

USO DE ALGORITMO GENÉTICO PARA SOLUÇÃO DE PROBLEMAS DE *NESTING* EM IMPRESSORAS 3D

Icaro Roberto Bedinotto

Universidade Federal de São Carlos
Rod. Washington Luís, km 235 - SP-310
icaro.roberto97@gmail.com

Roberto Fernandes Tavares Neto

Universidade Federal de São Carlos
Rod. Washington Luís, km 235 - SP-310
tavares@dep.ufscar.br

RESUMO

É notório o avanço e a disseminação da manufatura aditiva ao longo dos anos e quão impactante a mesma tem sido para a cadeia produtiva como um todo. Logo, como é habitual de todo processo de produção é de extrema relevância o controle e planejamento do mesmo, buscando manter a qualidade, reduzir tempo e custos. O presente trabalho, contextualiza, apresenta o problema e formas de solução já trabalhados na literatura. Sabendo do impacto que um melhor posicionamento das peças causa no processo (redução tempo e custos), é apresentada uma abordagem baseada em algoritmo genético para a resolução desse problema. Além disso, propõe-se uma busca local para assim encontrar uma melhor solução. Os resultados foram confrontados com os encontrados em procedimentos de solução da literatura.

PALAVRAS CHAVE. Manufatura aditiva, *Nesting*, Algoritmo genético.

Tópicos: AD& GP

ABSTRACT

The advancement and spread of additive manufacturing over the years is striking, and how shocking it has been for the production chain as a whole. Therefore, as is usual in all production processes, it is extremely important to control and plan the process, seeking to maintain quality, reduce time and costs. The present work, contextualizes, presents the problem and forms of solution already worked in the literature. Knowing the impact that a better positioning of the parts causes in the process (reducing time and costs), a genetic algorithm-based approach is presented to solve this problem. In addition, it is proposed a local search to find a better solution. The results were compared with those found in literature solution procedures.

KEYWORDS. Additive Manufacturing, *Nesting*, Genetic Algorithm.

Tópicos: AD & GP

1. Introdução

A manufatura aditiva ganhou um grande espaço no atual cenário de produção, principalmente pela capacidade de confecção de peças de alta complexidade e redução no número de etapas no processo, uma vez que a impressão consegue concentrar a eficiência que vários processos possuem. Dessa forma, a demanda por um controle e o planejamento da produção aumentou consideravelmente com o intuito de diminuir o tempo gasto e o custo na impressão de alguma peça. Algumas pesquisas foram realizadas com o propósito de criar um modelo para auxiliar no planejamento e controle da produção na manufatura aditiva, [Dvorak et al., 2018] e [Zhang et al., 2018] destacaram que uma forma de realizar redução no tempo do processo seria imprimir mais de uma peça de uma vez, já que reduziria o tempo de *setup* e o número de interrupções. Essa ideia se repetiu em outras pesquisas - assim o problema a ser resolvido era como criar um conjunto de peças com o intuito de reduzir custos e tempo. As máquinas usadas no processo (impressoras 3D) têm limitações que foram grandes entraves nas pesquisas, dentre essas limitações estão a altura máxima da peça, máxima área e volume de produção, tempo de *set-up* e de impressão.

Em algumas pesquisas, para o planejamento de qual peça a ser produzida foram consideradas as *deadlines* de cada peça. Em [Chergui et al., 2018], ficou claro que, quanto maior o número de peças, maior será a demanda por máquinas. Isso resulta em um aumento nos custos proporcionando assim em um *trade-off*. Dessa forma, no planejamento será de extrema importância saber balancear os custos e os prazos de entregas.

Um planejamento completo dos grupos de peças que devem ser impressas juntas deve-se levar em consideração as limitações do processo e as *deadlines*. Porém, isso não é tudo; um outro ponto a ser levantado é como posicionar essas peças na plataforma de impressão. A resolução desse problema pode ser determinante para o sucesso de algum processo, conforme [Piili et al., 2015] citado por [Chergui et al., 2018] somente o melhor posicionamento das peças poderia acarretar uma redução de 81-92% nos custos. Assim, esse item será de extrema importância no presente trabalho.

Dessa forma, o presente artigo objetiva desenvolver um método para resolver o problema de alocação de peças para produção em uma impressora 3D. São consideradas as seguintes características do problema: As peças serão representadas bidimensionalmente, o formato da peça é desprezado sendo todas encaradas como retângulos, o posicionamento inicial é dado pelo método *Shelf Next Fit* estendido [Jylänki, 2010] e sobreposição não é permitida, a forma de solução será por meio de algoritmo genético, além do uso de busca local.

