

**LA GESTIÓN DE PROYECTOS CORPORATIVOS Y LA PROGRAMACION LINEAL. CALCULO DE HOLGURAS Y OPTIMIZACIÓN DEL COSTE**

**THE MANAGEMENT OF CORPORATE PROJECTS AND THE LINEAR PROGRAMMING. CALCULATION OF CLEARANCES AND COST OPTIMIZATION**

Javier Parra Peña<sup>1</sup>  
Yeny Andrea Niño Villamizar<sup>2</sup>  
Darin Jairo Mosquera Palacios<sup>3</sup>

**Resumen**

La gestión de proyectos está inmersa dentro de las funciones organizacionales e implica la aplicación de conocimientos, habilidades, herramientas y técnicas para cumplir con los propósitos que lo originaron. En tal sentido, el presente artículo propone un conjunto de modelos que complementan a los que se han desarrollado desde la programación lineal, permitiendo el cálculo de las holguras las cuales son indispensables para controlar la ejecución del proyecto. A través de la programación lineal se construyeron modelos apropiados para calcular los tiempos tempranos y tardíos de los eventos y las holguras del proyecto, haciendo uso del software especializado GAMS®, el cual emplea el modelado algebraico y permite la ejecución de los modelos de manera secuencial, de tal forma, que es posible utilizar resultados previos de las variables. Se obtiene un conjunto de modelos que, ejecutados secuencialmente, calcula las holguras del proyecto y contribuye al desarrollo de los modelos tradicionales de programación lineal. Se concluye que la programación lineal continúa contribuyendo a la gestión de proyectos y por tanto, mejorar los modelos existentes generará beneficios tanto académicos como empresariales.

**Palabras clave:** Gestión de proyectos, Método de la ruta crítica, Cálculo de holguras, Programación lineal, Compensación tiempo – coste, Modelizado, Optimización.

**Abstract**

Projects management is inside of organizational functions and implies the application of knowledge, abilities, tools, and techniques to achieve the purposes which made projects born. In its sense, this paper proposes a set of models that complement the previous ones

Fecha de recepción: Agosto de 2019 / Fecha de aceptación en forma revisada: Noviembre 2019

---

<sup>1</sup> Ingeniero Industrial, Universidad Distrital Francisco José de Caldas, Colombia. Magíster en Ingeniería Industrial, Universidad de Los Andes, Colombia. Doctor en Ingeniería y Producción Industrial, Universitat Politècnica de València, España. Docente en Universidad Distrital Francisco José de Caldas, Colombia. [jparrap@udistrital.edu.co](mailto:jparrap@udistrital.edu.co), <https://orcid.org/0000-0001-9002-8610>

<sup>2</sup> Ingeniera Industrial, Universidad Distrital Francisco José de Caldas, Colombia. Magíster en Ingeniería Industrial, Universidad Nacional, Colombia. Docente en Universidad Distrital Francisco José de Caldas, Colombia. [yaninov@udistrital.edu.co](mailto:yaninov@udistrital.edu.co), <https://orcid.org/0000-0001-7535-9833>

<sup>3</sup> Ingeniero de Sistemas, Universidad Autónoma de Colombia, Magíster en Teleinformática, Universidad Distrital Francisco José de Caldas. Docente en Universidad Distrital Francisco José de Caldas, Colombia. [djmosquerap@udistrital.edu.co](mailto:djmosquerap@udistrital.edu.co), <https://orcid.org/0000-0002-4526-2683>

based on linear programming; these models let to calculate the project slacks, which are indispensables to control project execution. Through the linear programming, it builds appropriate models to calculate event early times, event late times and project slacks using GAMS. It uses algebraic modeling and let sequential model execution, in that way, it is possible to use previous results of its variables. It obtains a set of models that, executed sequentially, calculates the project slacks and contributes to developing traditional linear programming models. In conclusion, linear programming continues contributing to project management and, in consequence, to improve the current models will produce benefits as academic as organizational.

**Keywords:** Management Projects, Critical Path Method, Slack calculation, Linear programming, Time–cost tradeoff, Modelling, Optimization,

### Introducción

De acuerdo con el *Project Management Institute*, un proyecto constituye un esfuerzo temporal para el logro de un objetivo único (producto, servicio o resultado); este carácter temporal, significa que tiene un inicio y fin definidos (Project Management Institute & Project Management Institute, 2017).

La ejecución de proyectos está inmersa en diferentes funciones de la organización, incluyendo aquellas que conducen al desarrollo o crecimiento organizacional, por lo cual el uso de herramientas que viabilicen procesos adecuados de planeación y optimización, constituye un área de importancia estratégica en la gestión y toma de decisiones en los proyectos (Reyes, s. f.). Aunque existen varias medidas para evaluar el éxito de un proyecto, y esto ha sido objeto de estudio de diferentes investigaciones (Demirkesen & Ozorhon, 2017; Williams, 2015), los indicadores asociados a la eficiencia siguen ocupando un lugar muy importante en esta área (Serrador & Turner, 2015). Por lo tanto, este estudio contribuirá a los interesados en disponer de herramientas que apoyen la mejora de la eficiencia en la gestión de proyectos. En este sentido, la programación lineal (PL) ha sido considerada como una de las herramientas que provee un procedimiento con una solución eficiente (Hillier & Lieberman, 2010; Kalenatic et al., 1993; Taha, 2012; Winston, 2005).

Dentro de los trabajos previos, se destaca en la literatura el trabajo de (Babu & Suresh, 1996) cuyo modelo fue implementado posteriormente en el trabajo desarrollado por (Khang & Myint, 1999) . Estos autores han considerado simultáneamente el tiempo, el costo y la calidad; bajo la premisa que esta última debe maximizarse o por lo menos, no verse afectada por la reducción de los plazos en la ejecución de las actividades. Para especificar la calidad de cada actividad propusieron una escala continua de cero a uno, lo cual ha sido considerado por otros autores como teóricamente significativo, pero no aplicable a problemas reales (Kim et al., 2012).

