

APLICAÇÃO DA REGRESSÃO ESTATÍSTICA PARA AJUSTAR OS PARÂMETROS DO ALGORITMO GENÉTICO

1. Introdução

Localização de pontos de facilidades pode ser associada a melhor localização de hospitais, escolas, indústrias, depósitos, antenas, dentre outros, tornando sua investigação um problema atual. Encontrar a localização ideal para os pontos de facilidade é uma tarefa trivial quando o problema é de pequeno porte e o número de demandas é baixo, permitindo solucionar através do algoritmo exato, aplicando o método exaustivo, que combina todas as soluções possíveis (Garey & Johnson, 1979). Entretanto, para problemas de grande porte, no qual o número de demandas é elevado, faz-se necessário a utilização de métodos heurísticos e metaheurísticos em sua resolução, pois conseguem boas soluções em tempos aceitáveis, tendo em vista a complexidade do problema.

A tecnologia de redes locais sem fio, também denominada *Wireless Local Area Network* (WLAN), permite fornecer as mesmas funcionalidades das redes convencionais com fio, porém, com maior flexibilidade, mobilidade, simplicidade e compatibilidade com diversos equipamentos, como por exemplo: *notebooks*, *smartphones*, *tablets* e videogames. A diminuição de custos tem acelerado o seu crescimento, com isso, o Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo (IFES) *campus* Cachoeiro de Itapemirim elaborou uma meta de conceder acesso à rede interna e internet para alunos, professores e visitantes através da tecnologia de rede sem fio. Porém, as variáveis envolvidas para a tomada de decisão tornam o problema complexo.

Solucionar o problema de localização de antenas é o que motiva a implementação de um aplicativo com base em inteligência computacional. A finalidade é o atendimento do maior número de demandas (clientes) e na utilização do menor número de facilidades (antenas), considerando as restrições de alcance de transmissão das facilidades e também possibilitar que o aplicativo possa ser utilizado por profissionais sem nenhum conhecimento de programação, algoritmo ou heurísticas.

É possível localizar esses pontos utilizando o Problema das p -Medianas (*p-Median Problem* - PMP), que inicialmente foi proposto por Hakimi (1964). O objetivo do PMP é localizar p vértices (facilidades) em um grafo contendo n vértices (demandas), de tal forma a minimizar o somatório das distâncias de cada facilidade até a mediana mais próxima. Entretanto, de acordo com Church and ReVelle (1976), na versão original do PMP, não há restrição em relação à distância de um grupo de demandas para o ponto de facilidade mais próximo, mas sim a menor distância média de um grupo de demandas em relação ao seu ponto de facilidade. Dessa forma, há a exigência de que todos os pontos de demanda sejam atendidos. Contudo nem sempre a facilidade possui capacidade de alcance suficiente para garantir a cobertura total à demanda. Por isso, é necessário a utilização de um limite no raio de alcance da antena. Na Figura 1, é apresentada uma solução com o PMP e outra com o PMP com o raio de alcance máximo. Este problema tem sido denominado como o problema de p -Medianas com restrição de distância máxima ou Problema de Localização de Máxima Cobertura - PLMC (Khumawala, 1973).

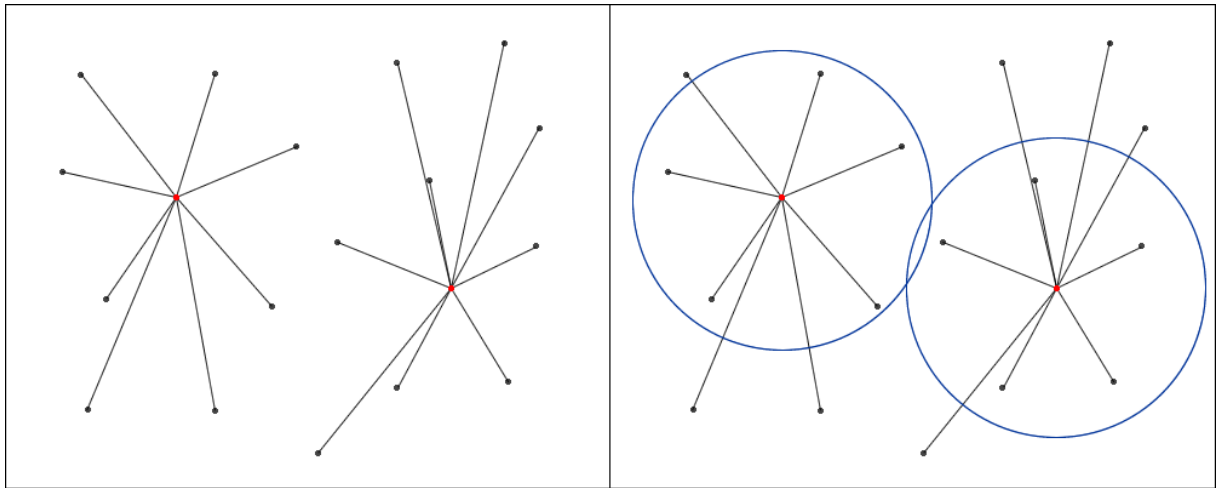


Figura 1 - Exemplo de solução do PMP sem e com o raio de alcance máximo

Fonte: Os próprios autores

O PLMC consiste em escolher a melhor localização para instalar os pontos de facilidade de forma que o maior número de pontos de demandas seja coberto, isto é, maximizar a cobertura dentro de uma desejada distância T localizando um número fixo de facilidades. Isso poderia assegurar que as demandas sejam atendidas com qualidade, pois não existem antenas de transmissão com alcance ilimitado.

