WPOM

Working Papers on Operations Management.
2016, Vol. 7, Nº1 (31-42)
ISSN: 1989-9068

http://dx.doi.org/10.4995/wpom.v7i1.5173

Definición y Evaluación de la dificultad del problema de secuenciación de unidades homogéneas

Definition and Evaluation of the difficulty of the Car Sequencing Problem

Julien Maheut[a], José Pedro Garcia-Sabater[b], Joan Morant[c], Federico Perea[d]

[a]Julien Maheut (EDEM Escuela de Empresarios, juma1@edem.es), [b] José P. Garcia-Sabater (ROGLE, Dpt. De Organización de Empresas, Universitat Politècnica de València, jpgarcia@upv.es), [c] Joan Morant (ROGLE, Dpt. De Organización de Empresas, Universitat Politècnica de València, joamollo@upv.es), and Federico Perea (Instituto Tecnológico de Informática, Grupo SOA, Universitat Politècnica de València, perea@eio.upv.es)

Recibido: 2016-03-22 Aceptado: 2016-04-26

Abstract

The Car Sequencing Problem is a relevant topic both in the literature and in practice. Typically, the objective is to propose exact or heuristic procedures that calculate, in a reduced computational time, a solution that minimizes the number of violated sequencing rules. However, reaching a solution that does not violate any sequencing rule is not always possible because although sequencing rules should be defined to smooth the workload, the evolution of the production mix or some other characteristics can influence the quality of the solutions. In this paper, a first definition of a sequencing rule difficulty is proposed and a statistical study is performed, which allow us to determine the impact of the number of rules, as well as to evaluate how difficult an instance is.

Keywords: *Car Sequencing Problem, Sequencing Rule, Dificulty.*

Resumen

El problema de secuenciación de unidades homogéneas es un caso muy tratado en la literatura donde en la mayor parte de los casos se intenta encontrar procedimientos exactos o heurísticos que permitan calcular en un tiempo computacional reducido una solución de la mejor calidad posible. La calidad de la solución se mide en función de las reglas de secuenciación violadas. Sin embargo, llegar a una solución que no viole ninguna restricción no siempre es posible ya que aunque las reglas de secuenciación se deberían definir para alisar la carga de trabajo, la evolución del mix de producción o las características de las reglas influyen sobre la calidad de las soluciones. En este articulo, se propone una medida para la dificultad de una regla de secuenciación cualquiera y determinar como el número de reglas de secuenciación y sus dificultades pueden servir para predecir en un conjunto de unidades a secuenciar como de difícil es conseguir una buena solución, y detectar los factores que hacen que un conjunto de productos sea más difícil de secuenciar.

Palabras clave: *Secuenciación de coches, Reglas de secuenciación, Dificultad.*

1. Introducción

El problema de secuenciación de unidades homogéneas (CSP) consiste en definir el orden en el cual un conjunto de unidades, como es el caso de coches, tiene que entrar en una línea de ensamblaje de mezcla de modelos. Este problema fue introducido por la primera vez hace más de 30 años en (Parrello, Kabat, & Wos, 1986). Cuando las unidades entran en una planta de montaje final, se pueden diferenciar por sus necesidades de montaje, es decir las opciones que se montan en la parte exterior (también denominada de Trim) o la interior (demoninada de chasis) (Fisher & Ittner, 1999). Algunas de las opciones pueden suponer la necesidad de esfuerzos adicionales en algunos puestos de trabajo, por ejemplo, porque el tiempo de ciclo de la operación es superior al tiempo de *takt* de la línea. Esta circunstancia se puede dar porque no todas las unidades llevan estas operaciones que suponen mucha carga de trabajo. Algunas de estas operaciones suelen ser las de montaje de los techos solares, el montaje del aire condicionado, los motores híbridos, etc. Si la aparición de las opciones en la secuencia no se hace de forma alisada, se podría generar una acumulación de modelos necesitando la misma operación y podría generar problemas en la línea (bloqueo de la línea necesidad puntual de mano de obra, retrabajos, etc.).

Con el fín de evitar estos contratiempos, el CSP plantea reglas de tipo L:M para evitar que más de L unidades de M consecutivas tengan la opción considerada (Briant, Naddef, & Mounié, 2008). Sin embargo, estas reglas que se definen en función de una carga de trabajo suelen obviar la consideración del mix de producción durante una franja temporal determinada. Es decir, dadas unas reglas de secuenciación, la dificultad/facilidad para encontrar una solución que no viole ninguna regla de secuenciación depende altamente del mix de producción determinado si todas las unidades están disponibles para secuenciar.

