

UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA

ÉWERTON CARLOS DE ARAÚJO ASSIS

**Heurísticas e metaheurísticas aplicadas
ao Problema de Escalonamento
Job-Shop**

Goiânia
2011

ÉWERTON CARLOS DE ARAÚJO ASSIS

Heurísticas e metaheurísticas aplicadas ao Problema de Escalonamento Job-Shop

Trabalho de Conclusão apresentado à Coordenação do Curso de Computação do Instituto de Informática da Universidade Federal de Goiás, como requisito parcial para obtenção do título de Bacharel em Computação.

Área de concentração: Otimização e Inteligência Computacional.

Orientador: Prof. Celso Gonçalves Camilo Júnior

Goiânia
2011

Sumário

Lista de Tabelas	3
1 Objetivos do projeto	4
2 Fundamentação teórica	5
2.1 Definição do problema de escalonamento job-shop	5
2.2 Critérios de otimização	6
2.2.1 Critérios baseados na data de entrega	8
2.2.2 Critérios baseados em custos	8
2.3 Pressupostos em problemas de escalonamento job-shop	9
2.4 Extensões à definição tradicional	10
2.5 Classificação das agendas	11
2.6 Métodos e abordagens de otimização aplicados ao problema de escalonamento job-shop	12
2.6.1 Métodos heurísticos e metaheurísticos	12
2.7 IVF-GA: Algoritmo auxiliar paralelo inspirado na fertilização in vitro	13
3 Metodologia	14
4 Cronograma de atividades	16
Referências Bibliográficas	17

Lista de Tabelas

4.1 Cronograma de atividades

16

Objetivos do projeto

O objetivo geral deste projeto de conclusão de curso é desenvolver uma melhor compreensão sobre as abordagens heurísticas e metaheurísticas atualmente utilizadas para solucionar problemas de escalonamento job-shop com o fim de propor uma nova abordagem (solução) que faça uso do IVF-GA [10] (*Algoritmo Auxiliar Paralelo inspirado na Fertilização in Vitro para melhorar o desempenho de Algoritmos Genéticos*).

Para que este objetivo geral seja alcançado, um dos objetivos específicos do projeto é realizar uma revisão bibliográfica das técnicas e ferramentas atualmente utilizadas para resolver problemas de escalonamento job-shop. Existem diversos métodos de otimização encontrados na literatura, que exploram abordagens distintas e, consequentemente, obtêm resultados distintos. A intenção é propor um método de otimização, com base no IVF-GA, que melhor se aplique às instâncias do problema analisado e que seja uma solução eficiente e efetiva. Portanto, outro objetivo específico do projeto é compreender em quais contextos os métodos descritos na literatura são aplicados e quais os resultados obtidos por estas pesquisas.

A partir da revisão bibliográfica e da análise e mapeamento das soluções de qualidade identificadas, será construída a nova solução metaheurística proposta com base no IVF-GA e um artigo será escrito com o fim de apresentar os resultados obtidos a partir desta proposta. Este constitui o objetivo específico final do projeto de conclusão de curso.

Os objetivos específicos são objetivamente listados a seguir:

1. Revisão bibliográfica das técnicas e ferramentas atualmente utilizadas para resolver problemas de escalonamento job-shop;
2. Mapear e analisar quais ferramentas estão sendo utilizadas para determinados contextos e aplicações e as instâncias do problema utilizadas como meios de comparação;
3. Propor uma nova solução metaheurística e realizar uma análise comparativa com as demais soluções identificadas na literatura;
4. Apresentar um artigo com os resultados obtidos a partir da solução proposta em comparação com as abordagens identificadas na literatura.

Fundamentação teórica

Problemas de escalonamento remontam desde a necessidade de organizar meios de produção para melhor satisfazerem a demanda de consumo dos mercados. A partir da Revolução Industrial houve uma crescente necessidade pela sistematização e otimização dos meios de produção, com fins de diminuir o tempo de produção e fazer melhor uso de recursos despendidos na manufatura. Problemas de escalonamento job-shop emergem nesse contexto de manufaturas, no qual tarefas ou produtos (jobs) devem ser processados por máquinas a fim de que todas as tarefas sejam concluídas. Contudo, problemas de escalonamento job-shop não se prendem a estes contextos de manufaturas; diversos problemas e contextos podem ser modelados e melhor compreendidos a partir dessa perspectiva.

