

EXAME DE UM ALGORITMO *DIFFERENTIAL EVOLUTION* PARA ALOCAÇÃO DE TRÁFEGO SUJEITO A CONGESTIONAMENTO

F. F. Cardoso, O. Magela Neto

Programa de Pós-Graduação em Engenharia Elétrica

Universidade Federal de Minas Gerais

31.2270-901 – Belo Horizonte – MG, Brazil

warriorssjufmg@ufmg.br, oriane@dee.ufmg.br

F. R. B. Cruz

Departamento de Estatística, Universidade Federal de Minas Gerais

31.2270-901 – Belo Horizonte – MG, Brazil

fcruz@est.ufmg.br

RESUMO

Neste trabalho examinamos o modelo *system optimum* de Wardrop (SO) para alocação de tráfego em uma rede de transportes. A formulação SO é equivalente a uma situação em que os usuários cooperam entre si, garantindo um funcionamento da rede com custo global mínimo, em termos do tempo total de deslocamento. Tais tempos são modelados por redes de filas gerais dependentes do estado do tipo $M/G/c/c$. Por um lado, os modelos $M/G/c/c$ fornecem expressões para o tempo de deslocamento que podem ser consideradas mais aderentes à realidade, mas por outro tais expressões trazem dificuldades, pela não-convexidade (forma de S), para a resolução do modelo SO. Um algoritmo *differential evolution* (DE) é então proposto para lidar com este problema. Mostramos, por meio de um planejamento de experimentos simples, que os principais parâmetros do algoritmo DE podem ser determinados de forma a garantir um desempenho eficiente.

PALAVRAS CHAVE. Logística e transportes, congestionamento, redes de filas.

ABSTRACT

In this paper, we examine the Wardrop system optimum (SO) model for traffic assignment in transportation networks. The SO formulation is equivalent to a situation in which users cooperate with each other in order to ensure the minimum overall cost in terms of total travel time. The travel times are modeled as $M/G/c/c$ state-dependent queueing networks. The $M/G/c/c$ model may be arguably more adherent to reality but will increase the difficulty to solve the SO model because of its non-convexity (an S -shaped form). A differential evolution (DE) algorithm is then proposed to deal with the SO problem. We show that the main parameters of the DE algorithm may be determined by means of a simple set of experiments to guarantee an efficient performance.

KEYWORDS. Logistics and transportation, congestion, queueing networks.

1. Introdução

Os problemas de alocação de tráfego ocorrem em várias situações encontradas na sociedade atual. Um exemplo interessante é o de um sistema para se produzir um automóvel. É necessário um grande número de estações de trabalho, como o almoxarifado, para armazenamento de materiais, as diversas linhas de montagem, o centro de inspeção do controle de qualidade e o pátio de armazenagem final, entre outras. Um serviço lento em qualquer uma dessas etapas pode reduzir a utilização de outros centros de trabalho ao longo da linha de produção levando à sua ociosidade (*starvation*) enquanto espera-se pela próxima entrada. Uma maior eficiência no roteamento pode evitar os ramos congestionados e aproveitar melhor aqueles folgados, ajudando a reduzir importantes medidas de desempenho, tais como o tempo de trabalho (*sojourn time*), o trabalho em processo (*work in process*), entre outras.

Na literatura sobre problemas de alocação de tráfego em uma rede congestionada (Prashker & Bekhor, 2000), dois modelos são comumente empregados, o modelo *system optimum* de Wardrop (SO) e o modelo *user equilibrium* (UE). Entretanto, neste artigo estamos interessados apenas no modelo SO, que assume que todos os usuários são capazes de (ou forçados a) cooperar entre si, a fim de minimizar os custos globais de viagem. O modelo SO pode ser formulado como se segue.