En este trabajo, se propone una medida para evaluar la dificultad de una regla de secuenciación de tipo "no más de". Esta medida se probará en instancias basadas en el desafio Renault (Solnon, Cung, Nguyen, & Artigues, 2008) para determinar si permite predecir con cierta precisión el número de violaciones que se producen en un set de unidades a secuenciar.

El resto del artículo se presenta de este modo. En primer lugar, en la sección 2, se revisa la literatura sobre la dificultad del problema de secuenciación de coches considerando las diferentes instancias presentes en la literatura del problema de secuenciación de coches y los límites asociados a las instancias dadas. En la sección 3, se proponen medidas para valorar la dificultad de una regla de secuenciación cualquiera. En la sección 4, se propone un plan experimental donde se evaluará el número de reglas violadas por unidad usando un procedimiento aleatorizado con memoria. Un ANOVA y la regresión lineal multiple se utilizarán para determinar primero cuáles son las variables significativas y proponer un modelo ajustado que permita predecir el número de violaciones esperadas por unidad de una instancia de unidades a secuenciar dadas unas reglas de secuenciación del CSP. Por último, se proponen conclusiones y futuras líneas de investigación.

2. Revisión de la literatura

La literatura sobre el CSP se centra en desarrollar métodos de resolución para encontrar soluciones óptimas o soluciones en tiempos cortos (Boysen, Fliedner, & Scholl, 2009), pues este problema es NP-Hard (Gent, 1998; Kis, 2004). Para ello, multipes técnicas de resolución para resolver el CSP están presentes en la literatura: la programación por restricciones (Bergen, Van Beek, & Carchrae, 2001), la programación matemática lineal (Drexl & Kimms, 2001; Gravel, Gagne, & Price, 2005), los métodos ad hoc (Drexl, Kimms, & Matthießen, 2006), la optimización de colonia de hormigas (Gottlieb, Puchta, & Solnon, 2003; Solnon, 2000), Branch & Bound (Fliedner & Boysen, 2008), los algoritmos voraces (Gottlieb et al., 2003) o la búsqueda local (Puchta & Gottlieb, 2002) son unos ejemplos.

En la literatura, estos métodos se prueban en diferentes conjunto de instancias como el desafio propuesto por Renault en el congreso ROADEF'05 (Solnon et al., 2008), Gravel et al. (Gravel et al., 2005), o Gent y Walsh (Gent & Walsh, 1999) entre otros.

El origen de estas reglas de secuenciación tiene relación con la carga de trabajo en las diferentes estaciones de trabajo. Bolat and Yano (Bolat & Yano, 1992) propronen un enfoque analítico para determinar cómo "derivar" reglas de secuenciación. Considerando dos tiempos de proceso (con y sin una opción) para un mismo centro de trabajo, el tiempo de ciclo así como la longitud de la estación, los autores proponen fijar una regla de secuenciación de tipo L:M. Con este enfoque, pretenden demostrar que las secuencias CS (*Car Sequencing*) factibles llevan a secuencias MMS (*Mixed-Model Sequencing*) factibles y viceversa. Golle et al. (Golle, Boysen, & Rothlauf, 2010) proponen un nuevo enfoque para la generación de reglas que clasifica correctamente secuencias como viables o inviables considerando una estación y dos tiempos de procesos diferentes. Este enfoque propone el uso de 2 reglas: "L_1:M_1" y "L_2:M_2" o incluso más. Tal y como se puede apreciar en estos trabajos, siempre se parte de la hipótesis de que las reglas se deben poder cumplir siempre ya que si no se puede cumplir, la solución propuesta podría llevar a considerar muchas violaciones. Por este motivo, técnicas como *Constraint Programming* se presentan en la literatura sobre el CSP para comprobar que un problema se puede resolver con optimalidad, entendido como posibilidad de no violar ninguna regla, como en (Benoist, 2008) por ejemplo.

Sin embargo, a pesar de que la literatura ofrece técnicas para determinar cómo establecer reglas de secuenciación en función de las características de las líneas de montaje, se ha obsersado que el mix de producción influye sobre la calidad de la mejor solución calculada y que no siempre es posible encontrar una solución que no viole reglas de secuenciación. Por lo que sabemos, no existen estudios donde se intenta estudiar o predecir el impacto de la dificultad de unas reglas de secuenciación dadas instancias dinámicas.

3. Propuesta de una medida para la dificultad de un *schedule*

La dificultad para resolver una instancia, llamado más en adelante *Schedule* como conjunto de unidades disponibles para secuenciar, del CSP se podría asimilar a la o las dimensiones que afectan la calidad de la solución encontrada en comparación con la solución óptima. Se propone pues cuatro aspectos fundamentales que afectan a dicha:

(1) La dificultad de las reglas de secuenciación consideradas;
(2) La calidad de la solución a obtener;
(3) La dificultad computacional;
(4) La dificultad del schedule;

A continuación, se pasa a explicar cada uno de los cuatros aspectos.