2.1 Definição do problema de escalonamento job-shop

A definição formal de um problema de escalonamento job-shop tradicional [5] é feita a seguir: são dadas n tarefas (ou *jobs*) $\{J_1, J_2, \dots, J_n\}$ a serem processadas por m máquinas $\{M_1, M_2, \dots, M_m\}$ — cada tarefa deve ser processada por cada máquina uma única vez. O processamento de uma tarefa em uma máquina é denominado operação: a operação da tarefa i na máquina j é denotada por o_{ij} . Restrições tecnológicas determinam a ordem de processamento de cada tarefa através das m máquinas. As tarefas em um problema de escalonamento job-shop não compartilham uma mesma ordem de processamento através das máquinas, como acontece em problemas de escalonamento flow-shop, no qual todas as tarefas compartilham uma mesma ordem de processamento. O ambiente analisado em um problema de escalonamento job-shop é geralmente denotado por oficina (*workshop*).

Cada tarefa i (J_i) apresenta um tempo de processamento para cada máquina, i.e. para cada operação o_{ij} , operação da tarefa i na máquina j , há um tempo de processamento p_{ij} . Qualquer tempo de ajuste ou de carregamento (*set-up time*) da máquina para processar a operação o_{ij} é incluído em p_{ij} . Além do tempo de processamento para cada operação, cada tarefa i (J_i) pode apresentar um tempo de lançamento, denotado por r_i ,

que determina a partir de quanto tempo ou em qual momento a tarefa J_i estará disponível para processamento pelas m máquinas da oficina.

A partir dessas restrições e definições surge a necessidade de construir um escalonador de tarefas de forma a agendar as n tarefas através das m máquinas de forma que o tempo total de processamento seja o menor possível. Além desse critério de otimização, outros critérios são usualmente estabelecidos, os quais serão apresentados a seguir. Com base nessa definição clássica do problema de escalonamento job-shop, outras definições também foram feitas, como extensão, as quais serão apresentadas em seguida.

O problema de escalonamento job-shop é considerado um problema numericamente intratável, NP-completo (para $m \geq 2$) [5] e que apresenta um limite superior de $(n!)^m$ soluções possíveis; portanto, objetivamente, a enumeração de todas as soluções possíveis para instâncias com dimensão relevante (por exemplo, 10×15 ou 10 tarefas, 15 máquinas) é impraticável [21].

2.2 Critérios de otimização

A partir da definição preliminar do problema de escalonamento job-shop podemos extendê-la de forma a obter outros critérios de otimização. Essa definição básica geralmente é estendida da seguinte forma [5] [21]:

d_i é a data de vencimento ou a data de entrega da tarefa J_i , o tempo ideal para que a tarefa seja completada;

a_i é a janela de tempo no qual a tarefa J_i pode ser processada, denotada por $a_i = d_i - r_i$;

W_{ik} é o tempo de espera da tarefa J_i no processamento da operação k . Não necessariamente a k -ésima operação refere-se à máquina M_k ; refere-se à k -ésima operação da ordem de processamento da tarefa J_i (determinada pelas restrições tecnológicas do problema específico);

W_i é o tempo total de espera da tarefa J_i , ou $W_i = \sum_{k=1}^m W_{ik}$;

C_i é o tempo total de processamento da tarefa J_i , ou $C_i = r_i + \sum_{k=1}^m (W_{ik} + p_{ij(k)})$;

F_i é o tempo total de processamento da tarefa J_i sem levar em conta o tempo de lançamento desta, i.e. é o tempo total em que a tarefa passou pelas máquinas e, assim, permaneceu na oficina ($F_i = C_i - r_i$);

L_i é o atraso no processamento da tarefa J_i ($L_i = C_i - d_i$). Quando a tarefa é completada antes do previsto na data de entrega (antecipação), L_i será negativo. Portanto, a partir de L_i são definidas duas outras variáveis de análise: $T_i = \max\{C_i - d_i, 0\}$, o atraso da tarefa J_i ; e $E_i = \max\{d_i - C_i, 0\}$, a antecipação da tarefa J_i ;

I_j é o tempo de ociosidade da máquina M_j .

