Tema 7: Optimización de redes logísticas.

Nota importante:

Los temas 7 y 8 están dedicados a las aplicaciones de la programación matemática a dos áreas muy significativas de la ingeniería de sistemas: las redes logísticas y las plantas industriales. Por motivos de eficiencia plantearemos estos temas del curso como dos dominios alternativos en los que los alumnos puedan desarrollar un proyecto de programación matemática que ponga de manifiesto la utilidad de esta tecnología para abordar problemas reales de la ingeniería de sistemas. Esto significa que aunque ambos temas deberán ser estudiados por los alumnos, los ejercicios correspondientes quedarán unificados en uno único, válido como ejercicio para ambos temas, en el que se abordará el desarrollo de un pequeño proyecto de optimización de una red logística o una aplicación industrial.

Objetivos del tema:

- ❑ Conocer el planteamiento de la gestión de una red de transporte logístico como un problema de optimización matemática
- ❑ Estudiar redes logísticas de transporte
- ❑ Estudiar redes logísticas de conducción de fluidos
- ❑ Analizar y plantear algunos ejemplos

Redes logísticas

La logística se ocupa del *proceso de planificación, implementación y control del flujo eficiente de mercancías, energía o información desde los puntos donde se originan hasta los puntos donde se consumen.*

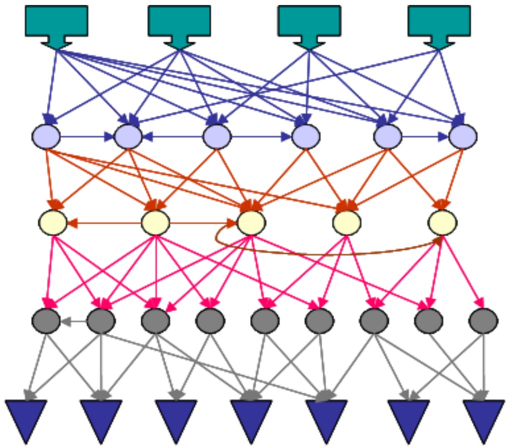
Una red logística podemos verla como un grafo compuesto por nodos y arcos. Los nodos representan los agentes de una organización (factorías, almacenes, centros de distribución, clientes, etc.), y los arcos son los diferentes medios de transporte entre nodos, por ejemplo trenes, barcos gasoductos, poliductos, etc.

En términos muy generales podemos decir que una red logística adquiere productos primarios (energía, información, materias primas, etc.), los transforma en productos finales y los distribuye a sus clientes.

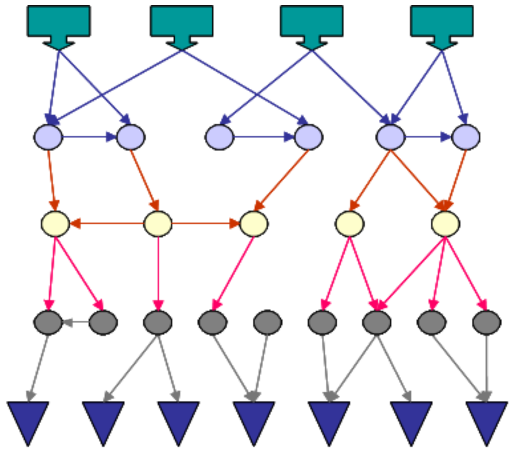
La gestión de una red logística consiste en tomar las decisiones que optimizan su funcionamiento. La función de óptimo se corresponde generalmente con una función de coste (minimización de gasto y maximización de beneficios) aunque pueden existir términos en esta función relacionados con otros aspectos del funcionamiento, como por ejemplo la garantía de niveles de seguridad de los *stocks*.

Un sistema de decisión logística parte del conocimiento de las alternativas de transporte y transformación de la red y determina el subconjunto que satisface unos objetivos preestablecidos. La calidad del subconjunto seleccionado se mide en términos de una función objetivo.

Red logística de alternativas



Red solución



Estos sistemas de decisión se plantean frecuentemente como problemas de optimización matemática y serán objeto de estudio en este tema.

Niveles en la gestión de una red logística

La gestión de una red logística se suele realizar a tres niveles diferentes: estratégico, táctico y operacional, dependiendo del horizonte temporal en el que se toman las decisiones. A estos tres niveles habría que añadir un cuarto, el nivel de control, que corresponde al funcionamiento en tiempo real de la red.

