

ALGORITMOS GENÉTICOS NA OTIMIZAÇÃO DA SEQUÊNCIA DE OPERAÇÕES EM MÁQUINAS CNC: UM ESTUDO DO DESEMPENHO DE OPERADORES

S . M. Barcelos ¹ , S. A. A. G. Cerqueira¹

¹Departamento De Engenharia Mecânica, Universidade Federal De São João Del Rei (correspondencia-samuelm2000@gmail.com)

Resumo. *Este trabalho é dedicado ao estudo do desempenho de combinações (dois a dois) de operadores de mutação e cruzamento de um algoritmo genético, com o objetivo principal de encontrar a melhor combinação destes operadores para resolução da minimização do tempo de percurso de ferramentas em máquinas CNC. Dentre os experimentos conduzidos variamos a quantidade de furos e o grau de simetria, fatores que geram substancial aumento na complexibilidade do problema. A escolha adequada da sequência de operações em máquinas CNC, minimizando o tempo de deslocamento, é particularmente importante quando um grande número de operações rápidas deve ser realizado pela máquina. A minimização de tal objetivo traz múltiplas consequências benéficas às operações, como maximização da produção e minimização do custo.*

Palavras - chave : *caixeiro viajante, genético, furações, CNC, caminho mínimo*

1.Introdução

O crescimento industrial, juntamente com a expansão do mercado consumidor, tornou essencial o estudo de métodos a fim de aperfeiçoar processos de fabricação, o CNC (comando numérico computadorizado) nasce no fim do século XX com o objetivo de suprir as necessidades ascendentes do mercado, como qualidade, velocidade de produção e quantidade de peças produzidas.

Os CNC's são máquinas controladas por computador que podem ter diferentes números de eixos: a partir dos tornos mecânicos, com dois eixos, passando pelas fresadoras, que possuem três, até os centros de usinagem que possuem diversos eixos. Tais máquinas desempenham tarefas com maior qualidade, melhor precisão, maior rapidez, dentre outras qualidades. A programação desses equipamentos é feita através de códigos, também chamados de linguagens de máquina, um conjunto de linhas de programação que desempenham tarefas, antes executadas manualmente pelo operador, com maior precisão e rapidez, envolvendo uma grande gama de escolhas, como a sequência de operações, o momento da troca de ferramenta, a escolha da velocidade de corte, dentre outros. O uso de sistemas CAD/CAM para geração automática dessas rotinas permite melhorar significativamente a produtividade das máquinas

ferramenta. Para tanto, diversos sistemas comerciais foram desenvolvidos e estão em uso corrente.

Durante muitos anos, tais sistemas dedicaram toda a atenção à otimização de apenas uma das parcelas do tempo de operação das máquinas, aquela dedicada à operação propriamente dita. Esta, entretanto, é apenas uma parcela do tempo total de manufatura, que se divide em três parcelas: tempo de deslocamento, tempo necessário para mover a árvore entre operações e tempo de troca da ferramenta para a próxima operação; e do tempo destinado à operação propriamente dita, durante o qual a ferramenta se movimenta com velocidade de corte no ar ou no material. Já há muito tempo se sabe que, em média, cerca de 70% do tempo total do processo de manufatura é dedicado às duas primeiras atividades [01].

Em operações com grande número de furações como circuitos impressos, placas de chicanas em trocadores de calor e flanges de conexão em estrutura metálica [02], a escolha adequada da sequência de operações se torna de grande importância, e sua minimização traz inúmeros benefícios como maximização da produção, diminuição do tempo peça, dentre outros.

São relatados na literatura duas versões desse problema [03], que diferem entre si, pela volta ou não da ferramenta ao ponto de partida. No primeiro caso, quando a ferramenta volta ao ponto de partida, o problema pode ser formulado como o clássico problema de otimização do caixeiro viajante, em especial, se as operações ocorrem em um mesmo plano, não havendo restrições ao movimento do cabeçote. O segundo caso é uma variante do mesmo problema, que se torna mais interessante quando há necessidade de movimentos tridimensionais do cabeçote.

O problema do caixeiro viajante, um dos mais estudados problemas de otimização, é um problema de otimização combinatorial, classificado em termos de complexidade como NP-difícil [04]. Há registros de sua formulação na literatura já no início do século XIX, mas recebeu proposição formal apenas na década de 30 do século passado. Em termos gerais, busca-se minimizar a distância percorrida por um caixeiro viajante que deve visitar um número conhecido de cidades, ligadas por estradas cujos comprimentos são conhecidos, sem passar por mais de uma vez em qualquer delas e retornando finalmente ao ponto inicial. Para um número reduzido de cidades, a solução do problema é trivial, mas a complexidade cresce rapidamente. Há sistemas comerciais que utilizam heurísticas desenvolvidas especialmente para o problema e são capazes de encontrar boas soluções para problemas particulares que envolvem milhares de cidades.

Vários trabalhos relatam a aplicação de métodos evolucionários na resolução desse problema, dentre esses destacamos o trabalho que estudou o processo de usinagem utilizando algoritmo genético[04], outro em destaque é o emprego do método da revoada de pássaros para otimizar uma série de diferentes processos, incluindo fresagem e furação[05]. Também pode ser evidenciado na literatura o desenvolvimento de uma técnica híbrida ao combinar o recozimento simulado e os algoritmos genéticos, para otimizar o caminho descrito pela ferramenta em operações de fresagem[06].

A minimização do custo total em operações de furação, incluindo todas as etapas, empregando a busca tabu, foi objeto do estudo por [07]. Estudando o problema da furação de componentes de brinquedos em parques de diversão, em metal e plástico,[08] empregaram algoritmos genéticos, comparando o desempenho de três operadores de cruzamento.

Por sua vez, os operadores de cruzamento e mutação, objetos do presente estudo, são de grande importância na busca do menor percurso entre operações, bem como na eficiência computacional do algoritmo, na medida em que permitem viabilizar a utilização de técnicas de otimização populacional em softwares proprietários, de maior abrangência comercial e que, em sua maioria, tem por objetivo a otimização somente do tempo de operação em si, omitindo a parcela entre operações.

