



OTIMIZAÇÃO COM ALGORITMO BIO-INSPIRADO DE CONTROLE DE TRÁFEGO EM SISTEMAS DE GRUPOS DE ELEVADORES

Msc. Juan P. Diago R.¹, Dr. Guilherme C. de Carvalho¹, Dr. Daniel M. Muñoz A¹.

(1) Faculdade de Tecnologia, Dpto. De Engenharia Mecânica. Programa de Pós-graduação Em Sistemas Mecatrônicos Universidade de Brasília UnB, Brasília DF- Brasil.

E-mail: jpdiago@unb.br, gccarval@unb.br, damaz@unb.br.

Resumo. Este artigo tem como objetivo apresentar a implementação de uma técnica de otimização bioinspirada como solução ao problema de controle de tráfego em sistemas de grupos de elevadores (EGCS). A técnica de controle usada é o algoritmo de otimização por inteligência de enxame (PSO - swarm optimization particle) de tipo binário. A ideia é que o algoritmo escolha o melhor elevador para um usuário que faz uma chamada de serviço em um sistema de controle destino (DCS – destination control system). Para a escolha do elevador o algoritmo tem uma função custo que considera as variáveis: (1) tempo de espera; (2) tempo de voo; (3) capacidade do elevador; (4) número de paradas alocadas; (5) número de paradas (baseado nas chamadas que são asignadas) para cada elevador. Estes parâmetros são ponderados de acordo com sua importância e inferência na seleção do melhor elevador. Assim, o sistema seleciona de todas as possíveis soluções encontradas a solução que apresente o melhor valor de aptidão (a solução representa o elevador ou os elevadores selecionado para atender a atual chamada).

Abstract. *This article has the objective present the results of implementing an Elevator Group Control System (EGCS). The technique of control used is the optimization algorithm based on swarm intelligence known as PSO (Particle Swarm Optimization). The main idea for process selection the best elevator for attending the hall calls in the building. It will make use of a DCS (destination control system), which provides the control system with an a-priori knowledge of for a service requests. The system elevator group has represented for a cost function, this function was build based on use of system parameters relevant, such as: (1) wait time; (2) travel time; (3) maximum attendant call time; (4) available capacity elevator; (5) number stops (based in the call allocated) for each elevator. These parameters were weighted according to their importance and inference in select a best solution. Thus, the system selects of the all possible found a solutions with a best cost value function (the solution represents the elevator or elevators that serve the present call).*

Keywords: EGCS elevator group control system, PSO Particle Swarm Optimization, DCS destination control system, Function cost.

1 INTRODUÇÃO

Desde o século passado tem havido um acelerado aumento da propriedade horizontal (prédios de vários andares) como resultado da intenção do homem em otimizar o espaço nas cidades, o incremento gera o problema de transporte vertical (Cortés, Larrañeta, Onieva, Muñuzuri, & Fernández, 2002), que consiste na necessidade de ter um meio adequado e eficaz que facilite o transporte nesse tipo de prédios. No início foi solucionada pelo uso das escadas (incluindo as elétricas da atualidade), até o momento em que essa alternativa foi se tornando rapidamente inadequada como solução quando os prédios apresentavam mais que cinco andares e dispunham de grande quantidade de pessoas se movimentando na edificação. As principais desvantagens do transporte vertical baseado em escadas são: tempos de deslocamento entre andares elevados, desconforto dos usuários, fadiga física das pessoas, probabilidade de acidentes, lesões físicas nas pernas, além de outros fatores próprios de cada pessoa tais como idade, uso de cadeira de rodas e limitações físicas

Desse modo, a necessidade de transporte foi solucionada com o uso do elevador, entendendo que ao se aumentar o tamanho do prédio (andares, capacidade, população) pode ser necessário o uso de sistemas de grupos de elevadores (três ou mais elevadores em um prédio) (Barney, 2003) (Markon K. H., 2006). Esses sistemas envolvem diferentes tipos de tecnologias no campo da instrumentação, controle e comunicação para a supervisão e intercâmbio de informações referentes ao funcionamento dos elevadores. Tal conjunto de tecnologias pertence a área denominada de automação predial (Barney, 2003).

Uma vez que as soluções para o problema de transporte vertical ainda não estão definidas, surge uma oportunidade interessante de pesquisa neste campo, pensando em uma possível solução com o uso de novas técnicas de controle e automação. Dessa forma, este trabalho apresenta os resultados obtidos com as implementadas com algoritmos de otimização bioinspirados como possível solução para fazer o controle de sistemas de grupos de elevadores (*EGCSs – Elevator Group Control Systems*) (Fernandez, Cortes, & Delgado, 2012).

2 FUNDAMENTACAO TEORICA

2.1 Sistemas de grupos de elevadores

Considera-se um sistema de grupo de elevadores quando em um prédio qualquer, há três ou mais elevadores como principal meio de transporte. As características desses sistemas e o seu funcionamento estão relacionados ao tipo de prédio onde são usados (residencial, comercial, hospital, shopping centers, entre outros). Cada tipo de prédio tem características e necessidades de transporte diferentes. Por exemplo, suponha-se um prédio do tipo residencial e outro do tipo comercial (Barney, 2003) (Markon K. H., 2006). No primeiro caso, a demanda do serviço dos elevadores pode ter uma intensidade menor que no segundo, dado que a demanda de transporte no prédio comercial é elevada, por causa da necessidade de uma quantidade maior de pessoas a serem transportadas.

Desse modo, a importância de definir as características dos prédios se foca em incrementar as possibilidades de criar um sistema de controle adequado para cumprir com as necessidades de transporte (tempo de espera adequado), elevando as possibilidades de gerar conforto para os usuários desses sistemas (Barney, 2003). Neste trabalho foi escolhido como

prédio de estudo um de tipo comercial. Este tipo de edificações tem características de alta complexidade, pela intensidade de fluxo de usuários precisando ser transportadas de acordo com a hora do dia a ser analisada. Independente da intensidade do fluxo o sistema de controle tem ter a capacidade de responder de maneira adequada com as necessidades de transporte dos usuários, garantindo que os tempos de serviço estejam de acordo com os entregados pela literatura científica para este tipo de prédios. Estes tempos são a base para determinar o bom funcionamento e prestação de serviço aos usuários, assim, tempos por abaixo dos característicos geram sensação de conforto e tempos por encima dos característicos geram sensação de desconforto e mal serviço. Vide figura 1.

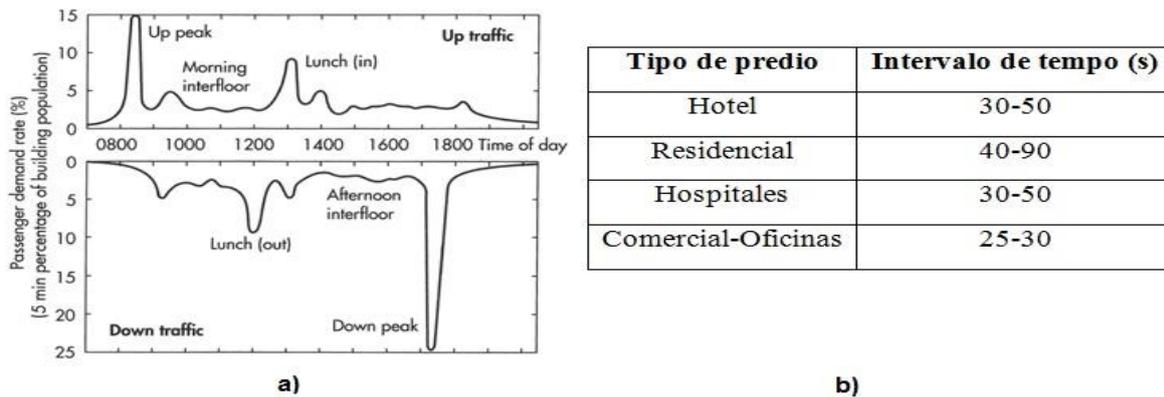


Figura 1 a) padrões de tráfego em prédios comerciais, b) tempos de espera padres em prédios comerciais

É importante também, a definição no Sistema de elevadores a forma como se caracteriza a chegada dos usuários e a forma como eles fazem a solicitude de transporte. No primeiro caso a chegada é representada matematicamente na literatura científica por um comportamento de um processo tipo Poisson, na qual a taxa de chegada é variável dependendo das mudanças de intensidade de fluxo de usuários e relacionada também com os padrões de trafego do prédio estudado (*Up-peak, Interfloor, lunch, Down-Peak*) (Bolata, Altunb, & Cortés, 2013). Por outro lado, para ter um melhor planejamento do funcionamento do sistema de elevadores se faz uso do DCS (*Destination Control System*) para realizar as solicitudes de transporte pelos usuários (Y Zhou, 2004). Este sistemas consiste em paineis que são colocados em cada andar fora dos elevadores, o usuário ingressa neles o andar de destino e espera um tempo corto para que o sistema entregue o número do elevador adequado para se transportar no menor tempo possível. Desta forma, o sistema tem conhecimento a-priori para planejar seu funcionamento. (Vide figura 2).