El **nivel estratégico** define la estructura de la red logística, es decir, los medios de producción, almacenamiento y transporte disponibles para un horizonte temporal amplio, de varios años. Los estudios estratégicos tienen por objetivo determinar la mejor estructura de una red logística a partir de datos históricos conocidos y de previsiones estimadas.

El **nivel táctico** planifica el funcionamiento de la red logística existente para satisfacer una demanda estimada en un horizonte temporal medio, del orden de meses. La planificación táctica de la red determina la utilización óptima de sus recursos en el período fijado.

El **nivel operacional** ejecuta los planes del nivel táctico sobre períodos temporales cortos, normalmente días.

Finalmente el **nivel de control** realiza el seguimiento en tiempo real de la planificación operacional.

En este tema nos centraremos en la planificación táctica.

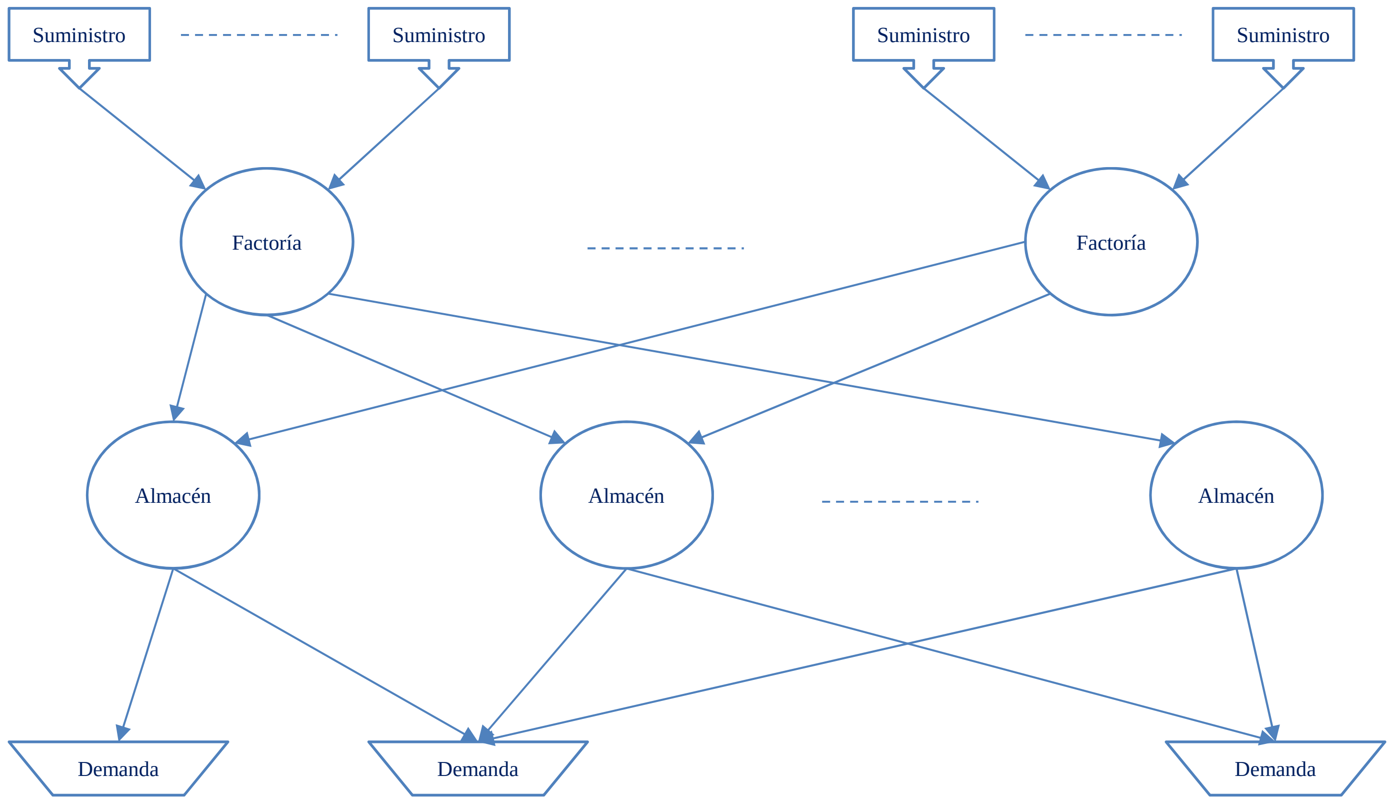
Desde el punto de vista matemático un sistema de planificación logística resolverá un problema de optimización con restricciones, es decir, permitirá expresar el modelo de programación matemática de la red por un conjunto de *variables de decisión*, un conjunto de *restricciones* sobre estas variables, y una *función objetivo* que mida la calidad de la solución:

$$\begin{aligned}
&\textit{Variables de decisión}:\ x_1, x_2, \ldots x_i, \ldots \\
&\textit{minimizar}\ \ f_o(\ldots x_i \ldots) \\
&\textit{sujeto a}\ R_j(\ldots x_i \ldots)
\end{aligned}$$

Contemplaremos en este tema dos tipos de redes logísticas: las de transporte y las de conducción de fluidos.