2 Materiais e métodos

2.1 Algoritmo Genético

Os métodos denominados otimização por populações destacam-se frente outros métodos devido a capacidade de trabalhar com informações a respeito de mais de um ponto, tratado como "informação corrente", [09]. Entre vários Algoritmos evolucionários, destacamos os (AG's) *Algoritmos Genéticos*, que reproduzem a evolução observada nos seres vivos, são caracterizados pelo desenvolvimento do conjunto de soluções-tentativas (populações) por meio da seleção natural e da genética,[10] adotando normas estocásticas de combinações e buscas, levando ao estabelecimento da população seguinte, em uma cadeia de gerações.

Os AG's possuem três operadores básicos, sendo: seleção, cruzamento e mutação. A seleção tem a capacidade de replicar e eliminar indivíduos da população a partir da aptidão produzida por cada indivíduo, observando a resposta à função objetivo, assim produzindo descendentes cada vez mais aptos, somando para melhora da população como um todo.

Já os operadores de cruzamento tem por finalidade a miscigenação entre dois ou mais indivíduos (soluções-tentativa). Por sua vez, os de mutação tem seu foco na inserção de novos indivíduos na população de forma estocástica, aumentando assim as chances de criação de novos indivíduos totalmente ou parcialmente novos.

Dividida em gerações, a execução do algoritmo genético é descrita da seguinte forma: é gerada uma população temporária, geralmente composta pelo mesmo número de indivíduos da população inicial, com base no processo descrito acima, da seleção natural, após tal processo é aplicado à população temporária os operadores de mutação e cruzamento, esses que podem ser binários, reais ou inteiros. Ao fim da geração vigente, é gerada uma população que será a entrada para próxima geração, o processo se repete até que a condição de parada seja satisfeita. Na figura (1) acima podemos observar o fluxograma básico de um algoritmo genético.

Reconhecidos pela capacidade de lidar com problemas complexos em espaços de dimensões elevadas, os algoritmos genéticos são dependentes somente da função objetivo, dispensando o cálculo de gradientes ou hessianas [11], tornando esse método apto a lidar, idealmente, com qualquer tipo de problema de otimização.

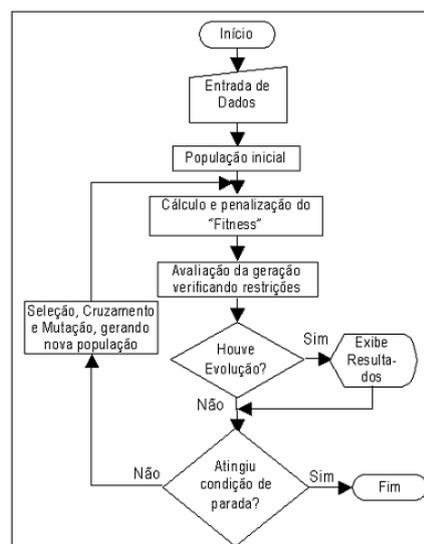


FIGURA 1. FLUXOGRAMA DE UMA AG

2.2 O problema do caixeiro viajante

O problema do caixeiro [12] tem por objetivo a minimização da soma do comprimento (d_{ij}) dos arcos percorridos pelo viajante ao visitar cada uma das n cidades do circuito, no trabalho corrente definiremos como cidades as furações que a máquina cnc deve executar. Definindo uma matriz de atribuição tal que

$$x_{ij} = \begin{cases} 1, & \text{se a cidade } j \text{ é alcançada a partir da cidade } i \\ 0, & \text{caso contrário} \end{cases}$$

O problema pode então ser escrito na forma:

$$\text{Minimizar } z = \sum_{i=1}^n \sum_{j=1}^n d_{ij}x_{ij} \text{ tal que } d_{ij} = \text{inf } \forall i = j, \quad (1)$$

$$\text{sujeito a } \sum_{i=1}^n x_{ij} = 1, \quad \forall j = 1, 2, \dots, n \quad (2)$$

$$\sum_{j=1}^n x_{ij} = 1, \quad \forall i = 1, 2, \dots, n \quad (3)$$

$$x_{ij} = \{0, 1\} \quad (4)$$

Sendo que as restrições 2 e 3 asseguram que cada cidade é visitada apenas uma vez. Ao comprimento do arco e dado valor infinito, caso não haja ligação direta entre duas cidades.

No problema da furação de placas planas, ou para peças não-planas quando não há obstáculos ao movimento da ferramenta entre dois furos coplanares, a distância entre dois pontos de coordenadas (x_i, y_i) e (x_j, y_j) é determinada pela norma euclidiana,

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \quad (5)$$

2.3 Operadores

Na solução do problema do caixeiro viajante podem ser empregados diversos operadores de cruzamento e mutação. Dentre vários descritos na literatura [13], foram escolhidos para o emprego no presente trabalho, quatro operadores de cruzamento ER,OX1,OX2 e POS e três de mutação DM, IVM e ISM, esses foram escolhidos por apresentarem bons resultados em vários trabalhos anteriormente propostos.

Os operadores de mutação distinguem entre si, em geral, pela forma de operação, enquanto DM (Displacement mutation) escolhe uma sequência de cidades e a recoloca aleatoriamente no filho mantendo a sequência, o IVM (Invetion mutation) inverte essa sequência antes de recoloca-la no individuo final, o ISM (Insetion mutation) se diferencia dos outros por

selecionar randomicamente e individualmente cada cidade a ser mutada, e recolocando-a no filho de forma individual, um a um e randômica.

Já os operadores de cruzamento possuem funcionamento mais complexo, o operador ER (Genetic Edge Recombination Crossover) é uma solução proposta ao se trabalhar com problemas de alto grau de simetria; já o OX1 (Order crossover) considera que a ordem das cidades selecionadas para a operação é a variável de maior importância, por consequência mantém nos filhos a ordem das cidades selecionados nos pais; em contraposição o OX2 (Order based crossover) não leva o fator da ordem em consideração; o POS (Position based crossover) seleciona um conjunto aleatório de posições na trajetória dos pais, porém ele impõe a posição das cidades selecionadas sobre as cidades correspondentes dos outros.

3 Parâmetros

3.1 Apresentação de casos

No presente artigo concentramos a atenção em quatro casos distintos que apresentam características distintas, a fim de encontrar a combinação do melhor par de operadores, variando a complexibilidade do problema em 10, 14, 20 e 30 furos e quanto à existência ou não de simetria.