2. Revisão bibliográfica

Manufatura aditiva (AM - *additive manufacturing*), ou também conhecida como impressão 3D, é o processo de união de materiais para realizar a confecção de objetos sendo a sua produção feita camada por camada, se opondo assim a manufatura convencional [Li et al., 2017]. O advento da manufatura aditiva se deu nos anos de 1980, todavia, recentemente que o processo ganhou maior relevância, a sua participação no mercado e aplicações industriais são em especial na indústria automotiva, aeroespacial e médica. De acordo com Bikas [Bikas et al., 2016], conforme citado por Chergui [Chergui et al., 2018] é esperado que o crescimento continue pelos próximos anos. Podemos verificar o avanço da manufatura aditiva com o exemplo da NASA, no qual 70 peças do *Mars Rover* foram impressas, além dos testes que vem sendo realizados na Estação Espacial Internacional no qual os astronautas podem imprimir as ferramentas que forem necessárias, [NASA, 2015] conforme citado por [Li et al., 2017].

A adoção da AM teve um crescimento acentuado nos últimos anos. Isso se deve ao fato que o processo proporciona a confecção de peças com alta complexidade, além de resultar em uma drástica redução no tempo de processo, teoricamente, uma vez que somente a impressão consegue

realizar a operação que várias máquinas da manufatura convencional realizariam, reduzindo assim o número de etapas e economizando tempo, mas em alguns casos a impressão é um processo muito demorado, podendo chegar até a dias o que muitas vezes não contribui para uma maior expansão do método [Dvorak et al., 2018]. Outros pontos positivos que existem na AM é a menor perda de material, maior eficiência no uso de recursos e a produção flexível [Li et al., 2017].

2.1. *Nesting*

O posicionamento das peças é fator chave para melhor utilização das máquinas usadas do processo podendo contribuir consideravelmente na redução de custos e de tempo. *Nesting* é a forma de otimizar o uso do espaço de impressão com o intuito de encaixar várias peças em um único processo sem causar falhas [Dvorak et al., 2018]. Em outras palavras, como posicionar as peças na máquina para que se utilize da melhor forma o espaço para assim, aumentar a produção, reduzir custos e tempo.

Os tipos de processos que existem na manufatura aditiva podem ser divididos em dois grupos [Zhang et al., 2016]: o primeiro grupo são aqueles que precisam de algum suporte para as estruturas como é o caso do SLA; já o segundo seriam os processos que se utilizam de matéria prima não depositada para fazer o suporte. Esse suporte tem o intuito de separar as peças ou até mesmo segurar elas durante o processo. Esse grupo de peças interferem no modo que devemos pensar no projeto, sendo que o intuito é elaborar uma forma de como encontrar um *layout* ideal, ou próximo a isso mantendo a qualidade da peça, buscando compactar os objetos atendendo as restrições pré-estabelecidas [Zhang et al., 2018], precisamos nos preocupar com a qualidade, uma vez que erro no posicionamento podem comprometer a estrutura da peça. No primeiro grupo não podemos colocar uma peça sobre a outra, uma vez que a estrutura do objeto de baixo pode ser danificada e quando usamos de suporte entre camadas a sua remoção no pós-processo é muito difícil, dessa forma processos que envolvem esse grupo devem ser encarados como bidimensionais. Sobre o segundo grupo, as peças podem ser sobrepostas, assim podemos encarar como um problema tridimensional, todavia, a transferência de calor entre outros fatores físicos e químicos podem comprometer os objetos em processos como o SLS.

Foi mencionado anteriormente a sobreposição de peças e seus efeitos na estrutura dos objetos, a sobreposição é conhecida como *overlap* e em muitos casos busca ser evitada. No estudo de [Zhang et al., 2016] o algoritmo somente termina quando o número de *overlaps* é o mínimo, já em [Zhang et al., 2018] o resultado somente é aceito quando não há sobreposição (*overlap*=0). Dessa forma, conseguimos notar a importância desse item no *nesting*.

Para simplificar o problema, os autores usados como base para o presente trabalho realizaram o *nesting* em duas dimensões, desconsiderando assim alguns pontos que tornariam o problema mais complexo e com maior exigência computacional. Para realizarem isso, ambos os autores fizeram a projeção de uma peça tridimensional em um plano, assim ficou mais simples saber se houve *overlap*, ou se alguma peça ultrapassou os limites da plataforma de impressão.