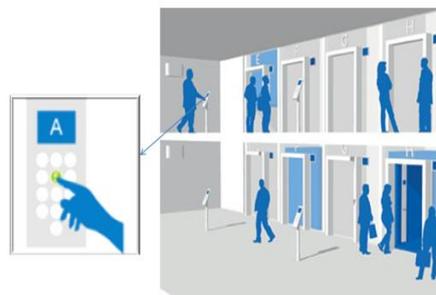


Figura 2 DCS (*Destination Control System*)

2.2 Sistemas de controle de grupos de elevadores

As características dos sistemas de elevadores apresentadas nos tópicos anteriores mostram a importância e a necessidade de se ter um ótimo controle do uso dos elevadores, com o fim de garantir tempos adequados de serviço prestado aos usuários de um prédio em qualquer situação de tráfego. Esta necessidade representa um desafio para os pesquisadores na área da automação predial, encontrando-se na literatura diferentes tipos de técnicas, baseados em técnicas de controle tipo convencional e controladores modernos, procurando sempre a diminuição de tempos de serviço ou a diminuição do consumo de potência.

Este trabalho se foca nas técnicas de controle moderno, na literatura encontram-se trabalhos interessantes que ajudaram na consolidação desta pesquisa (aspectos do conhecimento básico, aspectos de planejamento e abordagem, entre outros), assim pode-se citar os baseados em lógica nebulosa (Jafferi Jamaludin, 2010), Redes Neurais (Jian Liu, 2010), Algoritmos Genéticos (AG), Algoritmos Evolutivos (AE), Programação Evolutiva (PE) y Programação Genética (PG), inspirados basicamente no comportamento da teoria da evolução de Darwin (Atsuya Fujino, 1997) (Bailey, Ombuki-berman, & Asobiela, 2013). Nesta área está a técnica usada nesta pesquisa, a qual está baseada nos algoritmos evolutivos de otimização bioinspirada em inteligência de enxames, mais precisamente o uso do algoritmo *PSO* (*particle swarm optimization*) (R. & J., 1995) na sua versão desenvolvida para trabalhar com problemas discretos, este algoritmo será o encarregado de selecionar o elevador ou elevadores para atender uma chamada recebida.

2.3 BPSO (Binary Particle Swarm Optimization)

A técnica *PSO* (*Particle Swarm Optimization*) é um algoritmo de otimização inspirada no comportamento social do voo dos bandos de pássaros e também no movimento dos cardumes de peixes durante a busca por alimento. Este algoritmo foi desenvolvido em 1995 por dois pesquisadores de duas áreas diferentes o psicólogo James Kennedy e o engenheiro Russell Eberhart (Eberhart & Kennedy, 1995). A inspiração natural deste algoritmo pode ser explicada assim: seja um bando de pássaros que procura uma fonte de alimento em uma área delimitada. No início, os pássaros do bando voam aleatoriamente na área de busca e comunicam-se entre eles quando encontram uma nova fonte de alimento. Ao interagir entre si, o enxame segue o pássaro que esteja mais perto da melhor fonte de alimento. O algoritmo *PSO* emula este comportamento para resolver problemas de otimização, de forma que as características fundamentais para seu funcionamento são o conhecimento individual, representado pelo histórico de cada partícula, e o conhecimento social, representado pelo histórico das partículas vizinhas.

2.3.1 Algoritmo *PSO* básico

O algoritmo *PSO* pode ser descrito, de forma geral, como um conjunto de vetores que contém dados ligados à posição das partículas, as quais se movimentam em uma região ou área definida (vide figura 3) (R Eberhart, 1997). Neste algoritmo, a nova posição de uma partícula é definida por sua experiência particular, conhecida como memória individual (a partícula lembra-se da melhor posição determinada pela avaliação da função custo). Da mesma forma a partícula é influenciada pela experiência global do enxame, conhecida como memória coletiva (a partícula lembra a melhor posição do enxame determinada pela melhor aptidão ao ser avaliada na função custo)

A versão usada nesta pesquisa é a binária, porém o espaço de busca é igualmente binária, na qual as variáveis de decisão assumem dois possíveis valores 1 ou 0. Para fazer a adaptação foi preciso codificar as soluções encontradas como uma solução de tipo binária (R Eberhart, 1997). O novo algoritmo conserva a estrutura do PSO original, as equações de seu funcionamento (vide equação (1)), utilizando uma função sigmóide para normalizar os valores de velocidade de movimento das partículas (vide equação (2)). Para um espaço de busca N dimensional a posição de cada partícula é representada da forma $x = (x_1, x_2, \dots, x_N)$ e os valores possíveis de cada termo são calculados comparando-se o valor da função sigmóide com um valor aleatório r_{ij} no intervalo $[0,1]$ (vide equação(3)).

$$v_{ij}^{(t+1)} = wv_{ij}^{(t)} + c_1U_{1j}(y_{ij}^{(t)} - x_{ij}^{(t)}) + c_2U_{2j}(y_{sj}^{(t)} - x_{ij}^{(t)}) \quad (1)$$

$$v_{ij}^{(t+1)} = sig(v_{ij}^{(t+1)}) = \frac{1}{1 + e^{-v_{ij}^{(t+1)}}} \quad (2)$$

$$x_{ij}(t+1) = \begin{cases} 1 & \text{if } r_{ij} < sig(v'_{ij}(t)) \\ 0 & \text{para valores inferiores} \end{cases} \quad (3)$$

Para este tipo de algoritmo, faz-se a mesma avaliação da qualidade das possíveis soluções em uma função custo, obtendo a aptidão de cada uma delas, das quais o algoritmo seleciona a melhor de todas. Na figura 3 pode se observar o digrama de blocos do pseudocódigo do *BPSO*.