Caso 1:

O problema de 10 furos [14] apresentado abaixo é um problema altamente simétrico nos dois eixos, apresentando os seguintes pontos no sistema de referências cartesianas {1, 0, 20}, {2, 20,20}, {3, 20,0}, {4, 0,0}, {5, 0,10}, {6, 20,10}, {7, 8,12}, {8, 12,12}, {9, 12,8},{10, 8,8}.

Caso 2:

Problema de 14 furos [15] [16] apresenta espelhamento no eixo da abcissa, sua apresentação em coordenadas cartesianas é descrito: {1, 10.0, 10.0}, {2, 10.0, 60.0}, {3, 18.0, 53.5}, {4, 18.0, 42.5}, {5, 32.32, 12.66}, {6, 37.71, 43.6}, {7, 37.71, 43.6}, {8, 62.29, 43.6},{9, 62.29, 26.4}, {10, 90.0, 10.0}, {11, 82.0, 16.5}, {12, 82.0, 27.5},{13, 72.59, 55.75}, {14, 90.0, 60.0}

Caso 3:

Problema de 20 furos apresentado abaixo em coordenadas cartesianas[17], além de maior complexibilidade não apresenta qualquer grau de simetria {1, 5.294, 1.588}, {2, 4.286, 3.622}, {3, 4.719, 2.744}, {4, 4.185,2.23}, {5, 0.915, 3.821}, {6, 4.771, 6.041}, {7, 1.524, 2.871}, {8,3.447, 2.111}, {9, 3.718, 3.665}, {10, 2.649, 2.556}, {11, 4.399,1.194}, {12, 4.660, 2.949}, {13, 1.232, 6.440}, {14, 5.036, 0.244},{15, 2.710, 3.140}, {16, 1.072, 3.454}, {17, 5.855, 6.203}, {18,0.194, 1.862}, {19, 1.762, 2.693}, {20, 2.682, 6.097}.

Caso 4:

Problema de 30 furos, segue o exemplo acima {1, 41.0, 94.0}, {2, 37.0, 84.0}, {3, 54.0, 67.0}, {4, 25.0, 62.0},{5, 7.0, 64.0}, {6, 2.0, 99.0}, {7, 68.0, 58.0}, {8, 71.0, 44.0}, {9, 54.0, 62.0}, {10, 83.0, 69.0}, {11, 64.0, 60.0}, {12, 18.0, 54.0},{13, 22.0, 60.0}, {14, 83.0,

46.0}, {15, 91.0, 38.0}, {16, 25.0, 38.0}, {17, 24.0, 42.0}, {18, 58.0, 69.0}, {19, 71.0, 71.0}, {20, 74.0, 78.0}, {21, 87.0, 76.0}, {22, 18.0, 40.0}, {23, 13.0, 40.0}, {24, 82.0, 7.0}, {25, 62.0, 32.0}, {26, 58.0, 35.0}, {27, 45.0, 21.0}, {28, 41.0, 26.0}, {29, 44.0, 35.0}, {30, 4.0, 50.0}.

3.2 Parâmetros utilizados no algoritmo genético

Os parâmetros do algoritmo genético como probabilidade de mutação, probabilidade de cruzamento, quantidade de cidades a serem cruzadas ou mutadas, tamanho da população, e quantidade de gerações são fatores de total importância para o sucesso do trabalho. Enquanto alguns desses são valores usuais, outros não são citados na literatura corrente. Para determinar tais parâmetros, foi utilizado um método experimental a fim de encontrar os melhores parâmetros a serem utilizados no presente trabalho.

Foram conduzidos experimentos a partir de um caso geral para cada um dos problemas citados acima, nesses casos, foi variada a quantidade da população e a quantidade de gerações até que a curva melhor indivíduo por gerações não mais oscilasse, ao fim, definimos a população como sendo de 150 indivíduos para os casos de 10 furos e 14 furos, 400 indivíduos para o caso de 20 furos e 800 indivíduos para o caso de 30 furos, todos operando com 5000 gerações como critério de parada por estagnação em 5000 gerações da função objetivo.

No que se refere a probabilidades de mutação e cruzamento, adotamos valores usuais para casos similares, sendo 5% para mutação e 85% para cruzamento, um item que compõe esses parâmetros são as quantidades de cidades a serem cruzadas ou mutadas dentro de cada operador, esse dado de difícil estudo foi escolhido segundo a metodologia descrita a seguir.

Inicialmente, foi utilizado um valor fixo citado na literatura [13], em seguida foi aumentado o percentual de indivíduos a serem cruzados, como podemos observar na figura(2) abaixo, para um dos operadores. Esse procedimento foi realizado para cada um dos operadores apresentando resultados similares, mostrando que o aumento do percentual a ser escolhido para a operação causaram perturbações visíveis em relação a convergência do algoritmo. Logo, foi selecionado um valor fixo de 3 cidades a serem escolhidas para a operação de mutação ou cruzamento independente do tamanho do indivíduo.