A orientação das peças é um assunto que não foi muito trabalhado nos textos que serviram como base, com exceção de [Zhang et al., 2016], o processo empregado nesse estudo serviu como base para pesquisas posteriores. O método consiste em primeiramente identificar características relevantes de uma determinada peça, depois encontrar possíveis orientações e juntá-las em um grupo que posteriormente é refinado e por fim escolher qual a orientação é a melhor para aquele determinado problema, sendo respeitado as restrições pré-estabelecidas.

Para começar o *nesting* precisamos de um posicionamento inicial. Para isso utilizaremos do *Shelf Next Fit* (SNF) estendido [Jylänki, 2010], a escolha desse método foi por ser o mais simples apresentado, além de possuir bons resultados.

2.2. Algoritmo genético

O Algoritmo genético (GA - *Genetic Algorithm*), é uma metaheurística que tem por base a teoria da seleção natural de Darwin, e é usado para a solução de vários problemas de alta complexidade computacional [McCall, 2005]. Em ambos os artigos que serviram como base para o presente trabalho ([Zhang et al., 2016] e [Zhang et al., 2018]), o GA foi usado como forma de melhorar um *layout* que obteve um posicionamento menos adequado que um gerado anteriormente. Por estar baseado na biologia, muitas dos elementos desta são empregados nesse método, tais como: cromossomo - elemento do GA que representa uma instância da solução - indica um *layout*. Ainda, cada gene descreve a posição de perfil de cada peça, sendo os parâmetros a posição em x , y e o ângulo (θ) que a peça rotaciona com relação a posição inicial. Os passos que são seguidos por um algoritmo genético serão explicados abaixo, por exemplo, por [Zhang et al., 2018]:

- **Codificação:** Para todas as peças são atribuídas posições em coordenadas já destacadas, sendo x e y limitadas pela área de impressão e θ variando no intervalo $[0, 360)$.
- **Crossover:** Troca de genes de mesma posição entre dois cromossomos. Buscando assim uma melhora no *layout*, já que está relacionando duas soluções encontradas anteriormente.
- **Mutação:** Nessa operação ocorre a alteração de um determinado gene dentro de um intervalo de valores estabelecidos buscando a melhora do cromossomo.
- **Seleção:** Envolve a escolha de algum determinado cromossomo, para a realização de algum determinado processo de evolução, em outras palavras, forma de selecionar os cromossomos pais para que seja realizado algum processo de evolução (*crossover* ou mutação). [Zhang et al., 2018] expôs dois métodos de seleção. O primeiro é a política do melhor indivíduo, no qual se armazena o melhor resultado para ser usado no lugar do pior resultado da evolução, já o segundo é o método da roleta, no qual os pais são selecionados a partir de um sorteio sendo que sua probabilidade de escolha aumenta conforme a capacidade do mesmo [Zhang et al., 2018].
- **Função *Fitness*:** Conforme [Mallawaarachchi, 2017], é a forma usada para avaliar a qualidade de um determinado resultado. No texto de [Zhang et al., 2016] os autores usaram essa função buscando diminuir o número de *overlaps* (sobreposição de peças) melhorando assim a resposta.

Nos baseando nas definições expostas acima, podemos entender melhor o processo de um algoritmo genético. Em um primeiro momento as peças são posicionadas na plataforma de impressão e esse posicionamento é convertido para o formato de cromossomo (codificação), posteriormente é calculado a Função *Fitness* para determinar o atual estado, depois acontece uma seleção de quais cromossomos serão usados na evolução (seleção) e posterior a isso acontece o processo evolutivo propriamente dito (*crossover* e mutação) e por fim precisamos avaliar o quão bom foi esse avanço do cromossomo (Função *Fitness*).

3. O Algoritmo proposto

A figura 1 apresenta o fluxograma do algoritmo genético que foi elaborado para atender o presente trabalho:

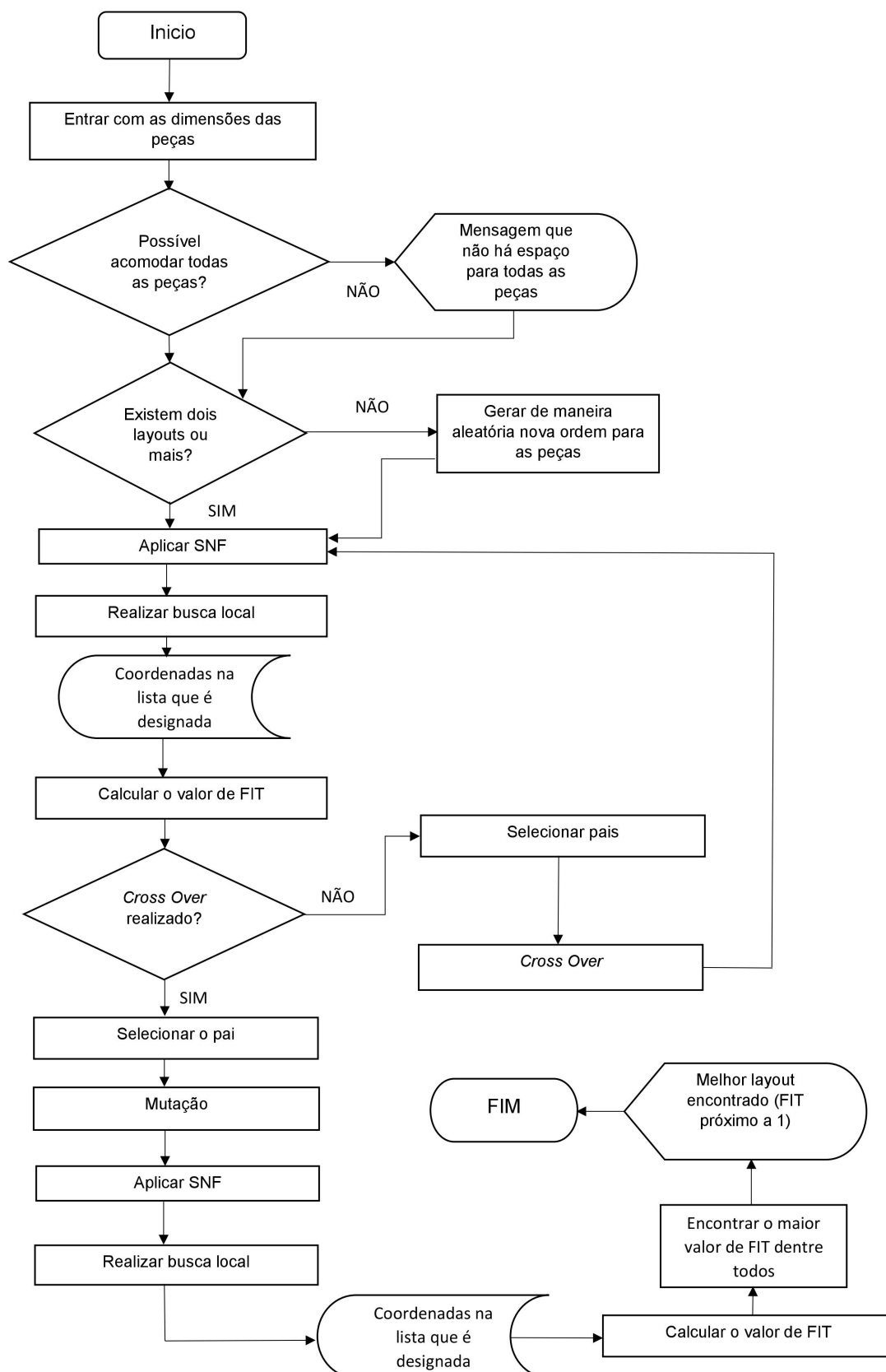


Figura 1: Algoritmo genético

Em termos gerais, esse algoritmo altera a ordem das peças e utiliza a heurística SNF em conjunto com a heurística desenvolvida. Inicialmente, são recebidas as dimensões das peças. Posteriormente, as peças são alocadas pelo método SNF sem nenhuma ordenação. Depois aplica-se a heurística proposta buscando aumentar o número de peças na máquina, pela melhoria do posicionamento. Por fim, é gerada uma nova ordem e os passos anteriores são repetidos. Tendo uma lista com dois pais, os usamos para realizar o *crossover*, e depois aplicamos o SNF e a heurística, nas ordens geradas. Agora com uma população de quatro membros, escolhemos o melhor para fazer a mutação, que após realizada, a ordem gerada passará pelo SNF e pela heurística, posterior a todo esse processo no total temos cinco *layouts* diferentes.