(SO):

$$\min z(\mathbf{x}) = \sum_a x_a c_a(x_a),$$

sujeito a:

$$\sum_k f_k^{rs} = q^{rs}, \quad \forall r, s,$$

$$x_a = \sum_r \sum_s \sum_k f_k^{rs} \delta_{ak}^r, \quad \forall a,$$

$$f_k^{rs} \geq 0, \quad \forall k, r, s,$$

definido em um grafo $G = (N, A)$, em que N é o conjunto de nós; A , o conjunto de arcos; $R \subseteq N$, conjunto de nós origem; $S \subseteq N$, conjunto de nós destino; K_{rs} , o conjunto de caminhos que conecta o par origem-destino $r-s$ (com $r \in R$, $s \in S$); x_a , o fluxo no arco a , com $\mathbf{x} = (\dots, x_a, \dots)$; $c_a(x_a)$, o tempo de viagem no arco a , com $\mathbf{c}(\mathbf{x}) = (\dots, c_a(x_a), \dots)$; f_k^{rs} , o fluxo no caminho k que conecta o par origem-destino $r-s$, com $\mathbf{f}^{rs} = (\dots, f_k^{rs}, \dots)$ e $\mathbf{f} = (\dots, \mathbf{f}^{rs}, \dots)$; c_k^{rs} , o tempo de viagem no caminho k que conecta o par origem-destino $r-s$, com $\mathbf{c}^{rs} = (\dots, c_k^{rs}, \dots)$ e $\mathbf{c} = (\dots, \mathbf{c}^{rs}, \dots)$; q^{rs} , a demanda entre origem r e destino s , com $(\mathbf{q})^{rs} = q^{rs}$; $\delta_{a,k}^{rs}$ é uma variável indicadora que assume o valor 1, se o arco a está no caminho k

entre o par origem-destino r - s , ou assume o valor 0, caso contrário.

Pode-se demonstrar que a solução ótima é atingida quando são iguais os custos *marginais* de viagem, em cada caminho que transporta algum fluxo positivo, o que em outras palavras equivale a:

$$f_k^{rs} (g_k^{rs} - g^{rs*}) = 0, g_k^{rs} - g^{rs*} \geq 0, \forall r, s,$$

em que g_k^{rs} é o custo marginal na rota k e g^{rs*} é o custo marginal ótimo, ambos entre o par origem-destino r - s . Pesquisadores na área de atribuição de tráfego têm sido bem sucedidos na resolução do modelo SO e similares, por meio de muitos algoritmos. Resultados animadores foram relatados tanto com algoritmos exatos duais (Hearn & Lawphongpanich, 1990), paralelos (Ho, 1990) e baseados em relaxação lagrangeana (Larsson & Patriksson, 1995), como também com heurísticas (Ceylan & Bell, 2005), incluindo algoritmos evolucionários (Cruz et al., 2010).

```

algoritmo

  leia grafo,  $G(N, A)$ 
  leia taxas de chegadas,  $\lambda_n, \forall n \in N$ 
  leia comprimento dos arcos,  $l_a, \forall a \in A$ 

  repita

    /* gerar probabilidades de roteamento */
    gerar  $p_a, \forall a \in A$ 
    /* calcular fluxo e tempo de percurso */
    calcular  $x_a, c_a(x_a), \forall a \in A$ 
    /* calcular função objetivo */
    calcular  $\sum_a x_a c_a(x_a)$ 

  até convergência ser atingida

  escrever  $p_a^{opt}, \forall a \in A$ 

fim algoritmo

```

Figura 1: Algoritmo para alocação de tráfego pelo modelo SO

O algoritmo da Figura 1 é baseado em um esquema simples e de fácil implementação. O algoritmo encontra iterativamente as probabilidades de roteamento p_a e a medida de desempenho da rede (tempo de percurso), para a minimização da função objetivo (tempo de percurso total). A heurística de otimização necessita estimar os fluxos x_a e os tempos de viagem correspondentes $c_a(x_a)$. Para encontrar estas estimativas, utilizaremos um modelo de redes de filas finitas recentemente proposto na literatura (Cruz et al., 2010), conforme descrito a seguir.

Na próxima seção, apresentaremos os conceitos fundamentais sobre a análise de

desempenho em redes de tráfego e o algoritmo de otimização utilizado. Em seguida apresentaremos alguns resultados computacionais obtidos com o algoritmo. Encerramos com conclusões e algumas observações finais.

2. Materiais e Métodos

Nosso objetivo principal é o estudo de problemas de alocação de tráfego em que os custos de viagem sejam modelados pela teoria de filas finitas e a alocação ótima seja encontrada por meio de algoritmos heurísticos. Serão utilizadas aqui filas do tipo $M/G/c/c$ dependentes do estado, para a modelagem das redes de tráfego. De acordo com a conhecida notação de Kendall (1953), M representa um processo de chegadas markoviano, G , tempos de serviço com distribuição geral (e, neste caso, dependentes do estado), c é o número de servidores em paralelo e o c final é a capacidade total do sistema, incluindo aqueles itens em serviço, o que é equivalente a dizer que não há áreas de espera (do inglês, *buffers*). O algoritmo heurístico de otimização utilizado é o conhecido *differential evolution* (DE), adaptado especialmente para o problema SO.