Redes de transporte

En este tipo de redes se parte de un conjunto de fuentes que suministran materias primas. Estas materias primas son transformadas en productos elaborados, almacenados temporalmente y transportados por la red hasta alcanzar los puntos de demanda (consumo). En la siguiente figura hemos representado un esquema general de este tipo de redes:



Dependiendo de su naturaleza estas redes se suelen planificar de dos modos:

1. Planificación **guiada por la demanda**. Operan con los datos de previsión de una demanda que tiene que satisfacerse al menor coste posible, es decir, utilizando las materias primas, almacenamientos, transformaciones y transporte que minimicen el coste.

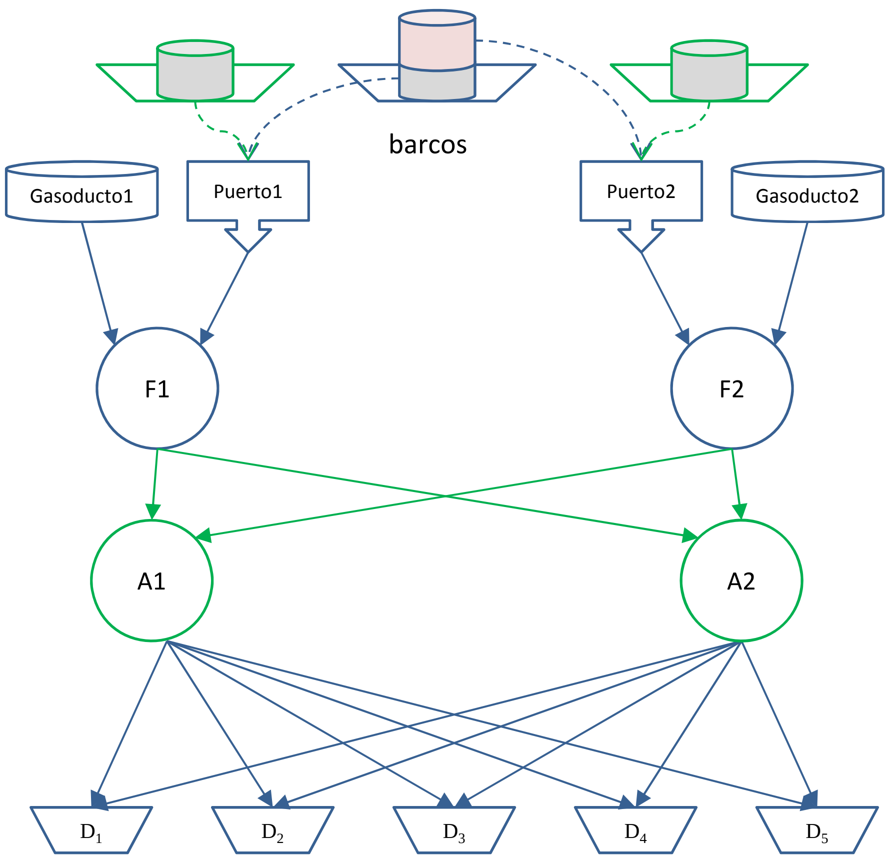
2. Planificación **guiada por la oferta**. Operan con los datos de una previsión de suministro que tiene que procesarse y transportarse a los puntos alternativos de demanda con el menor coste.

Ejemplo de red de transporte logístico guiada por la demanda.

El problema de barco a doble puerto planteado en el tema 2 (transparencia 26) es un ejemplo de red logística de transporte guiada por la demanda. Este ejemplo se puede ampliar en múltiples direcciones para dar lugar a una red logística de cierta complejidad.