Assim, algumas funções são básicas, a primeira é o SNF que somente é o posicionamento das peças em uma dada ordem, implicando na codificação de um cromossomo. Já a segunda, é busca local que melhora o resultado obtido na função anterior. Devemos sempre calcular o *fit* (resultado obtido na função *fitness*) de um cromossomo gerado, já que será de extrema relevância na escolha de quem participará das operações evolutivas, sendo essa seleção feita na função de mesmo nome. Para o *crossover* são escolhidos os dois cromossomos com melhores valores de *fit*, já para a mutação será somente um escolhido.

A seguir, para melhor compreensão iremos expor um exemplo numérico. Selecionados quem serão os cromossomos que participam do *crossover*, um número aleatório é gerado e esse corresponde a posição da peça que será trocada entre os cromossomos, por exemplo, o cromossomo A tem como ordem de peças [0, 1, 2, 3, 4] já o cromossomo B tem [2, 3, 1, 0, 4], então no *crossover* é escolhido de maneira aleatória a posição 1, assim é feita a troca e estabelecemos que C será [0, 3, 2, 1, 4] e D [2, 1, 3, 0, 4]. Note que foi trocado a posição das peças 1 (por conta do *crossover*) e 3 (para não repetir as peças presentes) dos cromossomos A e B. Dessa forma, ficamos com quatro novas formas de posicionamento. Após realizado o *crossover*, as novas ordens passam pelo SNF e pela busca local. A operação de *crossover* que foi escolhida para esse caso, pode ser definida como *Single Point Crossover* [Lim et al., 2017].

Já no operador de mutação, o cromossomo que possui melhor *fitness* é escolhido para passar pelo processo que consiste na escolha aleatória da posição de duas peças que serão transferidas para o final da ordem. Por exemplo, o cromossomo C foi o escolhido para passar pelo processo de mutação, inicialmente as peças estão ordenadas como [0, 3, 2, 1, 4], então de maneira aleatória são escolhidas as posições 2 e 3, assim é realizado o processo dando origem a ordem [0, 1, 4, 3, 2], que passará pelo SNF e pela heurística dando origem ao cromossomo E, concluindo assim o método e totalizando cinco cromossomos (cinco *layouts* diferentes).

4. Buscas locais

Um algoritmo de busca local tenta encontrar soluções melhoradas se baseando em perturbações locais [Johnson et al., 1988], em outras palavras, através de mudanças em um atual cenário se busca a chegar a um resultado melhor. Nesse presente artigo, foi criado uma heurística que usa dessa ferramenta para atingir o objetivo que é melhorar a alocação de peças em uma plataforma de impressão 3D. No algoritmo 1 está o pseudocódigo da busca local elaborado, sendo *lista* as dimensões de todas as peças, *plataforma* a posição de todas as peças e *Extrapolacoes* as peças que não couberam na plataforma aplicando o SNF.

Esse método em muitos casos não conseguiu chegar em um resultado ótimo, porém, foram atingidos resultados satisfatórios que serão abordados no próximo capítulo.

Algoritmo 1: BUSCA LOCAL

```
Entrada: lista, plataforma, Extrapolacoes
Saída: plataforma

1 início
2   repita
3     se  $x == 0$  e  $y \neq 0$  então
4       Encontrar o menor valor de y possível
5       Atribuir o novo valor a y
6       alterar na lista plataforma
7     fim
8     se  $x \neq 0$  e  $y \neq 0$  então
9       se Possível encaixar a peças ao mesmo nível y da anterior sem sobrepor
10        as demais então
11        Atribuir o valor de y da peça anterior a y da peça atual
12        Encontrar a nova coordenada x
13        Alterar na lista plataforma
14      fim
15      senão
16        Mantém onde está
17      fim
18  até  $i \leq \text{número de peças}$ ;
19  repita
20     $i = \text{número de peças} - 1$ 
21    repita
22      se Extrapolacoes[0] couber acima da peça i da plataforma sem sobrepor
23       nenhuma e passar do limite então
24       Atribui a y da peça que será encaixada o valor da coordenada y da
25       peça abaixo mais a altura
26       Colocar Extrapolacoes[0] na lista plataforma
27       Deletar Extrapolacoes[0]
28       Break
29     fim
30    até  $i == 0$  ou  $\text{tamanho(Extrapolacoes)} > 0$ ;
31    se Não encaixou nenhuma peça então
32      Break
33    fim
34  até  $\text{tamanho(Extrapolacoes)} == 0$ ;
35 fim
36 retorna plataforma
```

5. Resultados e análises

Para analisar o comportamento das técnicas propostas, essa seção compara os resultados na presença e na ausência da busca local. Em um primeiro momento, observando o comportamento

com relação à alteração do número de peças a serem posicionadas (figura 2) e à quantidade de cromossomos em uma população (número de gerações) (figura 3).