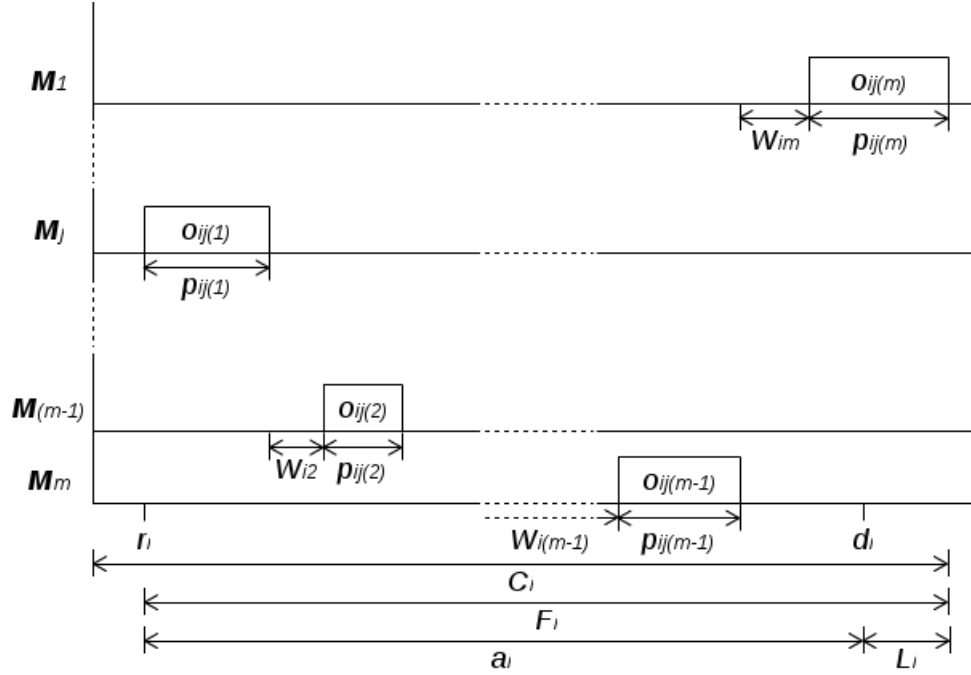


Figura 2.1: Diagrama de Gantt exemplificando o escalonamento de uma tarefa J_i . A ordem de processamento da tarefa é $\langle M_j, M_{m-1}, \dots, M_m, M_1 \rangle$. A variável W_{ik} , $k = 1, \dots, m$, indica o tempo de espera para o processamento da operação k . A tarefa J_i apresenta um tempo de lançamento r_i , indicado no canto inferior direito. No caso, $W_{i1} = r_i$. Não necessariamente $W_{ik} \neq 0$. A tarefa é concluída com atraso, já que $L_i > 0$.

Estas definições têm como propósito formalizar os critérios de otimização usualmente estabelecidos para problemas de escalonamento job-shop. O critério básico, comumente adotado pela literatura em problemas de escalonamento [3], tem como objetivo minimizar o C_{max} (*makespan*), i.e. minimizar o tempo total da tarefa de maior tempo de processamento. Em problemas nos quais as tarefas não têm um tempo de lançamento, estas já estão todas preparadas para serem executadas pelas máquinas, $C_{max} = F_{max}$, sendo F_{max} o tempo total de processamento da tarefa que permaneceu por mais tempo na oficina (passou por todas as máquinas). Em contexto distinto, no qual algumas tarefas apresentem algum tempo de lançamento, podemos ter $C_{max} \neq F_{max}$; portanto, outro critério de otimização é a minimização do valor F_{max} , o tempo gasto pelas tarefas na oficina, referente à tarefa com maior tempo de processamento pelas máquinas. Estes dois critérios de otimização são denotados como critérios baseados no tempo de conclusão das tarefas [5]. Outros critérios são apresentados a seguir.

2.2.1 Critérios baseados na data de entrega

A partir da definição de L_i alguns critérios de otimização baseados na data de entrega podem ser estabelecidos [5], notadamente T_{max} e E_{max} , respectivamente, o atraso máximo e a antecipação máxima. A minimização do valor T_{max} é apropriada em contextos nos quais o atraso no processamento de tarefas apresenta alguma penalidade. Já no caso de minimização do valor E_{max} , esta abordagem é apropriada nos contextos em que há um benefício ao antecipar-se o processamento das tarefas.

Por fim, outra abordagem de minimização relativa à data de entrega é a minimização do número de tarefas atrasadas (n_T). Esta abordagem é apropriada em contextos críticos, nos quais tarefas atrasadas apresentam um alto valor de penalidade.

2.2.2 Critérios baseados em custos

Em alguns ambientes, máquinas ociosas podem representar recursos ociosos ou, ainda, o processamento das tarefas está sujeito à disponibilidade de recursos para cada operação. Nestes contextos, outros critérios de otimização surgem. Esses critérios de otimização levam em conta as seguintes variáveis, para um determinado tempo t [5]:

- $N_{w(t)}$ o número de tarefas esperando para serem processadas por alguma máquina ou tarefas que ainda não estão prontas para serem processadas;
- $N_{p(t)}$ o número de tarefas sendo processadas no tempo t ;
- $N_{c(t)}$ o número de tarefas completadas no tempo t ;
- $N_{u(t)}$ o número de tarefas a serem ainda completadas no tempo t .