3.1. La dificultad de las reglas de secuenciación consideradas

La dificultad del CSP puede depender de las reglas consideradas en el problema. Los factores para caracterizar la dificultad de una regla de secuenciación, así como sus posibles valores, son los siguientes:

(1) El número de características/atributos/opciones con reglas de secuenciación (0, 1, más de 1)
(2) El número de reglas de secuenciación para cada unidad (0, 1, más de 1)
(3) El número de reglas para cada atributo (0, 1, más de 1)
(4) La severidad de la regla (valor de M menos el valor de L de la regla)
(5) El multiplicador de la regla (1, más de 1)
(6) La naturaleza de las reglas asociadas a una opción (no más de, no menos de, no más y no menos de)
(7) La dependencia de las reglas y de las unidades a secuenciar (reglas dedicadas a modelos de unidades, reglas compartidas entre modelos)

Se puede intuir que cuanto más grande es (1)-(2)-(3) más "difícil" es el problema; cuando más grande es (4)-(5), más "fácil" es el problema. También, se puede intuir que las reglas de tipo "*No más de y no menos de*" son más difíciles que las "*no menos de*" y las "*no más de*", que las reglas de tipo "*No más de"* son más difíciles que las "*no menos de".*

Debido a estas características, la interdependencia entre ellas, la importancia medida con su peso asociado (si existe) a violar éstas, el impacto resultante (económico, en recursos o en perdida de capacidad) de no cumplir una regla influyen en la dificultad del problema. Por ejemplo, en el desafio Renault, se proponen instancias de 6 colores a 19 colores para la planta de pintura junto con entre 11 y 275 configuraciones de unidades (Solnon et al., 2008). La naturaleza de las reglas es relevante y puede ser múltiple (Maheut & Garcia-Sabater, 2015). Pueden existir, por ejemplo, reglas que se aplican a una planta de pintura como en el problema de Renault, reglas que se aplican a las líneas de Trim en la planta de montaje o reglas que se aplican a las líneas de chasis (Valero Herrero & Molina Morte, 2012). La importancia de las reglas también impacta en la dificultad de las reglas. Las reglas se pueden considerar todas en un nivel de igualdad, o se puede separar las prioritarias de las noprioritarias.

3.2. La calidad de la solución a obtener

La calidad de la solución a obtener depende de lo aceptable que resulte para los "*stakeholders"*. En la literatura científica, se suele considerar la validez de un método en función de lo lejos que sus soluciones estén de una solución "optima", de la mejor solución encontrada por otros métodos en el menor tiempo posible.

Sin embargo, dependiendo de las reglas de secuenciación consideradas, de la instancia considerada y del tiempo de cálculo permitido, es posible que no exista ninguna solución que no respete todas las reglas de secuenciación consideradas. También es posible que respetar todas las reglas suponga una pérdida de capacidad productiva de la planta que se tenga que compensar con horas extras.

Por otro lado, la calidad de la secuencia obtenida no se suele percibir de la misma forma cuando varias unidades violan una regla que cuando unas pocas unidades violan muchas reglas, aunque el número total de reglas violadas fuera el mismo.

En la literatura del CSP, el principal criterio es minimizar el número de reglas violadas pero se podría también, como en el primer caso, minimizar el número máximo de reglas violadas por unidad y en el segundo caso, sería minimizar el número de unidades con opciones violadas.

3.3. La dificultad computacional

La dificultad de la solución depende del tamaño del problema, es decir, cuántas unidades se quieren secuenciar. Al ser un problema NP-Hard, los tiempos de cálculo necesarios para encontrar soluciones pueden crecer de forma exponencial con el número de unidades a secuenciar. Por otro lado, la dificultad computacional puede depender del tiempo de computación máximo disponible.

3.4. La dificultad de la instancia

La dificultad de una instancia del CSP depende de

(1) El tamaño de su espacio de búsqueda (del número de unidades diferentes disponibles para secuenciar),
(2) Las tasas de utilización de las diferentes opciones.
(3) La naturaleza de la instancia: estatica o dinámica.