Si bien es destacable la consideración de una condición de calidad y pudiera ser incluida dentro de las restricciones, el enfoque principal de este trabajo se orienta a proponer mejoras a los modelos tradicionales de PL en la gestión de proyectos, como son incluir los cálculos de los tiempos más tempranos, los tiempos más tardíos y las holguras de la red, además de hacer explícita su implementación en un software especializado, de tal forma, que se constituya en una herramienta para profesionales y académicos interesados en el área. Para cumplir con este propósito, el artículo ha sido estructurado en tres secciones; la primera, presenta algunos referentes teóricos básicos que orientan la siguiente

sección, en la cual se proponen los modelos de PL; posteriormente, se realiza su implementación en el software GAMS® (*General Algebraic Modelling Language*, 2020) y el análisis correspondiente de los resultados.

### **Algunos referentes teóricos**

Las técnicas de planeación de redes, definen la estructura de las relaciones entre las diferentes actividades involucradas para el cumplimiento de los propósitos del proyecto (Kalenatic et al., 1993). Los métodos incluyen el CPM, de las siglas en inglés *Critical Path Method*, si los tiempos de las actividades son conocidos y se utiliza para determinar el tiempo requerido para completar el proyecto y el tiempo en que se pueden retardar las actividades no críticas sin alterar la duración del proyecto cuando los tiempos son determinísticos; y el PERT, del inglés *Programming Evaluation and Review Technique*, si los tiempos de las actividades son desconocidos (estocásticos).

El método de la ruta crítica (CPM) empleado en este trabajo, se utiliza para todo tipo de proyectos. Este método está relacionado fundamentalmente con la compensación entre el tiempo de finalización y los costos del proyecto (Kelley y Walker, 1959 citado en (Kim et al., 2012), y es adecuado para la aplicación a condiciones deterministas en lugar de condiciones probabilísticas. El CPM se puede usar para determinar la compensación tiempo-costo para actividades que cumplen con los tiempos de finalización dados al costo mínimo, y es útil cuando hay experiencias similares de proyectos anteriores (Hillier & Lieberman, 2010).

Los requerimientos de la red incluyen la definición del proyecto; la lista de las actividades y sus relaciones (precedencia), ya que es preciso conocer que actividades deben haberse completado para que una actividad pueda iniciar; la duración (normal y límite) y los costes (normal y límite) de las actividades (Kalenatic et al., 1993).

La construcción de la red para la representación de un proyecto sigue las siguientes condiciones: las actividades se representan mediante arcos orientados y los eventos mediante nodos circulares, toda actividad está delimitada por un nodo en su inicio y otro nodo en su final. El proyecto posee un único inicio y un único final, el nodo uno representa el inicio del proyecto. Todos los arcos que representan actividades sin predecesoras (se pueden realizar desde el inicio del proyecto) parten del nodo 1. El nodo final es aquel al que convergen los arcos que no tienen actividades subsecuentes. Cada actividad se representa mediante un único arco. Dos actividades no pueden poseer nodos inicial y final comunes, de presentarse tal situación es necesario utilizar nodos adicionales y actividades ficticias con una duración igual a cero. Los nodos de la red se numeran de izquierda a derecha y de arriba a abajo de modo que para todas las actividades su nodo final tenga un número mayor que su nodo inicial (Kalenatic et al., 1993).

### **Cálculo de los tiempos**

#### **Tiempos tempranos**

Se calculan de izquierda a derecha en la red. El tiempo del nodo inicial es igual a cero  $E_1 = 0$  dado que representa el inicio del proyecto, y progresivamente se va calculando el tiempo para cada nodo  $j$  de la siguiente forma:

$$E_j = \max_{k \in K} \{E_k + t_{k,j}\}$$

Donde:

$K$  es el conjunto de nodos predecesores inmediatos del nodo  $j$

$E_k$  el tiempo en el nodo  $k$  y,

$t_{k,j}$  el tiempo asociado a la actividad que va de  $k$  a  $j$ .

Tiempos tardíos

Se calculan de derecha a izquierda a partir del nodo final. El tiempo tardío del nodo final es igual al tiempo temprano de dicho nodo. Los tiempos de los demás nodos se calculan progresivamente de acuerdo con:

$$L_i = \min_{k \in K} \{L_k - t_{i,k}\}$$

Donde:

$K$  es el conjunto de nodos sucesores inmediatos del nodo  $i$ ,

$L_k$  es el tiempo en el nodo  $k$  y,

$t_{i,k}$  el tiempo asociado a la actividad que va de  $i$  a  $k$ .

### **Tiempos de reserva u holguras (Kalenatic et al., 1993)**

Los tiempos de reserva u holguras son datos numéricos expresados en unidades de tiempo, en los cuales se puede aplazar (prorrogar) el comienzo o la terminación de actividades. Las holguras total, libre e independiente están asociadas a las actividades.

**Holgura total.** Es la diferencia entre el tiempo máximo permitido que permanece disponible para la ejecución de determinadas actividades y su tiempo de ejecución, es el resultado de restar al tiempo de la terminación tardía el valor correspondiente a la duración de la actividad y el tiempo de iniciación temprana. Es el máximo tiempo en el que puede desplazarse una actividad sin afectar la duración total del proyecto.

$$HT_{i,j} = L_j - t_{i,j} - E_i$$

**Holgura libre.** Es el resultado de restar al tiempo de la terminación temprana el valor correspondiente a la duración de la actividad y el tiempo de iniciación temprana. Representa el máximo tiempo en el que se puede desplazar una actividad sin perturbar el orden lógico y las dependencias en la red.

$$HL_{i,j} = E_j - t_{i,j} - E_i$$

**Holgura independiente.** Es la diferencia entre la holgura total y la holgura libre cuyo resultado puede ser positivo o negativo, siendo importantes sólo los resultados positivos. Muestra en cuanto tiempo se puede prolongar la duración de las actividades. Hasta que no se agote el tiempo de reserva independiente de determinadas actividades, no aparecerá ningún efecto en la relación de tiempo de otras actividades del diagrama de redes.

$$HI_{i,j} = E_j - t_{i,j} - L_i$$

**Holgura del evento.** Además de las holguras anteriores, existe una holgura asociada a los eventos que se calcula como se indica a continuación y que permite identificar los eventos críticos.

$$HE_i = L_i - E_i$$

**Ruta crítica.** Se llama actividad crítica a toda actividad cuya holgura total es igual a cero. La trayectoria que va desde el nodo inicial al nodo final y que está compuesta únicamente por actividades críticas se conoce como ruta crítica.