Segundo Lorena (2003), o objetivo do PLMC é localizar a configuração que maximiza a cobertura de um número de demandas. Cada ponto de demanda i deve ser atendido por apenas uma facilidade j , considerando d distância máxima de atendimento da facilidade. O PLMC pode ser modelado como um problema de programação inteira, a formulação em programação matemática, adaptada conforme Lorena (2003), é:

Parâmetros

n – Pontos de facilidade.

m - Pontos de demanda.

p - Número máximo de demandas.

d - Distância máxima de atendimento, entre a demanda e a facilidade.

Índices

i – Índices dos pontos de demanda. $i \in [1, m]$.

j – Índices dos pontos de facilidade. $j \in [1, n]$.

Variáveis de Decisão

X_j – Variável binária que representa as facilidades abertas no ponto i .

Y_i – Variável binária que representa as demandas atendidas no ponto j .

D_{ij} – Variável numérica que representa a distância entre o ponto i e o ponto j .

N_{ij} - Conjunto de facilidades que estão a menos da d distância máxima de atendimento do ponto de demanda i .

$$x_j = \begin{cases} 1 & \text{O ponto } j \text{ alocado como facilidade aberta.} \\ 0 & \text{Caso contrário.} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{O ponto } i \text{ alocado como demanda atendida por uma facilidade aberta.} \\ 0 & \text{Caso contrário.} \end{cases}$$

$$N_i = \{j \mid D_{ij} \leq d\}$$

Modelo

$$\text{Max} \quad \sum_{i=1}^m \int_i y_i \quad (1)$$

Sujeito a:

$$\sum_{j \in N_i} x_j \geq y_i \quad , i = 1, \dots, m \quad (2)$$

$$\sum_{j=1}^n x_j \leq p \quad (3)$$

$$x_j \in \{0,1\} \quad , j = 1, \dots, n \quad (4)$$

$$y_i \in \{0,1\} \quad , i = 1, \dots, m \quad (5)$$

No qual:

- 1) A função objetivo busca maximizar a cobertura das demandas de cada ponto \int_i .
- 2) A restrição indica que as demandas somente serão contabilizadas como cobertas se estiverem no raio de cobertura de alguma facilidade aberta.
- 3) A restrição estabelece um número máximo de facilidades a serem abertas.

Pode-se encontrar na literatura uma diversificada aplicação do PLMC para a localização de facilidades, dentre eles localização de: bases de ambulância para atendimento médico (Azizan, Lim, Hatta, & Gan, 2012), abrigos em caso de evento catastrófico em uma cidade (Ren, Chen, Guo, & Cai, 2009), postos de combustíveis alternativos no estado da Flórida (Lim & Kuby, 2010) e instalações para atendimento de emergência no caso de ataque bio-terrorista de grande escala de antraz na cidade de Los Angeles (Murali, Ordóñez, & Dessouky, 2012).

O PLMC pertence à classe NP-difícil (*non-deterministic polynomial time hard* - *NP-hard*) e possui ordem de complexidade exponencial, isto é, para sua resolução é necessário um esforço computacional que cresce exponencialmente com o tamanho do problema (Garey & Johnson, 1979; Shen, Zhan, & Zhang, 2011). Como uma alternativa para a resolução do PLMC, algoritmos evolucionários podem ser levados em consideração. Entre eles, o conceito de Algoritmo Genético (AG), inspirado na teoria de Darwin na sobrevivência do mais forte. Os AGs pertencem à classe dos algoritmos probabilísticos, mas eles são muito diferentes dos problemas aleatórios, pois combinam elementos da busca direta e estocástica. Devido a isso, os AGs são também mais robustos que os métodos existentes de pesquisa direta, e visam problemas complexos. Outra propriedade importante dos métodos de pesquisa do AG, é que eles mantêm uma população de potenciais soluções, distinta de todos os outros métodos que processam um único ponto do espaço de busca.

Neste trabalho, são propostas duas análises estatísticas, através de experimentos com o aplicativo desenvolvido com base no cenário real, com o intuito de encontrar os parâmetros ideais para o algoritmo genético, de forma que seu resultado e tempo de execução sejam otimizados.