Posibles ampliaciones:

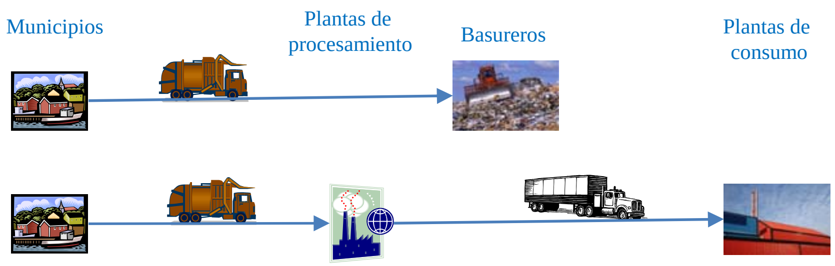
- Convertirla en **una red multi-etapa** introduciendo el tiempo (ver ejemplo "Producción multi-período" en la transparencia 26 del tema 3)
- Utilizar **barcos a puerto único** (además o en sustitución de los de doble puerto)
- Introducir **almacenamiento intermedio** entre factorías y demanda
- Utilizar **dos o más tipos de materias primas**
- Usar un **proceso de mezcla** en la producción de las factorías
- Introducir algún medio de **trasporte discreto**
- Imponer **suministro exclusivo** en la demanda desde un único almacén



Ejemplo de red de transporte logístico guiada por el suministro: red de eliminación de basura

Se trata de implementar una red de eliminación de basura para un conjunto de municipios. La basura se elimina por dos vías:

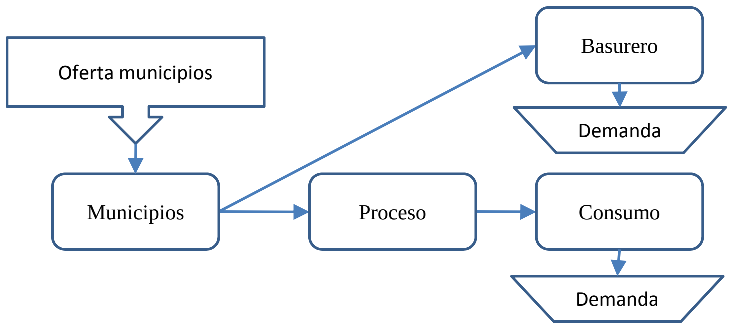
1) Traslado directo a los basureros
2) Traslado a plantas de procesamiento y posterior uso en plantas de consumo (combustible, abono, etc.).



Se disponen de los siguientes datos:

- Oferta en Tn de basura generada en cada municipio.
- Distancia en Km de los municipios a las plantas de procesamiento
- Distancia en Km de los municipios a los basureros.
- Distancia en Km de las plantas de procesamiento a las plantas de consumo.
- Coste del procesamiento en Euros/Tn
- Oferta mínima y máxima de cada basurero y precio pagado en Euros/Tn.
- Oferta mínima y máxima de cada planta de consumo y precio en Euros/Tn
- El costo del transporte es 1 euro/Km/Tn

Se trata de determinar a donde se lleva la basura de cada municipio a fin de obtener el máximo beneficio (o menor coste).



Red de eliminación de basura: datos

Municipios	Basura
M1	53
M2	17
M3	23
M4	79
M5	89
M6	23
M7	11
M8	61
M9	14
M10	59
M11	92
M12	21
M13	12
M14	77
M15	79
M16	21
	(oferta) Tn

Municipios Basureros	B1	B2	B3	B4	B5	B6	B7	B8	B9
M1	90	82	88	26	74	95	16	67	38
M2	50	49	66	46	36	92	36	15	53
M3	20	30	39	36	57	98	60	57	98
M4	90	73	76	56	68	38	64	25	16
M5	60	57	98	36	15	53	76	56	68
M6	10	23	69	66	94	25	90	28	41
M7	50	20	30	36	67	98	54	96	42
M8	50	20	30	36	67	98	30	66	34
M9	60	57	98	36	15	53	90	64	25
M10	10	23	69	66	94	25	66	34	42
M11	30	40	46	46	43	88	90	82	88
M12	90	64	25	16	35	48	16	67	38
M13	40	12	34	46	46	15	46	46	15
M14	60	47	52	86	52	43	30	40	46
M15	50	31	31	16	67	38	16	67	38
M16	50	31	31	16	67	38	16	67	38
	Distancia (Km)								

Demanda			
Basureros	Min	Max	Precio
B1	0	90	90
B2	0	82	82
B3	0	88	88
B4	0	26	26
B5	0	74	74
B6	0	95	95
B7	0	67	67
B8	0	89	89
B9	0	33	33
	Tn	Tn	Euros/Tn