### 1. Modelos de programación lineal

La determinación de la duración de la ruta crítica puede hacerse mediante Programación lineal de acuerdo con los modelos propuestos por Kalenatic et al. (1993), Taha (2012), Hillier & Lieberman (2010), Winston (2005). En los modelos citados, la función objetivo está definida para minimizar el tiempo del último nodo de la red, lo cual no garantiza que se calculen los tiempos de terminación tempranos ni tardíos de los demás eventos. Teniendo en cuenta lo anterior y que esta información es necesaria para el cálculo de las holguras y para los procesos de toma de decisiones que de ellas se derivan, este trabajo propone modelos en esta vía, y que complementan los desarrollos que desde la programación lineal se han realizado en la gestión de proyectos.

#### Conjuntos

Se definen dos tipos de conjuntos, para los arcos y los nodos.

$V$  Nodos o eventos de la red del proyecto

$A$  Arcos o actividades del proyecto, cada arco está asociado a dos nodos  $i \in V$  y  $j \in V$  que representan los eventos en los cuales inicia y termina la actividad.

#### Parámetros

$t_{i,j}$  Representa la duración de la actividad que va del evento  $i$  al evento  $j$

#### Variables de decisión

$E_j$  Tiempo temprano asociado al nodo  $j$  es decir, en el que inician o terminan las actividades.

#### Modelo 1: cálculo de la duración del proyecto y tiempos tempranos de los eventos

La función objetivo es la sumatoria de los tiempos en los diferentes nodos, la duración del proyecto está dada por el tiempo en el nodo final, y las restricciones obedecen al cálculo de los tiempos teniendo en cuenta la estructura de la red.

$$\text{Minimizar } z = \sum_{j \in V} E_j$$

Sujeto a:

$$E_j \geq E_i + t_{i,j} \quad \forall (i,j) \in A$$

$$E_j \geq 0 \quad \forall j \in V$$

La duración del proyecto será  $TF = E_{Card(Nodos)}$

### Modelo 2: cálculo de los tiempos tardíos y de las holguras

Los conjuntos y los parámetros son los mismos del modelo 1, y la función objetivo se establece para calcular los tiempos tardíos de los eventos.

#### Variables de decisión

$L_j$  Tiempo tardío asociado al nodo  $j$  es decir, en el que inician o terminan las actividades.

La función objetivo es la sumatoria de los tiempos en los diferentes nodos, y las restricciones obedecen al cálculo de los tiempos teniendo en cuenta la estructura de la red.

$$\text{Maximizar } z = \sum_{j \in V} L_j$$

Sujeto a:

$$L_{Card(V)} = E_{Card(V)}$$

$$L_j \leq L_i - t_{i,j} \quad \forall (i,j) \in A$$

$$L_j \geq 0 \quad \forall j \in V$$

Del modelo anterior se observa que los tiempos en los nodos finales de los dos modelos son iguales y por tanto si se ejecuta el modelo 1, el valor correspondiente a la duración del proyecto ( $E_{Card(V)}$ ) debe colocarse como un parámetro del modelo 2; así se hace uso de un resultado de una variable de un modelo anterior como parámetro del modelo actual, lo cual se ejemplifica en la implementación en el software GAMS®.

En este trabajo se propone además, la declaración de las holguras como variables auxiliares, así:

$HT_{i,j}$  Holgura total de la actividad que corresponde al arco  $(i,j) \in A$

$HL_{i,j}$  Holgura libre de la actividad que corresponde al arco  $(i,j) \in A$

$HI_{i,j}$  Holgura independiente de la actividad que corresponde al arco  $(i,j) \in A$

$HE_i$  Holgura del evento  $i \in V$

Tales variables fueron incluidas en el modelo haciendo uso de las ecuaciones correspondientes, con el fin de proporcionar mayor información útil en la gestión del proyecto.

$$HT_{i,j} = L_j - t_{i,j} - E_i \quad \forall (i,j) \in A$$

$$HL_{i,j} = E_j - t_{i,j} - E_i \quad \forall (i,j) \in A$$

$$HI_{i,j} = E_j - t_{i,j} - L_i \quad \forall (i,j) \in A$$

$$HE_i = L_i - E_i \quad \forall i \in V$$

### Modelo 3: Optimización de la duración del proyecto

Conocidas la duración normal y la duración límite del proyecto, se procede a optimizar la duración del proyecto, esto es a determinar las actividades cuya duración se debe reducir, y la magnitud de la reducción de modo que el incremento del coste sea mínimo. El modelo utilizado es el presentado a continuación (Hillier & Lieberman, 2010; Kalenatic et al., 1993; Taha, 2012; Winston, 2005):

#### Conjuntos

Se definen dos tipos de conjuntos, para los arcos y los nodos.

- $V$  Nodos o eventos de la red del proyecto  
 $A$  Arcos o actividades del proyecto, cada arco está asociado a dos nodos  $i \in V$  y  $j \in V$  que representan los eventos en los cuales inicia y termina la actividad.

#### Parámetros

- $t_{i,j}^N$  Representa la duración normal de la actividad que va del evento  $i$  al evento  $j$   
 $t_{i,j}^L$  Representa la duración límite de la actividad que va del evento  $i$  al evento  $j$   
 $c_{i,j}^R$  Representa el coste que implica reducir una unidad de tiempo la actividad que va del evento  $i$  al evento  $j$ , se calculó de acuerdo con la expresión:

$$c_{i,j}^R = \frac{C_{i,j}^L - C_{i,j}^N}{t_{i,j}^N - t_{i,j}^L}$$

- $C_{i,j}^L$  Representa el coste en tiempo límite de una actividad que va del evento  $i$  al evento  $j$   
 $C_{i,j}^N$  Representa el coste en tiempo normal de una actividad que va del evento  $i$  al evento  $j$   
 $TD$  Tiempo deseado de terminación del proyecto (el mínimo tiempo posible es el resultante de realizar todas las actividades en el tiempo límite, el cual puede calcularse al reemplazar las duraciones normales por las duraciones límites en el modelo anterior).