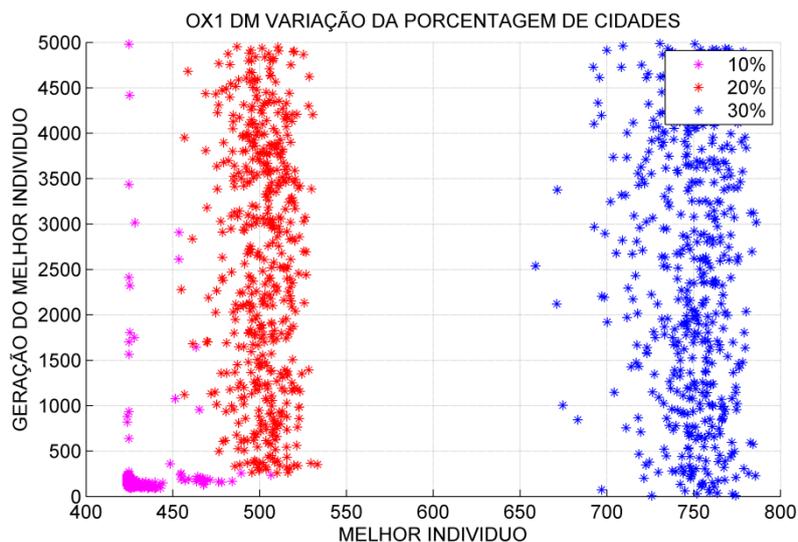
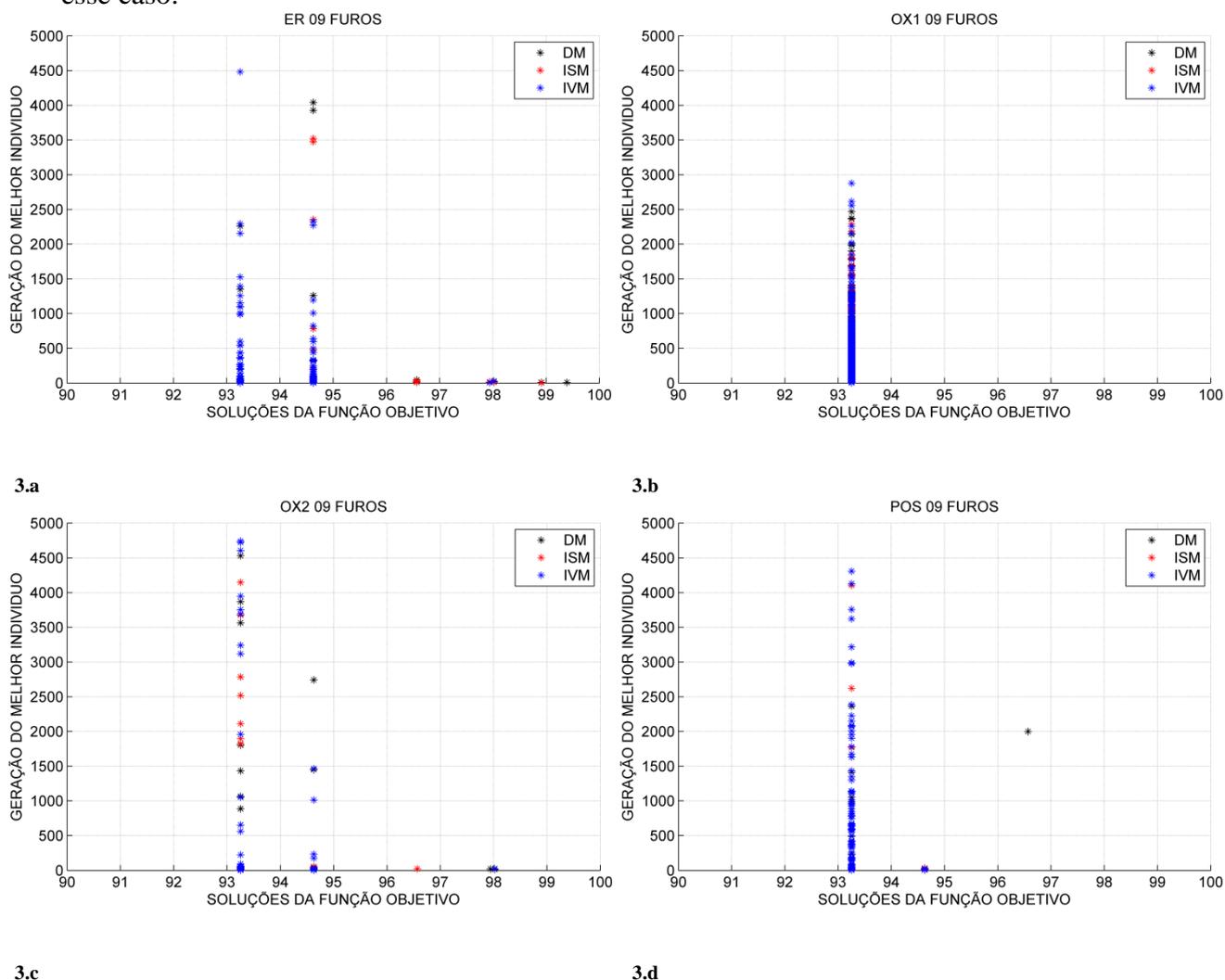


Figura 2. Variação da porcentagem de indivíduos a serem operado em relação a população.

4 Resultados

Para o problema de 10 furações, os resultados obtidos na literatura apresentam a melhor solução como 93.25 unidades de medida [18]. Para esse problema de simples solução, a maioria dos operadores de cruzamento quando combinados com os operadores de mutação, apresentaram bons resultados, isso pode ser constatado observando a figura 3, onde estão dispostos os gráficos que mostram o custo computacional bem como a eficiência da dupla de operadores. O OX1 figura 3.b apresenta os melhores resultados quando combinado com qualquer um dos operadores de mutação sendo que todos convergiram para o melhor resultado, não apresentando nenhuma discrepância neste quesito, diferenciando-se entre si somente pelo custo computacional, seguido de perto pela POS, que também apresentou bons resultados para esse caso.



3.c

3.d

Figura 3. gráficos para o problema de 09 furos , geração da melhor solução, correlacionado com solução da função objetivo, a geração pode indicar o custo computacional enquanto a solução da função indica o grau de eficiencia do par de operadores.

Com um pequeno aumento no número de furações e um grau a menos de simetria, o problema se tornou mais complexo, fato que pode ser observado na figura 4. Percebe-se grande discrepância entre as combinações de operadores nos resultados. A placa de 14 furos tem como melhor solução 290.40 unidades de medida [18], essa solução foi alcançada por todos as combinações, porém alguns tiveram maior percentual de acerto que outros, observando a figura 4.b OX1 quando combinado com DM apresentou melhor resultado dentre as outras combinações, seguido pelas combinações OX1 ISM e OX1 IVM