Assim, alguns critérios de otimização estabelecidos a partir destas variáveis são:

1. minimizar o número médio de tarefas esperando para serem processadas (\bar{N}_w) ou minimizar o número médio de tarefas não completadas (\bar{N}_u) de forma a minimizar os custos de inventário;
2. minimizar o número médio de tarefas completadas (\bar{N}_c), de forma a reduzir os custos dos bens produzidos;
3. maximizar o número médio de tarefas sendo processadas (\bar{N}_p) com o intuito de realizar um uso eficiente das máquinas. Essa maximização do valor \bar{N}_p relaciona-se à minimização do valor I_{max} , ou a minimização do tempo máximo que uma *máquina* fica ociosa.

2.3 Pressupostos em problemas de escalonamento job-shop

Pelas definições até então feitas é possível identificar a complexidade do problema de escalonamento job-shop e o quanto este pode se tornar complexo em ambientes reais, nos quais tarefas podem surgir aleatoriamente (ambientes dinâmicos) ou mesmo o tempo de processamento das tarefas pode não ser conhecido à priori (ambientes estocásticos) [5]. Assim, alguns pressupostos gerais são assumidos ao identificar e analisar problemas de escalonamento job-shop tradicionais [5]. Como será apresentado adiante, esses pressupostos podem ser ignorados ou modificados em extensões feitas ao problema de escalonamento job-shop tradicional.

cada tarefa é uma entidade as operações de uma determinada tarefa não podem ser processadas simultaneamente, i.e. cada operação constituinte de uma determinada tarefa deve ser processada isoladamente, sem paralelismo, em cada máquina;

não existe preempção enquanto uma operação é processada em uma máquina, esta continuará indisponível até que a operação seja concluída — não é admissível nesse contexto a interrupção do processamento;

cada tarefa é constituída de m operações distintas, uma para cada máquina esse pressuposto é uma consequência da definição anteriormente dada. Assim, temos que uma operação (ou, ainda, neste contexto, uma tarefa) não pode ser processada mais de uma vez em uma determinada máquina; e cada tarefa deve ser processada uma única vez em cada máquina;

tarefas não são canceladas cada tarefa deve ser processada pelas máquinas;

o tempo de processamento é independente da agenda construída o objetivo final do problema de escalonamento job-shop é a construção de uma agenda na qual as tarefas sejam escalonadas, i.e. uma ordem de processamento das tarefas será definida atendendo os critérios de otimização estabelecidos e as restrições envolvidas. Contudo, a agenda definida não influenciará os valores necessários para escalonar as tarefas; assim, o tempo de carregamento da tarefa ou o tempo de preparação para o processamento de uma tarefa não será influenciado pela agenda definida. Esse pressuposto caracteriza essa definição tradicional como independente da sequência ou agenda (*sequence-independent*);

tarefas podem ter que esperar para serem processadas este cenário pode ocorrer já que, como não é admissível a preempção ou o cancelamento de tarefas, e como o tempo de processamento (p_{ij}) é variável (entre tarefas e máquinas distintas), em alguns contextos tarefas podem ter que esperar para serem processadas em alguma máquina ocupada;

existe apenas um único tipo de cada máquina não existe diversas opções da mesma máquina na oficina analisada. Esse pressuposto elimina a possibilidade de evitar esperas entre as tarefas;

máquinas podem estar ociosas já que as diversas tarefas podem não ter concluído seu processamento em alguma das outras máquinas ou estão à espera de outras máquinas para serem processadas;

as máquinas não podem processar qualquer tarefa mais de uma vez

as máquinas da oficina estão sempre disponíveis para processamento das tarefas

as restrições tecnológicas são imutáveis e previamente conhecidas a ordem de processamento das operações de cada tarefa por cada máquina é previamente estabelecida e imutável ao longo do processo de escalonamento;

não existe aleatoriedade o número de tarefas, o número de máquinas, o tempo de processamento de cada operação, o tempo de lançamento de cada tarefa e outros valores usados para modelar um problema específico são previamente conhecidos e fixos.

Vale ressaltar que os pressupostos aqui estabelecidos aplicam-se todos ao problema de escalonamento job-shop tradicional, conforme definido em [5, French1982]. Sugestivamente, alguns destes pressupostos não são admissíveis em contextos e ambientes reais. As definições extendidas, apresentadas a seguir, geralmente modificam, estendem ou eliminam alguns destes pressupostos.