4. Propuesta de una fórmula para establecer la dificultad de una regla simple

En la literatura del CSP, las reglas de secuenciación suelen ser del tipo "$L_o^+ : M_o$" para la opción *o* con los siguientes parámetros:

Parámetros	*Descripción*
o	Característica, opción o atributo de una unidad
L_o^+	Límite Máximo (Upper bound) para la regla sobre la opción *o*
M_o	Número de unidades consecutivas para la regla sobre la opción *o*
α_o	Multiplicador de la regla de secuenciación asociada a la opción *o*
r_o	Ratio de presencia de la característica *o* en la instancia de unidades a secuenciar

A estas reglas, se les denominará R^+**.** Para evaluar la dificultad de una regla, se plantea el objetivo de encontrar una medida que dé un valor comprendido entre 0 y 1 para cada regla que se pueda no-violar. Cuanto más cerca de céro esté, más fácil será de respetar una regla. Cuanto más alto sea el valor de la medida, más difícil será no-violar la regla. En caso de qué el valor de la medida sea superior a 1, la no existe una secuencia que no viole la regla. Una primera aproximación para estimar la dificultad d_{R^+} para las reglas simples de tipo R^+ se propone a continuación:

$$d_{R^+} = \frac{r_o \cdot M_o}{\alpha_o \cdot L_o^+}$$

El multiplicador de la regla α_o es el factor que se aplica a una regla que no pueda ser simplificada. Una regla 1:3 por ejemplo tiene un $\alpha_o = 1$. Si $\alpha_o = 2$, la regla sería de tipo 2:6.

En la Tabla 1, se propone ver como evoluciona la medida propuesta de una regla de tipo R^+ con $\alpha_o = 1$ cuando la ventana M_o aumenta manteniendo una ratio $\frac{M_o}{L_o^+}$ constante. Según la fórmula propuesta, se observa que cuando aumenta la ventana, la dificultad de la regla va creciendo. También se puede observar que una restricción de tipo no más de 1 de cada 3 cuando la ratio de presencia de una opción es del 50% lleva obligatoriamente a violar esta regla de secuenciación.

Tabla 1. Evolución de los valores de la dificultad de una regla en función del tipo de regla y del mix de aparición de la opción

r_o	L_o^+	M_o	d_{R^+}
0.5	1	2	1
0.5	1	3	1.5
0.5	1	4	2
0.5	1	5	2.5
0.5	1	6	3
0.5	1	7	3.5
0.5	1	8	4
0.5	1	9	4.5
0.5	1	10	5
0.5	1	11	5.5
0.5	1	12	6
0.5	1	13	6.5
0.5	1	14	7
0.5	1	15	7.5

r_o	L_o^+	M_o	d_{R^+}
0.5	2	3	0.75
0.5	2	5	1.25
0.5	2	7	1.75
0.5	2	9	2.25
0.5	2	11	2.75
0.5	2	13	3.25
0.5	2	15	3.75

r_o	L_o^+	M_o	d_{R^+}
0.5	3	4	0.67
0.5	3	5	0,83
0.5	3	7	1,17
0.5	3	8	1,33
0.5	3	10	1,67
0.5	3	11	1,83
0.5	3	13	2,17
0.5	3	14	2,33

Con la fórmula propuesta, se ve que para una ratio de 1:2, la dificultad vale 1, lo que implica que es una restricción muy difícil de cumplir. De hecho, solamente existe una solución que cumpla sin violar este tipo de regla. Lo que se puede observar es que una regla de tipo 3:5 es más difícil que 2:3 que a su vez es más difícil que 3:4 dado una ratio de aparición del 50% del atributo.

5. Descripción de los experimentos

Con el fin de demostrar el impacto de las reglas de secuenciación, se ha utilizado el conjunto de instancias (o s*chedules* en adelante) propuesto en el desafío Renault. Dadas las 8 instancias existentes, se ha mantenido el número de unidades a secuenciar y el número total de unidades a producir de cada clase. Para el establecimiento de las reglas de secuenciación, dada la ratio de aparición de este atributo calculado gracias al mix de producción, se ha fijado el tipo L:M que mejor se acerque al valor de la dificultadad deseada.

Para ello, se ha pre-establecido un conjunto de reglas posibles de la manera siguiente: se ha creado un conjunto de 57 reglas de tipo "1:M" y de tipo "M-1:M" con M entre 2 y 30 como máximo. Para todos los atributos considerados del Schedule Renault, se ordenan los atributos según el valor absoluto de $(\frac{r_o}{d_{R^+}} - 0.5)$ creciente. A cada atributo se le asigna una regla de secuenciación si y solamente si el valor absoluto de $(d_{R^+} - d_{obj})$ es inferior a 0.1, siendo d_{obj} la dificultad buscada. En caso de no conseguir ninguna regla, no se valoran reglas de secuenciación para este atributo. Se para la asignación de reglas de secuenciación a atributos cuando $\frac{r_o}{d_{R^+}} > 0.99$ ó $\frac{r_o}{d_{R^+}} < 0.01$. De esta forma, para cualquier atributo, siempre se intenta asignar una regla que cumpla con la dificultad buscada.