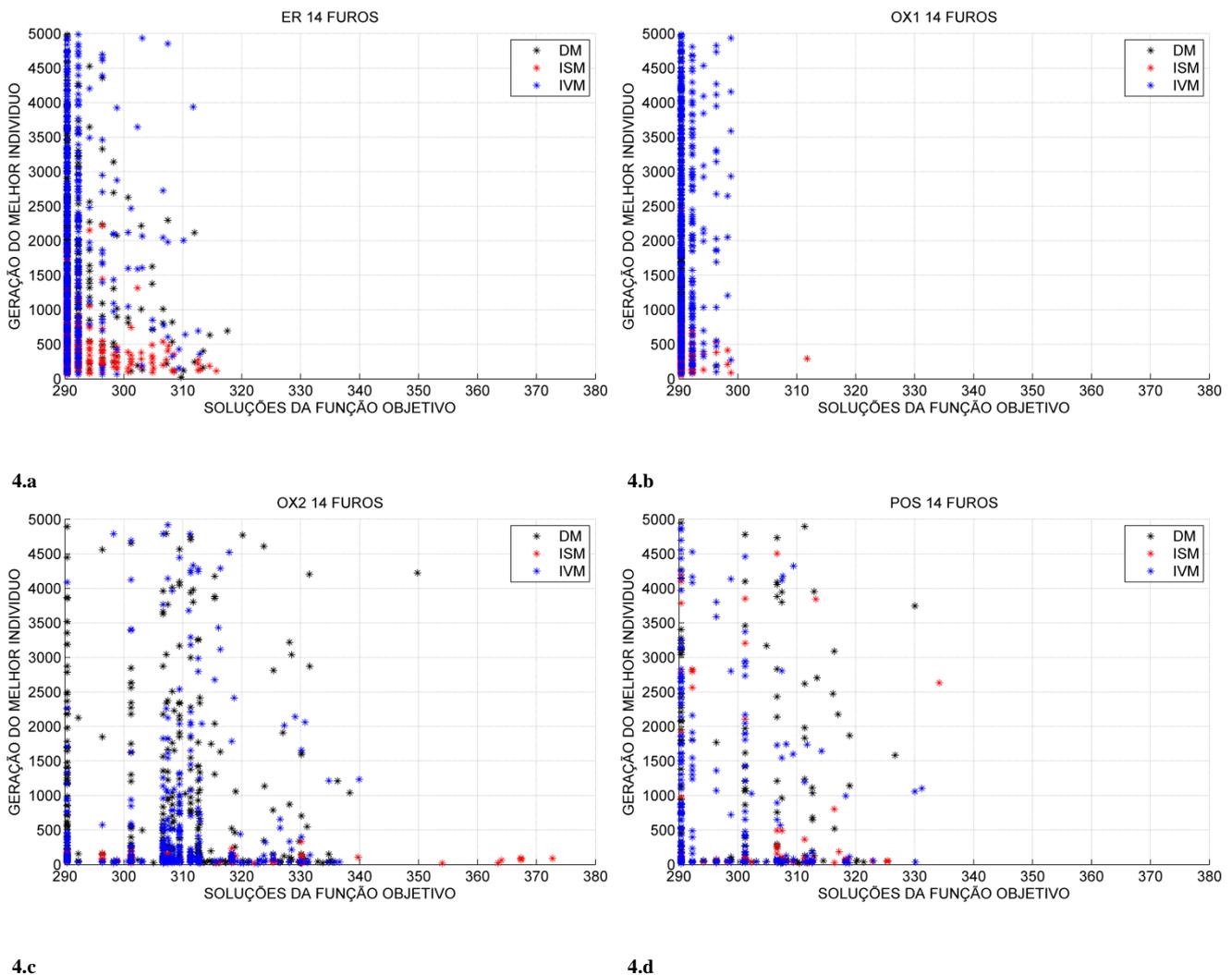


Figura 4. gráficos para o problema de 14 furos , geração da melhor solução, correlacionado com solução da função objetivo, a geração pode indicar o custo computacional enquanto a solução da função indica o grau de eficiência do par de operadores.

Analisando os resultados, percebe-se que nos dois primeiros casos avaliados, os operadores de cruzamento OX1 e ER, e os operadores de cruzamento ISM e IVM tiveram destaque. No caso de 20 furos, esses mesmos operadores também se destacaram, sendo que o operador ER requer atenção especial, pois com o crescimento da complexibilidade, sua eficiência aumentou consideravelmente. O operador de mutação ISM também merece destaque, pois além de apresentar bons resultados, também apresentou baixo custo computacional. Para este caso, o melhor resultado citado pela literatura é 24.526 unidades de medida,[17] sendo esse um problema altamente complexo e assimétrico. Por outro lado, o OX2 e POS mostraram-se ine-

ficientes perante esse problema, apresentando uma grande oscilação no conjunto de soluções do problema, como mostrado na figura 5. isso pode sugerir dificuldade de tais operadores com problemas de assimetria, pois apesar de apresentarem soluções com certo grau de discrepância nos problemas de 10 e 14 furos, ainda assim apresentaram melhor eficiência que no que se segue.