2.4 Extensões à definição tradicional

Com base na definição tradicional dos problemas de escalonamento job-shop, outras definições são feitas extendendo ou reinterpretando as restrições e definições aplicadas ao problema. Algumas dessas extensões são o problema de escalonamento job-shop flexível (FJSSP) e o problema de escalonamento job-shop multi-objetivo (MOJSSP).

Problema de escalonamento job-shop flexível (FJSSP) Essa extensão faz uma reinterpretação do que representa cada máquina e, conseqüentemente, aumenta o espaço de busca do problema. Além dos critérios de otimização analisados, um outro objetivo é acrescentado: além de determinar a ordem de execução das tarefas, o escalonador deve determinar em qual das máquinas determinada tarefa será executada em dado momento. Assim, não existe restrições tecnológicas quanto à ordem que as tarefas devem ser processadas pelas diversas máquinas. Por conseqüência, o espaço de busca é aumentado.

Problema de escalonamento job-shop multi-objetivo (MOJSSP) Essa extensão na verdade é uma categorização de problemas de escalonamento job-shop que estabelecem

mais de um critério de otimização no modelo do problema a ser solucionado. Portanto, existem algumas abordagens que podem ser estabelecidas para guiar a solução, como por exemplo estabelecer um sistema de pesos dentre os diversos critérios a fim de obter uma única função objetiva; ou a abordagem da eficiência à Pareto na resolução de problemas multi-objetivos.

Estas extensões apresentadas não têm como propósito serem definitivas, já que outras definições podem ser obtidas a partir de outras considerações sobre os pressupostos e restrições apresentadas.

2.5 Classificação das agendas

Como foi supracitado, o espaço de busca do problema de escalonamento job-shop é consideravelmente grande até mesmo para pequenas instâncias do problema ($m \geq 2$). Assim, é encontrado na literatura uma categorização das soluções com base em alguns dos critérios de otimização analisados. A seguir são apresentadas as categorizações mais usuais:

agendas ativas são agendas nas quais não é possível escalonar tarefas (ou operações) para serem processadas antecipadamente sem violar algumas das restrições tecnológicas [2]. Outra definição apresenta agendas ativas como agendas nas quais não é possível ter ao menos uma tarefa terminando antecipadamente e nenhuma tarefa atrasada [20];

agendas semiativas são agendas nas quais existe tempo ocioso e que não pode ser eliminado, i.e. algumas máquinas estarão ociosas [21]. Outra definição afirma que agendas semiativas são aquelas nas quais tarefas (ou operações) não podem ser processadas antecipadamente sem modificar a ordem de processamento ou violar as restrições tecnológicas [2] [20];

agendas sem atraso são agendas nas quais não existe tempo ocioso, i.e. máquinas nunca estão ociosas [21];

agendas ótimas as agendas ótimas são parte de um subconjunto das agendas sem atraso e que minimizam o tempo de processamento de todas as tarefas, i.e. minimizam o tempo da tarefa com maior tempo de processamento (C_{max}) [21].

Essa categorização tem como objetivo clarificar quais são os objetivos a serem almejados em uma solução proposta, além de delimitar o espaço de busca que está sendo usado como ambiente de busca e quais as agendas que podem ser efetivamente construídas em determinada abordagem de solução.

2.6 Métodos e abordagens de otimização aplicados ao problema de escalonamento job-shop

Várias abordagens têm sido utilizadas para apresentar soluções ao problema de escalonamento job-shop [3]. Esses métodos de resolução são usualmente caracterizados como métodos exatos, métodos com regras de prioridade, métodos heurísticos e metaheurísticos e métodos da inteligência artificial. Além do uso isolado desses métodos, é comum o uso híbrido destes com o propósito de melhorar uma abordagem específica. As formas híbridas geralmente tem como forma a combinação de um método de busca global acompanhado por um método de busca local [3].

O presente projeto de conclusão de curso não tem como objetivo fazer uma revisão sistemática e extensiva dos métodos de resolução atualmente empregados; contudo, a partir da análise das heurísticas e metaheurísticas empregadas e citadas na literatura os métodos de resolução analisados a seguir são aqueles citados como métodos heurísticos e metaheurísticos, foco primordial do projeto de conclusão de curso proposto.