Baseado no que foi apresentado na bibliografia, esse trabalho gerou um conjunto de 120 arquivos contendo as dimensões das peças divididos em 4 grupos, cada um possuindo diferentes quantidades de peças a serem posicionadas (15, 25, 30 e 50 peças).

Para identificarmos a evolução dos resultados vamos usar como parâmetro o valor calculado na equação 1:

$$Fitness = \frac{fit(x) - max(fit)}{max(fit)} \quad (1)$$

Sendo,

x - número de membros da população

$fit(x)$ - melhor valor da função objetivo para um dado x

$max(fit)$ - maior valor para função objetivo dentre todos encontrados [0, 1], independente do valor de x

Ainda para a análise dos resultados, como destacado anteriormente, as dimensões das peças, assim como da plataforma de impressão (10 x 10), foram obtidas na literatura [Amini, 2014], mantendo a grandeza física do texto original (polegadas).

Cada instância o problema foi executada considerando o número de cromossomos como sendo (5, 10, 100, 200, 300 ou 500) e ligando e desligando a busca local. Na figura 2, é possível observar que para todos os casos o uso da busca local resultou em uma mediana próximo a zero, diferente da execução sem o algoritmo que se distanciou consideravelmente. Na tabela 1 podemos verificar isso novamente, mas através da média, que em todos os casos para o uso da busca local foi mais próximo de zero quando compara a ausência da mesma, assim podemos garantir que quanto mais peças houverem no problema maior será a vantagem do uso da busca local, mostrando eficiência do método.

Referente à figura 3, podemos notar que para todos os casos a ação da busca local é significativa para um melhor resultado, além do que quanto mais gerações estão presentes maior é a distância do resultado da solução sem busca local do 0, diferente de quando a mesma é aplicada, permitindo um resultado de *fitness* sempre próximo a zero (com exceção de 5 gerações). Na tabela 2, podemos notar esse fenômeno novamente, já que sempre o valor obtido com aplicação de busca local é inferior (mais próximo a zero) do que quando a ferramenta está ausente, mostrando novamente efetividade no método desenvolvido.

Algo que ficou muito claro tanto na análise das figuras 2 e 3, quanto das tabelas 1 e 2 é quando não é aplicada a busca local ocorre uma maior discrepância entre os resultados, quando comparamos ao resultados que houve aplicação do algoritmo, isso deve-se ao fato que o método desenvolvido sempre está buscando compactar mais as peças deixando menos objetos de fora o que por consequência não permite uma grande variação de peças não encaixada diminuindo assim as discrepâncias.

