
REVISÃO SISTEMÁTICA DE META-HEURÍSTICAS PARA FLEXIBLE JOB SHOP SCHEDULING PROBLEM (FJSSP)

Rebeca Emi Ito¹
Simone Sawasaki Tanaka²

RESUMO

O *flexible job shop scheduling problem* (FJSSP) é um obstáculo presente na computação e na manufatura, onde ambos procuram otimizar o tempo de produção. Na computação ele apresenta uma complexidade do tipo *NP-Hard*, onde deve ordenar n jobs com m máquinas de maneira que o processamento seja o mais rápido e eficiente, e que a seleção da operação e máquina não convirjam com as outras jobs. Para solucionar o FJSSP são propostos a utilização de meta-heurísticas, que são algoritmos para resolver problemas diversos, diferente da heurística que visa resolver um problema em específico e hiper-heurísticas que selecionam heurísticas e meta-heurísticas que melhor soluciona o problema. Dentro da meta-heurística o algoritmo genético (GA) é a mais utilizada, devido a sua implementação simples, métodos como o *ant colony optimization* (ACO) e *quantum particle swarm optimization* (QPSO) são explorados, desenvolvendo novos algoritmos com melhores resultados. Este trabalho tem como objetivo realizar uma revisão sistemática de meta-heurísticas e hiper-heurísticas para o FJSSP. Através de leitura e análise de artigos e trabalhos foi possível observar que o GA é um dos métodos utilizado pelos pesquisadores dessa área para a resolução do FJSSP.

106

Palavras-chaves: FJSSP; meta-heurística; JSSP

ABSTRACT

The flexible job shop scheduling problem (FJSSP) is an obstacle in the computation and manufacture fields, having the objective of optimizing production time. The FJSSP has a complexity of NP-Hard in the computation, where it must schedule a sequence of n jobs and o operations with m machines resulting in a faster and more efficient processing time, without having the same operation or machine processing at the same time as the other jobs. To solve FJSSP it's often used meta-heuristics, they are algorithms that solve multiple problems, unlike heuristics that solve specific problems and hyper-heuristics that choose the best heuristic or meta-heuristic to solve the problem. In the meta-heuristic field, the genetic algorithm (GA) is more used among researchers. The reason it's most likely to be because the implementation of the GA is

¹ Graduanda do curso de Ciência da Computação do Centro Universitário Filadélfia - UniFil. emi.ito@edu.unifil.br

² Orientadora: Professora Mestre do curso de Ciência da Computação do Centro Universitário Filadélfia - UniFil. simone.tanaka@unifil.br

easier and simpler than the others in that field. Methods like ant colony optimization (ACO) and quantum particle swarm optimization (QPSO) are also used to solve FJSSP, they are also used to generate new algorithms that are faster and better. The objective of this paper is to make a systematic review of meta-heuristics and hyper-heuristics for FJSSP. After reading and analyzing papers and projects, it was possible to observe that GA was the most used and researched to solve FJSSPs.

Keywords: FJSSP, Meta-heuristic, JSSP.

INTRODUÇÃO

Job shop scheduling problem (JSSP) é um dos obstáculos presentes na computação com a complexidade *NP-Hard*, onde é retirado o problema da manufatura, cujo objetivo é encontrar uma sequência de operações e máquinas que processem todos os *jobs* o mais rápido e eficiente possível, possuindo o tempo de processamento de cada operação é pré-determinado para cada máquina.

Dentro do JSSP há outras áreas que surgiram dela com maiores níveis de complexidade, sendo um deles o *flexible job shop scheduling problem* (FJSSP) que possui um conjunto de possibilidades de máquinas que podem processar a operação ao invés de uma combinação estática.

Para a solução do FJSSP são utilizados meta-heurísticas, que são algoritmos responsáveis por resolver problemas mais genéricos, sendo possível solucionar diferentes problemas que são similares em alguns aspectos, sendo alguns dos métodos o *genetic algorithm* (GA), o *quantum particle swarm optimization* (QPSO), e a *ant colony optimization* (ACO).

As hiper-heurísticas são algoritmos que resolvem múltiplos problemas, mais especificamente, é um algoritmo que seleciona heurísticas e meta-heurísticas. Algoritmos como o *multi-armed bandit* (MAB) e *multi-swarm particle swarm optimization* (MSPSO), elas também são utilizadas para resolução de problemas com múltiplos objetivos.

