



CENTRO DE INVESTIGACIÓN EN MATEMÁTICAS, A.C.

Paisaje de búsqueda en el problema de planificación de producción tipo taller

Tesis que presenta

Juan Germán Caltzontzin Rabell

para obtener el Grado de

**Maestro en Ciencias con Especialidad en
Computación y Matemáticas Industriales**

Director de Tesis

Dr. Carlos Segura González

Guanajuato, Gto. Julio de 2021

Resumen

En este trabajo se analiza el efecto de distintas metodologías de modificación del paisaje de búsqueda del problema de planificación de producción tipo taller (JSSP por sus siglas en inglés)

Abstract

En este trabajo se analiza el efecto de distintas metodologías de modificación del paisaje de búsqueda del problema de planificación de producción tipo taller (JSSP por sus siglas en inglés) (En inglés)

Índice general

1. Introducción	1
1.1. Antecedentes y Motivación	1
1.2. Objetivo	1
1.3. Propuesta	1
1.4. Contribuciones	2
1.5. Panorama General	2
2. Marco teórico	3
2.1. Optimización	3
2.2. Metaheurísticas	4
2.3. Vecindad	4
2.4. Función de aptitud o fitness	5
2.5. Paisaje de búsqueda	5
2.6. Problema del planificación de producción tipo taller (JSSP)	6
2.7. Representación de planificaciones	7
2.8. Metaheurísticas aplicadas al JSP	8
2.9. Vecindades previamente propuestas	8
3. Propuestas	9
4. Validación experimental	10
5. Conclusiones y Trabajos a Futuro	11

Introducción

Sumario

1.1. Antecedentes y Motivación	1
1.2. Objetivo	1
1.3. Propuesta	1
1.4. Contribuciones	2
1.5. Panorama General	2

1.1. Antecedentes y Motivación

Actualmente los resultados del estado del arte para este problema se han obtenido mediante algoritmos meméticos con búsqueda tabú y con ejecuciones de entre 24 y 48 hrs paralelas. La literatura reciente se ha centrado en hacer más eficiente la búsqueda tabú sin considerar el paisaje de búsqueda.

1.2. Objetivo

El objetivo de este trabajo es plantear modificaciones que permitan el uso de mecanismos más simples y rápidos para encontrar soluciones comparables al estado del arte.

1.3. Propuesta

1. Se propone utilizar la búsqueda local iterada (ILS por sus siglas en inglés) por ser un algoritmo simple y rápido.

2. Se propone una extensión a la vecindad N7
3. Se planteó una función de fitness que no solo toma en cuenta el makespan sino también los tiempos de finalización de todas las máquinas.

1.4. Contribuciones

1.5. Panorama General

Marco teórico

Sumario

2.1. Optimización	3
2.2. Metaheurísticas	4
2.3. Vecindad	4
2.4. Función de aptitud o fitness	5
2.5. Paisaje de búsqueda	5
2.6. Problema de planificación de producción tipo taller (JSSP)	6
2.7. Representación de planificaciones	7
2.8. Metaheurísticas aplicadas al JSP	8
2.9. Vecindades previamente propuestas	8

2.1. Optimización

La optimización es una herramienta que nos ayuda a encontrar *la mejor* entre diferentes opciones elegibles. En nuestra vida diaria a menudo nos encontramos en este tipo de situaciones, por ejemplo al elegir entre diferentes rutas para llegar a algún lugar.

Formalmente, un problema de optimización consiste en hallar el mínimo o máximo de una función $f : X \rightarrow \mathbb{R}^n$ (llamada función objetivo) en un conjunto de soluciones X . De modo que el problema consiste en hallar:

$$\min_{x \in X} f(x) \tag{2.1}$$

Es importante mencionar que cualquier problema de maximización puede transformarse en un problema equivalente de minimización con el reemplazo $f(x) \leftarrow -f(x)$, por lo que se considera solo el caso de minimización sin pérdida de generalidad.

2.2. Metaheurísticas

Es muy común que en nuestra cotidianidad nos enfrentemos a problemas tan difíciles o para los que tengamos tan poco tiempo de decisión que no podamos hacer un análisis riguroso, en estos casos es muy común que utilicemos algún método (posiblemente basado en la experiencia) que nos permita hallar una solución aceptable, por ejemplo, es común que reemplacemos el problema por uno más simple que sí podemos responder y cuya respuesta está relacionada con nuestro problema original.¹

En el contexto de la optimización una metaheurística es una metodología de alto nivel que combina diferentes heurísticas y puede aplicarse para resolver de manera aproximada una gran cantidad de problemas. En la práctica existen numerosas metaheurísticas que pueden ser muy diferentes entre sí por lo que no hay un sistema de clasificación universalmente aceptado aunque se han propuesto diferentes criterios de clasificación así como características como:

- De trayectoria vs discontinua. Una metaheurística de trayectoria consiste en, dada una solución inicial, mejorarla de manera iterativa mediante algún operador que «mueve» a la solución a través del espacio de búsqueda.
- basadas en población vs basadas en una sola solución. En las metaheurísticas basadas en población se mantiene un conjunto de soluciones candidatas.
- basadas en búsqueda local vs constructivas. Como se explicará más adelante, en la búsqueda local, el proceso de mejora implica la evaluación de soluciones muy parecidas a una solución inicial dada mientras que en las constructivas se crean nuevas soluciones de acuerdo a una heurística o algoritmo preestablecido.
- Con uso de memoria vs sin uso de memoria. El uso de memoria consiste en almacenar información que nos ayude a explorar el espacio de búsqueda.

2.3. Vecindad

La definición de vecindad es crucial para las metaheurísticas de trayectoria y las basadas en una sola solución. Formalmente, una vecindad es un mapeo $N : X \leftarrow 2^X$ que le asigna a cada solución $x \in X$ un subconjunto de soluciones en X . Intuitivamente podemos pensar que es una forma de definir a las soluciones que «rodean» a otra. Se dice que la solución y es un vecino de x si $y \in N(x)$.

A partir de la definición de vecindad podemos también definir un operador de movimiento cuyo efecto al aplicarlo a una solución sea transformarla en una que pertenezca a su vecindad, i.e. este operador selecciona a un vecino de la solución inicial.

¹No podemos predecir con certeza si lloverá durante el día pero sí podemos responder si el cielo está plagado de nubes oscuras

2.4. Función de aptitud o fitness

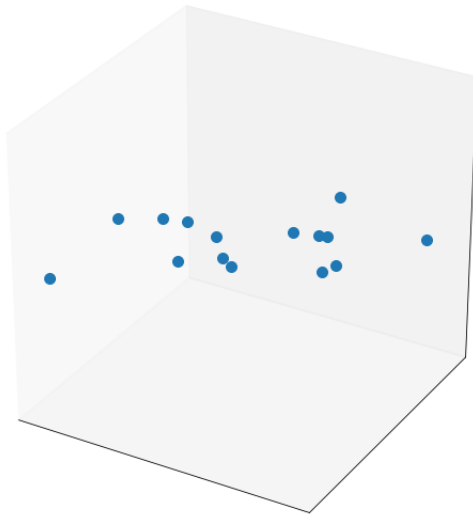
Aunque para un problema de optimización ya se tiene definida una función objetivo que se quiere minimizar, no siempre tendremos el mejor desempeño de las metaheurísticas con solo esta función por lo que resulta benéfico plantear una nueva función a minimizar con la que tengamos mejor desempeño. Por ejemplo puede suceder que aunque dos soluciones tengan asociado el mismo valor de la función objetivo, una de ellas sea un mejor punto de partida para una metaheurística de trayectoria.

Esta función debe asociar a cada solución un elemento de un espacio donde esté definido un ordenamiento total. En esencia esta función define un operador de comparación entre soluciones.

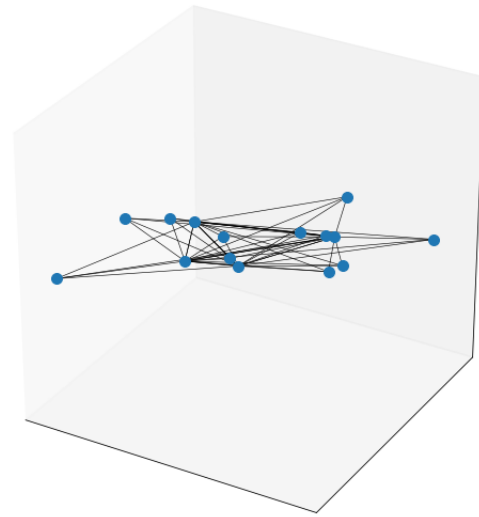
2.5. Paisaje de búsqueda

Una vez que tenemos el espacio de búsqueda y operadores de cambio para generar nuevas soluciones a partir de otras, se define el espacio de búsqueda como un grafo dirigido en el que los nodos son las soluciones al problema y una solución x está conectada a otra y si podemos generar a y aplicando los operadores de cambio a x .

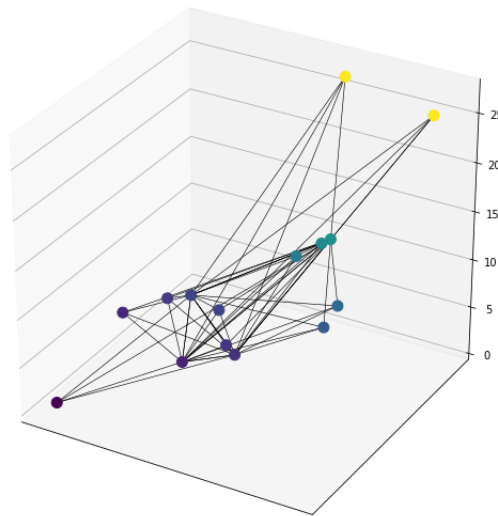
Podemos asociar a cada solución en el espacio un valor de aptitud o fitness que mide la calidad de dicha solución. La adición de esta función de aptitud al espacio de búsqueda genera al paisaje de búsqueda.