2.6.1 Métodos heurísticos e metaheurísticos

Existem atualmente várias abordagens heurísticas e metaheurísticas com o intuito de apresentar soluções para o problema de escalonamento job-shop [3]. Algumas heurísticas e metaheurísticas notáveis são: algoritmos genéticos [7] [8] [14] [15] [16] [17] [21] [23] [25] [26] [27], algoritmos meméticos [28], *particle swarm optimization* [6] [12], *simulated annealing* [12], colônia de formiga [13] [21] [24], *variable neighborhood search* [1] [11] e busca tabu [4] [27]. Atualmente, há uma tendência no uso híbrido dessas heurísticas e metaheurísticas com o intuito de explorar as vantagens e minimizar as desvantagens de cada algoritmo [3].

O uso de métodos heurísticos e metaheurísticos, também denominados métodos aproximados [21], em contrapartida aos métodos exatos (ou métodos de otimização clássicos, como *branch-and-bound* [5], *cutting-plane* e programação inteira [5]) tem como motivador a qualidade das soluções obtidas, embora não garantidamente ótimas. Portanto, são algoritmos relativamente efetivos, além de serem soluções eficientes, quando comparadas aos métodos exatos [3]. Os métodos exatos são ineficientes, embora efetivos, na resolução de instâncias de problemas com dimensão considerável (por exemplo, 15×15 ou 15 tarefas, 15 máquinas).

Os métodos aproximados são geralmente classificados como técnicas construtivas ou de busca local [21]. Aqueles têm o propósito de obter uma solução final a partir de um espaço de solução vazio e, iterativamente, melhorar essa solução inicial (ou soluções iniciais, no contexto das abordagens populacionais). A busca local tem como propósito

melhorar uma solução completa inicial na tentativa de obter uma solução ainda melhor. A performance dos métodos aproximados é geralmente analisada a partir dos resultados obtidos experimentalmente [9] [21].

2.7 IVF-GA: Algoritmo auxiliar paralelo inspirado na fertilização in vitro

A partir das considerações feitas anteriormente é perceptível a importância dos métodos aproximados como soluções ao problema de escalonamento job-shop. Uma dessas abordagens comumente utilizadas são os algoritmos genéticos. Contudo, embora alguns estudos asseverem a eficácia dos algoritmos genéticos como método de busca global [10], a ineficiência deste como um método de busca local [15] e a fragilidade deste, por fim, quanto às questões de convergência prematura e, consequentemente, baixa qualidade da solução final [10] são frequentemente salientadas na literatura. Várias abordagens complementares foram desenvolvidas com o intuito de superar essas limitações [10] [15] e assim obter soluções cada vez melhores. Dentre uma delas está o algoritmo auxiliar paralelo (AAP) inspirado na fertilização in vitro (FIV), ou aqui simplesmente denotado como IVF-GA, no contexto de sua aplicação em algoritmos genéticos.

O IVF-GA tem como propósito primordial garantir uma melhor manutenção das características cromossômicas de qualidade dos indivíduos—soluções a cada nova geração a partir dos submódulos do algoritmo responsáveis por recombinar uma seleção de indivíduos de qualidade; estes submódulos têm por objetivo obter no mínimo um novo indivíduo—solução artificial com qualidade igual ou superior ao melhor indivíduo de uma geração evidente (ou corrente).

A partir das considerações e exercícios de comparação feitos em [10, Celso2010] foi possível verificar que o desempenho e velocidade de convergência foi melhorado sem denegrir a qualidade das soluções finais conquistadas pelo algoritmo genético híbrido (IVF-GA). Como o problema da mochila multidimensional [10], o problema de escalonamento job-shop é altamente combinatório e multidimensional. Contudo, aplicações e análise da eficiência do IVF-GA para esta classe de problemas em escalonamento ainda não foram feitas. O intuito, portanto, é obter uma análise comparativa do IVF-GA em relação às demais metaheurísticas utilizadas para solucionar problemas de escalonamento job-shop.

