

Heurística para la resolución de problema de línea de montaje en la automoción

Serigne Gueye¹, Adnan Yassine¹, Alberto Gómez²

¹ Universidad Le Havre, calle Philippe Lebon, LP 540, 76058 Le Havre. Francia. serigne.gueye@univ-lehavre.fr, adnan.yassine@univ-lehavre.fr

² Escuela Técnica Superior de Ingenieros Industriales. Campus de Viesques, 33204 Gijón (Asturias). mgpast@etsiig.uniovi.es

Resumen

El objetivo de este trabajo es presentar el método de resolución que proponemos para la resolución de un problema de programación de fabricación de los vehículos de una planta para optimizar un cierto número de criterios.

Palabras clave: Heurística, optimización, multiobjetivo.

1. Introducción

Para definir el problema claramente, es recomendable especificar, como eso se hace siempre en la programación matemática, que son las variables del problema, las punterías tenía y los apremios.

1.1. Variables de decisión

Los tipos de vehículos a fabricar en un día así como su número n se saben por adelantado. Las decisiones que se tomarán se relacionan con solamente el orden en el cuál se fabrican los diferentes vehículos.

Matemáticamente, si $v = 1...n$ indica los números de coches y $l = 1...n$ su fila en las cadenas, programar el vehículo v con la posición l se puede representar por el x_{vl} variable = 1 si v están con la posición l y 0 si no.

1.2. Limitaciones

Los vehículos son sometidos a una serie de operaciones que condicionan el orden de fabricación.

La pintura, cada vez que un vehículo debe ser pintado con un color diferente del anterior se necesita una purga (lavado de la pistola de pintura). Por otra parte, después de un número máximo de la pintura del mismo color, en una serie de vehículos, la purgar es también necesario.

Para la instalación de los diferentes equipamientos, el aumento en la carga de trabajo de las máquinas tenía con las marcas de este equipo necesarias de los apremios adicionales. En

efecto, si diferentes vehículos son montados, las máquinas (y los hombres) dedicadas a estas operaciones se solicitan constantemente. "Para alisar" la carga, es decir para distribuir los esfuerzos autorizados en el tiempo, es recomendable que estas máquinas no estén seguidas toda la hora. Un espaciamiento entre los vehículos difíciles es necesario.

Las distancias entre vehículos diferente se traducen en el documento en un cociente N/P. Es decir que en una continuación sin especificar (pero contigua) de P vehículos en la cadena de vehículos, no más de N están autorizados para tener dificultades del equipo a ir para arriba.

1.3. Objetivos

La purgar es un coste económico pues el disolvente necesario para la purgar es costoso. Se pretende reducir al mínimo el número de purgas para así reducir los costes de la fabricación. Uno de los objetivos de programar es así reducir al mínimo el número de purgar. Además, debido al hecho de que el mismo vehículo presenta varias características, respetar un mínimo de espacio entre los vehículos difíciles, puede ser incompatible con la minimización del número de purgas. Luego es necesario llegar a un compromiso entre la necesidad de reducir al mínimo el número de purgas y la necesidad de mantener la regla adecuado entre vehículos difíciles. Éste consiste en aceptar, en caso de necesidad, de las violaciones del espaciamiento. El segundo objetivo del problema es así reducir al mínimo estas violaciones. El problema es así multi-objetivo con dos objetivos principales:

- minimización del número de purgas
- minimización del número de la violación de los apremios del cociente.

Para hacer la optimización posible, es necesario establecer el orden de los diferentes objetivos. En principio, se establece que el orden de un primer objetivo no debe ser cambiado por la optimización del segundo.

2. Resolución

Cada vehículo de programar en las plantas de fabricación tiene varias características: el color, los diferentes equipos que se irán encima,.... El objetivo del método es proponer un modelo genérico simple que tenga en cuenta estas características entonces que solucionan el problema asociado de la optimización. De una manera general, para una autoridad dada, una puede indicar por C el conjunto de las características de los vehículos.

2.1. Algoritmo 1-opt

A continuación se presenta un algoritmo greedy para la solución de este problema. Se representa la programación de los N vehículos por un vector donde cada vehículo es representado por un número.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | . | . | . | . | N |
| 1 | 2 | 3 | . | . | . | . | N |

es V (o) el valor del criterio que se reducirá al mínimo para programar O que consiste un algoritmo 1-opt tiene que explorar salir obtenido las soluciones de O permutando alternadamente los pares de 2 coches.