2. Algoritmo Genético

Man, Tang, and Kwong (1996) argumenta que o AG é inspirado nos mecanismos naturais, um processo biológico em que os indivíduos mais fortes são provavelmente os vencedores em um ambiente de competição, uma analogia direta à evolução natural. Presume-se que a solução de um potencial problema é um indivíduo e pode ser representado por um conjunto de parâmetros. Esses parâmetros são considerados como os genes de um cromossomo e pode ser estruturado por uma sequência de valores na forma binária. Um

valor positivo, comumente conhecido como aptidão ou *fitness*, é empregado para avaliar todos os cromossomos, e este valor está diretamente relacionado com a sua função objetivo.

Linden (2008) destaca em seu livro, a importância da representação cromossomial para o AG, pois através dela é possível uma maneira de traduzir a informação do nosso problema, em uma maneira viável, que possa ser calculada pelo computador. Cada pedaço indivisível desta representação é chamado de gene e formam um indivíduo ou cromossomo. Sendo assim, um problema que possui 40 pontos de demandas e 5 pontos de facilidades, será formado por um cromossomo de 45 genes, sendo diferenciado pelo valor binário do gene (0, 1), conforme exemplo mostrado na Figura 2. Caso o gene seja demanda terá o valor 0 (zero) e caso o gene seja facilidade terá valor 1 (um).

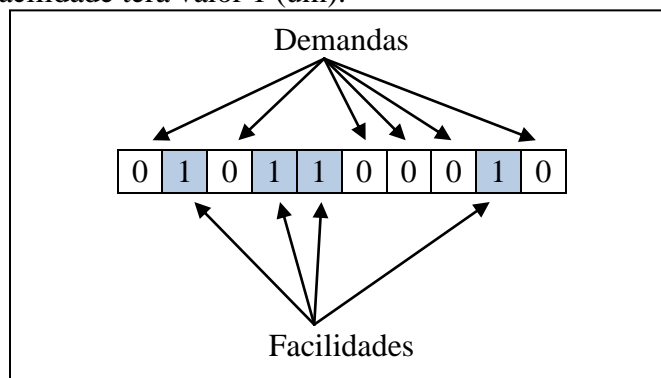


Figura 2 - Representação cromossomial

Fonte: Os próprios autores

O AG possui parâmetros que influenciam diretamente na qualidade do resultado e no tempo computacional gasto, por isso o ajuste dos parâmetros é de extrema importância. Os AGs utilizam, no mínimo, três parâmetros numéricos: probabilidade de cruzamento, probabilidade de mutação e tamanho da população (ou número de indivíduos). Devido a sua importância, existem diversos trabalhos científicos que elaboram experimentos acerca do ajuste dos parâmetros do AG, dos quais destacamos (Goldberg, 1989; Grefenstette, 1986; Jong, 1975; Schaffer, Caruana, Eshelman, & Das, 1989).

Jong (1975), testou combinações distintas nos parâmetros do AG com base em cinco funções com características diversas, incluindo contínua e descontínua, convexo e não convexo, uni modal e multi modal, determinística e não determinístico. Sua *suíte* de funções já foi aprovada por vários pesquisadores como base de teste padrão para avaliar projetos de AG (Giger, Keller, & Ermanni, 2007; Hendtlass, 2001; Horn & Goldberg, 1996; Kok & Sandrock, 2009; Paperin, 2008).

Jong (1975) utilizou AG simples com seleção por roleta, um único ponto de cruzamento e mutação simples para investigar os efeitos de quatro parâmetros: tamanho da população, taxa de cruzamento, taxa de mutação e o número de gerações. Suas principais conclusões foram:

- Aumentar o tamanho da população resultou em um desempenho melhor em longo prazo, mas a população menor respondeu mais rapidamente e, portanto, apresentaram resultados iniciais melhores.
- A mutação é necessária para restaurar alelos perdidos, mas isso deve ser mantido em probabilidades baixas, senão o AG degenera e proporciona em uma busca aleatória.
- A probabilidade de cruzamento em torno de 60% obteve o melhor resultado. Mas o aumento dessa probabilidade favoreceu a degradação do desempenho.
- O modelo de população não sobreposta funcionou melhor em geral.

Segundo Jong (1975), o conjunto de parâmetros que foram mais eficientes, nas funções por ele utilizadas. Seu trabalho foi muito importante na medida em que forneceu orientações práticas para a aplicação do AG. Suas recomendações têm sido amplamente adotado e muitas vezes referido como configuração padrão. Entretanto, a aplicação inconsequente dos valores dos parâmetros do AG, em alguns casos pode ser um erro grave.

Mais tarde, Grefenstette (1986) nota que, o AG pode ser utilizado como um procedimento de otimização dos parâmetros de outro AG. Nos experimentos, a meta-heurística AG evoluía uma população de 50 indivíduos, tendo como objetivo aperfeiçoar o conjunto de parâmetros tratados por (Jong, 1975), em sua *suíte* de teste. Em sua representação cromossômica, cada indivíduo representava seis parâmetros do AG: tamanho da população, taxa de cruzamento, taxa de mutação, *generation gap*, *scaling window* e estratégia de seleção (elitista ou não elitista). A aptidão de um indivíduo foi calculada em função do desempenho do AG usando os parâmetros codificados por esse indivíduo. A meta-heurística utilizava os parâmetros encontrados por Jong (1975).