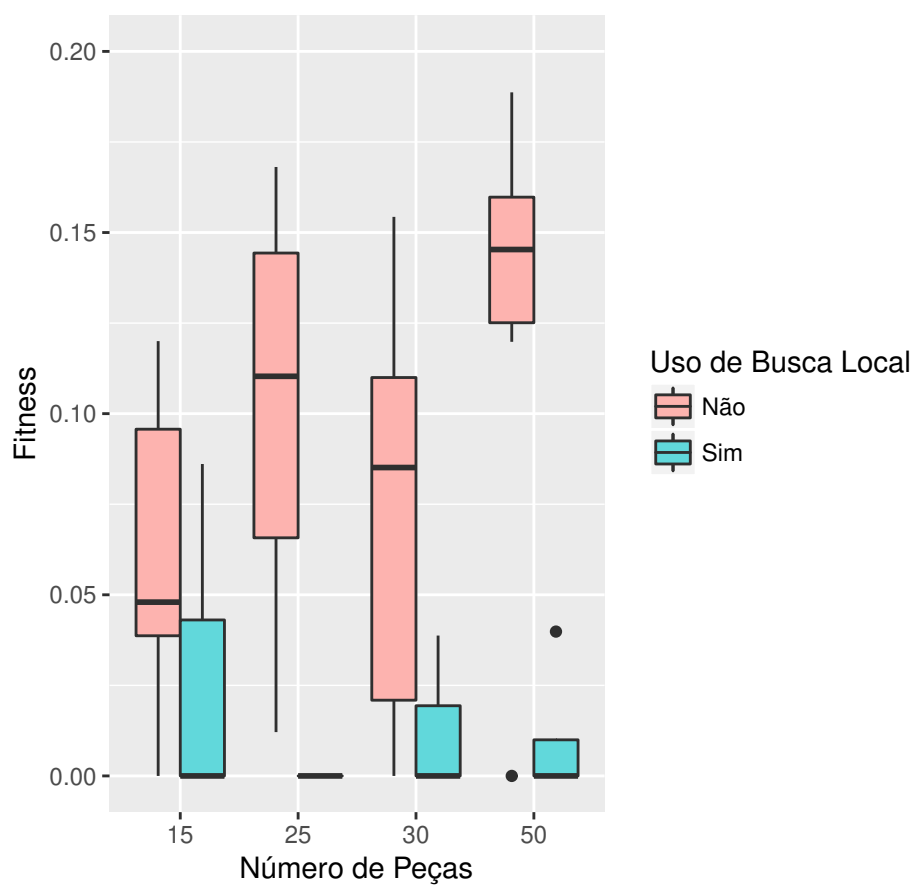


Figura 2: Comparação entre usar ou não a busca local, com relação ao número de peças

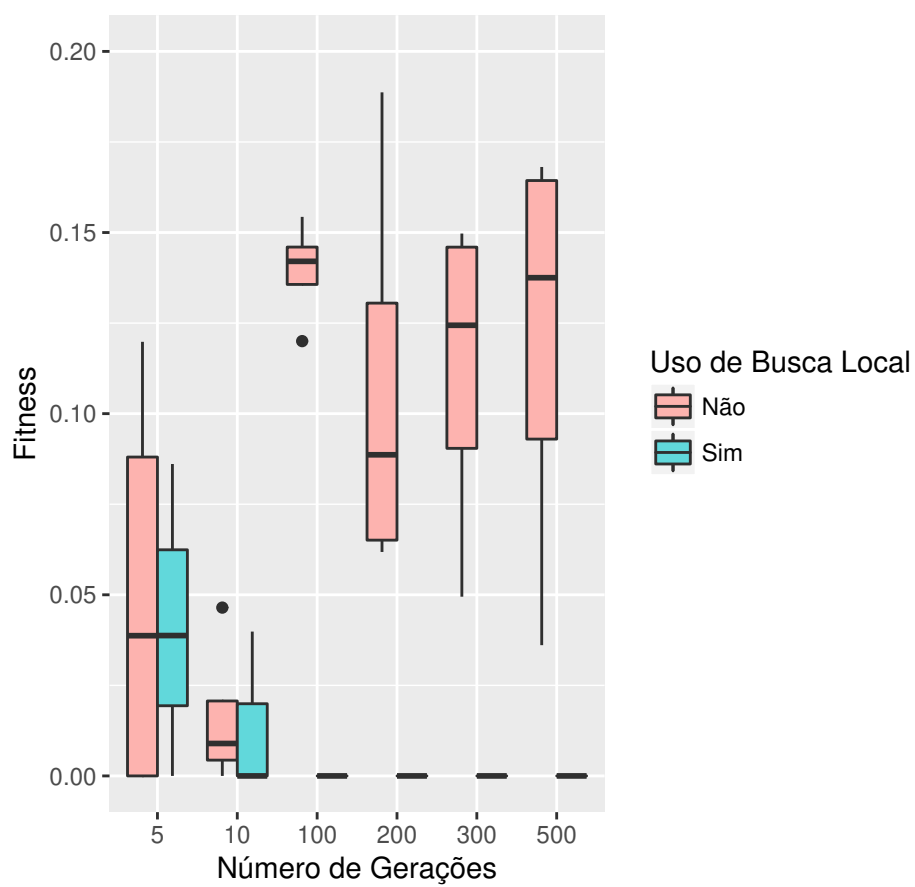


Figura 3: Comparação entre usar ou não a busca local, com relação ao número de gerações

Tabela 1: Comparação entre usar ou não a busca local (média e desvio padrão), com relação ao número de peças)

Nº de peças	Sem Busca Local		Com Busca Local	
	Média	Desvio	Média	Desvio
15	6,05E-02	0,0462	0,01435	0,03515
25	0,101122	0,0603	1,06E-07	6,74E-08
30	7,37E-02	0,0616	0,006458	0,01581
50	0,127042	0,0663	6,64E-03	0,01626

Tabela 2: Comparação entre usar ou não a busca local (média e desvio padrão), com relação ao número de gerações)

