

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

Juan Germán Caltzontzin Rabell
Asesor : Dr. Carlos Segura González

2 de febrero de 2021

1 Introducción

2 Resultados

3 Conclusiones

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)

Planteamiento del problema

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 predeterminada.

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.

Planteamiento del problema

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.

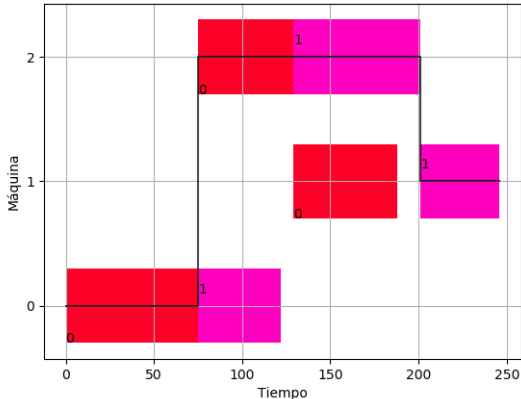
Se muestra un ejemplo de una instancia con 3 máquinas y 2 trabajos.

Cuadro: Instancia simple con 3 maquinas y 2 trabajos

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

Ejemplo

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.

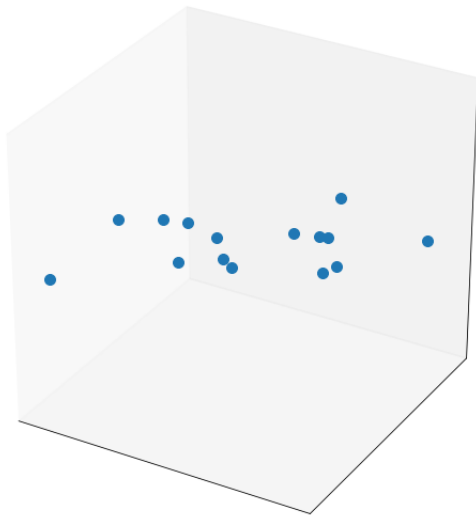


Las soluciones a este problema suelen representarse con un grafo dirigido en el que los nodos son las operaciones de cada trabajo y las aristas representan la secuencia en la que se procesan las operaciones. El problema de hallar una planificación para cada una de las m operaciones de los n trabajos en las m máquinas se reduce a elegir una permutación de los n trabajos en cada máquina por lo que el número de posibles soluciones es $O(n!^m)$.

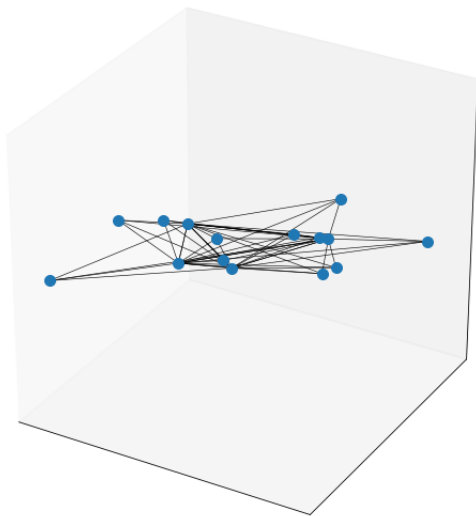
Una vez que tenemos una representación para las soluciones y operadores de cambio para generar nuevas soluciones a partir de otras, se define el espacio de búsqueda como un grafo dirigido en el que los nodos son las soluciones al problema y una solución x está conectada a otra y si podemos generar y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