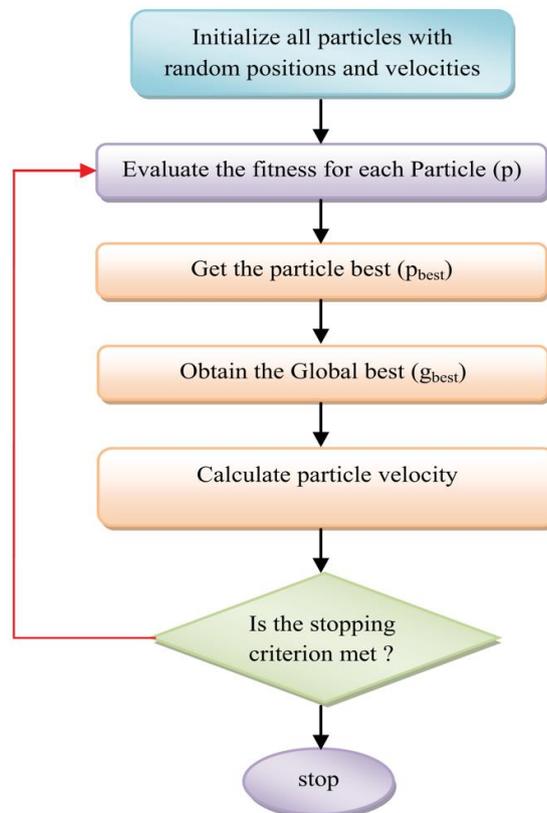


Figura 3 pseudocódigo *BPSO*

3 IMPLEMENTAÇÃO DO SISTEMA DE CONTROLE

3.1 Metodologia

A metodologia empregada neste trabalho consiste na combinação de duas metodologias complementares a *top down* e a *bottom up* (Valentino Crespi, 2007), vide figura 4. Com a metodologia *top down* foi desenvolvida a primeira etapa do trabalho de pesquisa, permitindo realizar um estudo da literatura existente sobre o problema do controle de elevadores e as técnicas usadas como possíveis soluções para ele. Na fase experimental se fez uso da metodologia *bottom up*, foi proposta uma abordagem do problema baseada em: a) geração de chamadas dos usuários; b) construção da função de custo que representa o sistema de elevadores; c) funcionamento do controlador com o algoritmo PSO binário. Para finalmente fazer os testes em um simulador desenvolvido no software *Arena* (ferramenta software da *Rockwell Automation* com vantagens para trabalhar com variáveis estocásticas e com análises de tempos) (Rockwell Automation, 2005), obtendo-se assim os resultados que são usados para a conclusão da pesquisa.

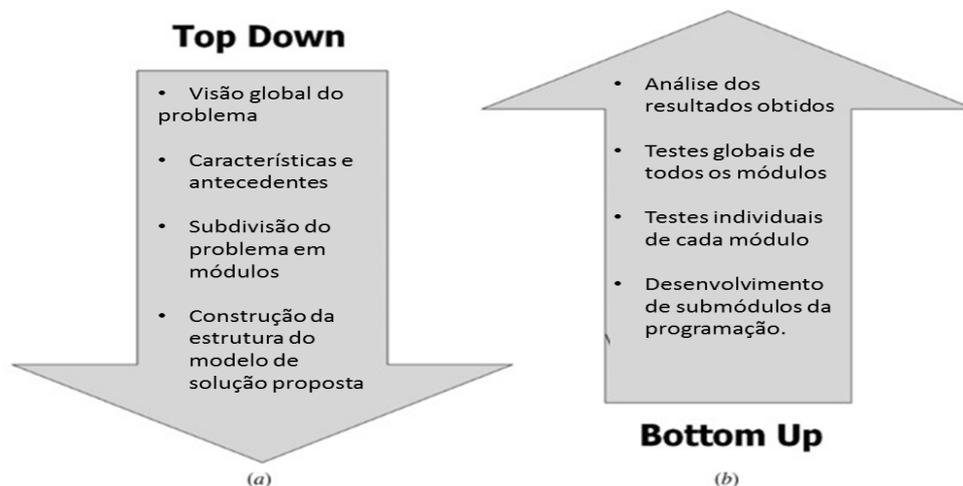


Figura 4 metodologia

3.2 Modelo Desenvolvido

Da aplicação da primeira metodologia foi desenvolvido um modelo no intuito de implementar um simulador de um grupo de elevadores com um controlador baseado no algoritmo de otimização bioinspirado BPSO, que permita avaliar seu funcionamento do controlador num prédio comercial, (vide figura 5). Este modelo pode-se dividir em três partes fundamentais: (1) o sistema gerador de chamadas (capacidade de gerar chamadas dos diferentes padrões de trafego); (2) o sistema de controle *BPSO* incluindo o alocação das chamadas ao elevador selecionado e (3) a dinâmica dos elevadores.

3.3 Gerador de chamadas

É importante se lembrar que o sistema de controle deve ter a capacidade de responder de maneira ótima, isto é, respostas com tempos de serviço de acordo aos padrões (tempos de espera pelo elevador e tempos da viagem ou tempo dentro do elevador para cada chamada gerada), no caso de estudo desta pesquisa se tem um prédio comercial com 4 tipos de padrões

de trafego, o padrão *up peak* é considerado o mais crítico de eles, é será usado para realizar os testes do controlador proposto (Barney, 2003). A escolha deste padrao é baseada na literatura, na qual se tem com o padrao que tem o maior desafio pra os controladores, porem, se um controlador tem a capacidade de atender as exigencias deste padrao o controlador podera responder de forma adequada aos outros padroes.

As características de intensidade do padrão *up peak* podem ser observadas na figura 6. as porcentagens das chamadas neste padrão de tráfego são de 90% *up-peak* y 10% para os outros padrões. Além disso, as chamadas deste tipo de tráfego tem uma característica comum. O andar de origem do 90% é o mesmo (1) e o andar de destino pode ser qualquer dos outros andares.