2.1 Modelagem de um Arco Único

Uma fila $M/G/c/c$ dependente do estado pode ser considerada adequada para descrever uma via simples em uma rede de transporte (Yuhaski & Smith, 1989). De fato, resultados empíricos comprovam que o tempo de serviço em uma rede de tráfego depende do número de usuários presentes no sistema (Yuhaski & Smith, 1989). O número de servidores pode ser considerado igual à capacidade do sistema (número de clientes que ele suporta). Logo, não existe espaço de espera.

As probabilidades limites para o número aleatório N de entidades em um modelo de fila $M/G/c/c$ dependente do estado, $p_n = \Pr[N = n]$, são dadas por:

$$p_n = \left\{ \frac{[\lambda E[T_1]]^n}{n! f(n) f(n-1) \cdots f(2) f(1)} \right\} / \left\{ 1 + \sum_{i=1}^c \frac{[\lambda E[T_1]]^i}{i! f(i) f(i-1) \cdots f(2) f(1)} \right\},$$

em que $n = 1, 2, \dots, c$, considerando c a capacidade do sistema, λ é a taxa de chegada, $E[T_1] = l/V_1$ é o tempo de serviço esperado para um veículo que trafega sozinho em um arco de comprimento l , considerando que V_1 é a velocidade de um veículo trafegando livre (sem redução na velocidade, causada por congestionamento) e $f(n) = V_n/V_1$ é a taxa de serviço, definida como:

$$f(n) = \exp \left[- \left(\frac{n-1}{\beta} \right)^\gamma \right], \text{ com } \gamma = \left[\frac{\log(V_a/V_1)}{\log(V_b/V_1)} \right] / \log \left(\frac{a-1}{b-1} \right) \text{ e } \beta = \frac{a-1}{[\log(V_1/V_a)]^{1/\gamma}}.$$

Os valores a e b são pontos arbitrários usados para ajustar a curva exponencial. Em aplicações relacionadas ao tráfego de veículos, valores comuns são $a = 20lw$ e $b = 140lw$, que correspondem às densidades de 20 e 140 veículos/milha-pista respectivamente. Valores razoáveis para estes pontos são $V_a = 48$ mph e $V_b = 20$ mph, utilizados com sucesso anteriormente (Cruz et al., 2010).

A capacidade do sistema é dada por $c = \lfloor klw \rfloor$, em que l é o comprimento do arco (em milhas), w é a largura (em número de pistas), k é a capacidade deste arco (por unidade de comprimento por pista) e $\lfloor x \rfloor$ é o maior inteiro não superior a x . Para aplicações relacionadas ao tráfego de veículos, k representa o parâmetro de densidade de tráfego (em veículos/milha-pista) e situa-se normalmente entre 185 e 265.

Finalmente, as medidas de desempenho podem ser deduzidas diretamente das probabilidades limite p_n . São elas a probabilidade de bloqueio, $p_c = \Pr[N = c]$, a vazão, em veículos/h, $x_a \equiv \theta = \lambda(1-p_c)$, o número esperado de veículos no arco, $L = \sum_{n=1}^c np_n$, e o tempo de serviço esperado (em horas), $c_a(x_a) \equiv W = L/\theta$, aqui obtido pela Lei de Little.

2.2 Modelagem de Redes de Filas

Note-se que o problema está resolvido apenas parcialmente. A dedução de medidas de desempenho para filas $M/G/c/c$ dependentes do estado configuradas em *redes* é uma tarefa significativamente mais complexa, devido às probabilidades de roteamento que irão definir a entrada em cada fila e por causa dos efeitos de bloqueio mútuo. O método da expansão generalizado (GEM, do inglês *generalized expansion method*) é uma ferramenta que foi proposta por Kerbache & Smith (1987), com a finalidade de estimar aproximadamente medidas de desempenho de redes de filas finitas. O método tem uma longa tradição nesta área e é uma combinação de métodos de tentativas repetidas e decomposição nó a nó, com a característica chave de adicionar um nó de espera artificial precedendo cada fila finita na rede, a fim de registrar consumidores bloqueados que tentam entrar neste nó finito, mas que são impedidos porque esta fila está com a sua capacidade esgotada. Maiores detalhes sobre o GEM não serão fornecidos aqui, mas podem ser obtidos na literatura (Cruz et al., 2010).