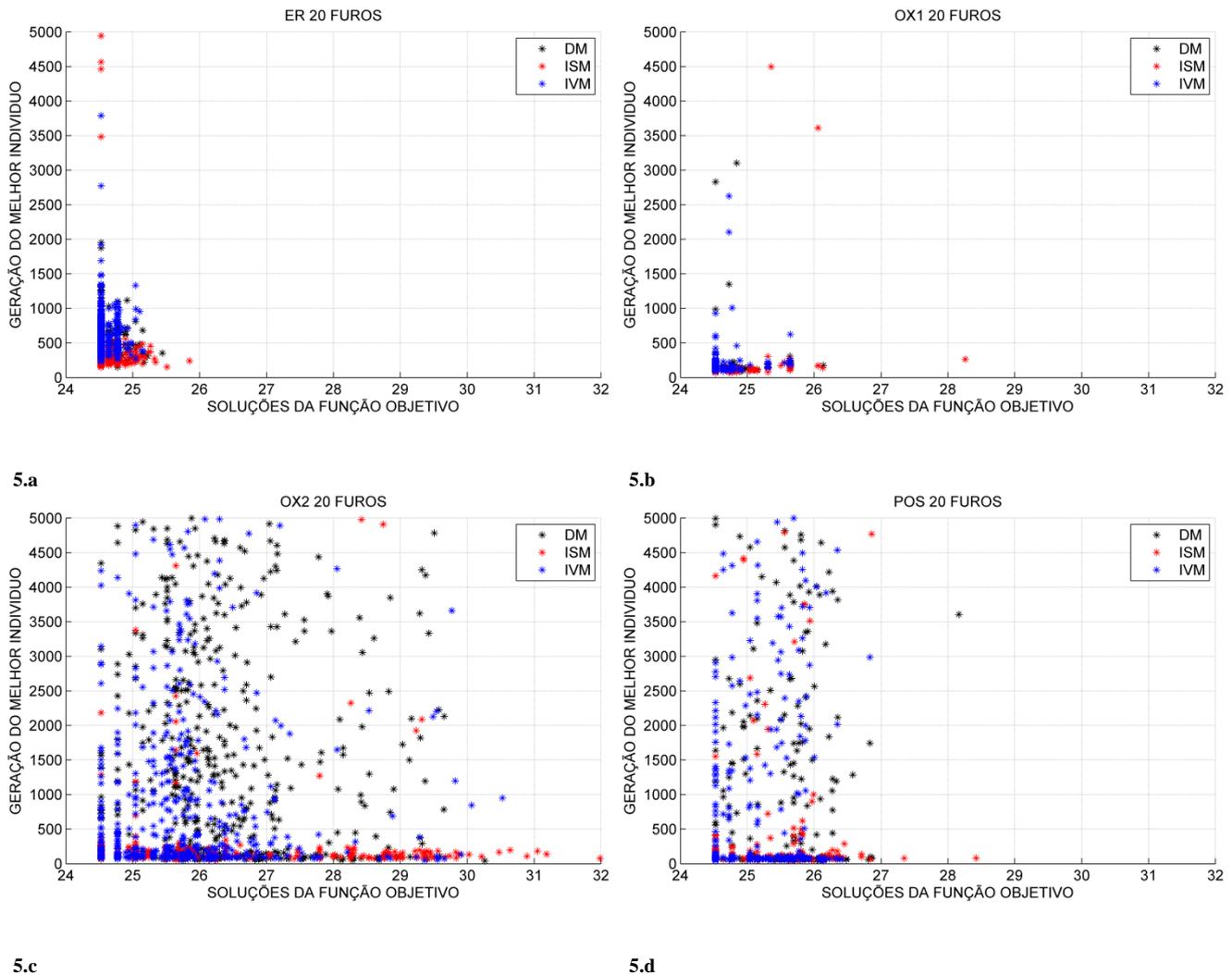


Figura 5. gráficos para o problema de 20 furos , geração da melhor solução, correlacionado com solução da função objetivo, a geração pode indicar o custo computacional enquanto a solução da função indica o grau de eficiência do par de operadores.

No caso de 30 furos, foi elevado o grau de complexibilidade seguido pela assimetria do problema, observamos um bom resultado das combinações ER-DM e OX1-DM, seguidos pela ineficiência dos operadores POS e OX2, como pode ser facilmente observado nas figuras (6.c) e (6.d), a discrepância na nuvem do conjunto de soluções sugere que tais operadores não são eficientes e possuem custo computacional elevado, resultado semelhante pode ser observado nas figuras (5.c) e (5.d).

Podemos ainda perceber que os operadores de mutação também apresentam alto grau de discrepância, o que pode ser constatado observando a miscigenação das cores que representam esses operadores, demonstrando grande variedade de soluções ruins.