Metodologia

Conforme os objetivos específicos apresentados, o plano de trabalho foi organizado de forma a atender estes objetivos. Inicialmente será feito um levantamento bibliográfico de artigos encontrados em bases científicas, atividade que consumirá dois meses. As bases científicas estabelecidas para este levantamento bibliográfico são: *IEEE Xplore*; *ScienceDirect*; *ACM Digital Library* e demais bases científicas indexadas pelo *Google Scholar* e acessíveis pelo Portal de Periódicos da Capes. O acesso a estas bases científicas está atualmente disponível através deste mesmo portal. O ano base da pesquisa é 2007, conquanto exceções existirão para textos de qualidade referenciados por outros textos atuais e relevantes (em termos de conteúdo) ou textos com relevância histórica nas abordagens e definições usadas ao problema. Os termos de busca usuais são: "*job-shop*", "*job-shop scheduling*", "*job-shop genetic algorithm*", "*job-shop random-key*", "*job-shop particle swarm optimization pso*", "*job-shop ant colony optimization aco*", "*flexible job-shop*", "*multiobjective job-shop*" e "*job-shop hybrid*". Outros termos de busca podem ser determinados a partir da análise de novas necessidades de pesquisa. Não é a intenção desta atividade analisar todos os artigos que estejam disponíveis nestas bases científicas; o critério de análise leva em consideração os artigos que são de maior relevância (maior número de citações). Não está pré-estabelecido um número máximo de artigos, mas considera-se como mínimo para análise vinte e cinco artigos.

Após feito o levantamento bibliográfico, será feito um mapeamento das ferramentas e abordagens de solução para o problema de escalonamento job-shop e os problemas, instâncias e contextos que estas ferramentas e abordagens são aplicadas, atividade que consumirá um mês. Esse mapeamento será possível graças ao levantamento bibliográfico feito à priori. A partir deste levantamento e mapeamento será feita uma proposta de melhoria das abordagens atuais com base no IVF-GA [10] (*Algoritmo Auxiliar Paralelo inspirado na Fertilização in Vitro para melhorar o desempenho de Algoritmos Genéticos*); assim, uma nova solução metaheurística será o objetivo desta atividade. Instâncias usualmente citadas na literatura como meios de comparação de soluções serão utilizadas; notadamente as instâncias de Taillard e instâncias disponíveis na OR-Library (82 instâncias encontradas em diversos textos). Estas atividades, com a intenção de propor uma

nova solução e definir as instâncias de exercício, consumirão dois meses.

Com a nova solução metaheurística construída e as instâncias de comparação determinadas, será feita a execução de testes com essas instâncias de forma a compor um quadro com os resultados obtidos; atividade que consumirá um mês. Cada instância será exercitada no mínimo dez (10) vezes, como é usualmente feito na literatura. A partir da análise dos resultados obtidos, será escrito um artigo descrevendo a nova proposta de solução metaheurística e os resultados apresentados por esta nos exercícios previamente executados. Todas estas atividades serão parte do texto final da monografia.

Os recursos necessários para a execução destas atividades encontram-se disponíveis no momento. Estes recursos são: acesso às bases científicas supracitadas e computadores para exercício das instâncias.

Cronograma de atividades

As atividades a serem desempenhadas neste projeto de conclusão de curso são objetivamente listadas a seguir e organizadas na tabela em seguida:

1. Levantamento bibliográfico
2. Mapeamento das ferramentas e abordagens de solução para o problema de escalonamento job-shop
3. Proposta de melhoria das abordagens atuais com base no IVF-GA [10]
4. Determinar as instâncias de exercício da nova solução metaheurística proposta
5. Execução de testes ou exercícios com as instâncias selecionadas
6. Análise dos resultados obtidos
7. Produção de artigo com a descrição da nova solução metaheurística proposta e os resultados obtidos com esta
8. Escrita e conclusão da monografia apresentando todas as atividades realizadas e os resultados obtidos

Tabela 4.1: *Cronograma de atividades*

Atividade	MAR	ABR	MAIO	JUN	AGO	SET	OUT	NOV
A_1	•	•						
A_2		•						
A_3			•	•				
A_4			•	•				
A_5				•				
A_6					•			
A_7					•	•		
A_8				•	•	•	•	•