2.3 Algoritmo de Otimização

O *differential evolution* (DE) é uma versão de algoritmo genético, da classe dos

algoritmos evolucionários (AE), que são baseados no princípio da sobrevivência do mais apto. Isto é basicamente um algoritmo de otimização e busca computadorizada baseada em populações. O DE difere de um AE na forma de uma mutação diferencial. Em um algoritmo evolucionário a mutação é baseada na saída de uma função de distribuição pré-definida, enquanto o DE usa a diferença de vetores-objeto aleatoriamente amostrados. Ao invés de usar apenas informação local de cada vetor-objeto (indivíduos da população), o DE muta todos os vetores-objeto com a mesma probabilidade. Deste modo o espaço de busca é coberto na tentativa de encontrar um ótimo global. Um algoritmo DE simplificado pode ser visto na Figura 2.

```

algoritmo
    /* inicializar parâmetros */
    D – dimensão do problema
    NP, F, CR – parâmetros de controle
    GEN – condição de parada
    L, H – limite das restrições
    /* inicializar população */
    Popij ← randij[L, H]
    /* avaliar adequabilidade */
    Fitj ← f(Popj) ∀ j
    para g = 1 até GEN faça
        para j = 1 até NP faça
            /* escolher índices aleatórios */
            r1,2,3 ∈ [1, ..., NP], r1 ≠ r2 ≠ r3 ≠ j
            /* criar indivíduo tentativa */
            X ← S(r, F, CR, Pop)
            /* verificar limites das restrições */
            se (xi ∉ [L, H]) então
                xi ← rand[L, H]
            fim se
            /* selecionar melhor solução */
            se (f(Xj) > Fitj) então
                Popj ← Xj
                Fitj ← f(Xj)
            fim se
        fim para
    fim se
fim algoritmo

```

Figura 2: Algoritmo *differential evolution* simplificado

As seguintes características do DE, válidas para problemas de otimização de espaço contínuo (Storn & Price, 1997), o justificam como um método de solução apropriado para a resolução do modelo SO: é simples, rápido e robusto; possui capacidade de otimização global elevada; pode ser implementado facilmente em um ambiente de computação paralela, o que acelera o processo de otimização; é efetivo em otimização não linear e pode ser facilmente adaptado para otimização com parâmetros mistos; não requer uma função objetivo diferenciável; opera em superfícies planas; pode

prover múltiplas soluções em uma única execução.

O DE é definido como um método de busca direta e paralela que opera em uma população P_G de tamanho constante que é associada com cada geração G e consiste de NP vetores, ou soluções candidatas, X_{pG} , $p = 1, 2, \dots, NP$. Cada vetor X_{pG} consiste de D variáveis de decisão $X_{o,p,G}$, com $o = 1, 2, \dots, D$. A variação dos parâmetros do DE é definida pela notação $DE/x/y/z$ (Price & Storn, 2010), em que x especifica o vetor a sofrer mutação (*rand*, vetor escolhido aleatoriamente na população, ou *best*, o vetor com menor custo da população atual), y é o número de vetores-diferença usados e z especifica o esquema de cruzamento (*bin*, cruzamento através de experimentos binomiais independentes, ou *exp*, cruzamento através de experimentos exponenciais).

3. Experimentos Computacionais e Discussão dos Resultados

Todos os algoritmos descritos anteriormente foram codificados em C++, pela sua eficiência e flexibilidade. O programa está disponível para fins educacionais e de pesquisa, com os autores ou diretamente de Cardoso (2010). Os experimentos foram conduzidos em um computador pessoal, rodando o sistema operacional Windows© Vista.

