x_2, x_3) = (0,5; 0,5; 0,5)$ , onde  $n_1 = 2$  e  $n_2 = 3$ . Esta configuração foi ilustrada na Figura 7. As Tabelas 8 e 9 apresentam os resultados obtidos no primeiro e segundo passos do algoritmo. Foram realizadas 20 rodadas, considerando minimizar  $\bar{T}$  e  $\sigma_p$ , respectivamente. Os parâmetros utilizados são:  $G_1 = 1000$ ;  $G_2 = 1000$ ;  $Pop_1 = 200$ ;  $Pop_2 = 100$ ;  $\Delta_1 = 0,01$  e  $\Delta_2 = 0,01$ . Os cromossomos que representam a solução da Tabela 8 são:  $y = (y_1, y_2, y_3, y_4, y_5) = (0,59; 0,87; 0,0; 0,55; 0,87)$  para localização das bases de ambulâncias e  $x = (x_1, x_2, x_3) =$

Figura 8: Curva de *trade-off* entre  $\bar{T}$  e  $\sigma_p$  algoritmo AG/hipercubo de localização

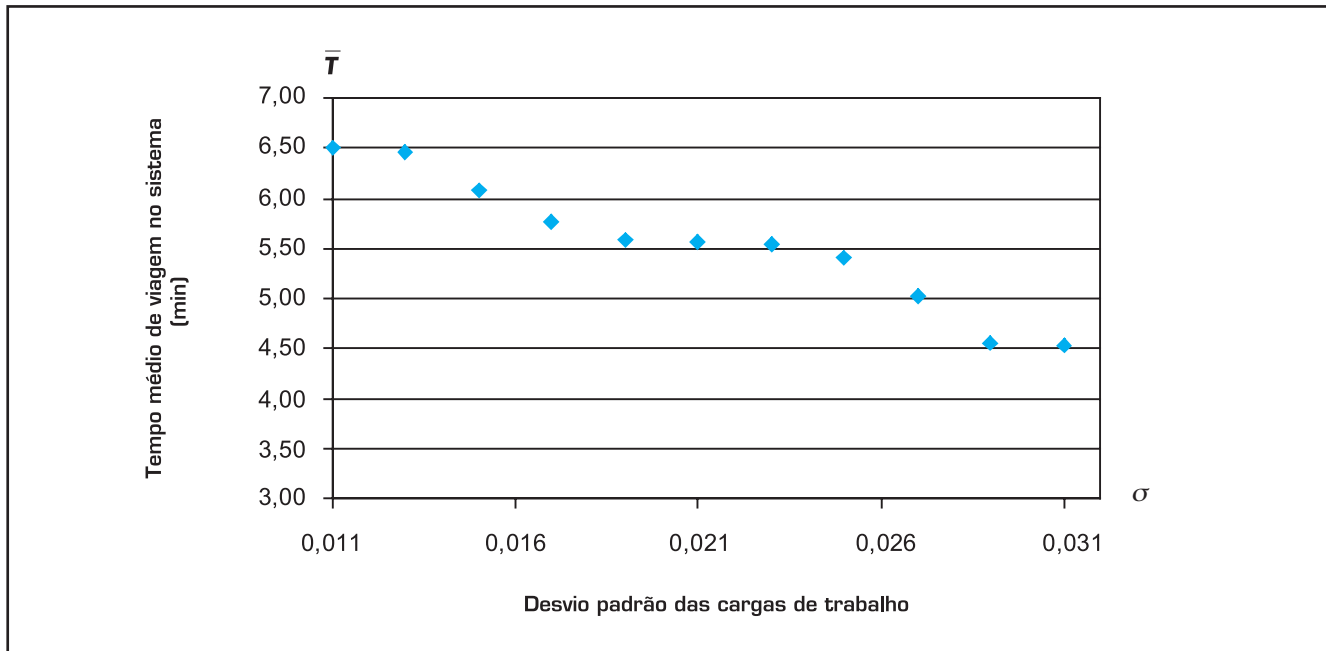


Tabela 5: Resultados do algoritmo AG/hipercubo de localização e dimensionamento, minimizando  $\bar{T}$ .