Demanda			
Consumos	Min	Max	Precio
C1	0	9	90
C2	0	82	82
C3	0	88	88
C4	0	26	26
C5	0	74	74
C6	0	95	95
C7	0	67	67
C8	0	89	89
C9	0	33	33
C10	0	95	44
C11	0	67	23
	Tn	Tn	Euros/Tn

Municipios Procesos	P1	P2	P3	P4	P5	P6
M1	90	82	88	26	74	95
M2	50	49	66	46	36	92
M3	20	30	39	36	57	98
M4	90	73	76	56	68	38
M5	60	57	98	36	15	53
M6	10	23	69	66	94	25
M7	50	20	30	36	67	98
M8	50	20	30	36	67	98
M9	60	57	98	36	15	53
M10	10	23	69	66	94	25
M11	30	40	46	46	43	88
M12	90	64	25	16	35	48
M13	40	12	34	46	46	15
M14	60	47	52	86	52	43
M15	50	31	31	16	67	38
M16	50	31	31	16	67	38
	Distancia (Km)					

Procesos	Coste
P1	90
P2	82
P3	88
P4	26
P5	74
P6	95
	Euros/Tn

Procesos Consumos	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11
P1	90	82	88	26	74	95	16	67	38	74	95
P2	50	49	66	46	36	92	36	15	53	74	95
P3	20	30	39	36	57	98	60	57	98	74	95
P4	90	73	76	56	68	38	64	25	16	74	95
P5	60	57	98	36	15	53	76	56	68	74	95
P6	10	23	69	66	94	25	90	28	41	74	95
	Distancia (Km)										

Red de eliminación de basura: modelo matemático

Variables de decisión

$transporte\_mb_{mb}$: toneladas de basura transportada directamente desde los municipios a los basureros

$transporte\_mp_{mp}$: toneladas de basura transportada desde los municipios a las plantas de proceso

$ransporte\_pc_{pc}$: toneladas de basura transportada desde las plantas de proceso a las plantas de consumo

$cantidad\_p_p$: toneladas de basura procesada por el proceso p

$cantidad\_b_b$: toneladas de basura recibidas en el basurero b

$cantidad\_c_c$: toneladas de basura procesada recibidas en la planta de consumo c

.

Restricciones

$\forall m \in municipios$

$$\sum_{b \in basureros} transporte\_mb_{mb} + \sum_{p \in procesos} transporte\_mp_{mp} = basura_m$$

$\forall p \in procesos$

$$cantidad\_p_p = \sum_{m \in municipios} transporte\_mp_{mp}$$

$$\sum_{c \in consumos} transporte\_pc_{pc} = cantdad\_p_p$$

$\forall b \in basureros$

$$\sum_{m \in municipios} transporte\_mb_{mb} = cantdad\_b_b$$

$$mínimo\_b_b \le cantidad\_b_b \le máximo\_b_b$$

$\forall c \in consumos$

$$\sum_{p \in procesos} transporte\_pc_{pc} = cantdad\_c_c$$

$$mínimo\_c_c \le cantidad\_c_c \le máximo\_c_c$$

Función de coste

$$\sum_{m \in municipios} \sum_{b \in basureros} transporte\_mb_{mb} \cdot distancia\_mb_{mb} +$$
$$\sum_{m \in municipios} \sum_{p \in procesos} transporte\_mp_{mp} \cdot distancia\_mp_{mp} +$$
$$\sum_{p \in procesos} \sum_{c \in consumos} transporte\_pc_{pc} \cdot distancia\_pc_{pc} +$$
$$\sum_{p \in procesos} coste\_p_p \cdot cantidad\_p_p -$$
$$\sum_{b \in basureros} coste\_b_b \cdot cantidad\_b_b -$$
$$\sum_{c \in consumos} coste\_c_c \cdot cantidad\_c_c$$