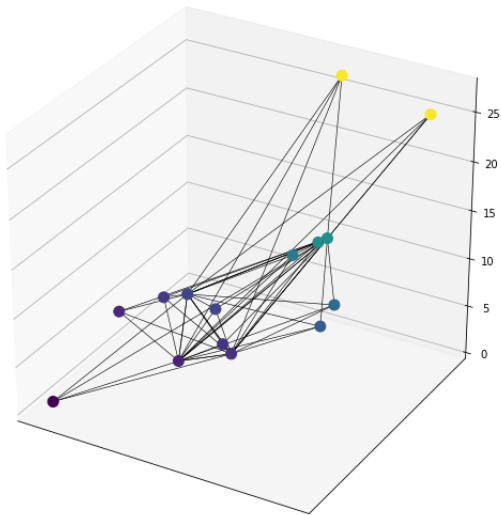
Paisaje de búsqueda



Paisaje de búsqueda



Paisaje de búsqueda



Para una solución dada al JSSP se define una vecindad como el conjunto de soluciones que se encuentran a una misma distancia de la solución inicial. Esta distancia normalmente se define a partir de los operadores de cambio.

Para este problema se han planteado varias estructuras de vecindad a partir de la denominada ruta crítica de la solución. La ruta crítica es la secuencia de trabajos más larga en la planificación. Puede probarse que para reducir el makespan de una solución necesariamente tienen que intercambiarse al menos dos operaciones que pertenezcan a la ruta crítica. La vecindad con la que se han obtenido los resultados del estado del arte se conoce como N7.

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.

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.

Es posible conseguir resultados comparables al estado del arte con algoritmos sencillos y rápidos si modificamos el paisaje de búsqueda.

- 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.

Aquí se presentan los resultados de las dos variantes de ILS, modificar la función objetivo y agrandar la vecindad. También se muestran los mejores resultados reportados a la fecha en la literatura para 80 instancias de prueba comúnmente utilizadas. Las 80 instancias se muestran en dos partes. A cada instancia se le aplicó ILS por 5 min, para 100 soluciones generadas aleatoriamente. Se muestran los mejores resultados obtenidos.

Resultados

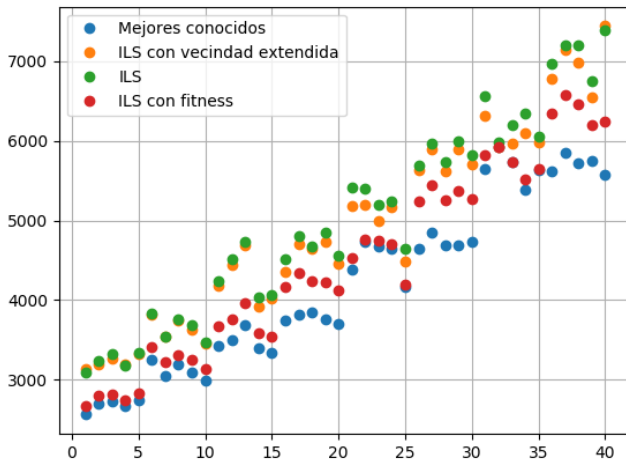


Figura: Resultados de cada método en la primer mitad de instancias

Resultados

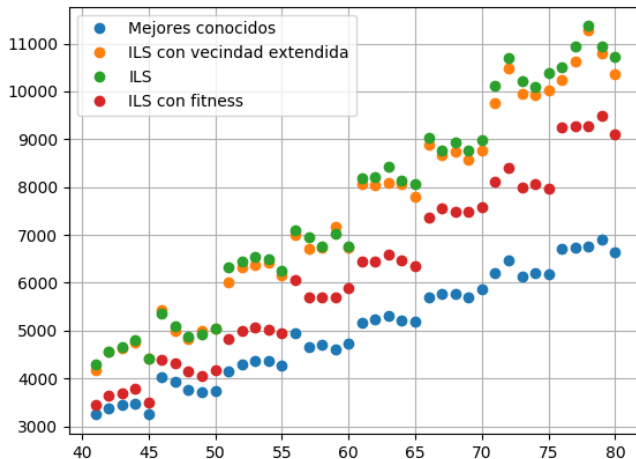


Figura: Resultados de cada método en la segunda mitad de instancias

La modificación de la función objetivo obtuvo mejores resultados que la extensión de vecindad aunque ambos presentan una mejora respecto a la ILS sin modificaciones.

La segunda mitad de las instancias utilizadas tienen una estructura en la que es más fácil estancarse en óptimos locales de peor calidad.

Trabajo a futuro