Nº de gerações	Sem Busca Local		Com Busca Local	
	Média	Desvio	Média	Desvio
5	4,93E-02	0,059517	0,031212	0,040898
10	1,61E-02	0,02084	0,009961	0,019922
100	1,40E-01	0,014315	-3,5E-08	1,1E-07
200	1,07E-01	0,058875	9,08E-08	3,48E-08
300	1,12E-01	0,046423	-3,2E-08	1,04E-07
500	1,20E-01	0,061309	-6,67E-09	7,03E-08

6. Considerações finais

Esse trabalho apresenta um algoritmo para posicionamento de peças em impressoras 3D. Como foi visto na literatura, o melhor posicionamento das peças na plataforma de impressão pode contribuir consideravelmente para um melhor resultado com relação ao custos e tempo de produção. Assim, o presente trabalho teve como objetivo encontrar uma melhor forma de realizar o posicionamento, usando de algoritmo genético e de uma busca local desenvolvida. Em um primeiro momento foi realizada a revisão da literatura, que permitiu conhecer trabalhos já desenvolvidos que foram de grande importância para a elaboração desse. Depois, foi exemplificado método criado, abordando desde posicionamento inicial (*Shelf Next Fit*), passando pelo algoritmo genético até chegarmos na busca local. Por fim, executamos alguns testes que nos permitiram observar que o algoritmo desenvolvido é eficiente, uma vez que, quando comparamos o resultados na presença e ausência da busca local obtivemos resultados muito mais satisfatórios no primeiro caso.

Nossos resultados mostram que a busca local proposta influencia de forma positiva e significativa o algoritmo genético adotado. A influência do número de gerações, embora presente, se estabilizou em 200 indivíduos. O algoritmo de busca local não se comporta de forma linear quanto ao número de peças, obtendo resultados significativamente melhores quando busca resolver instâncias com 25 peças.

Como futuros trabalhos, espera-se realizar outras comparações com algoritmos da literatura, aprimorar os procedimentos de busca local e tornar o problema mais realista usando peças com formatos mais complexos.

Referências

Amini, M. (2014). Time estimation for additive manufacturing. Master's thesis, Texas State University.

- Bikas, H., Stavropoulos, P., e Chrysosolouris, G. (2016). Additive manufacturing methods and modelling approaches: A critical review. *The international Journal of advanced Manufacturing Technology*, 83:389–405.
- Chergui, A., Hadj-Hamou, K., e Vignat, F. (2018). Production scheduling and nesting in additive manufacturing. *Computers and Industrial Engineering*, 126:292–301.
- Dvorak, F., Micali, M., e Mathieu, M. (2018). Planning and scheduling in additive manufacturing. *Inteligencia Artificial*, 21:40–52.
- Johnson, D. S., Papadimitriou, C. H., e Yannakakis, M. (1988). How easy is local search? *Journal of Computer and System Sciences*, 37:79 – 100.
- Jylänki, J. (2010). A thousand ways to pack the bin - a practical approach to two-dimensional rectangle bin packing. <https://bit.ly/2XVazXk>. Acessado: 2019-01-22.
- Li, Q., Kucukkoc, I., e D., Z. Z. (2017). Production planning in additive manufacturing and 3d printing. *Computers and Operations Research*, 83:157–172.
- Lim, S. M., Sultan, A. B. M., Sulaiman, M. N., Mustapha, A., e Leong, K. Y. (2017). Crossover and mutation operators of genetic algorithms. *International Journal of Machine Learning and Computing*.
- Mallawaarachchi, V. (2017). How to define a fitness function in a genetic algorithm? <https://bit.ly/2Dct1MI>. Acessado: 2019-02-05.
- McCall, J. (2005). Genetic algorithms for modelling and optimisation. *Computacional and Applied Mathematics*, 184:205 – 222.
- NASA (2015). Space station 3-d printer builds ratchet wrench to complete first phase of operations. https://www.nasa.gov/mission_pages/station/research/news/3Drachet_wrench/. Acessado: 2015-03-21.
- Piili, H., Happonen, A., Vaisto, T., Venkataramanan, V., Partanen, J., e Salminen, A. (2015). Cost estimation os laser additive manufacturing of stainless steel. *Physics Procedia*, 78:388–396.
- Zhang, Y., Bernard, A., Harik, H., e Fadel, G. (2018). A new method for single-layer-part nesting in additive manufacturing. *Rapid Prototyping Journal*, 24:840–854.
- Zhang, Y., Gupta, G. R., e Bernard, A. (2016). Two-dimensional placement optimization for multi-parts production in additive manufacturing. *Robotics and Computer-Integrated Manufacturing*, 38:102–115.