Reconhecendo que os valores dos parâmetros podem ter um impacto significativo sobre o desempenho de um AG e que uma análise mais profunda é necessária, Schaffer et al. (1989) expandiram o experimento realizado por Jong (1975). Além das cinco funções que eles tinham estudado, introduziram mais cinco. Uma observação notável foi que bons resultados no desempenho do AG resultam de uma relação inversa entre o tamanho da população e a taxa de mutação, ou seja, as taxas de mutação elevadas associado com populações menores obtiveram bons resultados, como também taxas de mutação baixas associados com grandes populações conseguiram resultados bons.

3. Experimentos

Nesta seção são apresentados os experimentos computacionais da heurística proposta, no qual foram desenvolvidas em PHP com banco de dados *Mysql* e foram executados em um servidor DELL R610 com dois processadores Intel Xeon de 2.13GHz e 6GB de memória.

3.1. Primeiro experimento

A fim de obter o melhor desempenho do AG, foram realizados testes com o aplicativo utilizando faixas de valores para os parâmetros empregados no AG, como pode ser observado na Tabela 1, no qual os valores foram escolhidos aleatoriamente em tempo de execução. A taxa de cruzamento teve o intervalo definido de 10% a 80%, a taxa de mutação teve o intervalo de 0,01% a 20%, o tamanho da população teve o intervalo definido entre 4 e 40 indivíduos e por final a quantidade de gerações de 10 a 5000. Os intervalos foram selecionados, com o intuito de abranger e ampliar os parâmetros propostos por (Grefenstette, 1986; Jong, 1975; Schaffer et al., 1989)

Tabela 1 - Parâmetros utilizados no primeiro experimento

Parâmetro	Valor Mínimo	Valor Máximo
Cruzamento	10%	80%
Mutação	0,01%	20%
Indivíduos	4	400
Gerações	10	5000

Fonte: Os próprios autores

Com a definição da faixa de valores, foram realizadas simulações com o aplicativo e os dados obtidos em cada execução foram salvos em banco de dados *Mysql*, com a finalidade de realizar análises estatísticas posteriores, a fim de obter os melhores parâmetros.

No primeiro experimento foram obtidos 49.386 resultados com a execução do aplicativo com valores aleatórios, na Tabela 2 são apresentados alguns dados reais do experimento, no qual são informados: o id para controle dos dados, o valor de aptidão do melhor indivíduo alcançado dentre todas as gerações, a taxa de cruzamento usada (valor em decimal), a taxa de mutação empregada (valor em decimal), o número de indivíduos da população e o número de gerações.

Tabela 2 - Alguns dados obtidos do primeiro experimento

Id	Aptidão	Cruzamento	Mutação	População	Gerações
1	23340	0.580000	0.005600	24	1081
2	19840	0.750000	0.119500	263	1553
3	18980	0.440000	0.133500	394	4413
4	22700	0.210000	0.123600	342	1545
5	20760	0.110000	0.170400	327	3268
6	19980	0.590000	0.023800	54	4559
7	20040	0.770000	0.133700	75	2494
8	19540	0.620000	0.172700	123	604
9	19340	0.340000	0.153600	50	2636
10	22040	0.470000	0.156900	377	2844

Fonte: Os próprios autores

Os dados do primeiro experimento foram inicialmente exportados do banco de dados *Mysql* e tabulados no software Microsoft Office Excel versão 2007. E, posteriormente, analisados estatisticamente utilizando o software SAEG (Sistema para Análises Estatísticas e Genéticas) versão 9.1, a fim de obter relação dos parâmetros empregados e o valor de aptidão. Foi obtida a regressão entre a aptidão (*fitness*) em função da taxa de mutação. Os valores de aptidão e mutação foram normalizados, a fim de possibilitar a análise pelo software SAEG. A escolha do modelo estatístico da regressão, dentre os apresentados pelo SAEG (Linear, Quadrático, Cúbico, Raiz Quadrada, Potencial, Exponencial, Cúbico-Raiz, dentre outros), foi feita mediante a análise dos parâmetros da regressão (significância e coeficiente de determinação), como pode ser visualizado na Figura 3, no qual foi escolhido o Cúbico-Raiz, por apresentar o maior valor de significância (R2) e coeficiente de determinação abaixo de 5% (F. sig.).