Un primero

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | . | . | . | . | N |
| 2 | 1 | 3 | . | . | . | . | N |

Un segundo

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | . | . | . | . | N |
| 3 | 2 | 1 | . | . | . | . | N |

El funcionamiento es el siguiente, si la calidad de una solución es mejor que la anterior se acepta, sino se rechaza.

Una vez alcanzado un óptimo local se plantea la resolución con un segundo objetivo, partiendo de la solución alcanzada en el primero. La forma de actuar es idéntica, lo único que no se puede producir, es que una solución nueva estropee la solución alcanzada con el objetivo primero.

3. Resultados

Se realizaron diferentes pruebas con una base de datos suministrada por Renault, cuando los problemas a resolver son sencillos y con menos de 100 vehículos el algoritmo propuesto funciona de forma óptima.

Para problemas de mayores dimensiones el algoritmo comienza a ser lento, por lo cuál se están investigando nuevas formas de tratar este problema. Teniendo en cuenta la codificación empleada, se piensa que se podrán utilizar algoritmos genéticos

4. Conclusiones

En este trabajo se aborda un problema planteado por una Renault. Se plantea una heurística de resolución, la cual se muestra eficaz para problemas catalogados por la empresa de sencillos, para los problemas “complejos” ya no resulta factible su empleo siendo necesario recurrir a algoritmos genéticos.

Referencias

Descripción del problema a resolver:

<http://www.prism.uvsq.fr/%7Evdc/ROADEF/CHALLENGES/2005/#sujet>.

Davis, L. (1991): Handbook of Genetic Algorithms. Van Nostrand Reinhold.

Goldberg. D. (1989): Genetic Algorithms in Search, Optimization & Machine Learning. Addison-Wesley Co., Inc, Reading, MA.

Ejemplo de fichero de datos:

Exemple de fichier « *vehicles.txt* » :

*Date;SeqRank;Ident;Paint Color;HPRC1; HPRC2; HPRC3; HPRC4; LPRC1;
LPRC2; LPRC3*

2003 18 3;997;025031830079;5;1;0;0;0;0;0

2003 18 3;998;025031850339;2;0;0;0;0;1;0;0

2003 18 3;999;025031920802;1;1;0;1;1;1;1

2003 18 3;1000;025031830269;1;0;0;1;0;1;0
2003 18 3;1001;025031850094;2;0;0;0;0;0;0
2003 18 3;1002;025031910010;2;1;1;0;0;1;0
2003 18 3;1003;025031850332;2;0;0;0;0;1;0
2003 18 3;1004;025031820317;1;1;0;1;1;1;1
2003 18 5;1;025031910825;3;0;0;1;0;1;0;1;0
2003 18 5;2;025031820293;7;1;0;1;1;1;0;1;0
2003 18 5;3;025031850497;2;0;0;0;0;0;0;0;0
2003 18 5;4;025031911085;5;1;0;1;0;1;1;1;0
2003 18 5;5;025031850299;5;0;0;1;0;1;0;1;0
2003 18 5;6;025031850082;2;1;0;0;0;0;0;0;0
2003 18 5;7;025031850783;5;0;0;1;0;1;1;1;0
2003 18 5;8;025031920415;5;1;1;1;1;0;0;1;0
2003 18 5;9;025032010270;5;0;0;0;0;0;0;0;0
2003 18 5;10;025031911083;5;1;0;1;0;1;1;1
2003 18 5;11;025031910060;5;0;0;0;0;0;0;0;1
2003 18 5;12;025031970126;5;1;0;1;1;0;0;1
2003 18 5;13;025031910182;2;0;0;0;0;1;0;0
2003 18 5;14;025031850845;1;1;0;1;0;1;1;1
2003 18 5;15;025031910373;1;0;0;1;0;0;0;1
2003 18 5;16;025031850597;1;1;0;0;0;1;0;0
2003 18 5;17;025031830304;1;0;0;1;1;1;0;1
2003 18 5;18;025031850802;1;1;0;1;0;1;1;1
2003 18 5;19;025031850666;2;0;0;0;0;0;0;0
2003 18 5;20;025031850746;2;1;0;0;0;1;0;0
2003 18 5;21;025031850856;2;0;0;0;0;1;0;0
2003 18 5;22;025031830202;9;1;0;1;1;1;0;1
2003 18 5;23;025031850767;2;0;0;0;0;0;0;0