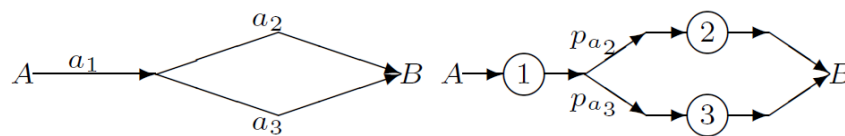


Figura 3: Rede de três trechos e o respectivo modelo $M/G/c/c$

A rede utilizada nos nossos experimentos pode ser vista na Figura 3. Esta é uma rede bastante simples, composta de três trechos de estrada. Os pontos A e B são conectados pelo trecho a_1 e dois trechos alternativos, a_2 e a_3 . Uma destas rotas é mais longa (e de percurso mais demorado) do que a outra. O ajuste do modelo $M/G/c/c$ dependente do estado é apresentado na Tabela 1. O algoritmo foi rodado para a rede de três trechos (Figura 3), para diferentes taxas de chegada λ . Os resultados da alocação de tráfego não serão apresentados aqui, por fugir aos nossos propósitos, mas podem ser encontrados em Cardoso et al. (2010).

As Tabelas 2 a 4 exibem resultados para a variação dos parâmetros do algoritmo DE para a rede da Figura 3. São exibidas a média e o erro-padrão da média do número total de avaliações de função, para trinta execuções por variação dos parâmetros. É importante ressaltar que todas as execuções atingiram o resultado ótimo para o tempo total de percurso segundo o modelo SO para uma taxa de chegada de 1000 usuários.

Experimentos foram realizados com outras taxas de chegada e os resultados (não apresentados) não alteraram significativamente. Os parâmetros iniciais para o algoritmo de otimização são os seguintes: tamanho da população igual a 8; fator de escala de mutação igual a 1; constante de cruzamento igual a 0,8. Na Tabela 2 ocorre a variação do tamanho da população, NP . Na Tabela 3, mostramos a variação do fator de escala de mutação, F . Finalmente, na Tabela 4, a variação da constante de cruzamento, CR .

Tabela 1: Configuração da rede de três trechos

Rota	Comprimento*	Largura [†]	V_1^{\ddagger}	V_a^{\ddagger}	V_b^{\ddagger}	c (veíc.)	$E[T_1]^{\S}$
a_1	0,80 (0,50)	5	23 (37)	25 (40)	10 (16)	800	0,0320 (115)
a_2	2,50 (1,55)	2	20 (32)	18 (29)	6 (10)	1.000	0,1250 (450)
a_3	1,85 (1,15)	2	20 (32)	18 (29)	6 (10)	740	0,0925 (333)

Obs.: *em milhas (km); [†]em # pistas; [‡]em mph (km/h); [§]em h (s)

Tabela 2: Variação do parâmetro *tamanho da população* para a rede de três trechos

NP	média*	erro-padrão da média*
4	617	59
8	545	52
12	837	29
16	1.114	32
20	1.388	34
24	1.779	38
28	2.049	31
32	2.330	37
36	2.627	45
40	3.029	52

Obs.: *sobre trinta avaliações de função

Tabela 3: Variação do parâmetro *fator de escala de mutação* para a rede de três trechos

F	média*	erro-padrão da média*
0,25	315	12
0,50	343	11
0,75	455	13
1,00	499	48
1,25	626	26
1,50	652	23
1,75	848	29
2,00	897	43

Obs.: *sobre trinta avaliações de função

Pelos resultados apresentados na Tabela 2, podemos afirmar que tamanho da população deve ser pequeno, mas não demais. O menor número de avaliações de função ocorre quando o tamanho da população é de 8 indivíduos, que volta a crescer quando o tamanho da população cai para 4. O aumento do tamanho da população apenas aumenta o número de avaliações de função e, conseqüentemente, o tempo de execução do algoritmo. Quanto ao fator de escala de mutação, F , a menor média foi para o valor 0,25 (Tabela 3). Notamos, porém, que o menor erro-padrão da média ocorreu com o valor do fator um pouco maior, igual a 0,50. Portanto, um fator de escala de mutação um pouco maior (como o valor de 0,50) pode piorar um pouco a média de avaliações, mas pode ser

preferível por fornecer um resultado mais robusto (com menor variabilidade). Finalmente, observamos pela Tabela 4 que a constante de cruzamento deve possuir um valor alto, próximo ao seu limite superior. De fato, a Tabela 4 mostra que o menor número de avaliações de função é obtido para o valor da constante de cruzamento 0,9.