Referências Bibliográficas

- [1] ADIBI, M. A.; ZANDIEH, M.; AMIRI, M. **Multi-objective scheduling of dynamic job shop using variable neighborhood search**. *Expert Systems with Applications*, 37:282–287, 2010.
- [2] BRUCKER, P. **Scheduling Algorithms**. Springer, Osnabrück, Deutschland, 2007.
- [3] CHEN, Y.; GUAN, Z.; SHAO, X. **A comparative analysis of job scheduling algorithm**, 2011.
- [4] FATTAHI, P.; SAIDI-MEHRABAD, M. **Flexible job shop scheduling with tabu search algorithms**, 2006.
- [5] FRENCH, S. **Sequencing and Scheduling: An introduction to the Mathematics of the Job-Shop**. Ellis Horwood, West Sussex, England, 1982.
- [6] GAO, L.; ZHANG, G.; SHAO, X.; LI, P. **An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem**. *Computers Industrial Engineering*, 56:1309–1318, 2009.
- [7] GONCALVES, J. F.; DE MAGALHÃES MENDES, J. J.; RESENDE, M. G. C. **A hybrid genetic algorithm for the job shop scheduling problem**. *ATT Labs Research Technical Report*, 2002.
- [8] HUANG, M.; LIU, P.; LIANG, X. **An improved multi-population genetic algorithm for job shop scheduling problem**, 2010.
- [9] JONG, K. A. D. **Evolutionary computation: a unified approach**. MIT Press, Cambridge, 2006.
- [10] JÚNIOR, C. G. C. **Algoritmo Auxiliar Paralelo inspirado na Fertilização in Vitro para melhorar o desempenho de Algoritmos Genéticos**. PhD thesis, Universidade Federal de Uberlândia, Março 2010.
- [11] KHALILI, M.; ROSHANAIEI, V.; NADERI, B.; JOLAI, F. **A variable neighborhood search for job shop scheduling with set-up times to minimize makespan**. *Future Generation Computer Systems*, 25:654–661, 2009.

- [12] LIN, T.-L.; HORNG, S.-J.; KAO, T.-W.; CHEN, Y.-H.; RUN, R.-S.; CHEN, R.-J.; LAI, J.-L.; KUO, I.-H. **An efficient job-shop scheduling algorithm based on particle swarm optimization.** *Expert Systems with Applications*, 37:2629–2636, 2010.
- [13] LIOUANE, N.; SAAD, I.; HAMMADI, S.; BORNE, P. **Ant systems local search optimization for flexible job shop scheduling production.** *International Journal of Computers, Communications Control*, 2:174–184, 2007.
- [14] LIU, M.; JIANG SUN, Z.; WEI YAN, J.; SONG KANG, J. **An adaptive annealing genetic algorithm for the job-shop planning and scheduling problem.** *Expert Systems with Applications*, 2011.
- [15] MA, J.; MA, J.; ZHU, Y.; SHI, G. **Solving the flexible job-shop scheduling problem by immune genetic algorithm.** In: *2010 3rd International Conference on Advanced Computer Theory and Engineering*, 2010.
- [16] MANIKAS, A.; CHANG, Y.-L. **Multi-criteria sequence-dependent job shop scheduling using genetic algorithms.** *Computers Industrial Engineering*, 56:179–185, 2009.
- [17] MIANZHOU, C.; JINGHUA, W. **Research of an improved genetic algorithm for job shop scheduling.** In: *2010 International Conference on Measuring Technology and Mechatronics Automation*, 2010.
- [18] NAZARATHY, Y.; WEISS, G. **A fluid approach to large volume job shop scheduling**, 2010.
- [19] PARDALOS, P. M.; SHYLO, O. V. **An algorithm for the job shop scheduling problem based on global equilibrium search techniques**, 2006.
- [20] PINEDO, M. L. **Scheduling: Theory, Algorithms, and Systems.** Springer, New York, 2008.
- [21] RONDON, R. L. A.; CARVALHO, A. S. **Solving a real job shop scheduling problem**, 2009.
- [22] SHI, L.; PING WANG, H. **Improved genetic algorithm for solving the fuzzy multiobjective job shop problem**, 2010.
- [23] XIAOMEI, Y.; JIANCHAO, Z.; JIYE, L.; JIAHUA, L. **A genetic algorithm for job shop scheduling problem using co-evolution and competition mechanism.** In: *2010 International Conference on Artificial Intelligence and Computational Intelligence*, 2010.

- [24] XING, L.-N.; CHEN, Y.-W.; WANG, P.; ZHAO, Q.-S.; XIONG, J. **A knowledge-based ant colony optimization for flexible job shop.** *Applied Soft Computing*, 10:888–896, 2009.
- [25] YIN, H.; HU, E.; WANG, Y.; XIAO, N.; JIANG, Y. **A three-dimensional encoding genetic algorithm for job shop scheduling.** In: *2007 International Conference on Computational Intelligence and Security Workshops*, 2007.
- [26] ZHANG, G. **Using matrix-coded genetic algorithm for solving the flexible job-shop scheduling**, 2010.
- [27] ZHANG, G.; GAO, L.; SHI, Y. **A genetic algorithm and tabu search for multi objective flexible job shop scheduling problems.** In: *2010 International Conference on Computing, Control and Industrial Engineering*, 2010.
- [28] ZHANG, G.; GAO, L.; ZHANG, L.; LI, X. **An efficient memetic algorithm for solving the job shop scheduling problem.** *Computers Industrial Engineering*, 2011.