Red de eliminación de basura: OPL (datos)

```
//  CONJUNTOS
{string} basureros = {"B1","B2","B3","B4","B5","B6","B7","B8","B9"};
{string} municipios = {"M1","M2","M3","M4","M5","M6","M7","M8","M9","M10","M11","M12","M13","M14","M15","M16"};
{string} procesos = {"P1","P2","P3","P4","P5","P6"};
{string} consumos = {"C1","C2","C3","C4","C5","C6","C7","C8","C9","C10","C11"};
```

```
// DATOS

float basura[municipios] = [53, 17, 23, 79, 89, 23, 11, 61, 14, 59, 92, 21, 12, 77, 79, 21];

float distancia_mb[municipios][basureros]= [
[90, 82, 88, 26,74,95,16,67,38],
[50, 49, 66, 46,36,92,36,15,53],
[20, 30, 39, 36,57,98,60,57,98],
[90, 73, 76, 56,68,38,64,25,16],
[60, 57, 98, 36,15,53,76,56,68],
[10, 23, 69, 66,94,25,90,28,41],
[50, 20, 30, 36,67,98,54,96,42],
[50, 20, 30, 36,67,98,30,66,34],
[60, 57, 98, 36,15,53,90,64,25],
[10, 23, 69, 66,94,25,66,34,42],
[30, 40, 46, 46,43,88,90,82,88],
[90, 64, 25, 16,35,48,16,67,38],
[40, 12, 34, 46,46,15,46,46,15],
[60, 47, 52, 86,52,43,30,40,46],
[50, 31, 31, 16,67,38,16,67,38],
[50, 31, 31, 16,67,38,16,67,38]
];
```

```
float distancia_mp[municipios][procesos]= [
[90, 82, 88, 26,74,95],
[50, 49, 66, 46,36,92],
[20, 30, 39, 36,57,98],
[90, 73, 76, 56,68,38],
[60, 57, 98, 36,15,53],
[10, 23, 69, 66,94,25],
[50, 20, 30, 36,67,98],
[50, 20, 30, 36,67,98],
[60, 57, 98, 36,15,53],
[10, 23, 69, 66,94,25],
[30, 40, 46, 46,43,88],
[90, 64, 25, 16,35,48],
[40, 12, 34, 46,46,15],
[60, 47, 52, 86,52,43],
[50, 31, 31, 16,67,38],
[50, 31, 31, 16,67,38]
];

float distancia_pc[procesos][consumos]= [
[90, 82, 88, 26,74,95,16,67,38,74,95],
[50, 49, 66, 46,36,92,36,15,53,74,95],
[20, 30, 39, 36,57,98,60,57,98,74,95],
[90, 73, 76, 56,68,38,64,25,16,74,95],
[60, 57, 98, 36,15,53,76,56,68,74,95],
[10, 23, 69, 66,94,25,90,28,41,74,95]
];

float coste_p[procesos]= [90, 82, 88, 26, 74, 95];

float coste_b[basureros] = [90, 82, 88, 26, 74, 95, 67, 89, 33];
float minimo_b[basureros]= [0, 0, 0, 0, 0, 0, 0, 0, 0];
float maximo_b[basureros]= [90, 82, 88, 26, 74, 95, 67, 89, 33];

float coste_c[consumos] = [90, 82, 88, 26, 74, 95, 67, 89, 33, 44, 23];
float minimo_c[consumos]= [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0];
float maximo_c[consumos]= [90, 82, 88, 26, 74, 95, 67, 89, 33, 95, 67];
```