Solução do passo 1: localização	6,2311 min
Solução do passo 2: dimensionamento	6,1616 min
Melhoria c/relação ao passo 1	1,11%

Tabela 6: Resultados do algoritmo AG/hipercubo de localização e dimensionamento, minimizando  $\sigma_p$ .

Solução do passo 1: localização	0,0218
Solução do passo 2: dimensionamento	0,0152
Melhoria c/relação ao passo 1	30,27%

(0,51, 0,43, 0,57) para o tamanho dos átomos entre as bases adjacentes. Os cromossomos que representam a solução da Tabela 9 são:  $y = (y_1, y_2, y_3, y_4, y_5) = (0,01; 0,89; 0,08; 0,53; 1,0)$  e  $x = (x_1, x_2, x_3) = (0,50, 0,21, 0,48)$ . Também realizamos experimentos com a variação do algoritmo AG/hipercubo de localização e dimensionamento com  $nc = 10$  soluções, minimizando  $\bar{T}$ . Os resultados obtidos nos passos 1 e 2 do algoritmo estão apresentados na Tabela 10. Similarmente ao SAE *Anjos do Asfalto*, a melhor solução do passo 2 (solução 2 - 4,4533) não é gerada pela melhor solução do passo 1 (solução 1 - 4,5117). O algoritmo com  $nc = 1$  consome em média 243 segundos (4,1 minutos) e com  $nc = 10$  consome em média 1168 segundos (19,5 minutos).

## CONCLUSÕES

Neste estudo combinamos variações do modelo hiper-cubo com algoritmos genéticos para otimização da configuração e operação de SAEs em rodovias. Em uma primeira abordagem (algoritmo AG/hipercubo de localização) tratamos da localização das bases de ambulâncias ao longo da rodovia, de forma a minimizar o tempo médio de resposta aos usuários, o desbalanceamento das cargas de trabalho das ambulâncias e a fração de chamadas atendidas em mais que um tempo limite. Dado que estas medidas podem ser conflitantes, mostramos como esta abordagem pode ser adaptada para realizar uma análise

**Tabela 7: Resultados da variação do algoritmo AG/hipercubo de localização e dimensionamento (com  $nc = 10$  configurações no passo 1), minimizando  $\bar{T}$ .**

	SOLUÇÃO DO PASSO 1 (min)	SOLUÇÃO DO PASSO 2 (min)
1	6,2311	6,1616
2	6,2351	6,1632
3	6,2354	6,1577
4	6,2362	6,1589
5	6,2364	6,1661
6	6,2377	6,1665
7	6,2378	6,1574
8	6,2398	6,1487
9	6,2401	6,1695
10	6,2405	6,1548

**Tabela 8: Resultados do algoritmo AG/hipercubo de localização e dimensionamento, minimizando  $\bar{T}$ .**

Solução 1: melhor localização	4,5117 min
Solução final: localização e dimensionamento	4,4561 min
Melhoria c/relação ao passo 1	1,23%

**Tabela 9: Resultados do algoritmo AG/hipercubo de localização e dimensionamento, minimizando  $\sigma_p$ .**

Solução 1: melhor localização	0,01021
Solução final: localização e dimensionamento	0,00972
Melhoria c/relação ao passo 1	4,80%

de *trade-off* e gerar uma fronteira eficiente de Pareto entre estas medidas.

Esta abordagem também foi estendida (algoritmo AG/hipercubo de localização e dimensionamento) para tratar decisões combinadas de localização das bases das ambulâncias e dimensionamento das regiões de cobertura de cada ambulância ao longo da rodovia. Tal abordagem envolve dois passos, sendo que o segundo utiliza o algoritmo AG/hipercubo de dimensionamento apresentado em Iannoni e Morabito (2006a). Resultados computacionais foram apresentados aplicando-se as abordagens em dois estudos de caso: o SAE *Anjos do Asfalto* (com único despacho e *backup* parcial) na rodovia Presidente Dutra, que foi inicialmente estudado por Mendonça e Morabito (2001), e o SAE *Centrovias* (com múltiplo despacho e *backup* parcial) em trechos de rodovias no interior de São Paulo, recentemente estudado em Iannoni e Morabito (2006c).