Este artigo tem como objetivo demonstrar uma revisão sistemática sobre as meta-heurísticas e hiper-heurísticas para FJSSP.

TRABALHOS CORRELATOS

O GA, por ser um algoritmo de fácil implementação, é uma meta-heurística muito utilizado dentro do FJSSP. Trabalhos como o do Rooyani e Liang (2019) utilizam o GA em suas pesquisas, mas possuindo duas aproximações distintas, onde um modifica o GA para acomodar o problema, e o outro implementa um algoritmo híbrido entre GA e *Simulated Annealing Algorithm* (SAA), respectivamente.

O método de hiper-herística apresentado por Xuewen (2018) são estratégias para otimizar o MSPSO, onde possui como base o método de *particle swarm optimization* (PSO), já Almeida (2020) apresenta o MAB, que utiliza o princípio de *Markov Decision Process* (MDP), onde utiliza-se de recompensas e punições para alcançar um objetivo.

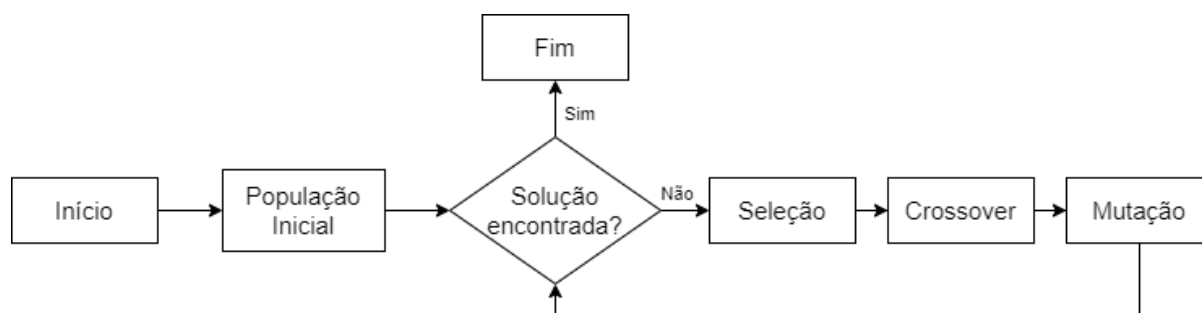
DESENVOLVIMENTO

Durante o desenvolvimento da pesquisa, foram pesquisados tópicos que envolvem o JSSP e FJSSP, que são problemas presentes na computação e na manufatura, onde ambos procuram otimizar o tempo de produção, organizando os *jobs* e máquinas de maneira que o processamento seja o mais rápido e eficiente.

Analisando o trabalho de Coelho (2021), é possível observar que o GA é muito explorado, sendo o terceiro tópico mais presente nos trabalhos (palavras-chaves) de pesquisadores, perdendo apenas por FJSSP e *scheduling*. Nesses últimos anos, assuntos como otimização de multi-objetivo, *Particle Swarm Optimization* (PSO), busca local vêm ganhando significância dentro da área.

Para a resolução do FJSSP são utilizados meta-heurísticas, tais como o GA (Figura 1), onde ele é baseado na teoria de evolução de Charles Darwin, assim como citado no trabalho de Monique (2016), onde o algoritmo foi executada dentro do JSSP com sucesso em muitos problemas de programação da produção, e para a resolução de problemas mais complexos, como a do FJSSP, são realizadas modificações para que ele se “adapte” ao problema.

Figura 1 - Algoritmo Genético.



Fonte: Autoria Própria.