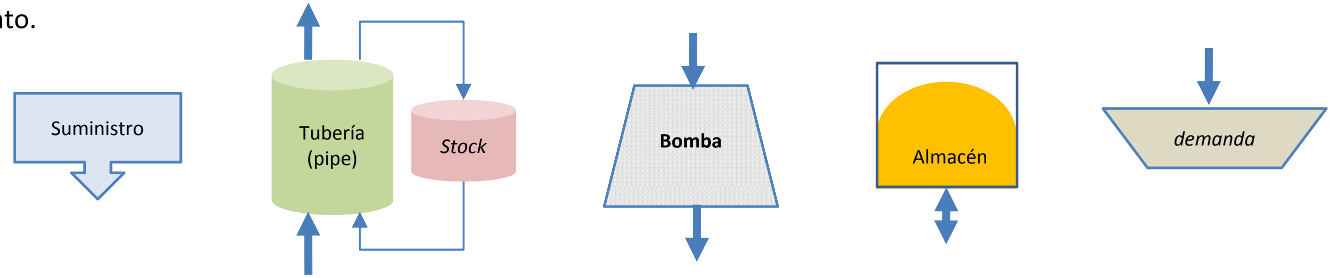
Red de eliminación de basura: OPL (modelo y resultados)

```
// VARIABLES DE DECISION
dvar float+ transporte_mb[municipios][basureros];
dvar float+ transporte_mp[municipios][procesos];
dvar float+ transporte_pc[procesos][consumos];

dvar float+ cantidad_p[procesos];
dvar float+ cantidad_b[basureros];
dvar float+ cantidad_c[consumos];

// FUNCION OBJETIVO
minimize sum(m in municipios, b in basureros)transporte_mb[m][b]*distancia_mb[m][b]+
        sum(m in municipios, p in procesos)transporte_mp[m][p]*distancia_mp[m][p]+
        sum(p in procesos, c in consumos)transporte_pc[p][c]*distancia_pc[p][c]+
        sum(p in procesos)coste_p[p]*cantidad_p[p]-
        sum(b in basureros)coste_b[b]*cantidad_b[b]-
        sum(c in consumos)coste_c[c]*cantidad_c[c];

// RESTRICCIONES
subject to
{
 forall(m in municipios)
   sum(b in basureros)transporte_mb[m][b]+sum(p in procesos)transporte_mp[m][p]== basura[m];

 forall(p in procesos)
{
   cantidad_p[p] == sum(m in municipios)transporte_mp[m][p];
   sum(c in consumos)transporte_pc[p][c] == cantidad_p[p];
}

forall(b in basureros)
 {
   sum(m in municipios)transporte_mb[m][b] == cantidad_b[b];
   minimo_b[b] <= cantidad_b[b] <= maximo_b[b];
 }

 forall(c in consumos)
 {
   sum(p in procesos)transporte_pc[p][c] == cantidad_c[c];
   minimo_c[c] <= cantidad_c[c] <= maximo_c[c];
 }
}
```

```
transporte_mb = [
              [ 0  0  0 0  0  0 53  0  0]
              [ 0  0  0 0  0  0  0 17  0]
              [23  0  0 0  0  0  0  0  0]
              [ 0  0  0 0  0  0  0 60 19]
              [ 0  0  0 0 74  0  0  0  0]
              [ 0  0  0 0  0 23  0  0  0]
              [ 0 11  0 0  0  0  0  0  0]
              [ 0 61  0 0  0  0  0  0 0]
              [ 0  0  0 0  0  0  0  0 14]
              [52  0  0 0  0  7  0  0  0]
              [15 10 67 0  0  0  0  0  0]
              [ 0  0 21 0  0  0  0  0  0]
              [ 0  0  0 0  0 12  0  0  0]
              [ 0  0  0 0  0 53 12 12  0]
              [ 0  0  0 0  0  0  2  0  0]
              [ 0  0  0 0  0  0  0  0  0]];

transporte_mp = [
              [0 0 0  0 0 0]
              [0 0 0  0 0 0]
              [0 0 0  0 0 0]
              [0 0 0  0 0 0]
              [0 0 0 15 0 0]
              [0 0 0  0 0 0]
              [0 0 0  0 0 0]
              [0 0 0  0 0 0]
              [0 0 0  0 0 0]
              [0 0 0  0 0 0]
              [0 0 0  0 0 0]
              [0 0 0  0 0 0]
              [0 0 0  0 0 0]
              [0 0 0  0 0 0]
              [0 0 0 77 0 0]
              [0 0 0 21 0 0]];

transporte_pc = [
              [0 0 0 0 0  0 0  0 0 0 0]
              [0 0 0 0 0  0 0  0 0 0 0]
              [0 0 0 0 0  0 0  0 0 0 0]
              [0 0 0 0 0 24 0 89 0 0 0]
              [0 0 0 0 0  0 0  0 0 0 0]
              [0 0 0 0 0  0 0  0 0 0 0]];

cantidad_p = [0 0 0 113 0 0];

cantidad_b = [90 82 88 0 74 95 67 89 33];

cantidad_c = [0 0 0 0 0 24 0 89 0 0 0];
```

Redes de conducción de fluidos

Las redes logísticas de conducción de fluidos están compuestas por un conjunto de tuberías (*pipes*) interconectadas por las que circulan fluidos de baja viscosidad (gases o líquidos no viscosos). Como en el caso de las redes de mercancías existen unas fuentes de materia prima que alimenta la red (*suministro*) y unos puntos de consumo que hacen de sumideros de la red (*demanda*). Repartidos por toda la red existen unas unidades de bombeo encargadas de transmitir al fluido la energía necesaria para su transporte por la red (*bombas*). Esta energía se consume en vencer la resistencia de las tuberías y los desniveles de las mismas. Además, suelen existir unos elementos que permiten el almacenamiento temporal de fluido (*almacenamientos*). En la siguiente figura se presentan los símbolos gráficos de cada elemento.



Las propias tuberías se contemplan en muchos problemas como elementos de almacenamiento distribuidos por toda la red. Por ello se suele representar gráficamente como un *stock* asociado a la función pura de trasmisión de fluido.