O estudo mostrou que as principais medidas de desempenho analisadas podem ser melhoradas realocando-se as bases de ambulâncias e variando-se o tamanho das áreas de cobertura das ambulâncias do sistema. As abordagens resolvem em tempos computacionais satisfatórios problemas de tamanho moderado (por exemplo, com menos de 10 ambulâncias). Para problemas maiores, as abordagens, nas suas versões atuais, tornam-se custosas computacionalmente, dado que envolvem resolver muitas vezes modelos hipercubos (modelos 1 e 2) com sistemas lineares de  $O(2^N)$  equações, para avaliar as medidas de desempenho de cada cromossomo. Desta forma, uma perspectiva interessante para pesquisa futura é adaptar e utilizar métodos aproximados para resolver modelos hipercubos, tais como os métodos em Larson (1975), Jarvis (1985) e, mais recentemente, em Atkinson *et al.* (2006, 2007).

**Tabela 10: Resultados da variação do algoritmo AG/hipercubo de localização e dimensionamento (com  $nc = 10$  configurações no passo 1), minimizando  $\bar{T}$ .**

	<b>SOLUÇÃO DO PASSO 1 (min)</b>	<b>SOLUÇÃO DO PASSO 2 (min)</b>
1	4,5117	4,4561
2	4,5131	4,4533
3	4,5196	4,4666
4	4,5198	4,4634
5	4,5209	4,4604
6	4,5210	4,4666
7	4,5228	4,4673
8	4,5231	4,4677
9	4,5245	4,4649
10	4,5252	4,4705