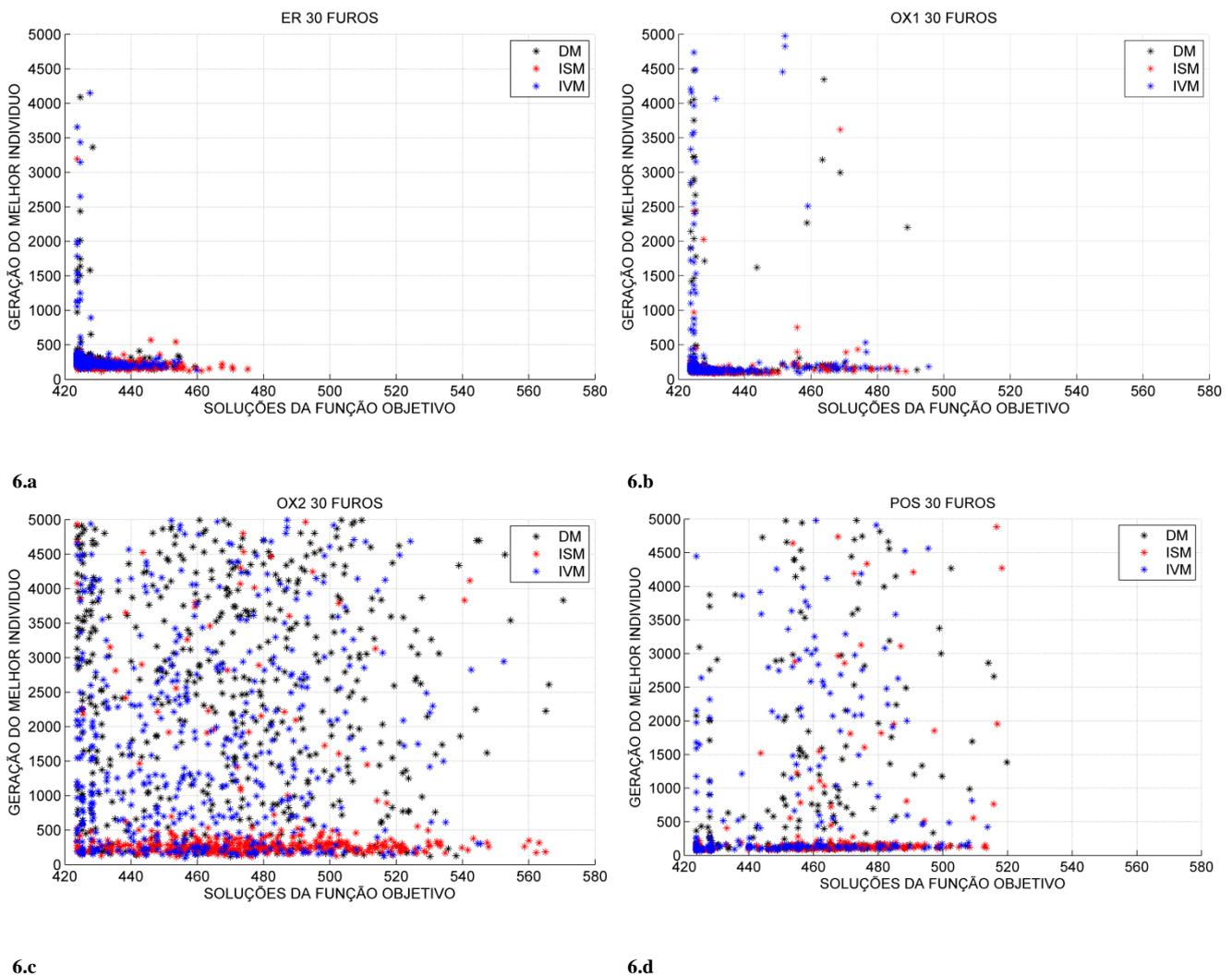
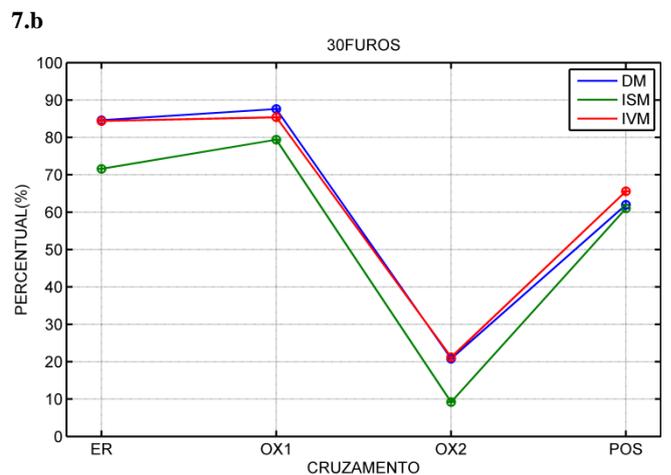
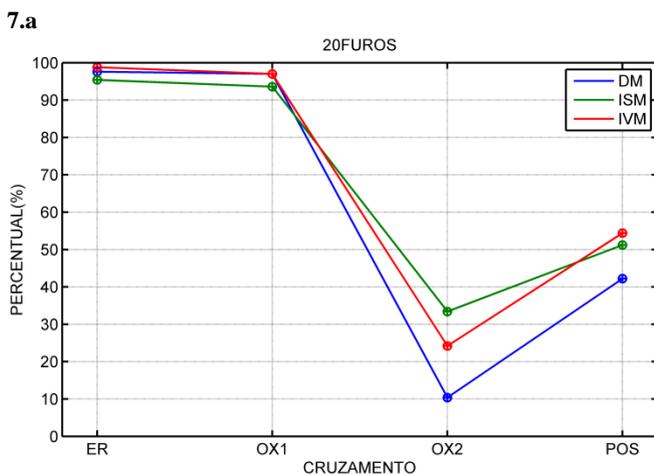
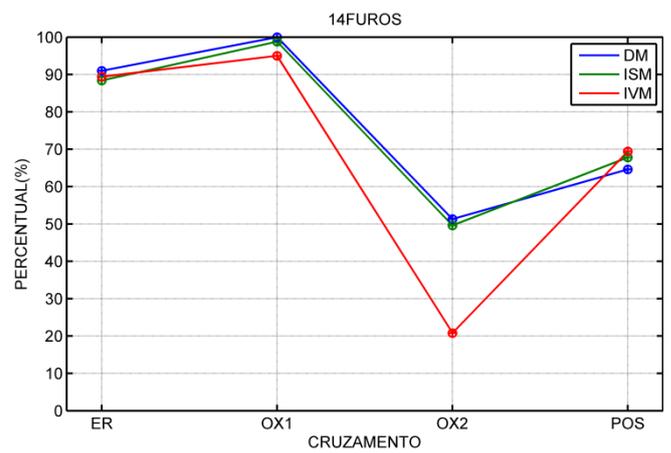
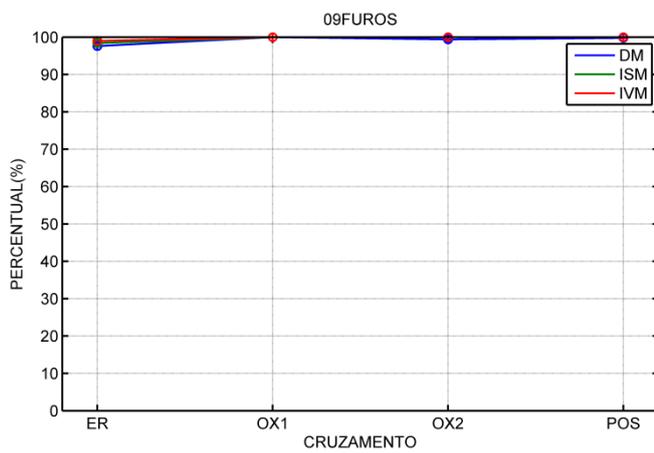


Figura 6. gráficos para o problema de 30 furos , geração da melhor solução, correlacionado com solução da função objetivo, a geração pode indicar o custo computacional enquanto a solução da função indica o grau de eficiência do par de operadores.

Para o estudo do melhor par de soluções, fez-se necessário conhecer o percentual de convergência para cada par de combinação entre operadores, sendo um de mutação e outro de cruzamento. Para tal, tabelamos tabelas 1,2,3,4, e graficamos gráficos 6.a,6.b,6.c,6.d os resultados ao longo das rodadas permitindo o erro de 2% para mais.

Ao analisar tais tabelas, podemos dizer que a combinação OX1- DM é a melhor dentre todas apresentando resultados entre 87.6 e 100%, seguido de perto pela combinação ER-IVM, que apresentou uma variação entre 84.6 e 97.6%. Esse fato se torna altamente visível ao se analisar os gráficos 6., composta pelos gráficos de percentual de convergência das combinações, é também importante perceber que os operadores de cruzamento OX1 e ER, quando combinados com IVM apresentaram boas soluções na faixa de 84.4 a 100%.

Também podemos observar o baixo índice de convergência dos operadores de cruzamento POS e OX2, que apresentam resultados na faixa de 9% a 70%, mesmo combinado com os três tipos de operadores de mutação não conseguiram apresentar resultados apresentáveis, esse fato pode ser observado nas figuras 7.c,7.d.



7.c

Figura 7. gráficos do percentual de acerto do par de operadores, as diferentes cores das linhas indicam os operadores de mutação enquanto os de cruzamento são indicados no eixo das abcissas, o percentual pode ser observado no eixo das coordenadas.

