

Proyecto de Curso

Planificación de los ensayos en una telenovela*

Programación por Restricciones

Escuela de Ingeniería de Sistemas y Computación

Facultad de Ingeniería



Profs. Juan Francisco Díaz Frias, Robinson Duque

Febrero del 2020

1. Objetivos

- Enfrentar al estudiante al análisis de un problema abordado desde la óptica del paradigma de Restricciones y a la elaboración de un Modelo de Restricciones producto de dicho análisis.
- Implementar el Modelo de Restricciones propuesto en el sistema de programación MiniZinc.

2. (50 %) Parte 1 - El problema de la planificación de los ensayos en una telenovela

En este trabajo se trata de resolver el problema de los ensayos en una telenovela. Es bien sabido que las escenas de las telenovelas no necesariamente se ensayan en el orden en que se van a emitir (en algunas telenovelas el orden de las escenas se podría variar de cualquier manera sin que la trama sufriera, porque no la hay). No todos los actores participan en todas las escenas. Como a los actores les pagan por el tiempo total en que deben estar en el set, es muy bueno no dejarlos esperando después de una escena en que participan durante varias en que no lo hacen, hasta que llegue otra en que vuelven a participar.

Para simplificar, supondremos que se conoce de antemano el número de escenas que componen la telenovela (lo que en Colombia es falso porque la duración de una telenovela puede ser infinita). El cuadro siguiente muestra un ejemplo de la participación en escenas para la telenovela “Desenfreno de Pasiones”. Para efectos prácticos utilizaremos “Desenfreno de Pasiones” para explicar detalladamente el trabajo.

*Idea original de Camilo Rueda

| Escena | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | Costo |
|---------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|-------|
| Actor 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 10 |
| Actor 2 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 4 |
| Actor 3 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 5 |
| Actor 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 5 |
| Actor 5 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 5 |
| Actor 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 40 |
| Actor 7 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 4 |
| Actor 8 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 20 |
| Durac. | 2 | 1 | 1 | 1 | 1 | 3 | 1 | 1 | 1 | 2 | 1 | 1 | 2 | 1 | 2 | 1 | 1 | 2 | 1 | 1 | |

Cuadro 1: Escenas de “Desenfreno de Pasiones”

Para cada actor, un uno en la columna de la escena i indica que ese actor debe ensayar esa escena. Por ejemplo, el actor 6 (Narciso Perfiles, venezolano) participa en las escenas de la 14 a la 18. La última columna indica lo que cobra cada actor (en cientos de miles) por cada unidad de tiempo que le toque estar en el estudio. La última fila establece la duración de cada escena. Por ejemplo, la escena 6 dura 3 unidades de tiempo.

El problema consiste en encontrar el orden en que se deben ensayar (o filmar) las escenas de modo que se minimice el costo total que se debe pagar a los actores. Hay que tener en cuenta que un actor debe estar presente al comienzo de la primera escena que le toca y solamente puede salir después de la última que le toca. Por ejemplo, si la secuencia de escenas se programara en el orden en que están en la tabla, el actor 3 debería estar en el estudio desde la escena 2 hasta la escena 15, o sea durante $1 + 1 + 1 + 1 + 1 + 3 + 1 + 1 + 1 + 1 + 2 + 1 + 1 + 2 + 1 + 2 = 19$, y cobraría $19 \times 5 = 95$, o sea \$9,500,000 pesos.

2.1. Modelo e Implementación

Cada grupo (**máximo dos integrantes por grupo**) es libre de diseñar el modelo del problema de la manera que considere más adecuada e implantar dicho modelo con el algoritmo que desee, siempre y cuando se ajuste al estilo de programación del paradigma de restricciones y específicamente a la forma de programación en MiniZinc.

Ud. deberá implementar una interfaz de entrada/salida, ojalá gráfica. La entrada se leerá de archivos de texto (*.dzn); la interfaz de salida debe permitir ver claramente que la solución sí satisface el pedido de entrada.

La entrada del problema estará normalizada para permitir que todos los grupos prueben sus algoritmos con los mismos datos. La salida del problema, de la misma manera, estará normalizada para permitir la verificación de los resultados obtenidos.

El proyecto consiste, entonces, en:

- Escribir un CSP que modele el problema: Diga cuáles son sus:
 - Datos de entrada.
 - Variables: Explique qué representa cada una.
 - Restricciones: para cada una explique su semántica intuitiva.
 - Función objetivo: Explique su significado.
- Implementar su modelo utilizando MiniZinc; definir, sustentar e implementar la estrategia de distribución a utilizar; correrlo con diferentes ejemplos, y mostrar la solución (preferiblemente de manera gráfica).

2.2. Entrada y salida

1. Entrada

La entrada del programa para esta primera parte debe consistir de un archivo de texto (*.dzn) así:

- El primer parámetro de entrada (ACTORES) consiste del conjunto de nombres de los actores que *deberá ser tratado como una enumeración dentro del modelo* para conservar el orden de los nombres (i.e., Actor1 = 1, Actor2 = 2,...,Actorn=n).
- El segundo parámetro de entrada (Escenas) utiliza 1's para indicar por cada actor (uno por línea) si este debe participar en la escena i y 0's en otro caso. La última columna de cada una de estas líneas indica el costo por cada unidad de tiempo para el actor.
- La línea siguiente contiene el arreglo “Duración” seguida de las duraciones de cada escena i .
- Por ejemplo, la siguiente sería la entrada para “Desenfreno de Pasiones”:

```

ACTORES = {Actor1 , Actor2 , Actor3 , Actor4 , Actor5 , Actor6 , Actor7 , Actor8} ;

Escenas=[| 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 10 % Actor 1
          | 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 0, 1, 4  % Actor 2
          | 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 5  % Actor 3
          | 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 5  % Actor 4
          | 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 5  % Actor 5
          | 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 40 % Actor 6
          | 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 4  % Actor 7
          | 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 20]; %Actor8

Duracion = [2 , 1 , 1, 1, 1, 3, 1, 1, 1, 2, 1, 1, 2, 1, 2, 1, 1, 2, 1, 1];

```

- Salida: la salida se debe escribir en un archivo texto y consiste en una línea con los números de las escenas en el orden en que se van grabar, más un número al final indicando el costo de la solución.