#### Variables de decisión

- $T_j$  Tiempo asociado al nodo  $j$  es decir, en el que inician o terminan las actividades.  
 $R_{i,j}$  Tiempo a reducir a la actividad que va entre el nodo  $i$  y el nodo  $j$ .

$$\text{Minimizar } z = \sum_{(i,j) \in A} c_{i,j}^R R_{i,j}$$

Sujeto a:

$$T_j \geq T_i + t_{i,j}^d - R_{i,j} \quad \forall (i,j) \in A$$

$$R_{i,j} \leq t_{i,j}^N - t_{i,j}^L \quad \forall (i,j) \in A$$

$$T_{card(V)} = TD$$

$$T_i \geq 0 \quad \forall i \in V$$

La función objetivo es el incremento del coste total de ejecución del proyecto, que es la sumatoria de los costes de reducción de las actividades; las restricciones obedecen al cálculo de los tiempos teniendo en cuenta la estructura de la red incluyendo las reducciones consideradas. Las reducciones deben estar dentro del rango de duración posible y la duración final del proyecto es igual a la duración deseada.

## 2. Implementación de los modelos en el *software* GAMS®

De acuerdo con los propósitos de este artículo, se desarrolla a continuación la implementación de los modelos de programación lineal empleando el *software* Gams®, para lo cual se toma como referencia el siguiente ejemplo.

*Una firma consultora se encuentra planeando el desarrollo del proyecto denominado Alfa, para ello, ha recabado la información que se presenta en la tabla 1, el diagrama de PERT/CPM del proyecto se presenta en la figura 1.*

**Tabla 1.**  
*Información del proyecto Alfa*

Actividad	Arco	Predecesoras	Tiempo Normal	Tiempo Límite	Coste Normal	Coste Límite
A	(1, 2)	-	12	8	12	20
B	(1, 3)	-	7	4	11	14
C	(1, 4)	-	10	7	15	21
D	(1, 6)	-	8	6	10	15
E	(3, 7)	B	6	3	8	13
F	(4, 8)	A, C	7	5	16	20
G	(5, 8)	A, B	11	9	16	21
H	(6, 10)	D	10	8	18	23
J	(7, 9)	D, E	14	12	20	25
I	(7, 10)	D, E	6	3	12	16
K	(8, 9)	F, G	8	7	9	12
L	(9, 10)	K, J	4	3	7	9



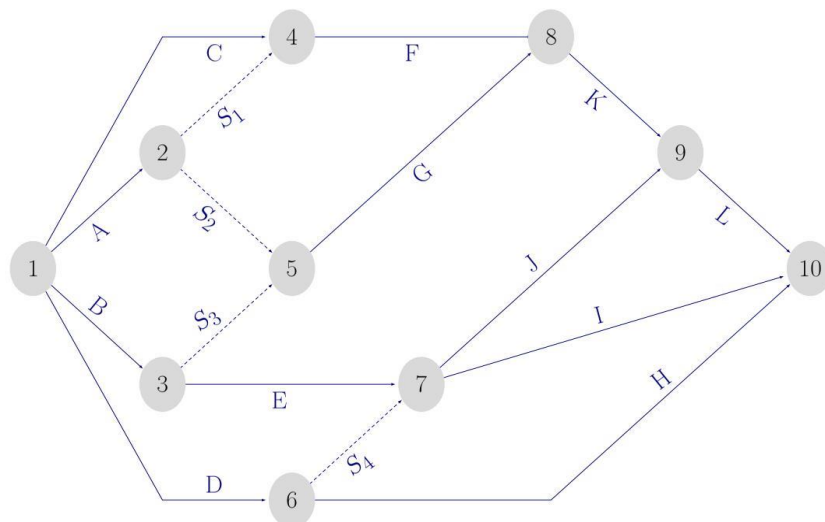


Figura 1. Red del proyecto Alfa

A manera de resumen se presentan en la tabla 2 los principales comandos del lenguaje GAMS®.

**Tabla 2.**  
Elementos básicos del lenguaje de Modelado en GAMS®

Elemento	Lenguaje GAMS
Comentario	\$Ontext \$Offtext
Conjunto	<b>set</b> nombreConjunto <b>Comentario</b> ;
Alias	<b>alias</b> (conjunto, alias1, alias2);
Parámetro	<b>parameter</b> NombreParametro(Conjunto) <b>Comentario</b> ;
Variable	<b>variable</b> NombreVariable(Conjunto) <b>Comentario</b> ; <b>positive variables</b> NombreVariable; <b>integer variables</b> NombreVariable; <b>binary variables</b> NombreVariable;
Ecuaciones	<b>equation</b> NombreEcuación(Conjunto) <b>Comentario</b> ; NombreEcuación(Conjunto).. Ecuación;
Incluir datos	\$include Ruta\NombreArchivo
Modelo	<b>model</b> NombreModelo/ecuación1, ecuación2,... /;
Resolver	<b>solve</b> NombreModelo <b>minimizing</b> NombreVariableResultados using LP;
Visualizar	<b>display</b> NombreVariable.l;