Para medir la dificultad para encontrar una mejor solución, se ha decidido crear secuenciaciones aleatorias y evaluar en cada iteración el número de unidades que violan reglas así como el número de reglas de secuenciación violadas. Una función de memoria guarda en cada caso la mejor solución conseguida. El procedimiento para cuando transucurren 5 minutos sin mejorar la solución sin tiempo máximo total.

Se ha definido para el estudio estadístico la variable dependiente

Y = Número total de reglas de secuenciación violadas por unidad secuenciada medido según el criterio **act** planteado por Boysen (Boysen et al., 2009).

Las variables independientes que se han registrado son $n_{j,i}$ = el número máximo de reglas de secuenciación de dificultad j, con $j \in \{0.2,0.4,0.6,0.8\}$, asociado al schedule i con $i \in \{1,2,3,\ldots,8\}$ cuyos valores máximos vienen expresados en la Tabla 2.

Tabla 2. Número de reglas de secuenciación consideradas con una dificultad fijada para los diferentes schedules disponibles

$n_{j,i}$	*j=0.2*	*j=0.4*	*j=0.6*	*j=0.8*
i=1	8	6	5	5
i=2	10	11	11	12
i=3	11	11	11	10
i=4	13	15	19	19
i=5	3	5	5	4
i=6	11	14	17	17
i=7	5	6	7	7
i=8	5	6	7	7

Un total de $\sum_{i,j} n_{j,i}$ =303 casos se ha considerado. Para cada uno de ellos, se generarán 1,000 réplicas de manera aleatoria y para cada réplica, se ha registrado el mejor valor de Y así como la iteración en que se encontró esta solución. Un ejemplo del programa diseño se presenta en la Ilustración 1 a continuación.

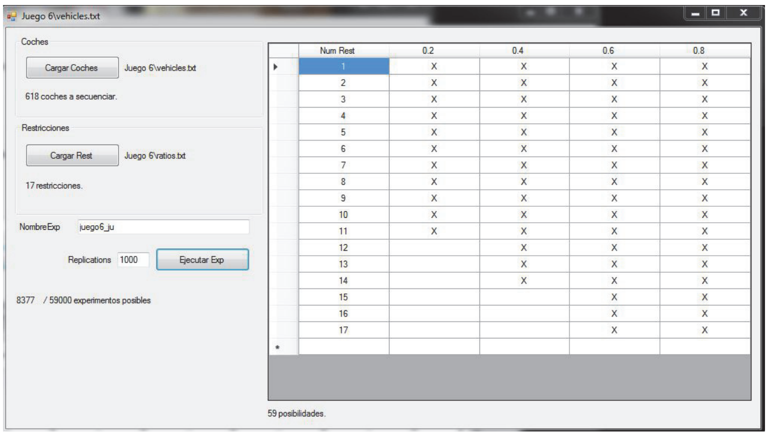

Ilustración 1. Screenshot de la aplicación diseñada para los experimentos

Con este experimento, se pretende demostrar que las reglas de secuenciación de $d_{R^+} = 0.2$ no suponen un incremento de la dificultad de la instancia, es decir, se puede encontrar con relativa facilidad una secuencia creada aleatoriamente que no viole ninguna restricción. También se quiere demostrar que existe una correlación positiva entre el número de restricciones y el número de violaciones por unidad de la mejor solución encontrada.

6. Resultados experimentales

Se ha realizado un análisis de varianza multifactorial para comprobar la significatividad de los grupos que pueden ser considerados como cualitativos. Para ello, se ha definido que la variable dependiente es el número de violaciones por unidad y los factores son "*Schedule*", el número de restricciones (NRest) así como los grupos de d_{R^+} . Se han estudiado un total de 153,626 combinaciones diferentes de estos factores. En la Tabla 3, se pueden observar los valores-P que prueban la significatividad estadística de cada uno de los factores. Puesto que los 6 valores-P son menores que 0,05, estos factores tienen un efecto estadísticamente significativo sobre el número de violaciones por unidad con un 95,0% de nivel de confianza.

Tabla 3. Análisis de Varianza para Y - Suma de Cuadrados Tipo III

Fuente	*Suma de Cuadrados*	*Gl*	*Cuadrado Medio*	*Razón-F*	*Valor-P*
EFECTOS PRINCIPALES					
A:NRest	243,6	4	60,8999	277107,53	0,0000
B: d_{R^+}	1069,16	3	356,387	1621637,22	0,0000
C: *Schedule*	56,9767	7	8,13952	37036,57	0,0000
INTERACCIONES					
AB	143,346	12	11,9455	54354,69	0,0000
AC	19,8862	28	0,710222	3231,66	0,0000
BC	92,0314	21	4,38245	19941,08	0,0000
RESIDUOS	33,7457	153550	0,00021977		
TOTAL (CORREGIDO)	1694,83	153625			

En las ilustraciones siguientes, se presentan los gráficos de interacciones. Se observa que el aumento de d_{R^+} no afecta de manera igual los diferentes schedules. Se observa por ejemplo que los Schedules nº 1, 5 ó 6 tienen un comportamiento muy diferente al 7,8,2 por ejemplo. Esto tiene que ver con el número de reglas de secuenciación y el número de unidades diferentes consideradas en los diferentes schedules.