Tabela 4: Variação do parâmetro *constante de cruzamento* para a rede de três trechos

<i>CR</i>	<i>média*</i>	<i>erro-padrão da média*</i>
0,0	1.551	139
0,1	1.209	83
0,2	968	76
0,3	911	47
0,4	719	46
0,5	738	59
0,6	648	47
0,7	547	32
0,8	540	48
0,9	452	23
1,0	453	24

Obs.: *sobre trinta avaliações de função

4. Conclusões e Observações Finais

O modelo *system optimum* de Wardrop (SO) foi tratado aqui por meio de uma expressão estocástica não usual para o tempo de percurso. Esta expressão é baseada em um modelo de filas finitas do tipo $M/G/c/c$ dependente do estado. Esta nova forma de modelar o tempo de percurso traz vantagens sobre as expressões comumente utilizadas, com possivelmente uma melhor aderência à realidade, por modelar os efeitos de congestionamento (isto é, a redução no tempo de percurso com o aumento do congestionamento nos trechos da rede). Por outro lado, a função resultante para o tempo de percurso tem formato S (Cardoso et al., 2010), o que traz novos desafios para o desenvolvimento de algoritmos de otimização do modelo SO, pela possibilidade de ótimos locais. Resultados computacionais aqui apresentados atestam que heurísticas do tipo *differential evolution* (DE) podem ser uma ferramenta efetiva na resolução de problemas SO. Um planejamento de experimentos simples possibilitou a determinação dos parâmetros que garantem a eficácia do algoritmo DE

Possíveis direções para trabalhos futuros nesta área de pesquisa incluem a análise de redes mais complexas e a aplicação dos algoritmos na modelagem de redes de pedestres, uma vez que também se aplicam ao tráfego de pedestres muitas das características da função de tempo de percurso aqui tratada.

Agradecimentos

Este trabalho contou com apoio do CNPq (processos 553019/2009-0 e 303388/2010-2) e da FAPEMIG (processo CEX-PPM-00390-10).

Referências

- Cardoso, F. F.** (2010). *Problemas de alocação de tráfego sujeitos a congestionamento*, Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia Elétrica – UFMG, Belo Horizonte, MG. (disponível em <http://www.cpdee.ufmg.br/defesas/68M.PDF>)
- Cardoso, F. F., Magela Neto, O. & Cruz, F. R. B.** (2010). Computational experience with a state-dependent traffic assignment problem, *Proceedings of 'The IV Southern Conference on Computational Modeling - IV MCSUL'*, Rio Grande, Brazil, 172-177. (disponível em <http://www.est.ufmg.br/ftp/fcruz/publics/mcsul-so.pdf>)
- Ceylan, H. & Bell, M. G. H.** (2005). Genetic algorithm solution for the stochastic equilibrium transportation networks under congestion, *Transportation Research Part B*, 39: 169-185.
- Cruz, F. R. B., van Woensel, T., Smith, J. M. & Lieckens, K.** (2010). On the system optimum of traffic assignment in $M/G/c/c$ state-dependent queueing networks, *European Journal of Operational Research*, 201(1): 183-193.
- Hearn, D. W. & Lawphongpanich, S.** (1990). A dual ascent algorithm for traffic assignment problems, *Transportation Research Part B*, 24(6): 423-430.
- Ho, J. K.** (1990). Solving the dynamic traffic assignment problem on a hypercube multicomputer, *Transportation Research Part B*, 24(6): 443-451.
- Kendall, D. G.** (1953). Stochastic processes occurring in the theory of queues and their analysis by the method of imbedded Markov chains, *Annals Mathematical Statistics*, 24: 338-354.
- Kerbache, L. & Smith, J. M.** (1987). The generalized expansion method for open finite queueing networks, *European Journal of Operational Research*, 32: 448-461.
- Larsson, T. & Patriksson, M.** (1995). An augmented Lagrangean dual algorithm for link capacity side constrained traffic assignment problems, *Transportation Research Part B*, 29(6): 433-455.
- Prashker, J. N. & Bekhor, S.** (2000). Some observations on stochastic user equilibrium and system optimum of traffic assignment, *Transportation Research Part B*, 34: 277-291.
- Storn, R. & Price, K.** (1997). Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces, *J. of Global Optimization*, 11(4): 341-359.
- Yuhaski, S. J. & Smith, J. M.** (1989). Modeling circulation systems in buildings using state dependent models, *Queueing Systems*, 4: 319-338.