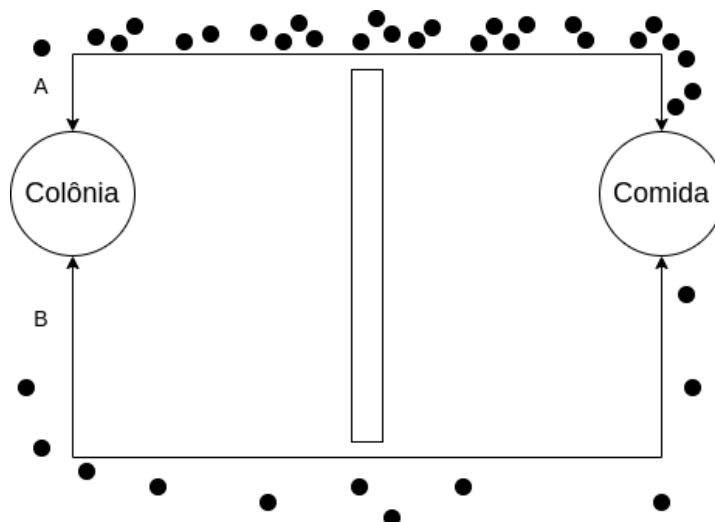
Assim como dito por Viana (2016), foram surgindo métodos derivados do GA para se adaptarem ao FJSSP, tais como apresentam no trabalho de Liang (2019) onde é implementado um algoritmo híbrido entre GA e SAA. Outro exemplo de métodos desenvolvidos é por Rooyani (2019), que apresenta o GA de dois estágios, que como dito no nome possui dois estágios, nos quais utilizam o GA, sendo a primeira para a organização do de *job*-operação, e a segunda para a organização do resultado da primeira fase com as máquinas.

109

Para melhorar a otimização do GA, Luo (2019) propõe uma série de mudanças possíveis para o algoritmo, como a organização do conjunto de máquina e tempo, população inicial, estratégias e novos métodos para a mutação que podem melhorar a performance do algoritmo, e Lin (2021) propõe mudanças na representação de cromossomos, onde são separados em dois grupos, as representações completas e incompletas, sendo o objetivo de ambas encontrar novas soluções.

Ant Colony Optimization (ACO) tem como base o comportamento das formigas, onde para chegar ao objetivo (alimento), elas liberam um hormônio chamado de feromônio para criar um rastro de cheiro da colônia até o objetivo, fazendo com que as outras formigas sigam esse rastro (Figura 2). Esse hormônio é utilizado para determinar qual o caminho mais rápido até o alimento, sendo determinada pela concentração do hormônio, quanto mais vezes uma formiga passa, maior a quantidade de feromônio na trilha (ANDRADE, 2020).

Figura 2 - Ilustração da ideia do ACO.



Fonte: Autoria própria.

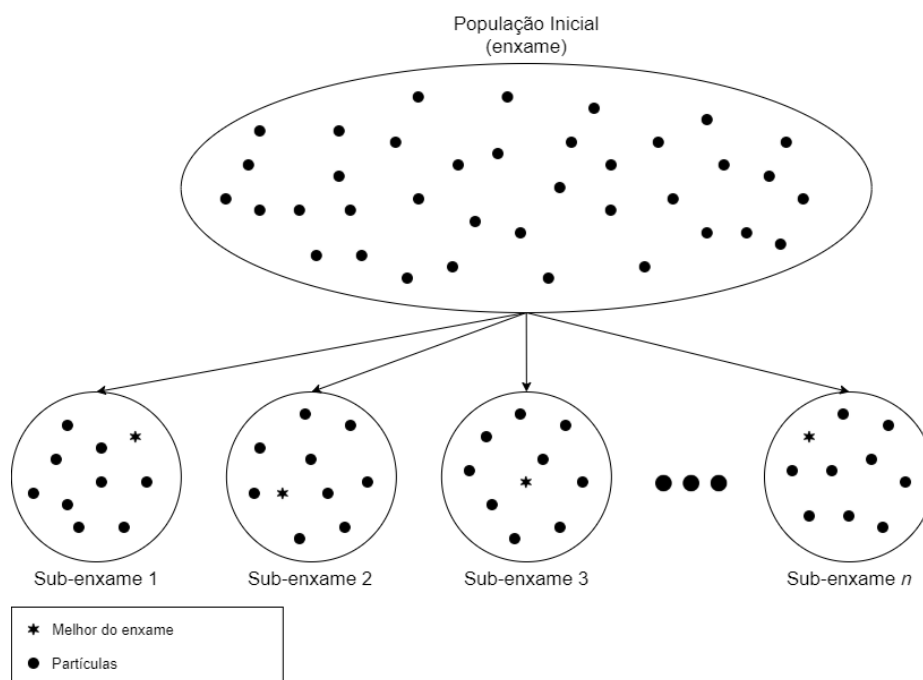
Segundo Chandra (2011), em 2010 houve uma grande demanda no o mercado para a variação para resolver o JSSP, a estratégia apresentada é o *Knowledge-Based Ant Colony Optimization* (KBACO), onde é composto por uma base de dados (conhecimento) e um modelo de busca, que neste caso seria o ACO. Além disso, diz que se o ACO for utilizado com um aproveitamento mais agressivo, mostram melhores resultados, e ajuda a evitar a estagnação do algoritmo.