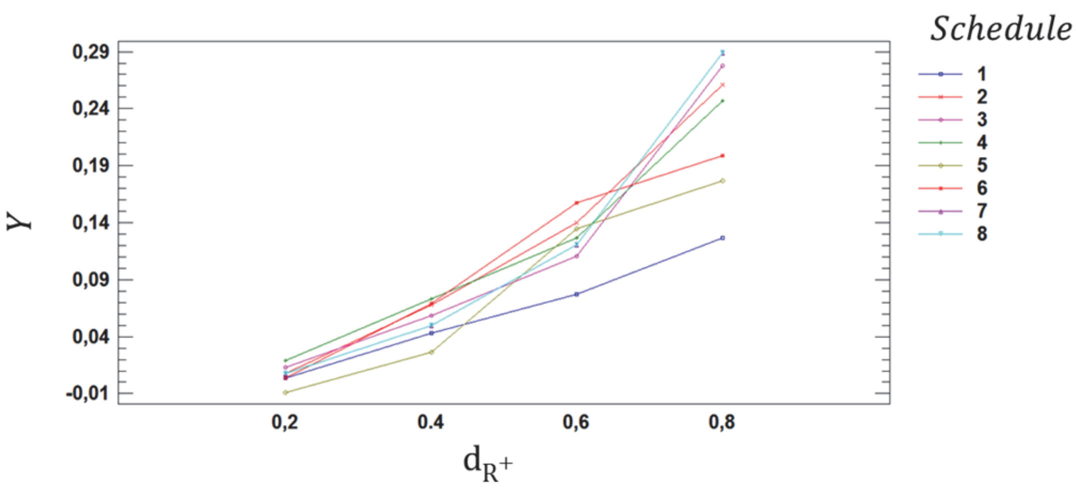

Ilustración 2. Gráfico de interacciones de d_{R^+} y Schedule sobre Y

El gráfico de interacciones muestra que el número de restricciones tiene mucho más impacto sobre Y cuanto más importante es la dificultad de las reglas.

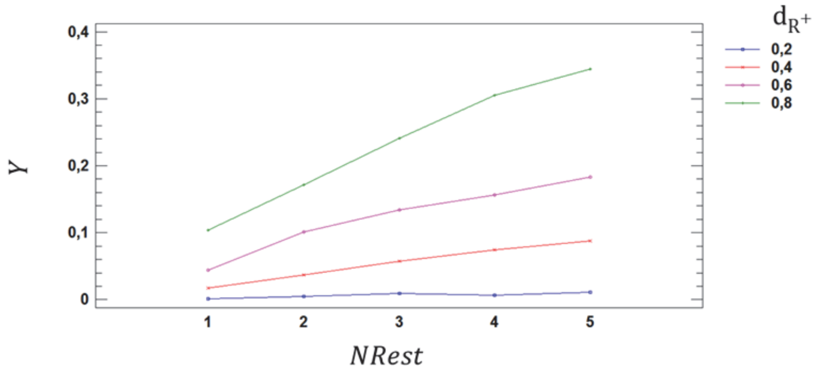

Ilustración 3. Gráfico de interacciones de d_{R^+} y NRest sobre Y

El gráfico de interacciones en la figura siguiente muestra que el número de restricciones no afecta de manera similar a los diferentes juegos. Una de las posibles razones es que hay reglas que son exclusivas de un modelo de unidad en concreto cuando otras reglas se comparten entre modelos. Además de ellos, es posible que no todas las reglas tienen en mismo comportamiento en función del ratio de aparición.