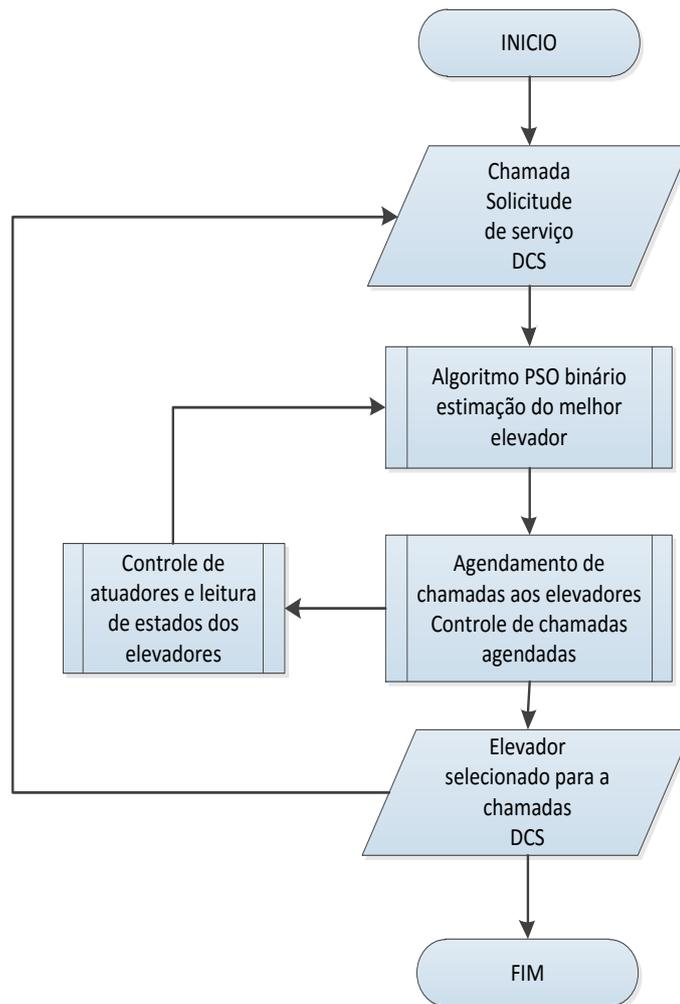


Figura. 5 Módulo do simulador

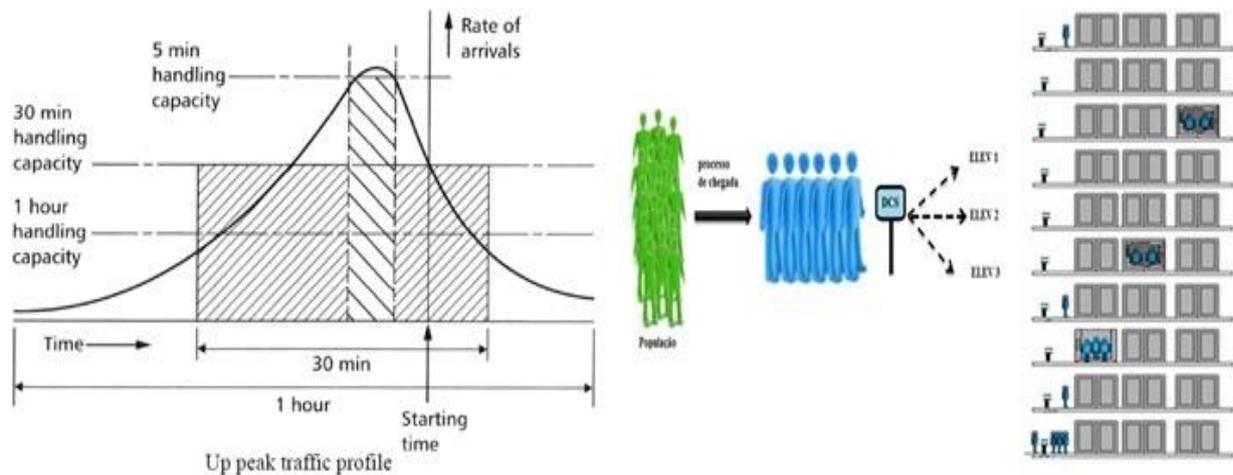


Figura. 6 tráfego *Up-peak*

O programa desenvolvido para simular a chegada de pessoas ao prédio foi simulado no Software Arena. Baseado na distribuição tipo Poisson foram geradas chegadas de usuários e chamadas para um prédio comercial com uma população de 460, e de acordo ao padrão *Up-peak* a intensidade máxima em 5 min é de aproximadamente do 15% desta população, vide figura 7. A taxa de chamadas por tempo é de 0,24 pessoas por segundo, gerando-se 69 chamadas das quais 64 foram do tipo *up-peak* e 5 foram de outros tipos de tráfego, garantindo as porcentagens estipuladas na literatura.

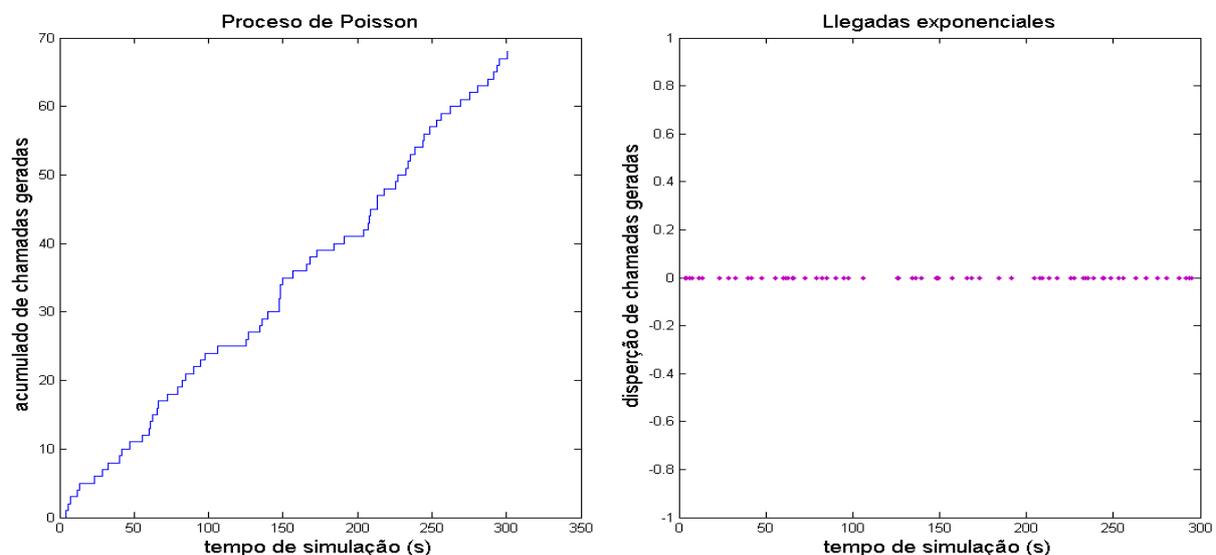


Figura 7 distribuição de chamadas geradas

3.4 Controlador Bioinspirado BPSO

O controlador proposto foi desenvolvido de forma modular para facilitar seu entendimento, vide figura 8.

3.4.1 Funcionamento do BPSO

Para determinar que elevador vai atender uma chamada gerada, foi selecionado o algoritmo BPSO na sua versão binária, pela sua facilidade de implementação, pela pesquisa na literatura e pelos resultados obtidos em testes com este algoritmo e outros similares (*ABC Artificial Bee Colony*, *FA Firefly algorithm*, *SFLA shuffled frog-leaping algorithm*), em funções de prova do tipo *benchmark*. Com tudo, e de acordo ao problema dos elevadores o qual tem uma função custo na lineal e de tipo discreta (tendo em conta que o número de elevadores é discreta ou binária), dependente de parâmetros dinâmicos respeito ao fluxo de passageiros.

Nos sistemas de elevadores a solução pode ser representada como na figura 9. Uma palavra binária na qual os elevadores que vão ser enviados para atender uma chamada tem um valor de 1 e os que tem o valor de 0 são os que não vão a atender essa chamada. Deste modo, o algoritmo BPSO como controlador do sistema será o encarregado de encontrar a melhor combinação binária (o elevador com o menor tempo de espera e menor tempo de viagem), depois de avaliar muitas possíveis soluções numa função custo que representa o sistema de elevadores

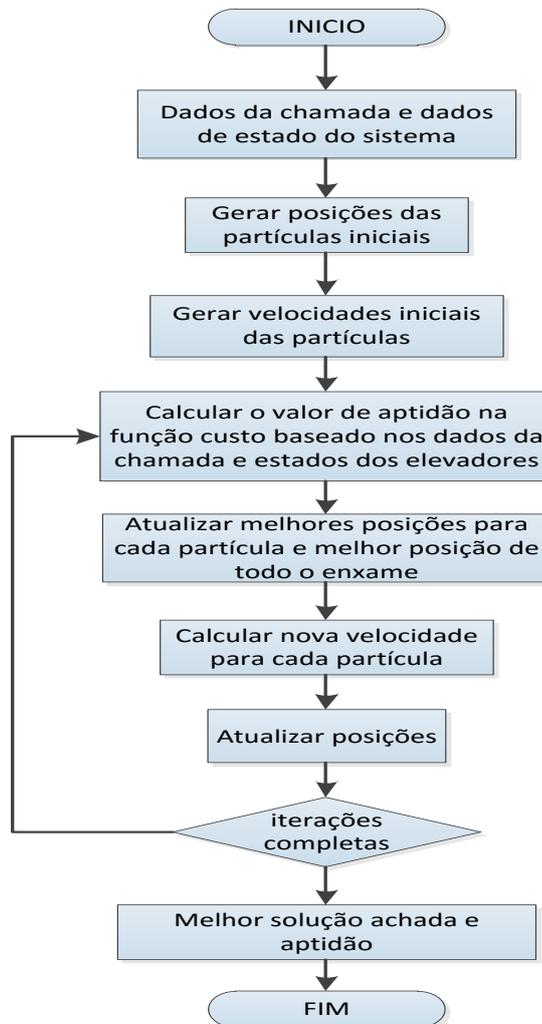


Figura. 8 diagrama do sistema de controle BPSO

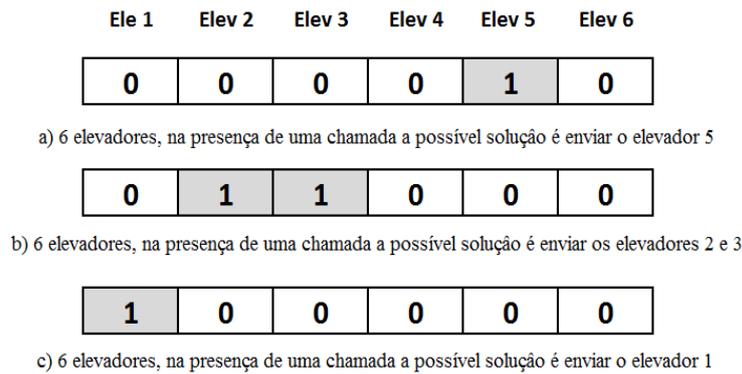


Figura. 9 representação das soluções

3.4.2 Função custo do sistema de grupo de elevadores

Para o desenho da função custo do sistema foram usados fatores relacionados com os tempos de espera e tempos de voo do elevador, assim como também, foram usados outros parâmetros importantes que tem influência nas possibilidades de alocar uma chamada num elevador determinado. Para o qual, a função de custo do sistema se represento como a soma de 6 parâmetros relevantes ponderados (vide equação 4). Esta abordagem de desenho foi feita sem ter em conta o consumo de potência (energia) e só se foco minimização de tempos de transporte.