7.d

PLACA COM 09 FUROS			
CRUZ\MUT	DM	ISM	IVM
ER	97.6	98.4	99
OX1	100	100	100
OX2	99.4	100	100
POS	99.8	100	100

Tabela 1. Percentual de convergência 09 furos

PLACA COM 14 FUROS			
CRUZ\MUT	DM	ISM	IVM
ER	91	88.4	89.4
OX1	100	98.8	95
OX2	51.3	49.6	20.8
POS	64.6	67.8	69.4

Tabela 2. Percentual de convergência 14 furos

PLACA COM 20 FUROS			
CRUZ\MUT	DM	ISM	IVM
ER	97.6	95.4	98.8
OX1	97	93.6	97
OX2	10.4	33.4	24.2
POS	42.2	51.2	54.4

Tabela 3. Percentual de convergência 20 furos

PLACA COM 30 FUROS			
CRUZ\MUT	DM	ISM	IVM
ER	84.6	71.6	84.4
OX1	87.6	79.4	85.4
OX2	20.8	9.2	21.2
POS	62	61	65.6

Tabela 3. Percentual de convergência 30 furos

4 Conclusão

Portanto, podemos concluir que os operadores, quando combinados de forma correta, apresentam boa eficiência e conseguem convergir para o melhor resultado na função objetivo. Os operadores de cruzamento OX1 e ER apresentam melhores resultados perante o OX2 e o POS em todos os casos estudados, o que os torna aptos a resolução de uma grande quantidade de problemas. Com relação ao operador de mutação, pode-se concluir que o DM e IVM apresentam melhores resultados diante do ISM, que em poucos casos superou os outros dois.

Observando os gráficos (6) e as tabelas (1,2,3,4), percebe-se uma leve superioridade no casal de operadores OX1- DM, que mostraram maior eficiência tanto na convergência quanto no custo computacional, seguido de perto pelo par ER- IVM.

O uso da técnica de otimização por populações, aliada a escolha correta dos operadores de cruzamento e mutação, mostra-se uma ferramenta poderosa na solução do problema proposto, sendo que se torna importante a continuidade desse trabalho com testes práticos e comparativos para comprovação do estudo teórico nesse trabalho proposto.

Além do mais, o uso de tal artifício pode ser de grande auxílio nas técnicas de produção moderna, pois a otimização do percurso da ferramenta pode ser um forte combatente de gargalos em processos produtivos, auxilia na diminuição tempo peça e na redução de gastos energéticos.

Porém, recomenda-se o emprego de outros recursos a fim de aumentar o índice de convergências do algoritmo e diminuir o custo computacional. Soluções para tal, seria o emprego de algoritmos de busca local e populações iniciais controladas. Importante ressaltar que o emprego dos mesmos neste artigo não foi necessário, pois sairíamos do escopo do trabalho.

5 Referências

[01] Ghaiebi H. and Solimanpur M.,. An ant algorithm for Optimization of hole-making operations. *Computers & Industrial Engineering*, Vol. 52, pp. 308–319 , 2007.

[02]Adam, A., Abidin, A. F. Z., Ibrahim, Z., Husain, A. R., Yusof, Z. M., E Ibrahim,I. A particle swarm optimization approach to robotic drill route optimization. In *Proceedings of the Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation*. 2010.

[03]Qudeiri, J. A., Yamamoto, H., E Ramli, R. Optimization of operation sequence incnc machine tools using genetic algorithm. *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, 1(2):272–282, 2007.

[04]Balic, J., Kovacic, M., Vaupotic, E Bostjan. Intelligent programming of cnc turning operations using genetic algorithm. *Journal of Intelligent Manufacturing*, 17(3):331–340, 2006.

[05]Guo, Y., Mileham, A., Owen, G., E li, W. Operation sequencing optimization using a particle swarm optimization approach. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 220(12):1945–1958, 2006.

[06]Oysu, C. E Bingul, Z. Application of heuristic and hybrid-gasa algorithms to tool-path optimization problem for minimizing airtime during machining. *Engineering Applications of Artificial Intelligence*, 22(3):389 – 396, 2009. ISSN 0952-1976. doi:DOI: 10.1016/j.engappai.2008.10.005.

[07]Kolahan, F. E Liang, M. Optimization of hole-making operations: a tabu-searchapproach. *International Journal of Machine Tools and Manufacture*, 40(12):1735 – 1753,. ISSN 0890-6955. doi:DOI: 10.1016/S0890-6955(00)00024-9, 2000.

[08]Sigl, S. E Mayer, H. A. Hybrid evolutionary approaches to CNC drill route optimization. In *Proceedings of the Int. Conf. on Computational Intelligence for Modelling, Control and Automation and Int. Conf. on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06)*, volume 1, páginas 905–910. IEEE Computer Society, Washington, DC, USA, 2005. ISBN 0-7695-2504-0-01.

[09] Carrano, E. G.; Cardoso, E. P.; Takahashi, R. H. C.; Fonseca, C. M.; Neto, O. M.. Power distribution network expansion scheduling using the dynamic programming genetic algorithm (DP-GA). *IEEE Proceedings on Generation, Transmission and Distribution*, 2008.

[10] Goldberg, D., *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Company, INC, 1989.

[11] Linden, Ricardo. (2006), *Algoritmos Genéticos*, 2 ed. Brasport.

[12] Taha, H. A. *Operations Research: An Introduction (8th Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006. ISBN 0131889230.

[13] P. Larránaga, C.M.H. Kuijpers, R.H. Murga, I. inza and S. Dizdarevic.. *Genetic Algorithms for the Travelling Salesman Problem: A Review of Representations and Operators*, 1999.

[14]Huang M., Hsieh C., Arora J.S. A genetic algorithm for sequencing type problems in engineering design. *International Journal for Numerical Methods in Engineering*, 40, 3105-3115, 1997.

[15]Zhu G.-Y. Drilling Path Optimization Based on Swarm Intelligent Algorithm. *Proceedings of IEEE International Conference on Robotics and Biomimetics*, 193-196, 2006.

[16]Zhu G.-Y. and Zhang W.-B. Drilling path optimization by the particle swarm optimization algorithm with global convergence characteristics. *International Journal of Production Research*, 46 (8), 2299-2311, 2007.

[17]Wei-Bo Zhang, Guang-Yu Zhu., Comparison and application of four versions of particle swarm optimization algorithms in the sequence optimization. *Expert Systems with Applications*, *Expert Systems with Applications*, Elsevie, 38 8858–8864, 2011

[18]Aykut Kentli, Ali Fuat Alkaya.. Deterministic Approach to Path Optimization Problem. Ozean Journal of Applied Sciences 2 (2) 1943-2429, 2009.