110

As hiper-heurísticas são utilizadas para solucionar problemas de multi-objetivos, assim como nos FJSSPs, onde o algoritmo pode selecionar meta-heurísticas e heurísticas para ser utilizada. Alguns desses métodos são o MAB e o MSPSO, onde ambos têm a mesma base, divide um problema maior em problemas menores.

Xuewen (2018) são utilizados duas estratégias para otimizar o MSPSO, que são o *purposeful detecting strategy* (PDS) e o *sub-swarm regrouping strategy* (SRS), onde eles ajudam o algoritmo a não ficar preso no melhor local. O MSPSO é composto de sub-enxames que são chamados de *dynamic sub-swarm number strategy* (DNS), sendo elas grupos menores de enxame que estão em um mesmo ambiente, com o tempo a quantidade de sub-enxames irá diminuir, e os métodos PDS e SRS sendo aplicadas constantemente (Figura 3).

Figura 3 - Representação do MSPSO.



Fonte: Autoria própria.

Outro método de hiper-heurística Almeida (2020) é o MAB, termo baseado na ação de apostas em um cassino, onde o jogador deve escolher em qual máquina ele irá jogar, a quantidade de vezes, e em que ordem, em geral esse método utiliza a mesma ideia que o *Reinforcement Learning* (RL), que utiliza o princípio de *Markov Decision Process* (MDP), onde são selecionados as máquinas que possuem maior chances de sucesso, pois são aqueles que têm mais “recompensas” (Figura 4).

Figura 4 - Representação do MAB

Máquina 1	Máquina 2	Máquina 3	Máquina 4	Máquina 5
70% Sucesso	32% Sucesso	83% Sucesso	85% Sucesso	17% Sucesso

Fonte: Autoria própria.

CONCLUSÃO

Através de leitura e análise de artigos e trabalhos foi possível observar que o GA é um dos métodos utilizados pelos pesquisadores dessa área para a resolução do

FJSSP, possuindo também uma grande quantidade de variações, tais como os apresentados neste trabalho.

Por ser um algoritmo simples de se analisar e compreender, principalmente por possuir vários artigos falando a respeito dele, o GA é mais fácil implementá-lo. Recentemente, outros métodos como o ACO, PSO e QPSO vêm ganhando popularidade, a quantidade de estudos sobre eles também estão aumentando, e conseqüentemente variantes destes algoritmos. Assim como as meta-heurísticas, as hiper-heurísticas vem tendo destaque, algumas delas possuem como base as meta-heurísticas, tais como o MSPOS, que por sua vez tem como base o PSO.

Assim como observado no trabalho de Coelho (2021), é possível observar que a área do FJSSP e seus métodos de resolução, tais como meta-heurísticas como o GA, PSO, QPSO e ACO vêm ganhando mais interesse e conseqüente mente mais publicações comentando em relação a esse assunto, e a tendência é só de aumentar.

Como trabalhos futuros, sugiro explorar mais a fundo as meta-heurísticas como o ACO, PSO e QPSO, e hiper-heurísticas como o MAB e MSPSO, e também procurar estudar outros algoritmos para as meta-heurísticas e hiper-heurísticas.

112

REFERÊNCIA

ANDRADE J. V. C. **Desenvolvimento de Uma Hiper-Heurística Aplicada ao Escalonamento em Problemas de Job Shop**. Applied Soft Computing Journal. 2020.