$$f(i) = k_1WT + k_2JT + k_3MWT + k_4NP + k_5NC + k_6RRI \quad (4)$$

Onde WT : tempo de espera pelo elevador i . JT : tempo de voo do elevador i , (desde o andar origem ao andar de destino). MWT : tempo máximo em atender uma chamada alocada no elevador i . NP : número de pessoas no elevador i . NC : número de chamadas alocadas no elevador i . RRI : número de pessoas esperadas no elevador i quando chega ao andar de origem. A ponderação dos parâmetros da função foi inspirada no trabalho desenvolvido por (Yu, Zhou, Mabu, Hirasawa, Hu, & Markon, 2007), os valores selecionados de cada constante de ponderação foram escolhidos pela sua importância ($k_1=0,4$, $k_2=0,3$, $k_3=0,1$, $k_4=0,08$, $k_5=0,04$ e $k_6=0,08$) com maior valor ou com maior peso as constantes relacionadas com tempos de espera e serviço (WT e JT), a escolha destes valores foi feita pelo médio de ensaio e erro (podem ser otimizados em próximas melhoras da pesquisa). Para o caso de NP , foi decidido ter uma condição especial denominada de penalidade, com o fim de evitar que os elevadores que tenham um número de pessoas dentro dele igual ou perto da sua capacidade, tenham uma possibilidade mais baixa de ser selecionados, para isto o valor de NP é acrescentado de forma considerável.

Para o cálculo de cada parâmetros foram criados vetores com os dados de paradas alocadas em cada andar do prédio, para as chamadas na quais o transporte é para subir como para os quais o transporte é para baixar. Ademais, se tem vetores associados para cada elevador com o número de pessoas que vão ingressar e sair em cada andar do prédio (relacionados diretamente com as chamadas já alocadas). Desta forma, basta com mexer na informação contida nestes vetores e junto com os dados do prédio para calcular os parâmetros da função custo, também, se tem a equação 5, desenvolvida para o cálculo de (WT , JT , MWT), relacionando tempos de transporte (tempos de um andar a outro, tempos de aceleração e desaceleração, e tempos em velocidade constante), e de outra parte, a equação 6 relaciona

os tempos do elevador num andar (tempo de saída y entrada de pessoas, tempo de abertura y fechado de portas).

$$tt_i = Df * (|F_{o-a} - F_d|) / [N_{rc} * V_{ac} + (N_{rc} + 1) * V_{dac} + (|F_{o-a} - F_d| - N_{rc}) * V_{max}] \quad (5)$$

$$WT / JT / MWT = N_{rc} ((T_{ip} + 2 * (ran() - 0,5)) + (T_{op} + 2 * (ran() - 0,5)) + (T_{od} + T_{dc})) + (N_{rc} + 1) * T_{op} + tt_i \quad (6)$$

Onde: tt_i = tempos de transporte do elevador i ; Df = a distância entre os andares do prédio; F_o = andar de origem da chamada analisada; F_a = andar atual do elevador i (o uso de estas variáveis depende do parâmetro a ser calculado WT e MWT). F_d = andar de destino da chamada analisada, N_{rc} = número de paradas entre o andar de origem e o andar de destino, ou número de andares entre el piso atual e o andar de origem; V_{ac} = velocidade de aceleração do elevador; V_{dac} = velocidade de desaceleração do elevador, V_{max} = velocidade máxima quando o elevador vai de um andar para o outro (constante). T_{ip} = tempo que vai demorar uma pessoa em entrar ao elevador, a qual vai acompanhada de um término de tempo aleatório emulando a demora de uma pessoa em entrar (pessoa falando, caminhando divaga). T_{op} = tempo que demora uma persona em sair do elevador, também acompanhada de um termino de tempo aleatório emulando uma demora adicional na saída do elevador. T_{od} = tempo que o elevador demora em abrir as portas. T_{dc} = tempo que o elevador demora em fechar as portas.

3.5 Cenários possíveis no momento de ser gerada uma chamada.

Analisando as possíveis situações nas que pode se encontrar um só elevador quando é gerada uma chamada, considerem-se 8 possíveis cenários, dos quais 6 de eles podem observar-se na figura 10. Onde os pontos vermelhos representam a chamada atual, os pontos azules são as chamadas alocadas, a flecha vermelha mostra a direção da chamada e da flecha azul da direção do elevador.

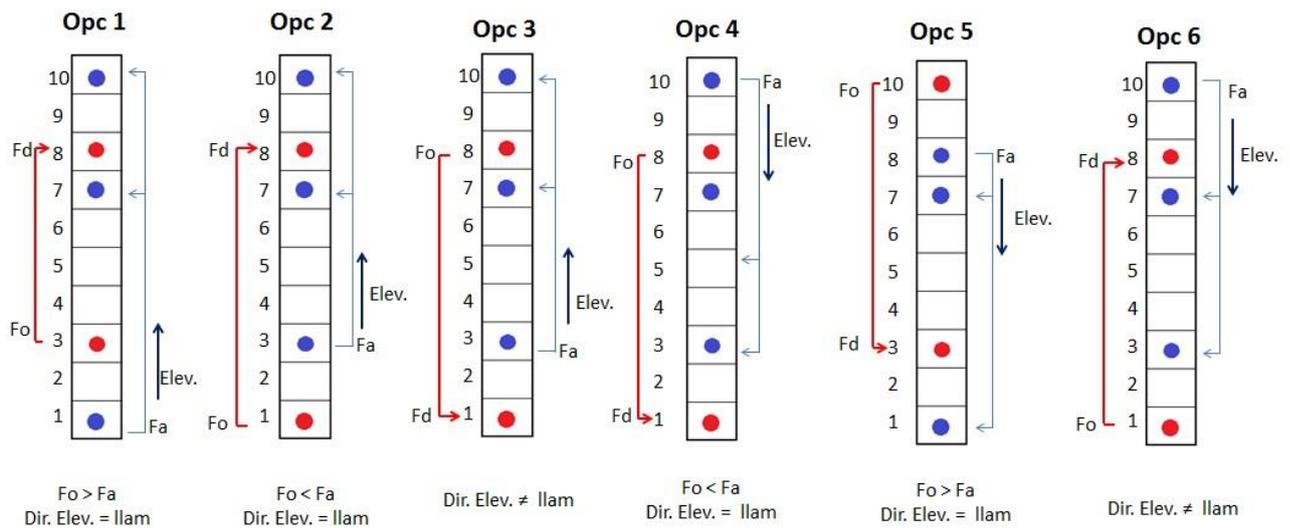


Figura. 10 cenários analisados

De exemplo pode se pensar no análises do cenário 1, onde o elevador está no primer andar ($F_a=1$) e gera-se uma chamada no andar 3 ($F_o=3$), com destino no andar 8 ($F_d=8$). A

direção do elevador é a mesma da chamada. Para o caso de *WT* se tem o tempo em que o elevador demora em deslocar-se do andar 1 até o andar 3. *JT* será o tempo que demora o elevador em deslocar-se desde o andar 3 até o andar 8, tendo em conta o tempo da parada no andar 7. *MWT* será o tempo que demora o elevador em atender a última chamada considerando da mesma forma as distancias, paradas e demais. Por último, para o cálculo de *NC*, *NP* e *RRi* foram usados vetores que armazenam as paradas alocadas, o número de pessoas atuais, junto com as que sobem e as que baixam do elevador. Vide a figura 11.

```

RC vetor de paradas alocadas
RC=[   subindo           descendo   ]
RC=[0010011001  0000000000; elev. 1
    0000000000  0000100010; elev. 2
    0100101101  0000000000]; elev. 3

C_EST vetor de estado dos elevadores
C_EST= [ SUB(1) OU BAIXA , RUN(1) OU STOP , CARGA, ANDAR ATUAL ]
C_EST=[1 0 3 3; elev.1
      0 0 1 9; elev.2
      1 1 4 2]; elev.3
    
```