Fuente: Los autores con base en Gams User extraído de <http://www.gams.com>

Se presentan los diferentes modelos, los cuales se han construido secuencialmente en el archivo PERTCPM.gms, en el primer modelo se declaran sus elementos (conjuntos, parámetros y variables) y sus relaciones o ecuaciones, como se observa en la figura 2. La función objetivo propuesta no representa una medida de desempeño propia del proyecto, pero permite determinar los tiempos tempranos en que se completan sus actividades. En este caso la variable auxiliar TiempoTotalN representa el tiempo en el evento final y en consecuencia la duración del proyecto.

```

$ontext
*****
Cálculo de la duración del proyecto y tiempos tempranos de los eventos
*****
$offtext

set Nodos  Nodos de la red del proyecto;
alias (Nodos, i, j);
set Arcos(i, j)  Aristas o arcos de la red;

parameter
DuracN(i, j)  Duración normal de una actividad en un arco;

variables
Tiempo(Nodos)  Tiempos tempranos de los eventos
TiempoTotalN  Tiempo normal de duración del proyecto
SumaTiempos  Suma de los tiempos;
positive variables Tiempo;

equations
TiemposTempr  Suma de los tiempos tempranos
TiempoNodo(i, j)  Restricción de la estructura de la red
TiempoT(Nodos)  Tiempo total del proyecto;

TiemposTempr.. SumaTiempos =e= sum(i, Tiempo(i));
TiempoNodo(Arcos(i, j)).. Tiempo(j) =g= Tiempo(i) + DuracN(i, j);
TiempoT(Nodos)$ (ord(Nodos)=card(Nodos)).. TiempoTotalN =e= Tiempo(Nodos);

$include DatosPert

model TiemposRed /TiemposTempr, TiempoNodo, TiempoT/;
option limrow = 20;
solve TiemposRed minimizing SumaTiempos using LP;
display TiempoTotalN.l, Tiempo.l;

```

Figura 2. Modelo 1: cálculo de la duración del proyecto y tiempos tempranos de los eventos

Para el segundo modelo, se incluyen los parámetros duración normal del proyecto que es igual a la duración del proyecto hallada con la variable auxiliar del primer modelo TiempoTotalN y los tiempos tempranos de los eventos TiemTemp(Nodos) a los cuales se les asignan los valores de las variables Tiempo.l(Nodos), obtenidos al ejecutar el primer modelo. Se incluyen además la variable TiempoT(Nodos) con la cual se calculan los tiempos tardíos de cada uno de los eventos, y las variables auxiliares para calcular cada una de las holguras asociadas, tanto a las actividades como a los eventos, como se presenta en la figura 3.

```

$Ontext
*****
Cálculo de los tiempos tardíos y de las holguras
*****
$Offtext

parameter
DuracNormal Duración normal del proyecto
TiemTemp(Nodos) Tiempos tempranos de los eventos;

DuracNormal = TiempoTotalN.l;
TiemTemp(Nodos) = Tiempo.l(Nodos);

variables
Tiempt(Nodos) Tiempos tardíos de los eventos
HolgT(i,j) Holgura total de la actividad i j
HolgL(i,j) Holgura libre
HolgI(i,j) Holgura independiente
HolgE(i) Holgura del evento;
positive variables TiempT;

equations
TiemposTard Sumatoria de los tiempos tardíos de los eventos
TiemptFinal(Nodos) El tiempo tardío en el último nodo es igual al temprano
TiempoNodoA(i,j) Cálculo de los tiempos tardíos de los eventos
HolguraT(i,j) Cálculo de la holgura total
HolguraL(i,j) Cálculo de la holgura libre
HolguraI(i,j) Cálculo de la holgura independiente
HolguraE(i) Cálculo de la holgura del evento;

TiemposTard.. SumaTiempos =e= sum(i, TiempT(i));
TiemptFinal(Nodos)$ (ord(Nodos)=card(Nodos)).. TiempT(Nodos)=E= DuracNormal;
TiempoNodoA(Arcos(i,j)).. TiempT(i) =l= TiempT(j) - DuracN(i,j);
HolguraT(Arcos(i,j)).. HolgT(i,j) =e=TiempT(j) - TiemTemp(i) - DuracN(i,j);
HolguraL(Arcos(i,j)).. HolgL(i,j) =e=TiempTemp(j) - TiemTemp(i) - DuracN(i,j);
HolguraI(Arcos(i,j)).. HolgI(i,j) =e=TiempTemp(j) - TiempT(i) - DuracN(i,j);
HolguraE(i).. HolgE(i) =e=TiempT(i) - TiemTemp(i);

model TiemposTardios
/TiemposTard,TiemptFinal,TiempoNodoA,HolguraT,HolguraL,HolguraI,HolguraE/;
solve TiemposTardios maximizing SumaTiempos using LP;
display TiempT.l, Tiempo.l, HolgT.l, HolgL.l, HolgI.l, HolgE.l;

```

Figura 3. Modelo 2: cálculo de los tiempos tardíos y de las holguras

El tercer modelo implementado, tiene como propósito establecer la duración del proyecto en tiempo límite, incluye las ecuaciones básicas del primer modelo, pero se cambia el parámetro correspondiente a la duración normal de las actividades por su duración límite, obteniendo en consecuencia el menor tiempo posible en que el proyecto puede realizarse, ver figura 4.

```

$Ontext
*****
Cálculo de la duración del proyecto en tiempo límite
*****
$Offtext

parameter
DuracL(i,j)      Duración límite de una actividad en un arco;

variable
TiempoTotalL    Tiempo límite de duración del proyecto;

equation
TiempoL(Nodos)  Tiempo total límite del proyecto
TiempoNodoL(i,j) Restricción de la estructura de la red;

TiempoL(Nodos)$ (ord(Nodos)=card(Nodos)).. TiempoTotalL =e= Tiempo(Nodos);
TiempoNodoL(Arcos(i,j)).. Tiempo(j) =g= Tiempo(i) + DuracL(i,j);

model TiemposRedL /TiempoL,TiempoNodoL/;

solve TiemposRedL minimizing TiempoTotalL using LP;
display TiempoTotalL.1;