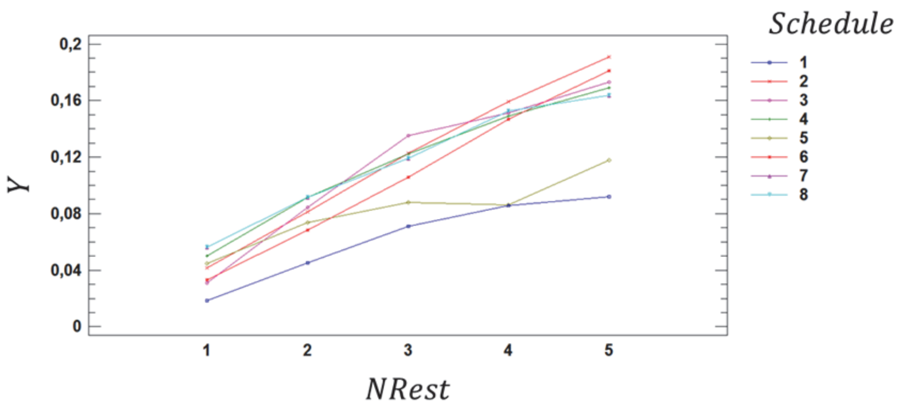

Ilustración 4. Gráfico de interacciones de Schedule y NRest sobre Y

A partir de este análisis, en la figura siguiente se muestra la comparación de la medias para los diferentes schedules.

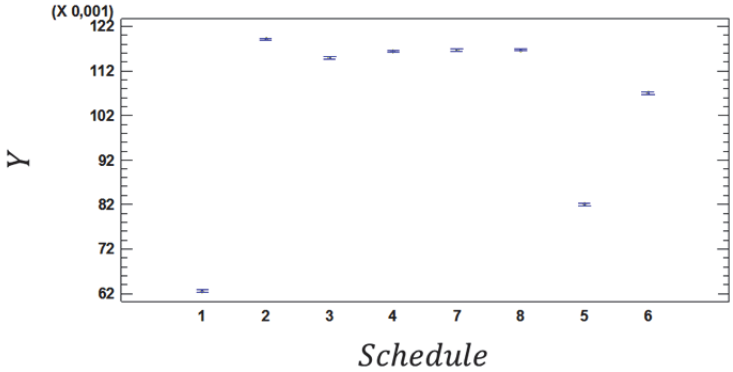

Ilustración 5. Medias y 95,0% de Tukey HSD del factor Schedule

Se observa que los schedules 1, 2, 3, 5 y 6 tienen un promedio estadísticamente diferente salvo los schedules 4, 7 y 8 ya que los intervalos correspondientes estos schedules se solapan. Los schedules 7 y 8 son bastantes similares ya que consideran el mismo número de unidades, las mismas opciones con una diferencias mínimas sobre las reglas de secuenciación. Al ser los schedules significativos, los resultados demuestran que hay componentes aún no explorados que pueden explicar que la dificultad de una instancia no puede ser explicada únicamente por d_{R^+} y que los diferentes factores planteados anteriormente pueden ser relevantes.

Sin embargo, en el caso de las dificultades o de NRest, no existen grupos homogéneos ya que las medias son distintas entre cada grupo de valores.

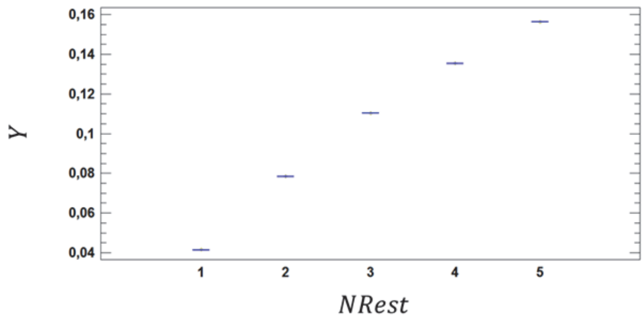

Ilustración 6. Medias y 95,0% de Tukey HSD del factor NRest

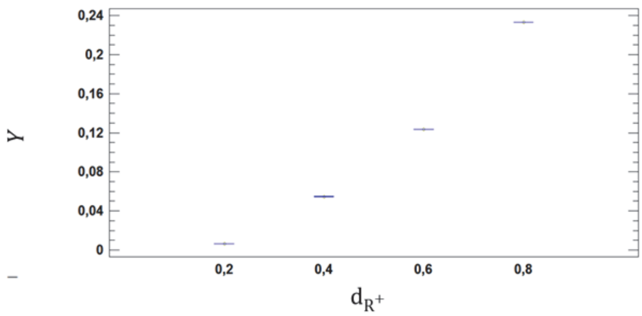

Ilustración 7. Medias y 95,0% de Tukey HSD del factor d_{R^+}

7. Conclusiones

En este artículo, se propone una medida para caracterizar la dificultad de una regla de secuenciación en función del mix de aparición de una opción asociada a esta regla en una instancia dada. El impacto del número de restricciones y de su dificultad sobre el número de violaciones por unidades se ha evaluado generando secuencias aleatorias para las diferentes instancias del desafio Renault. Un ANOVA ha demostrado que tanto el número de restricciones como su dificultad tienen un impacto relevante sobre el número de violaciones por unidad. Estudiar el impacto del factor multiplicador de la regla, evaluar el impacto del número de reglas por vehículos o analizar combinaciones de reglas con dificultades distintas son varias líneas de trabajo que quedan pendientes.