ALMEIDA C. P., GONÇALVES R. A., VENSKE S., LÜDERS R., DELGADO M. **Hyper-heuristics using multi-armed bandit models for multi-objective optimization**. Applied Soft Computing Journal. 2020. DOI: <https://doi.org/10.1016/j.asoc.2020.106520>

CHANDRA M. B., BASKARAN R. **Survey on Recent Research and Implementation of Ant Colony Optimization in Various Engineering Applications**. International Journal of Computational Intelligence Systems, Vol. 4, No. 4. 2011. DOI: <http://dx.doi.org/10.2991/ijcis.2011.4.4.14>

COELHO P., PINTO A., MONIZA S., SILVA C.. **Thirty Years of Flexible Job-Shop Scheduling: A Bibliometric Study**. Procedia Computer Science. 2021. DOI: <https://doi.org/10.1016/j.procs.2021.01.329>

DORIGO M.; CARO D. G.. **Ant Colony Optimization: A New Meta-Heuristic**. 1999. IEEE. 1999. DOI: <https://doi.org/10.1109/CEC.1999.782657>

KATO E. R. R., MORANDIN Jr. O.; FONSECA M. A. S. **A Max-Min Ant System Modeling Approach for Production Scheduling in a FMS.** Department of Computer Science Federal University of São Carlos (UFSCar). 2010. DOI: <https://doi.org/10.1109/ICSMC.2010.5642232>.

LIANG J., WANG Q., XU W., GAO Z., YAN Z.; YU F. **Improved Niche GA for FJSP.** 2019 IEEE 6th International Conference on Cloud Computing and Intelligence Systems (CCIS). 2019. DOI: <https://doi.org/10.1109/CCIS48116.2019.9073748>.

LIN C., LEE I., WU M. **Merits of using chromosome representations and shadow chromosomes in genetic algorithms for solving scheduling problems.** Robotics and Computer Integrated Manufacturing. 2021. DOI: <https://doi.org/10.1016/j.rcim.2019.01.005>.

LIU G., CHEN, W., CHEN H., XIE J.. **A Quantum Particle Swarm Optimization Algorithm with Teamwork Evolutionary Strategy.** Multiscale and Multiphase Computational Particle Technology. 2019. DOI: <https://doi.org/10.1155/2019/1805198>.

LUO X., QIAN Q., FU Y.. **Improved Genetic Algorithm for Solving Flexible Job Shop Scheduling Problem.** Procedia Computer Science, Volume 166, Issue C. 2019. DOI: <https://doi.org/10.1016/j.procs.2020.02.061>.

113

ROOYANI D., DEFERSHA F. M. **An Efficient Two-Stage Generic Algorithm for Flexible Job-Shop Scheduling.** School of Engineering, University of Guelph, Guelph, Ontario, Canada. 2019. DOI: <https://doi.org/10.1016/j.ifacol.2019.11.585>.

SRIBOONCHANDR P., KRIENGGORAKOT N., KRIENGGORAKOT P. **Improved Differential Evolution Algorithm for Flexible Job Shop Scheduling Problems.** Industrial Engineering, Department, Faculty of Engineering, Ubon Ratchathani University, Ubon Ratchathani 34190, Thailand. 2019. DOI: <https://doi.org/10.3390/mca24030080>

VIANA M. S. **Algoritmo Genético Com Operador De Transgenia Para Minimização De Makespan Da Programação Reativa Da Produção.** Centro De Ciências Exatas e De Tecnologia Programa De Pós-Graduação Em Ciência Da Computação. 2016.

XUEWEN H., ISLAM S., ZHOU Y.. **Chromosome Encoding Schemes in Genetic Algorithms for the Flexible Job Shop Scheduling: A State-of-art Review Useful for Artificial Intelligence Applications.** 2020 5th International Conference on Innovative Technologies in Intelligent Systems and Industrial Applications (CITISIA). 2020. DOI: <https://doi.org/10.1109/CITISIA50690.2020.9371789>.

XUEWEN X., GUI L., ZHAN Z. **A multi-swarm particle swarm optimization algorithm based on dynamical topology and purposeful detecting.** Applied Soft

Computing, Volume 67, June 2018, Pages 126-140. 2018. DOI:
<https://doi.org/10.1016/j.asoc.2018.02.042>

ZHANG Q., HU, S. **An Improved Hybrid Quantum Particle Swarm Optimization Algorithm for FJSP**. Proceedings of the 2019 11th International Conference on Machine Learning and Computing. 2019. DOI:
<http://dx.doi.org/10.1145/3318299.3318359>.