```

Figura 4. Modelo 3: cálculo de la duración del proyecto en tiempo límite

El último modelo implementado consiste en la optimización del coste del proyecto en función del tiempo de ejecución del mismo, se incluyen como parámetros los costes normal y límite, los tiempos normal y límite de las actividades, a partir de los cuales se calculan tanto la reducción máxima posible de cada una de las actividades como el coste de reducción por unidad de tiempo; también se incluye el tiempo deseado para el desarrollo del proyecto (que debe estar entre los tiempos normal y límite). Como variable de decisión se tiene la reducción en el tiempo en cada una de las actividades y como variable de resultado el incremento resultante en el coste, ver figura 5.

```

$Ontext
*****
Optimización del coste del proyecto en función de la reducción del tiempo
*****
$Offtext

parameters
CostoN(i,j)      Coste normal de una actividad en un arco
CostoL(i,j)      Coste límite de una actividad en un arco
CostRed(i,j)     Coste total reducción
RedMaxi(i,j)     Reducción máxima de la actividad i j
CosteRed(i,j)    Coste por unidad de tiempo reducida
Tdeseado        Tiempo deseado de terminación del proyecto;

RedMaxi(i,j)$ (Arcos(i,j))= DuracN(i,j)-DuracL(i,j);
CostRed(i,j) = CostoL(i,j)-CostoN(i,j);
CosteRed(i,j)$ (RedMaxi(i,j) gt 0) = CostRed(i,j)/RedMaxi(i,j);
Tdeseado = TiempoTotalL.l;

variables
Red(i,j) Reducción en la duración de la actividad i j
CostReduc Costo de reducir la duración del proyecto;
positive variables Red;

equations
CostoRed        Incremento del coste por reducción en la duración
TiempoR(i,j)    Estructura de la red con reducción de tiempo
RedMax(i,j)     Reducción máxima de tiempo en la actividad
TiempDes(Nodos) Tiempo deseado de ejecución del proyecto;

CostoRed.. CostReduc =e= sum(i, sum(j, CosteRed(i,j) * Red(i,j)));
TiempoR(Arcos(i,j)).. Tiempo(j) =g= Tiempo(i) + DuracN(i,j)-Red(i,j);
RedMax(Arcos(i,j)).. Red(i,j) =l= DuracN(i,j)-DuracL(i,j);
TiempDes(Nodos)$ (ord(Nodos)=card(Nodos)).. Tiempo(Nodos) =e= Tdeseado;

model PertCost /CostoRed, TiempoR, RedMax, TiempDes/;
solve PertCost using lp minimizing CostReduc;
display Red.l, CostReduc.l;

```

Figura 5. Modelo 4: optimización del coste del proyecto en función de la reducción del tiempo

La figura 6 contiene los datos empleados en el modelo, de acuerdo con la estructura de la red y el lenguaje GAMS®.

```

$Ontext
Datos del Modelo
$Offtext

set Nodos Nodos de la red del
proyecto /1, 2, 3, 4, 5, 6, 7, 8, 9,
10/;
alias (Nodos, i, j);
set Arcos(i, j) Aristas o arcos de la
red
/ (1.2), (1.3), (1.4), (1.6), (3.7),
(4.8), (5.8), (6.10), (7.10), (7.9),
(8.9),
(9.10), (2.4), (2.5), (3.5), (6.7)/;

parameter
DuracN(i,j) Duración normal de una
actividad en un arco /
1.2 12
1.3 7
1.4 10
1.6 8
2.4 0
2.5 0
3.5 0
3.7 6
4.8 7
5.8 11
6.7 0
6.10 10
7.10 14
7.9 6
8.9 8
9.10 4
/;

parameter
DuracL(i,j) Duración límite de una
actividad en un arco /
1.2 8
1.3 4
1.4 7
1.6 6
2.4 0
2.5 0
3.5 0
3.7 3
4.8 5
5.8 9
6.7 0
6.10 8
7.10 12
7.9 3
8.9 7
9.10 3 /;

parameter
CostoN(i,j) Coste normal de una
actividad en un arco /
1.2 12
1.3 11
1.4 15
1.6 10
2.4 0
2.5 0
3.5 0
3.7 8
4.8 16
5.8 16
6.7 0
6.10 18
7.10 20
7.9 12
8.9 9
9.10 7
/;

parameter
CostoL(i,j) Coste límite de una
actividad en un arco /
1.2 20
1.3 14
1.4 21
1.6 15
2.4 0
2.5 0
3.5 0
3.7 13
4.8 20
5.8 21
6.7 0
6.10 23
7.10 25
7.9 16
8.9 12
9.10 9
/;

```

Figura 6. Archivo de datos DatosPert.gms

## 2.1. Resultados de la compilación en GAMS

El informe de salida de GAMS®, en este caso el archivo PERTCPM.lst, está estructurado de la siguiente forma: Archivos incluidos (*Include File Summary*), Compilación del código, Resultados de cada uno de los modelos y resumen de archivos (de entrada y de salida).

Por su parte los resultados de cada uno de los modelos ejecutados incluyen: los listados de las ecuaciones (*Equation Listing*) y de las variables (*Column Listing*), las estadísticas del modelo (*Model Statistics*) y el reporte de soluciones, el cual incluye los resultados de las ecuaciones y de las variables, y los datos (variables o parámetros) cuya presentación se solicita explícitamente en el código (*display*). La tabla 3 presenta un resumen de la ejecución de los modelos respectivos

**Tabla 3.**

Resumen de los resultados de los modelos

	Modelo 1: Cálculo de los tiempos tempranos de los eventos y duración del proyecto	Modelo 2: Cálculo de los tiempos tardíos de los eventos y de las holguras	Modelo 3: Cálculo de la duración del proyecto en tiempo límite	Modelo 4. Optimización del coste
<b>Estadísticas del modelo</b>				
Bloques de ecuaciones	3	7	2	4
Cantidad de ecuaciones	18	76	17	34
Bloques de variables	3	6	2	3
Cantidad de variables	12	69	11	27
Elementos diferentes de cero	45	144	34	78
<b>Resumen de la solución</b>				
Modelo	TiemposRed	TiemposTardios	TiemposRedL	PertCost
Objetivo	SumaTiempos	SumaTiempos	TiempoTotalL	CostReduc
Tipo	LP	LP	LP	LP
Sentido de optimización	Minimizar	Maximizar	Minimizar	Minimizar
Solver	CPLEX	CPLEX	CPLEX	CPLEX
Desde la línea	133	176	201	240
Estado de la solución	1 Finalización normal	1 Finalización normal	1 Finalización normal	1 Finalización normal
Estado del modelo	1 Óptimo	1 Óptimo	1 Óptimo	1 Óptimo
Valor objetivo	153.0000	183.0000	27.0000	18.0000

Los resultados para el ejemplo propuesto, correspondientes a los modelos 1 y 2 se resumen en las tablas 4 y 5, la primera muestra los tiempos tempranos y tardíos, y las holguras de los eventos; y la segunda, las holguras total, libre e independiente. El diagrama de la red con los tiempos se muestra en la figura 7.