References

Benoist, T. (2008). Soft car sequencing with colors: Lower bounds and optimality proofs. *European Journal of Operational Research*, *191*(3), 957–971. http://doi.org/http://dx.doi.org/10.1016/j.ejor.2007.04.035

Bergen, M. E., Van Beek, P., & Carchrae, T. (2001). Constraint-based vehicle assembly line sequencing. In *Advances in Artificial Intelligence* (pp. 88–99). Springer.

Bolat, A., & Yano, C. A. (1992). Scheduling algorithms to minimize utility work at a single station on a paced assembly line. *Production Planning & Control*, *3*(4), 393–405.

Boysen, N., Fliedner, M., & Scholl, A. (2009). Sequencing mixed-model assembly lines: Survey, classification and model critique. *European Journal of Operational Research*, *192*(2), 349–373. http://doi.org/http://dx.doi.org/10.1016/j.ejor.2007.09.013

Briant, O., Naddef, D., & Mounié, G. (2008). Greedy approach and multi-criteria simulated annealing for the car sequencing problem. *European Journal of Operational Research*, *191*(3), 993–1003. http://doi.org/http://dx.doi.org/10.1016/j.ejor.2007.04.052

Drexl, A., & Kimms, A. (2001). Sequencing JIT mixed-model assembly lines under station-load and part-usage constraints. *Management Science*, *47*(3), 480–491.

Drexl, A., Kimms, A., & Matthießen, L. (2006). Algorithms for the car sequencing and the level scheduling problem. *Journal of Scheduling*, *9*(2), 153–176.

Fisher, M. L., & Ittner, C. D. (1999). The impact of product variety on automobile assembly operations: Empirical evidence and simulation analysis. *Management Science*, *45*(6), 771–786.

Fliedner, M., & Boysen, N. (2008). Solving the car sequencing problem via branch & bound. *European Journal of Operational Research*, *191*(3), 1023–1042.

Gent, I. P. (1998). Two results on car-sequencing problems. *Research Reports of the APES Group, APES-02-1998, Available from Http://www.dcs.stand.ac.uk/~ Apes/apesreports.html*.

Gent, I. P., & Walsh, T. (1999). CSPLib: a benchmark library for constraints. In *Principles and Practice of Constraint Programming–CP'99* (pp. 480–481). Springer.

Golle, U., Boysen, N., & Rothlauf, F. (2010). Analysis and design of sequencing rules for car sequencing. *European Journal of Operational Research*, *206*(3), 579–585.

Gottlieb, J., Puchta, M., & Solnon, C. (2003). A study of greedy, local search, and ant colony optimization approaches for car sequencing problems. In *Applications of evolutionary computing* (pp. 246–257). Springer.

Gravel, M., Gagne, C., & Price, W. L. (2005). Review and comparison of three methods for the solution of the car sequencing problem. *Journal of the Operational Research Society*, *56*(11), 1287–1295.

Kis, T. (2004). On the complexity of the car sequencing problem. *Operations Research Letters*, *32*(4), 331–335.

Maheut, J., & Garcia-Sabater, J. P. (2015). Reglas de secuenciación en el problema de secuenciación en línea de montaje con mezcla de modelos. *WPOM-Working Papers on Operations Management*, *6*(2), 39. http://doi.org/10.4995/wpom.v6i2.3525

Parrello, B. D., Kabat, W. C., & Wos, L. (1986). Job-shop scheduling using automated reasoning: A case study of the car-sequencing problem. *Journal of Automated Reasoning*, *2*(1), 1–42.

Puchta, M., & Gottlieb, J. (2002). Solving car sequencing problems by local optimization. In *Applications of Evolutionary Computing* (pp. 132–142). Springer.

Smith, B. M. (1996). Succeed-first or fail-first: A case study in variable and value ordering.

Solnon, C. (2000). Solving permutation constraint satisfaction problems with artificial ants. In *ECAI* (Vol. 2000, pp. 118–122).

Solnon, C., Cung, V. D., Nguyen, A., & Artigues, C. (2008). The car sequencing problem: Overview of state-of-the-art methods and industrial case-study of the ROADEF'2005 challenge problem. *European Journal of Operational Research*, *191*(3), 912–927. http://doi.org/http://dx.doi.org/10.1016/j.ejor.2007.04.033

Valero Herrero, M., & Molina Morte, P. (2012). CSP Dinámico: Un algoritmo dinámico para la resecuenciación en un almacén de líneas en paralelo. *Working Papers on Operations Management*, *4*(1).