Modelos disponíveis - Pré-definidos				
Modelo	Equação	R2	R2ajust.	F.sig.
<input checked="" type="radio"/> Linear	$Y=a+b*X$	11.33	11.33	0.00
<input type="radio"/> Quadrático	$Y=a+b*X+c*X^2$	19.91	19.91	0.00
<input type="radio"/> Cúbico	$Y=a+b*X+c*X^2+d*X^3$	23.55	23.55	0.00
<input type="radio"/> Raiz Quadrada	$Y=a+b*X^{1/2}+c*X$	23.92	23.92	10.55
<input type="radio"/> Potencial	$Y=a*X^b$	14.95	14.94	4.06
<input type="radio"/> Exponencial	$Y=a*b^X$	8.61	8.60	0.00
<input type="radio"/> Hiperbólico-1	$Y=a+b/X$	3.09	3.09	0.00
<input type="radio"/> Hiperbólico-2	$Y=1/(a+b*X)$	5.50	5.50	0.00
<input type="radio"/> Logarítmico E	$Y=a+b*\ln(X)$	21.22	21.22	12.90
<input type="radio"/> Logarítmico 10	$Y=a+b*\log(X)$	21.22	21.22	12.90
<input type="radio"/> Log-Recíproco	$\log(Y)=a+b/X$	1.98	1.98	0.00
<input type="radio"/> Cúbico-Raiz	$Y=a+b*X^{1/2}+c*X+d*X^{1.5}$	24.28	24.28	0.00
<input type="radio"/> Log-Log	$\log(Y)=a+b*\log(X)$	14.95	14.94	4.06
<input type="radio"/> Ln-Ln	$\ln(Y)=a+b*\ln(X)$	14.95	14.94	4.06
		Abandonar	Processar	

Figura 3 - Parâmetros da regressão da aptidão em função da mutação

Fonte: Os próprios autores

Através da análise estatística do primeiro experimento, foi possível indicar uma taxa de mutação ótima, como pode ser visualizada pela Figura 4, no qual pode-se concluir que quanto menor a taxa de mutação maior é o *fitness* do aplicativo. A taxa de mutação empregada inicialmente foi de 0,01% a 20%, assim a taxa selecionada foi de 0,01%.

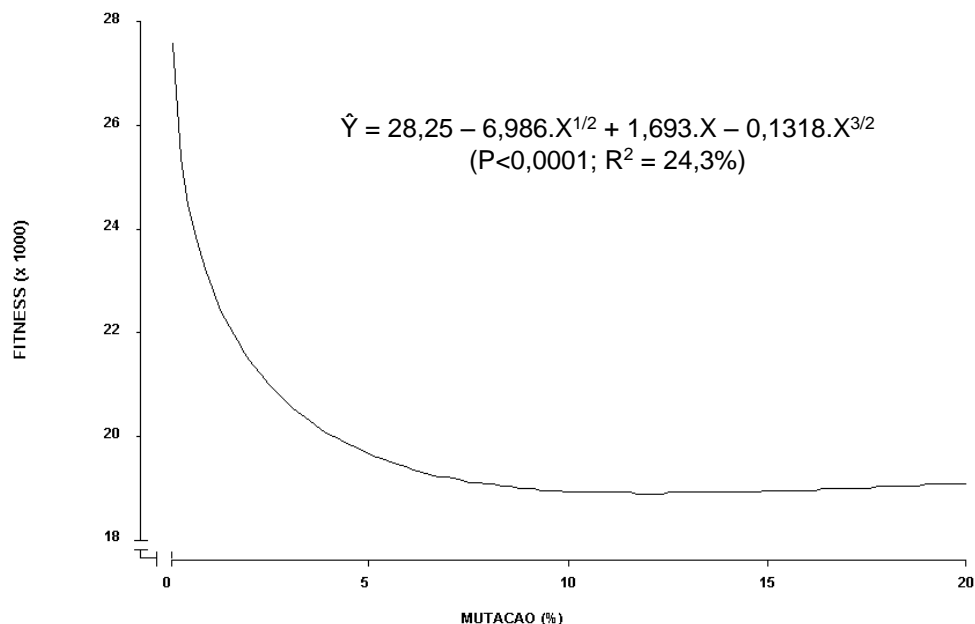


Figura 4 - Gráfico Cúbico-Raiz entre aptidão em função da taxa de mutação
Fonte: Os próprios autores

3.2. Segundo Experimento

Realizou-se um segundo experimento, com os mesmos intervalos de valores, mas com uma taxa de mutação fixa, obtida através do primeiro experimento. Na Tabela 3 são apresentados os parâmetros utilizados no segundo experimento, sendo que a taxa de mutação foi fixada em 0,01%.

Tabela 3 - Parâmetros utilizados no segundo experimento

Parâmetro	Valor Mínimo	Valor Máximo
Cruzamento	10%	80%
Indivíduos	4	400
Gerações	10	5000

Fonte: Os próprios autores

No segundo experimento foram obtidos 36.818 resultados com a execução do aplicativo, os dados do segundo experimento foram submetidos ao mesmo processo do primeiro experimento e foram posteriormente analisados pelo SAEG, com intuito de obter relação dos parâmetros empregados e o valor de aptidão. Foi possível obter a regressão entre a aptidão em função da taxa de crossover e da aptidão em função do número de gerações.