**Tabla 4.**

Resumen de los resultados: tiempos tempranos, tiempos tardíos y holgura de los eventos

**Tiempo normal de duración del proyecto** (TiempoTotalN.l) = 35.000

Nodo	Tiempos tempranos de los eventos - Tiempo.l (E <sub>j</sub> )	Tiempos tardíos de los eventos TiempT.L (L <sub>j</sub> )	Holgura del evento HolgE.L (HE <sub>j</sub> )
1	0.000	0.000	
2	12.000	12.000	
3	7.000	12.000	5.000
4	12.000	16.000	4.000
5	12.000	12.000	
6	8.000	21.000	13.000
7	13.000	21.000	8.000
8	23.000	23.000	
9	31.000	31.000	
10	35.000	35.000	

**Tabla 5.**

Resumen de los resultados: holguras total, libre e independiente

Actividad	Arco	Holgura Total- HolgT(i,j)	Holgura Libre - HolgL(i,j)	Holgura Independiente HolgI(i,j)
A	(1, 2)			
B	(1, 3)	5.000		
C	(1, 4)	6.000	2.000	2.000
D	(1, 6)	13.000		
S_1	(2, 4)	4.000		
S_2	(2, 5)			
S_3	(3, 5)	5.000	5.000	
E	(3, 7)	8.000		-5.000
F	(4, 8)	4.000	4.000	
G	(5, 8)			
S_4	(6, 7)	13.000	5.000	-8.000



Actividad	Arco	Holgura Total- HolgT(i,j)	Holgura Libre - HolgL(i,j)	Holgura Independiente HolgI(i,j)
H	(6, 10)	17.000	17.000	4.000
J	(7, 9)	12.000	12.000	4.000
I	(7, 10)	8.000	8.000	
K	(8, 9)			
L	(9, 10)			

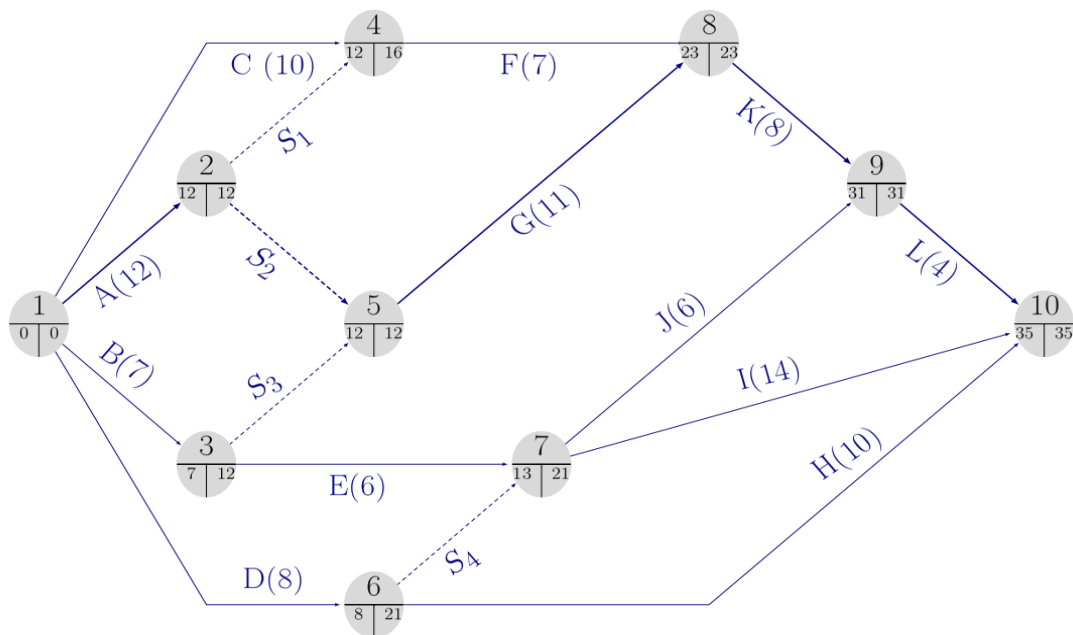


Figura 7. Diagrama de red con cálculo de los tiempos del proyecto

## 2.2. Análisis comparativo de la relación tiempo - coste

El coste total de realizar todas las actividades en tiempo normal es de 154.000, mientras que el de realizar todas las actividades en tiempo límite es de 209.000, con el modelo implementado se obtiene un coste total de realizar el proyecto en su duración límite igual a 172.000 con un ahorro de 37.000. La tabla 6 relaciona para diferentes duraciones factibles, el aumento de los costos y el costo total del proyecto

**Tabla 6.**

Análisis comparativo de la relación tiempo – coste en función de la duración del proyecto

Duración deseada (semanas)	35	34	33	32	31	30	29	28	27
Coste Reducción (miles de dólares)	0	2	4	6	8	10	12.5	15	18

Coste Total (miles de dólares)		154	156	158	160	162	164	166.5	169	172
Actividad	Arco									
A	(1, 2)	0	1	2	3	3	4	4	4	4
G	(5, 8)							1	2	2
K	(8, 9)									1
L	(9, 10)					1	1	1	1	1

De los resultados puede observarse que se puede llegar a realizar el proyecto en su duración límite de 27 semanas con un sobrecosto de 18.000 dólares, y que solo es necesario reducir la duración de las actividades determinadas por los arcos (1,2), (5,8), (8,9) y (9,10) en 4, 2, 1 y 1 unidades de tiempo, respectivamente.

La figura 8 muestra las relaciones entre el tiempo y el coste total, reflejando que el proceso de optimización desarrollado lleva a un coste total de 172.000 dólares con una duración de 27 semanas, lo que implica un ahorro de 37.000 dólares.