Artigo recebido em 16/11/2006

Aprovado para publicação em 25/10/2007

## Referências

- ARROYO J. C.; ARMENTANO V. A. Genetic local search for multi-objective flowshop scheduling. *European Journal of Operational Research*, 167, p. 717-738, 2005.
- ATKINSON J. B.; KOVALENKO I. N.; KUZNETSOV N.; MYKHALEVYCH K. V. Heuristic solution methods for a hypercube queueing model of the deployment of emergency systems. *Cybernetics and Systems Analysis*, v. 42, n. 3, p. 379-391, 2006.
- \_\_\_\_\_. A hypercube queueing loss model with customer-dependent service rates. *European Journal of Operational Research*. doi:10.1016/j.ejor.2007.08.014, 2007.
- BATTAR.; DOLAN J. M.; KRISHNAMURTHY N. N. The maximal expected covering location problem: Revisited. *Transportation Science*, 23, p. 277-287, 1989.
- BROTCORNE L.; LAPORTE G.; SEMET F. Ambulance location and relocation models. *European Journal of Operational Research*, 147, p. 451-63, 2003.
- CHELST K.; BARLACH Z. Multiple unit dispatches in emergency services: models to estimate system performance. *Management Science*, v. 27, n. 12, p.1390-1409, 1981.
- CHIYOSHI F.; GALVÃO R. D.; MORABITO R. O uso do modelo hipercubo na solução de problemas de localização probabilísticos. *Gestão & Produção*, v. 7, n. 2, p.146-174, 2000.
- CHIYOSHI F.; GALVÃO R. D.; MORABITO R. A note on solutions to the maximal expected covering location problem. *Computers and Operations Research*, v. 30, n. 1, p. 87-96, 2003.
- COHON J. L. *Multiobjective programming & planning*. New York: Academic Press, 1978.
- COSTA D. M. B. Uma metodologia iterativa para determinação de zonas de atendimento de serviços emergenciais. Universidade Federal de Santa Catarina – UFSC. Tese (doutorado em Engenharia de Produção) – Departamento de Engenharia de Produção, 2004.
- GALVÃO R. D.; CHIYOSHI F.; MORABITO R. Towards unified formulations and extensions of two classical probabilistic location models. *Computers & Operations Research*, n. 32, p. 15-33, 2005.
- GOLDBERG D. E. *Genetic Algorithms in Search, Optimization and Machine Learning*. New York: Addison Wesley, 1989.
- HOLLAND J. H. *Adaptation in Natural and Artificial Systems*. Cambridge, MA: MIT Press, 1975.
- IANNONI A. P.; MORABITO R. Modelo hipercubo integrado a um algoritmo genético para análise de sistemas médicos emergenciais em rodovias. *Gestão & Produção*, v. 13, n. 1, p.93-104, 2006a.
- IANNONI A. P.; MORABITO R. A discrete simulation analysis of a logistics supply system. *Transportation Research E*, v. 42 n. 3, p. 191-210, 2006b.
- IANNONI A. P.; MORABITO R. Modelo de fila hipercubo com múltiplo despacho e backup parcial para análise de sistemas de atendimento médico emergenciais em rodovias. *Pesquisa Operacional*, v. 26, n. 3, p. 493-519, 2006c.
- JASZKIEWICZ A. Genetic local search for multi-objective combinatorial optimization. *European Journal of Operational Research*, n. 137, p. 50-71, 2002.
- JARVIS J.P. Approximating the equilibrium behavior of multi-server loss systems. *Management Science*, n. 31, p. 235-239, 1985.
- LARSON R.C. A Hypercube queueing model for facility location and restricting in urban emergency services. *Computers and operations research*, n. 1, p. 67-95, 1974.
- LARSON R. C. Approximating the performance of urban emergency service systems. *Operations Research*, n. 23, p. 845-868, 1975.
- LARSON R. C.; ODONI A. R. *Urban operations research*. New Jersey: Prentice Hall, 1981.
- LARSON R.C. OR models for homeland security. *OR/MS Today*, n. 31, p. 22-29, 2004.
- MENDONÇA F. C.; MORABITO R. Analyzing emergency service ambulance deployment on a Brazilian highway using the hypercube model. *Journal of the Operation Research Society*, n. 52, p. 261-268, 2001.
- MICHALEWICZ Z. *Genetic Algorithms + Data Structures = Evolution Programs*, 3. ed. Berlin: Springer-Verlag, 1996.
- OWEN, S. H.; DASKIN, M. S. Strategic facility location: A review. *European Journal of Operational Research*, n. 111, p. 423-447, 1998.
- RAJAGOPALAN H. K.; SAYDAM C.; XIAO J. A multiperiod set covering location model for a dynamic redeployment of ambulances. *Computers & Operations Research*, n. 35, p. 814-826, 2008.
- SAYDAM C.; AYTUG H. Accurate estimation of expected coverage: revisited. *Socio-Economic Planning Sciences*, n. 37, p. 69-80, 2003.
- SWERSEY A. J. *Handbooks in OR/MS*. Amsterdam: Elsevier Science B.V., v. 6, p. 151-200, 1994.
- TAKEDA R. A.; WIDMER J. A.; MORABITO R. Analysis of ambulance decentralization in urban emergency medical service using the hypercube queueing model. *Computers & Operations Research*, v. 34, n. 3, p. 727-741, 2007.

## Agradecimentos

Os autores agradecem aos revisores anônimos pelos úteis comentários e sugestões, e o apoio financeiro da FAPESP (processo: 05/53126-5) e do CNPQ (processo: 522973/95-4).

## Sobre os autores

### Ana Paula Iannoni

E-mail: aiannoni@dep.ufscar.br

### Reinaldo Morabito

E-mail: morabito@ufscar.br

Departamento de Engenharia de Produção  
Universidade Federal de São Carlos  
End.: Rod. Washington Luis km 235 – São Carlos – São Paulo – CEP 13565-905  
Tel.: (16) 3351-8236, Fax: (16) 3351-8240