Por meio da análise estatística do segundo experimento, foi possível indicar duas taxas como sendo ideais. Na Figura 5 é apresentado um gráfico do valor de aptidão em função da taxa de crossover, e a taxa de crossover que obteve o maior valor de fitness é de 55%. Já na Figura 6 é apresentado um gráfico do valor de aptidão em função do número de gerações, e como era esperado quanto maior o número de gerações, maior será o valor da aptidão, entretanto quanto maior o número de gerações maior será o custo computacional e consequentemente maior será o tempo para o resultado. Diante disto o número de gerações selecionado foi 1000, pois obtém boas soluções em tempo aceitável, em 2 minutos e 30 segundos.

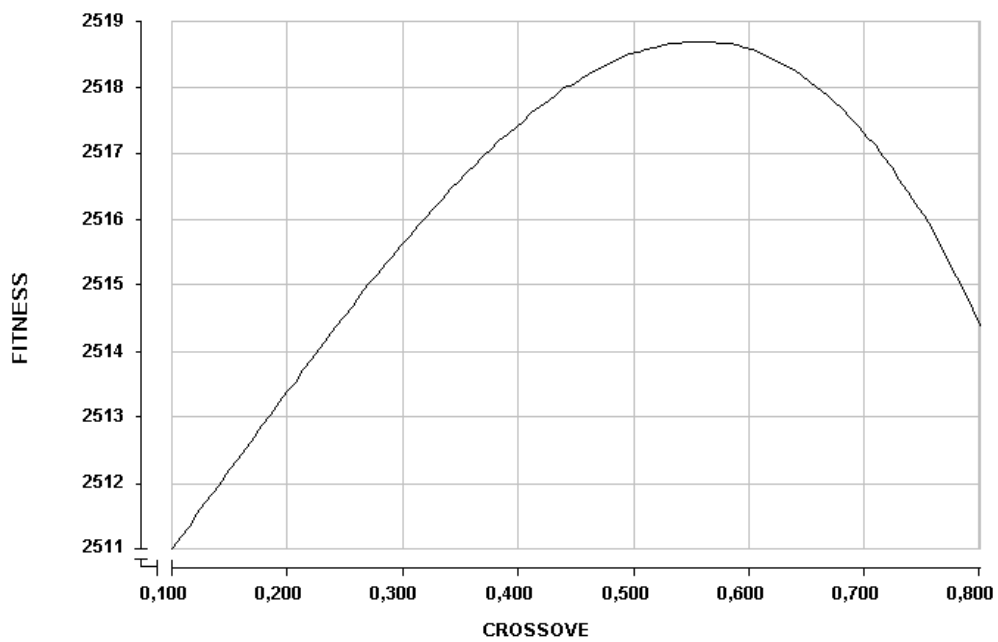


Figura 5 - Gráfico Cúbico entre aptidão em função da taxa de crossover
Fonte: Os próprios autores

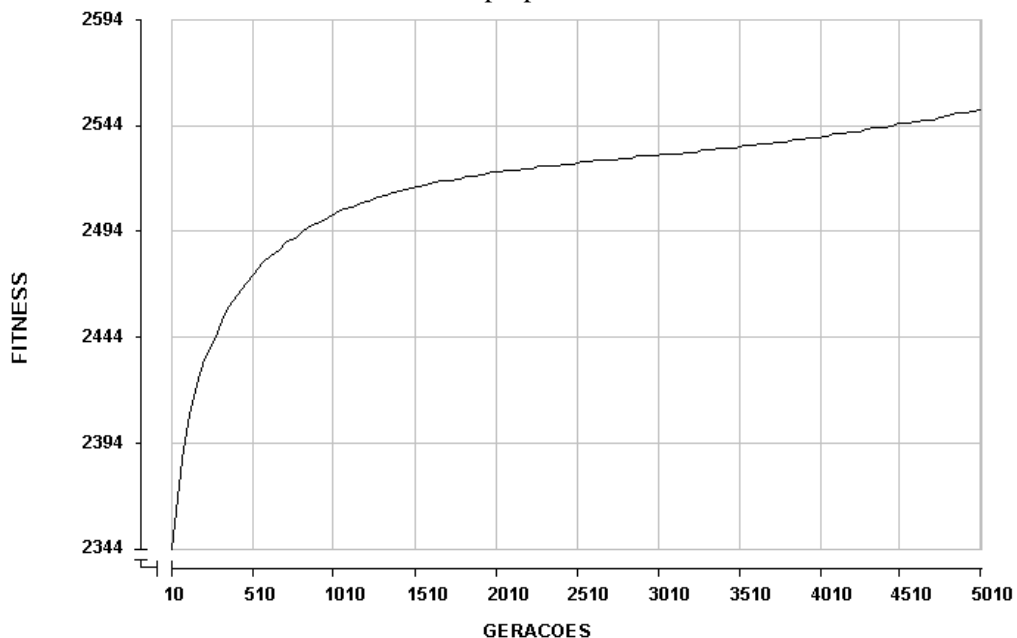


Figura 6 - Gráfico Cúbico-Raiz entre aptidão em função do número de gerações
Fonte: Os próprios autores