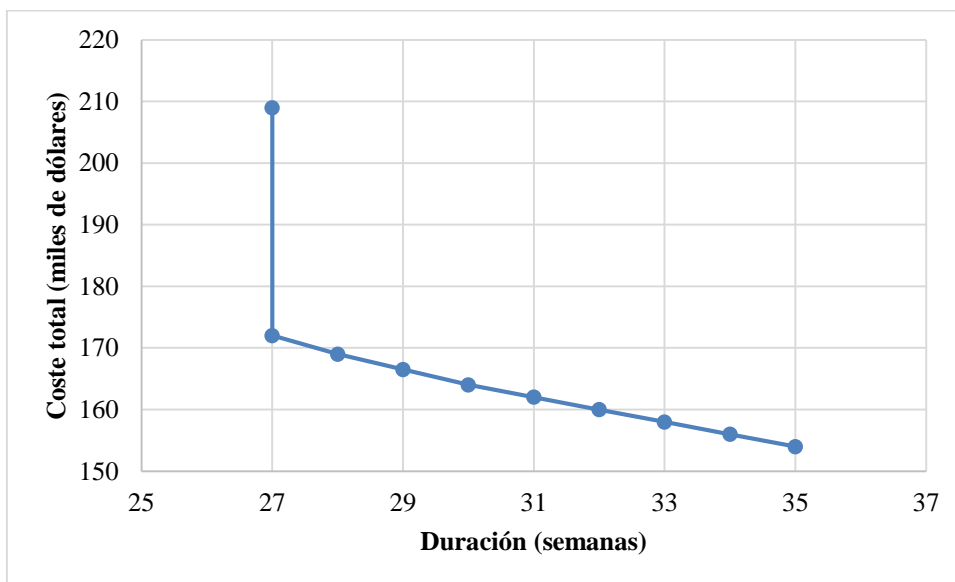


Figura 8. Comparación entre costo total y duración

### Conclusiones

La programación lineal es una herramienta muy poderosa y de gran utilidad en múltiples problemas de ingeniería. En esta oportunidad se presenta una aplicación a la gestión de proyectos en la cual se determinan tanto los tiempos tempranos como los tiempos tardíos de los eventos y a partir de ellos se calculan los diferentes tipos de holgura, los cuales sirven, como soporte para la toma de decisiones.

A pesar de los avances en la utilización de modelos de programación lineal a la gestión de proyectos, aún existen oportunidades de mejora como la que propone este artículo.

Asimismo, se hace una implementación de modelos de manera secuencial haciendo uso de la herramienta de optimización GAMS®, el cual es un *software* de modelado algebraico. Se resalta, la posibilidad de integrar varios modelos en un solo archivo, algunos de los cuales pueden estar basados en resultados previos. Además se resalta que este enfoque puede ser aplicado a proyectos de gran tamaño y por tanto, mayor complejidad como el presentado por (Renaud & Kumral, 2020).

Así este trabajo desarrolla de manera detallada los modelos y su implementación en el *software*, con el ánimo de motivar al lector al uso de herramientas matemáticas en el proceso de toma de decisiones.

Por último y tal como lo plantea (Dasović et al., 2020) la optimización en gestión de proyectos es un tema aún no resuelto en las herramientas informáticas disponibles en esta área, por lo cual trabajos como el que se presenta tienen particular importancia. Sin embargo, como lo señalan estos mismos autores, realizar este tipo de integración entre los lenguajes de modelado y los *software* de gestión de proyectos es aún una asignatura pendiente. También, se propone como un tema de investigación futura la construcción de modelos matemáticos para la gestión de los recursos en el marco de los proyectos.

### Referencias bibliográficas

- Babu, A. J. G., & Suresh, N. (1996). Project management with time, cost, and quality considerations. *European Journal of Operational Research*, 88(2), 320-327. [https://doi.org/10.1016/0377-2217\(94\)00202-9](https://doi.org/10.1016/0377-2217(94)00202-9)
- Dasović, B., Galić, M., & Klanšek, U. (2020). A survey on integration of optimization and project management tools for sustainable construction scheduling. *Sustainability (Switzerland)*, 12(8). Scopus. <https://doi.org/10.3390/SU12083405>
- Demirkesen, S., & Ozorhon, B. (2017). Measuring Project Management Performance: Case of Construction Industry. *Engineering Management Journal*, 29(4), 258-277. <https://doi.org/10.1080/10429247.2017.1380579>
- General Algebraic Modelling Language* (31.2.0). (2020). [Computer software]. GAMS Development Corporation. <http://www.gams.com>.
- Hillier, F. S., & Lieberman, G. J. (2010). *Introducción a la investigación de operaciones*. McGraw-Hill.
- Kalenatic, D., Blanco Rivero, L. E., Universidad Distrital Francisco José de Caldas (Bogotá), & Centro de de Investigaciones y Desarrollo Científico. (1993). *Aplicaciones computacionales en producción*. Fondo Editorial Universidad Distrital Francisco José de Caldas.
- Khang, D. B., & Myint, Y. M. (1999). Time, cost and quality trade-off in project management: A case study. *International Journal of Project Management*, 17(4), 249-256. [https://doi.org/10.1016/S0263-7863\(98\)00043-X](https://doi.org/10.1016/S0263-7863(98)00043-X)
- Kim, J., Kang, C., & Hwang, I. (2012). A practical approach to project scheduling: Considering the potential quality loss cost in the time–cost tradeoff problem. *International Journal of Project Management*, 30(2), 264-272. <https://doi.org/10.1016/j.ijproman.2011.05.004>

- Project Management Institute, & Project Management Institute. (2017). *Guía de los fundamentos para la dirección de proyectos: (Guía del PMBOK)*.
- Renaud, M., & Kumral, M. (2020). Planning a Complex Mine Construction Project under Price Cyclicity. *Engineering Management Journal*, 32(2), 120-129. <https://doi.org/10.1080/10429247.2020.1718461>
- Reyes, J. N. E. (s. f.). *Análisis de la gestión de proyectos a nivel mundial*. 38.
- Serrador, P., & Turner, R. (2015). The Relationship Between Project Success and Project Efficiency. *Project Management Journal*, 46. <https://doi.org/10.1002/pmj.21468>
- Taha, H. A. (2012). *Investigación de operaciones*. Pearson Educación.
- Williams, T. (2015). Identifying Success Factors in Construction Projects: A Case Study. *Project Management Journal*, 47, n/a-n/a. <https://doi.org/10.1002/pmj.21558>
- Winston, W. L. (2005). *Investigación de operaciones: Aplicaciones y algoritmos*. Cengage Learning.