(a) Soluciones



(b) Relaciones inducidas por los operadores de cambio



(c) Paisaje

Figura 2.1: caption

2.6. Problema del planificación de producción tipo taller (JSSP)

Una instancia del JSSP consiste en n trabajos diferentes constituidos cada uno por m operaciones que deben procesarse por un tiempo determinado en m máquinas en una secuencia predefinida.

El objetivo es hallar la planificación que minimiza el tiempo que toma terminar todos los trabajos dado que cada máquina puede procesar solo un trabajo a la vez.

Una planificación consiste en asignar tiempos de inicio y fin a cada operación, respetando el orden requerido para cada trabajo. El tiempo que toma terminar todos los trabajos se conoce como makespan y la secuencia de trabajos que toma el mayor tiempo en completarse se conoce como ruta crítica.

En general se considera que el tiempo requerido para procesar cada operación puede expresarse como un entero.

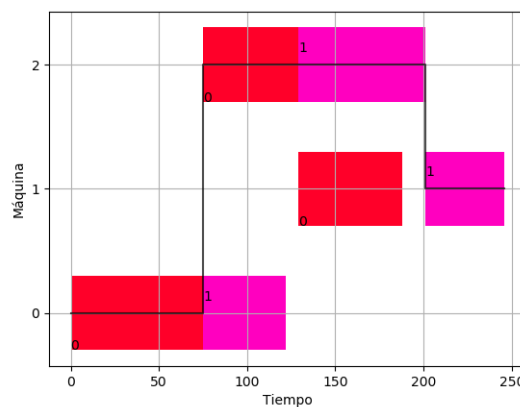
Ejemplo

Se muestra un ejemplo de una instancia con 3 máquinas y 2 trabajos.

Tabla 2.1: Instancia simple con 3 máquinas y 2 trabajos

Trabajo	Secuencia de procesamiento (máquina, tiempo)		
0	0, 75	2, 54	1, 59
1	0, 47	2, 72	1, 45

La siguiente es una posible planificación para la instancia de ejemplo, visualizada mediante un diagrama de gantt. En negro se marca los trabajos que conforman la ruta crítica.



2.7. Representación de planificaciones

Existen varias formas de representar las planificaciones, en este trabajo se utilizaron dos: el grafo disyuntivo y las reglas de prioridad.

Modelo de grafo disyuntivo

En este modelo las planificaciones se representan con un grafo dirigido $G = (V, A, E)$ en el que V es un conjunto de nodos que representa las operaciones, las aristas A representan la secuencia que deben seguir las operaciones dentro de un mismo trabajo y E es otro conjunto de aristas que indica el orden de procesamiento en cada una de las máquinas. Es importante mencionar que con este modelo podemos representar planificaciones no factibles, esto se da cuando el grafo G contiene un ciclo.

Reglas de prioridad

En esta representación una planificación se construye al aplicar un proceso de simulación en el que para cada maquina se construye una cola con las operaciones cuyas dependencias ya han sido procesadas. Inicialmente se tienen en las colas solo las operaciones iniciales de cada trabajo. Una vez que se tiene esto se utiliza una regla de prioridad para elegir qué operación debe planificarse en qué máquina. Se actualizan las colas para las máquinas que lo requieran y se continua con este proceso hasta completar la planificación (vaciar las colas)

Tipos de planificaciones

Dentro de el conjunto de planificaciones factibles se pueden distinguir dos subconjuntos de interés para el presente trabajo: el conjunto de las planificaciones óptimas y el conjunto de las planificaciones activas.

2.8. Metaheurísticas aplicadas al JSP

Dada la complejidad del JSP se han aplicado y desarrollado multiples metaheutrísticas para hallar soluciones razonables. Actualmente las más exitosas son algoritmos meméticos que combinan técnicas poblacionales con metaheurísticas de trayectoria

2.9. Vecindades previamente propuestas

Se han propuesto varias estructuras de vecindad al JSP, a continuación se describen las más importantes a la fecha:

- N1 Fue propuesta en 1992 y consistia en considerar todas las soluciones que se crean al intercambiar cualquier par de operaciones adyacentes que pertenecen a la ruta critica. Esta vecindad es muy grande y considera muchos cambios que no mejoran el makespan.

■

Capítulo 3

Propuestas

Capítulo 4

Validación experimental

Capítulo 5

Conclusiones y Trabajos a Futuro

Conclusiones