Figura. 11 vetores de estados

3.6 Estratégia de seleção do elevador

Como os cenários definidos e com o uso da função custo, se faz a escolha do melhor elevador para atender uma chamada da seguinte maneira, assim: o algoritmo BPSO recebe os dados da chamada (*Fa* e *Fo*), junto com os parâmetros de estados de cada elevador (andar atual, direção do movimento, capacidade disponível, paradas alocadas, etc.). As partículas do algoritmo procuram possíveis soluções em cada iteração, as quais são avaliadas na função custo (gerando um valor de aptidão), o algoritmo lembra qual é a melhor solução de cada partícula e a melhor solução de tudo o enxambre (a melhor é a que tem o valor mínimo de aptidão, ou seja a solução que tem uma minimização nos tempos de espera e viagem, com possibilidades elevadas de conforto). A informação da melhor solução encontrada é comunicada as outras partículas, no intuito de que iteração a iteração sejam influenciadas por esta solução. Desta forma, tende-se a melhorar progressivamente as soluções postuladas por parte de cada partícula, no melhor dos casos quando o algoritmo chega ao final das iterações se tem encontrado a solução mais ótima (combinação binária que representa o elevador ou os elevadores para atender uma chamada).

Na tabela 1 apresenta-se os parâmetros de ajuste usados no *PSO* binário para a realização dos testes. As dimensões do algoritmo foram escolhidas para estes testes baseado no número de elevadores do prédio (3) de estudo, o número de partículas e outros parâmetros foram escolhidos de acordo com o desempenho do algoritmo *PSO* em testes de ajuste realizados em Matlab.

Tabela 1. Parâmetros do PSO binário

PSO BINARIO	
parâmetros	valores
Partículas	10
dimensões	3
# iterações	50
peso Inercia (w)	[0,8,0,1]
coeficiente cognitivo e social	C1 =C2=2
velocidade máx.	[-3,3]

O controlador de alocação de chamadas com o algoritmo *PSO* binário foi implementado em *VBA* (possibilidade brindada pelo software *Arena*). No simulador desenvolvido foi representado por um bloco (contém o pseudocódigo do algoritmo), o qual tem como parâmetros de entrada os dados da chamada realizada por um usuário, os parâmetros atuais de posição, de capacidades, direção e paradas dos elevadores. Como saída desse bloco *PSO* produz o número do elevador que vai a servir a chamada analisada, também é realizada a atualização da agenda de paradas, de capacidade, de entrada e saída dos usuários do elevador escolhido. Na figura 12, pode-se observar cópia das telas do software *Arena* com o gerador de chamadas e com o bloco *PSO* binário. Na Figura 13, pode-se observar a simulação de espera, entrada e saída de usuários no elevador. É importante entender que pelo ambiente do software *Arena*, em muitos casos não é possível obter boas imagens dos sistemas simulados (resolução).

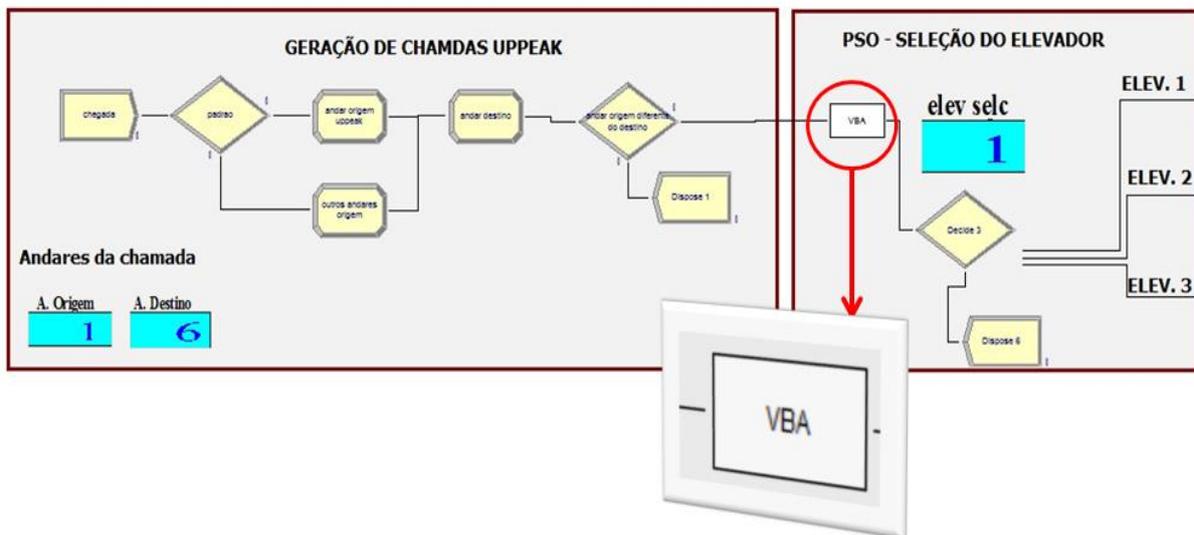


Figura 12. Bloco VBA do BPSO

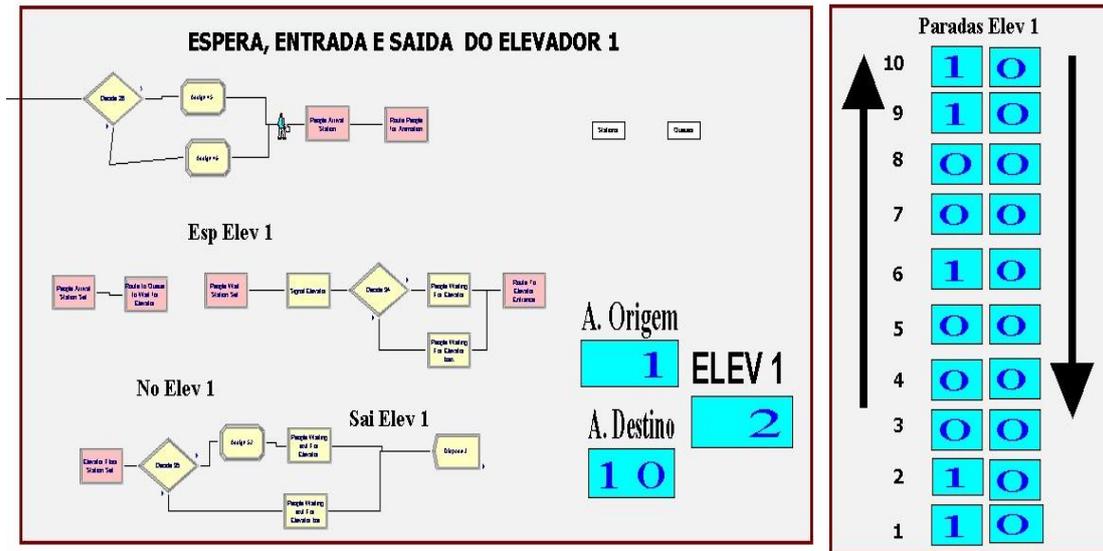


Figura 13. Simulação de usuários no elevador 1.

4 RESULTADOS

Foram realizados muitos testes com simulações no software de processos industriais *Arena* do sistema de elevadores, simulando chamadas sucessivas em períodos de tempo de 5 min em trafego *up-peak*. De acordo com o comportamento deste padrão, a porcentagem de população simulada, foi variando da seguinte forma: 2%, 4%, 6%, 8%, 10%, 12%, 14 % e 16% (o último valor é considerado a faixa critica deste padrão). Para esses testes, foi tomado o tempo médio de espera para o atendimento pelo elevador selecionado pelo algoritmo de otimização, junto com o tempo médio de voo para cada chamada. Vide a tabela 2.