3.3. Comparações

Com o intuito de validar os parâmetros obtidos com os experimentos, executou-se o aplicativo desenvolvido vinte vezes: dez vezes com os parâmetros propostos por Michalewicz (1996) e dez com os parâmetros propostos através dos experimentos, que podem ser visualizados na Tabela 4. Em cada execução do aplicativo, efetuou-se uma média da aptidão da população em todas as 1000 gerações. Os valores obtidos foram tabulados no software Microsoft Office Excel versão 2007, depois efetuou-se uma média da aptidão da população antes e depois do ajuste dos parâmetros do AG e nota-se uma diferença significativa entre os valores, isto é, a aptidão da população após o ajuste dos parâmetros do AG apresentou um aumento de 28,09% em relação a aptidão da população utilizando os parâmetros propostos por (Michalewicz, 1996). Na figura 7 pode-se visualizar um gráfico comparativo entre a média da aptidão da população antes e depois do ajuste dos parâmetros do Algoritmo Genético.

Tabela 4 - Parâmetros propostos através dos experimentos

Parâmetro	Valor Mínimo
Cruzamento	55%
Mutação	0,01%
Gerações	1000

Fonte: Os próprios autores

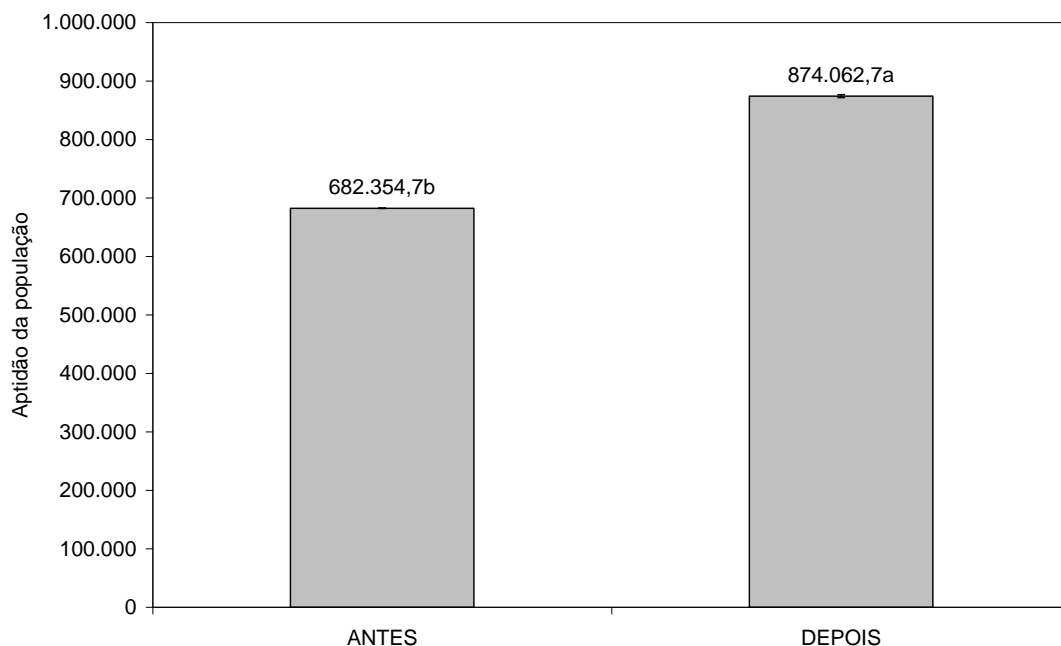


Figura 7 - Gráfico comparativo da média da aptidão da população antes e depois do ajuste nos parâmetros

Fonte: Os próprios autores

4. Conclusões

Neste trabalho, foram propostas e analisados dois experimentos, com base no algoritmo genético, a fim de aperfeiçoar os parâmetros de funcionamento do aplicativo responsável por encontrar uma solução para o problema de localização de antenas no IFES campus Cachoeiro de Itapemirim, visando o atendimento do maior número de demandas (clientes) e na utilização do menor número de facilidades (antenas), considerando as restrições de alcance de transmissão das facilidades.

No primeiro experimento foram obtidos 49.386 resultados, que corresponde a mais de quatro meses de execução do aplicativo, isto devido a variação do número de indivíduos e do número de gerações, pois as duas variáveis afetam diretamente no custo computacional e consequentemente no tempo de execução, apesar da alta capacidade de processamento do servidor e das diversas *threads* que executavam concorrentemente para aproveitar ao máximo os *multi-cores* presentes nos dois processadores. No segundo experimento, foram obtidos 36.818 resultados, que corresponde a três meses de execução do aplicativo, aproveitando ao máximo o *hardware* disponível.