3. (20 %)Parte 2- El problema de la planificación de los ensayos en una telenovela

3.1. Restricciones Adicionales

Eliminación de simetrías:

1. Este problema tiene simetrías, que pueden eliminarse para que la solución sea más eficiente. Diga qué restricciones añadiría para eliminar simetrías y sustente su escogencia.
2. Incluir estas restricciones en su programa y córrerlo, comparándolo con el original.

Adicionalmente se deben considerar las siguientes restricciones:

1. Disponibilidad de los actores: en general todos los actores deben estar disponibles todo el tiempo, pero hay ciertos actores que por ser estrellas pueden imponer restricciones como por ejemplo un máximo número de unidades de tiempo en el estudio. Por ejemplo Narciso Perfiles podría decir 10, y el actor 3 podría decir 15.
2. Puede haber actores que no la van bien con otros actores y sería ideal que el tiempo que estuvieran en el estudio al mismo tiempo fuera el menor posible (pero tenga en cuenta que la prioridad es reducir costo).

Esta parte consiste, entonces, en:

- Extiender el CSP de la Parte 1 que modele el problema con las restricciones adicionales: Diga cuáles son sus:

- Datos de entrada.
 - Variables: Explique qué representa cada una.
 - Restricciones: para cada una explique su semántica intuitiva.
 - Función objetivo: Explique su significado.
- Implementar su modelo utilizando MiniZinc; definir, sustentar e implementar la estrategia de distribución a utilizar; correrlo con diferentes ejemplos, y mostrar la solución (preferiblemente de manera gráfica).

3.2. Entrada y salida

La entrada del modelo deberá extenderse así:

- “Disponibilidad” especifica el nombre del actor seguido por la restricción de tiempo (0 quiere decir tiempo ilimitado). Esta estructura contiene una entrada por cada uno de los actores.
- “Evitar” especifica los actores que le gustaría evitar encontrarse en las grabación. Esta estructura puede contener un número de restricciones variable.
- Salida: la salida se debe escribir en un archivo texto y consiste en una línea con los números de las escenas en el orden en que se van grabar; más un número indicando el costo de la solución; más un número indicando el tiempo que comparten en las grabaciones los actores incluidos en la restricción “Evitar”.

```

ACTORES = {Actor1, Actor2, Actor3, Actor4, Actor5, Actor6, Actor7, Actor8} ;

Escenas=[| 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 10 % Actor 1
| 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 0, 1, 4 % Actor 2
| 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 5 % Actor 3
| 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 5 % Actor 4
| 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 5 % Actor 5
| 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 40 % Actor 6
| 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 4 % Actor 7
| 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 20|]; %Actor8

Duracion = [2 , 1 , 1, 1, 1, 3, 1, 1, 1, 2, 1, 1, 2, 1, 2, 1, 1, 2, 1, 1];

Disponibilidad = [| Actor1, 0
| Actor2, 0
| Actor3, 23
| Actor4, 0
| Actor5, 80
| Actor6, 10
| Actor7, 5
| Actor8, 100|];

Evitar = [| Actor1, Actor3
| Actor1, Actor4
| Actor2, Actor3
| Actor2, Actor4 |];

```

4. (30 %) Informe con Modelos y Conclusiones

Esta es una de las partes más interesantes del trabajo (pero no por ello la que más vale). En ella se espera que usted:

- Describa sus modelos en términos de: datos de entrada, variables, restricciones y función objetivo
- Analice los resultados obtenidos en la experimentación y **justifique** claramente sus afirmaciones.

5. Sustentación y calificación

El trabajo debe ser sustentado por los autores en día y hora por definir. La calificación del proyecto se hará teniendo en cuenta los siguientes criterios:

1. Informe.
2. Implementación.
3. Pruebas.
4. Desempeño personal en la sustentación, lo cual incluye la capacidad de cada uno de navegar en el código y realizar cambios rápidamente en él.

En todos los casos la sustentación será pilar fundamental de la nota asignada. Cada persona de cada grupo, después de la sustentación, tendrá asignado un número real (el factor de multiplicación) entre 0 y 1, correspondiente al grado de calidad de su sustentación. Su nota definitiva será la nota del proyecto, multiplicada por ese valor. Si su asignación es 1, su nota será la del proyecto. Pero si su asignación es 0.9, su nota será 0.9 por la nota del proyecto. La no asistencia a la sustentación tendrá como resultado una asignación de un factor de 0.

La idea es que lo que no sea debidamente sustentado no vale así funcione muy bien!!!
Éxitos!!!

6. Instrucciones finales

Se debe entregar una carpeta (en formato zip, gz, rar, etc) de nombre PPR-CodEstudiante que contenga los siguientes archivos:

1. Un archivo **readme.txt** que enumere y describa el contenido de la carpeta. Debe describir cómo compilar y poner a correr su programa.
2. Un informe (en formato PDF) que indique los aspectos más importantes de su modelo y de su implementación.
3. Todos los archivos fuentes.
4. Archivos de prueba. Al menos “Desenfreno de pasiones”.

La fecha límite de entrega del proyecto es el **21 de Abril del 2020** a las 23:59.