Pode-se observar o comportamento e a tendência do tempo médio de espera pelo elevador selecionado e do tempo de voo do andar de origem até o andar de destino, tudo com respeito às porcentagens de população atendidas. É importante lembrar que 90% das chamadas atendidas pelos elevadores têm como andar de origem o número 1 (padrão de trafego *up-peak*). Também, é relevante entender que para os cálculos destes valores foram feitos diferentes testes (10 para cada porcentagem, lembrando que por cada teste têm-se dados de chamadas baseados nas porcentagens e nos valores esperados, sendo assim, um valor de dados considerável e representativo). Por outro lado, para evitar os períodos transitórios no sistema durante as simulações (Barney, 2003) (Siikonen, 2000), foram simulados faixas de tempo de 6 min em cada teste, no qual os primeiros 30 seg. e os últimos 30 seg. se descartarão.

Na figura 14 pode se observar os tempos de espera e de voo, também na cor azul pode-se observar o tempo de serviço que é a soma destes dois tempos.

Tabela 2. Resultados tempos com diferentes porcentagens de população.

Resultados obtidos				
% pop.	Pessoas esperadas	Tempo de espera (s)	Tempo de voo (s)	Tempo de destino (s)
2%	12	15,14	14,35	29,49
4%	21	24,6	20,86	45,46
6%	27	23,42	19,9	43,32
8%	36	29,35	18,36	47,71
10%	48	33,69	22,4	56,09
12%	54	35,89	26,07	61,96
14%	64	37,07	24,34	61,41
16%	73	43,37	25,5	68,87

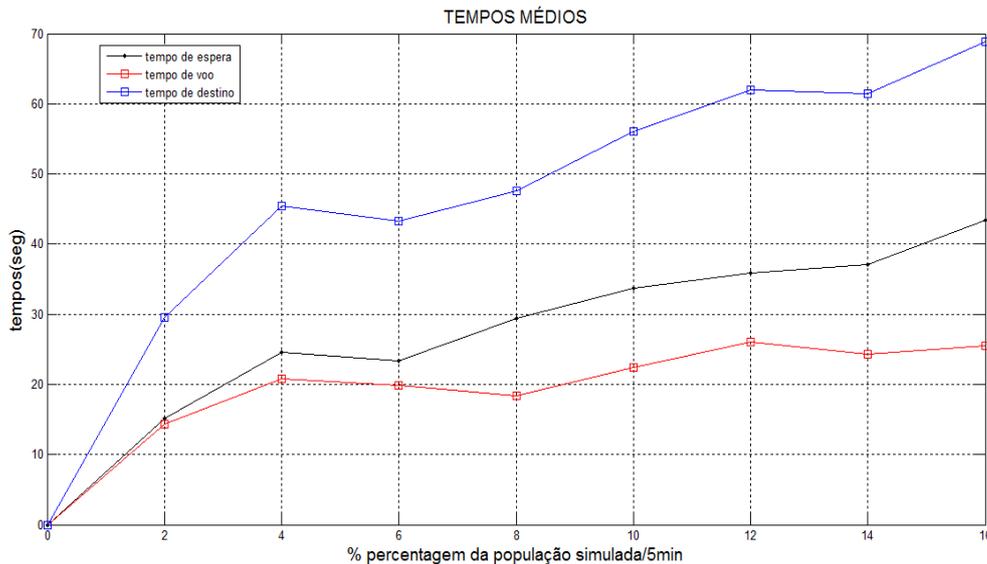


Figura 14. Tempos médios do sistema de elevadores

4.1 Comparação Com Outros Autores

A importância e a relevância dos resultados obtidos nas simulações podem ser observadas mais facilmente quando são comparados com os resultados obtidos por outros autores que abordam o mesmo problema na literatura científica. Pontos a serem observados incluem a forma geral das curvas obtidas e tendências, em uma abordagem qualitativa, uma vez que uma comparação em termos absolutos se faz difícil, devido às diferenças entre simuladores utilizados, assim como aos parâmetros utilizados na definição dos problemas simulados, cujas informações não estão completas na literatura. Com a finalidade de realizar uma avaliação qualitativa, faz-se a comparação dos resultados deste trabalho com os relatados nas referências (Patiño, 2010) e (Siikonen, 2000), e entre outros (Gu, 2012) (Berna Bolata, 2013) (Zhou & Ye, 2004) (Forero, Muñoz, Carvalho, & Llanos, 2011) (Wu & Wu, 2012).

Na comparação com outros trabalhos relacionados, não se tem a possibilidade de fazer de forma gráfica, mas pode-se fazer a comparação numérica com os tempos de espera, de voo e de destino obtidos. Na Tabela 3 observa-se que os resultados obtidos nesta pesquisa estão em concordância com os obtidos em pesquisas similares. Estas referências têm datas de publicação recentes e outras não tão recentes, relevantes nesta área de trabalho. Esta análise e comparação avalia a postulação dos algoritmos bioinspirados como possível solução para este tipo de problemas.

Tabela 3 A comparação de resultados com outros autores

	ano	Autor	trabalho	tempo médio de espera	Tempo médio de voo	tempo médio de destino
1	2000	Siikonen	<i>On traffic planning methodology</i> (Siikonen, 2000)	37	x	x
2	2004	Jian Liu	<i>Dynamically dispatching method aiming to reduce the servicing time in the egcs</i> (Y Zhou, 2004)	23,3	27,9	51,2
3	2010	Morkon	<i>Design of elevator group control system simulation platform based on shortest distance algorithm</i> (Chuansheng & Chunping, 2010)	35.45	30.46	65.91
4	2011	Patiño	<i>Fuzzy elevator group control system using technology for industrial automation</i> (Forero, Muñoz, Carvalho, & Llanos, fuzzy elevator group control system using technology for industrial automation, 2011)	47	20	67
5	2011	Sheng	<i>A Novel Elevator Group Control Scheduling Algorithm based on Pseudo Differential Feedback</i> , (Wu & Wu, 2012)	35,85	15,83	51,68
6	2012	Y.Gu	<i>Multi-objective optimization of multi-agent elevator group control system based on real-time particle swarm optimization algorithm</i> (Gu, 2012)	44.87	38.34	83.21
7	2013	Cortes	<i>A particle swarm optimization algorithm for optimal car-call allocation in elevator group control systems</i> (Berna Bolata, 2013)	x	x	33
8	2015	Diago	<i>Otimização com algoritmos bioinspirados de controle de tráfego em grupo de elevadores</i> (Proposto)	43,37	25,5	68,87

Contudo, pode-se observar que os resultados aqui alcançados, em termos de tempos médios, situaram-se dentro das faixas de variação reportadas na literatura específica. No intuito e na tentativa de se melhorarem os tempos que caracterizam o comportamento do sistema. Na Figura 15, pode-se observar que os tempos médios de espera obtidos (cor vermelha) são similares aos tempos mais elevados obtidos por os outros autores. Fazendo uma comparação só qualitativa, pois as considerações de construção das simulações nos trabalhos apresentados são desconhecidas, o que não permite fazer uma comparação direta.

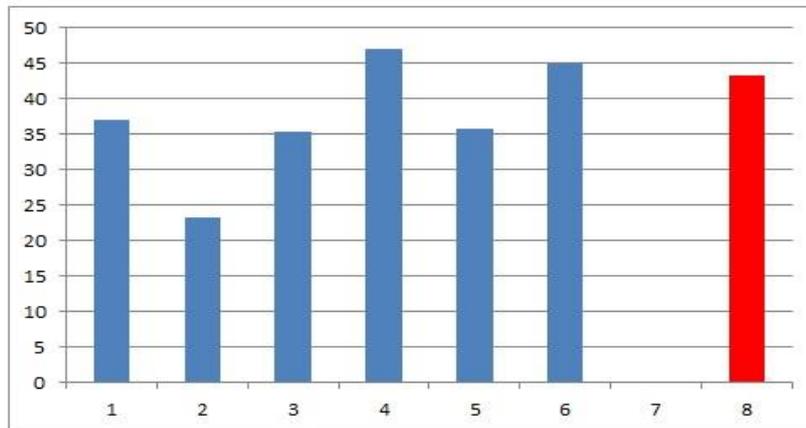


Figura 15. Tempos médios de espera

Na Figura 16, pode-se observar que os tempos médios de voo obtidos (cor vermelha) estão sobre na faixa média dos resultados comparados com outros autores. O simulador desenvolvido para avaliar a técnica de otimização *BPSO* apresentou desempenho adequado, respondendo às variações das condições do sistema e selecionando por parte do controlador proposto o elevador com melhores características para atender às chamadas. Entretanto, os resultados obtidos (tempo de espera e tempo de voo) nas simulações pode se observar que o sistema de simulação realizado satisfaz os requerimentos desta pesquisa. Na comparação feita observasse a compatibilidade dos resultados obtidos com os antecedentes e avaliam o controlador proposto (novamente é só qualitativamente). Desta comparação, o controlador proposto tem a mesma tendência dos resultados de pesquisas similares, o que pode ser observado como um desempenho adequado para o sistema de controle