Na análise dos resultados computacionais do primeiro experimento foi possível propor uma taxa de mutação que potencializa o valor de aptidão. Na análise dos resultados computacionais do segundo experimento foi possível escolher mais dois parâmetros que aprimoram ainda mais o valor de aptidão, são eles a taxa de crossover e o número de gerações, sendo que este último o resultado já era esperado, pois quanto maior o número de gerações, maior será a possibilidade de o algoritmo genético encontrar um valor da aptidão

superior, entretanto o número de gerações influencia diretamente no custo computacional e consequentemente no tempo de execução do aplicativo. Diante disto o número de gerações selecionado foi 1000, pois obtém boas soluções em tempo aceitável. Na Tabela 4 são apresentados os parâmetros sugeridos através da análise dos parâmetros. O número de indivíduos da população não obteve significância em função da aptidão, através da análise estatística realizada pelo software SAEG, dessa forma, nenhum valor é sugerido.

Referências

- Azizan, M. H., Lim, C. S., Hatta, W. A. L. W. M., & Gan, L. C. (2012). *Application of OpenStreetMap data in ambulance location problem*.
- Church, Richard L., & ReVelle, Charles S. (1976). Theoretical and Computational Links between the p-Median, Location Set-covering, and the Maximal Covering Location Problem. *Geographical Analysis*, 8(4), 406-415. doi: 10.1111/j.1538-4632.1976.tb00547.x
- Garey, Michael R., & Johnson, David S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*: W. H. Freeman & Co.
- Giger, M., Keller, D., & Ermanni, P. (2007). AORCEA - An adaptive operator rate controlled evolutionary algorithm. *Comput. Struct.*, 85(19-20), 1547-1561. doi: 10.1016/j.compstruc.2006.12.002
- Goldberg, David E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*: Addison-Wesley Longman Publishing Co., Inc.
- Grefenstette, J. J. (1986). Optimization of Control Parameters for Genetic Algorithms. *Systems, Man and Cybernetics, IEEE Transactions on*, 16(1), 122-128. doi: 10.1109/tsmc.1986.289288
- Hakimi, S. L. (1964). Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph. *Operations Research*, 12(3), 450-459.
- Hendtlass, Tim. (2001). *A Combined Swarm Differential Evolution Algorithm for Optimization Problems*. Paper presented at the Proceedings of the 14th International conference on Industrial and engineering applications of artificial intelligence and expert systems: engineering of intelligent systems.
- Horn, Jeffrey, & Goldberg, David E. (1996). *Natural niching for evolving cooperative classifiers*. Paper presented at the Proceedings of the First Annual Conference on Genetic Programming, Stanford, California.
- Jong, Kenneth Alan De. (1975). *An analysis of the behavior of a class of genetic adaptive systems*. University of Michigan.
- Khumawala, Basheer M. (1973). An Efficient Algorithm for the p-Median Problem With Maximum Distance Constraints. *Geographical Analysis*, 5(4), 309-321. doi: 10.1111/j.1538-4632.1973.tb00493.x
- Kok, Schalk, & Sandrock, Carl. (2009). Locating and characterizing the stationary points of the extended rosenbrock function. *Evol. Comput.*, 17(3), 437-453. doi: 10.1162/evco.2009.17.3.437
- Lim, Seow, & Kuby, Michael. (2010). Heuristic algorithms for siting alternative-fuel stations using the Flow-Refueling Location Model. *European Journal of Operational Research*, 204(1), 51-61. doi: <http://dx.doi.org/10.1016/j.ejor.2009.09.032>
- Linden, R. (2008). *Algoritmos Genéticos* (2 ed.): Rio de Janeiro: Brasport.
- Lorena, Luiz Antonio Nogueira. (2003). Análise Espacial de Redes com Aplicações em sistemas de Informações Geográficas. *Revista Produção Online*, 3(2).
- Man, K. F., Tang, K. S., & Kwong, S. (1996). Genetic algorithms: concepts and applications [in engineering design]. *Industrial Electronics, IEEE Transactions on*, 43(5), 519-534. doi: 10.1109/41.538609
- Michalewicz, Z. (1996). *Genetic algorithms + data structures*: Springer-Verlag.
- Murali, P., Ordóñez, F., & Dessouky, M. M. (2012). Facility location under demand uncertainty: Response to a large-scale bio-terror attack. *Socio-Economic Planning Sciences*, 46(1), 78-87.

- Paperin, Gregory. (2008). *Using holey fitness landscapes to counteract premature convergence in evolutionary algorithms*. Paper presented at the Proceedings of the 2008 GECCO conference companion on Genetic and evolutionary computation, Atlanta, GA, USA.
- Ren, X. Q., Chen, J., Guo, G., & Cai, W. P. (2009). *The application of the maximal coverage and partial coverage model in the shelter location problem*.
- Schaffer, J. David, Caruana, Richard A., Eshelman, Larry J., & Das, Rajarshi. (1989). *A study of control parameters affecting online performance of genetic algorithms for function optimization*. Paper presented at the Proceedings of the third international conference on Genetic algorithms, George Mason University, United States.
- Shen, Zuo-Jun Max, Zhan, Roger Lezhou, & Zhang, Jiawei. (2011). The Reliable Facility Location Problem: Formulations, Heuristics, and Approximation Algorithms. *INFORMS J. on Computing*, 23(3), 470-482. doi: 10.1287/ijoc.1100.0414