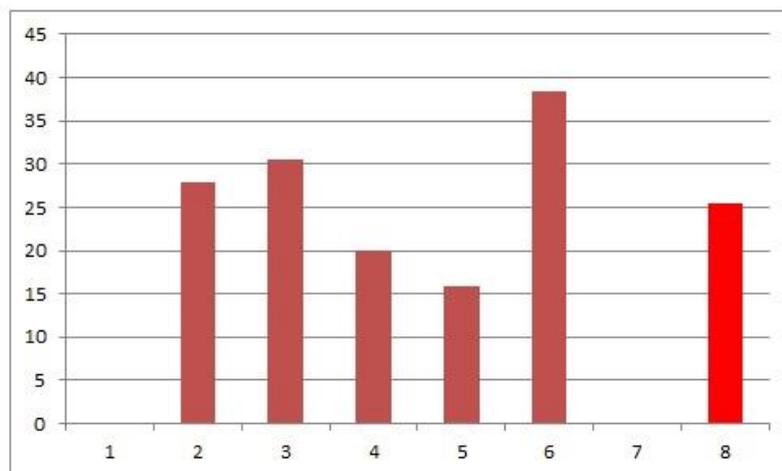


Figura 16. Tempos médios de voo

5 CONCLUSÕES

As conclusões mais importantes deste trabalho de pesquisa apresentam-se a seguir:

- Um resultado importante desta pesquisa é formação e definição da função custo que representa o problema de otimização. Os resultados de simulação demonstraram que o algoritmo *PSO* binário consegue selecionar o melhor elevador para diversas chamadas de serviço, comprovando a escolha do elevador mais apropriado para diferentes cenários de acordo com o melhor valor de aptidão.
- Foi implementado no software Arena o simulador de grupo de elevadores com o controlador proposto baseado em inteligência de exames *BPSO*. Este simulador foi testado na faixa crítica de fluxo de usuários em um prédio comercial (*up-peak*), concluindo a partir dos resultados obtidos, que o algoritmo de controle fornece desempenho satisfatório ao tentar controlar o tráfego presente no sistema. E de acordo com a literatura poderia controlar o tráfego dos outros padrões de tráfego sem problema.
- Os resultados obtidos pelo controlador proposto além de não serem os melhores quando foram comparados e analisados com os resultados encontrados em pesquisas similares, apresentam uma tendência e concordância similar aos da literatura científica estudada. O que pode estabelecer que o sistema de controle proposto pode se postular como uma solução potencial para problema continuando ajustado ele para melhorar seu desempenho. Também pode-se apresentar como solução para outros problemas de pesquisa similares.

6 REFERENCIAS

- Rockwell Automation. (2005). *Arena Basic User's guide*.
- Atsuya Fujino, T. T. (1997). An Elevator Group Control System with Floor-Attribute Control Method and System Optimization Using Genetic Algorithms. *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS*.
- Bailey, A., Ombuki-berman, B., & Asobiela, S. (2013). Discrete PSO for the Uncapacitated Single Allocation Hub Location Problem. *European Journal of Operational Research*.
- Barney, G. (2003). *Elevator Traffic Handbook Theory And Practice*.
- Berna Bolata, O. A. (2013). A particle swarm optimization algorithm for optimal car-call allocation in elevator group control systems. *Applied Soft Computing*.
- Chuansheng, W., & Chunping, C. (2010). Design of Elevator Group Control System Simulation Platform Based on Shortest Distance Algorithm. *International Conference on Electrical and Control Engineering*.
- Cortés, P., Larrañeta, J., Onieva, L., Muñuzuri, J., & Fernández, I. (2002). Algoritmos de Optimización en Sistemas de Transporte Vertical. *II Conferencia de Ingeniería de Organización*.
- Crespi, V., Galstyan, A., & Lerman, K. (2008). Top-Down vs Bottom-up Methodologies in Multi-Agent System Design. *Autonomous Robots*.

- Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. *International Symposium Micro Machine and Human Science. Nagoya, Japan: IEEE.*
- Eberhart, R., & Kennedy, J. (1997). A discrete binary version of the particle swarm algorithm. *International Conference on Systems, Man, and Cybernetics.*
- Forero, A. P., Muñoz, D. M., Carvalho, G. C., & Llanos, C. (2011). fuzzy elevator group control system using technology for industrial automation. *21st Brazilian Congress of Mechanical Engineering - COBEM.*
- Gu, Y. (2012). Multi-Objective Optimization of Multi-Agent Elevator Group Control System Based on Real-Time Particle Swarm Optimization Algorithm. *Engineering, Vol. 4 No. 7.*
- Jafferi Jamaludin, N. A. (2010). An Elevator Group Control System With a Self-Tuning Fuzzy Logic Group Controller. *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS.*
- Jian Liu, C. W. (2010). A Hybrid Control for Elevator Group System. *Third International Workshop on Advanced Computational Intelligence.*
- joaquin . fernandez, p. c. (2009). *fuzzy logic-based controller for vertical traffic detection purpose.*
- Marja-Liisa S. (s.f.). “Planning and Control Models for Elevators in High-Rise Buildings”. *KONE Corporation P.O. Box 8 SF-00331 Helsinki, Finland.*
- Markon, K. H. (2006). “ *Control of Traffic Systems*”.
- Markon, S., Kita, H., Kise, H., & Bartz-Beielstein, T. (2006). *Control of Traffic Systems in Buildings.*
- P Cortés, J. L. (2002). Algoritmos de Optimización en Sistemas de Transporte Vertical. *II Conferencia de Ingeniería de Organización.*
- Patiño, A. (2010). *Estudo e simulação de técnicas de controle de tráfego de grupo de elevadores usando automação industrial.* Unb- Universidade de Brasília.
- R., E., & J., K. (1995). “A new optimizer using particle swarm theory”. *International Symposium Micro Machine and Human Science. Nagoya, Japan: IEEE.*
- Serapião, A. B. (2009). FUNDAMENTOS DE OTIMIZAÇÃO POR INTELIGÊNCIA DE ENXAMES: UMA VISÃO GERAL. *Revista Controle & Automação/Vol.20.*
- Siikonen, M.-L. (2000). On traffic planning methodology. *International Congress on Vertical Transportation.* Berlin.
- Valentino Crespi, A. G. (2007). Top–Down vs Bottom–up Methodologies in Multi–Agent System Design.
- Wu, S., & Wu, G. (2012). A Novel Elevator Group Control Scheduling Algorithm based on Pseudo Differential Feedback. *Proceeding of the IEEE, International Conference on Automation and Logistics.*
- Y Zhou, Q. Y. (2004). Dynamically Dispatching Method Aiming to Reduce the Servicing Time in the EGCS. *Proceedings of the 5'World Congress on Intelligent Control and Automation.*

- Yu, L., Zhou, J., Mabu, S., Hirasawa, K., Hu, J., & Markon, S. (2007). Double-Deck Elevator Group Supervisory Control System Using Genetic Network Programming with Ant Colony Optimization with Evaporation. *IEEE Congress on Evolutionary Computation*.
- Zhou, Y., & Ye, Q. (2004). Dynamically Dispatching Method Aiming to Reduce the Servicing Time in the EGCS. *Proceedings of the 5'World Congress on Intelligent Control and Automation*.