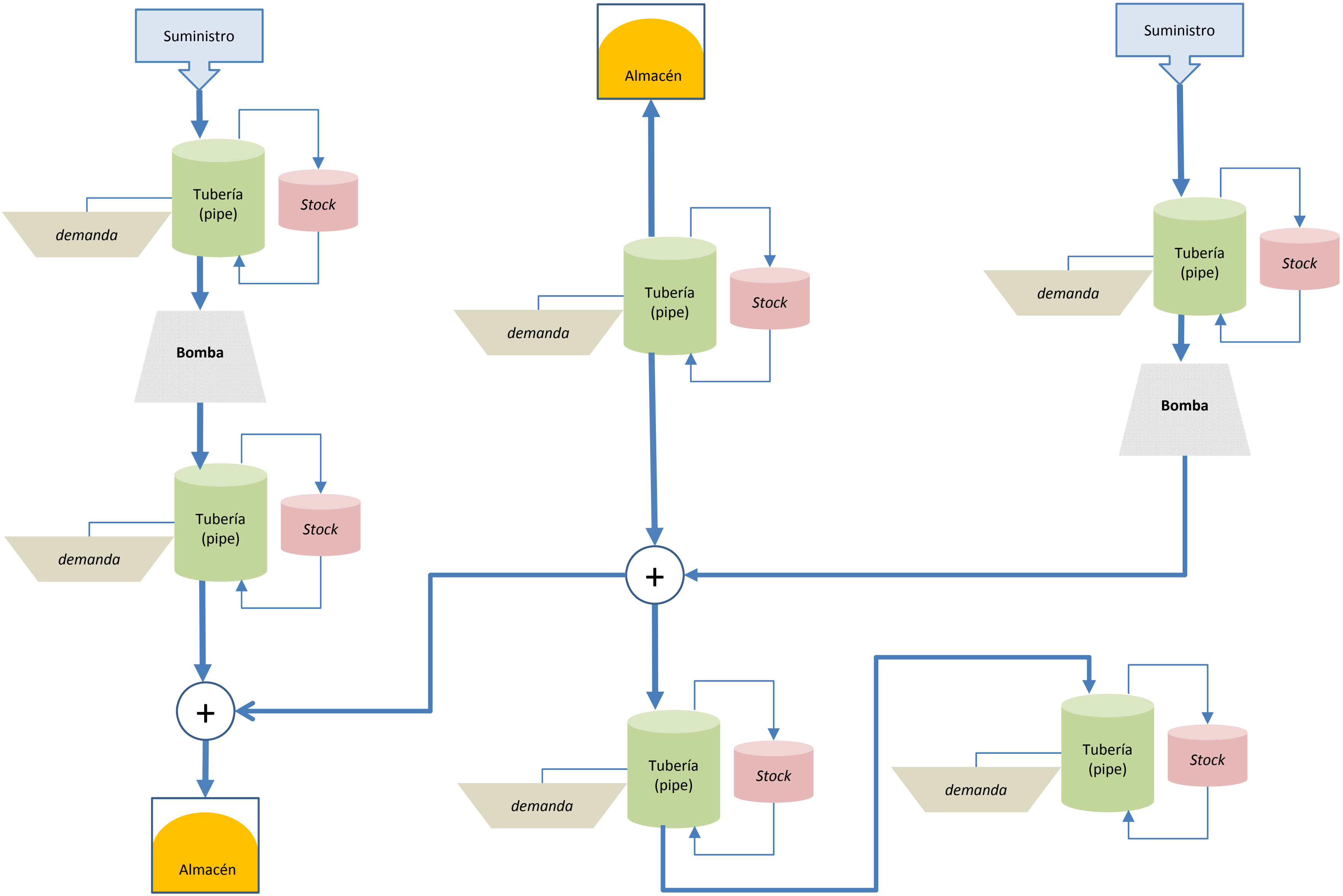
El modelo matemático de un sistema de planificación y optimización de una red de conducción de fluidos podemos organizarlo como una red de *componentes relacionales* interconectados entre sí, con una topología idéntica a la presentada por la red real, de manera que exista una correspondencia biunívoca entre los componentes relacionales del modelo y los elementos físicos de la red.

Cada componente relacional vendrá definido por un conjunto de variables externas (variables universales) cuyos valores deberán ser siempre compatibles con el comportamiento físico del elemento al que pertenecen, y un conjunto de variables internas (variables existenciales) sometidas a las restricciones que implementan el comportamiento externo del elemento.

De esta forma conseguimos que el modelo se construya siguiendo una estrategia de componentes, es decir, encapsulando las restricciones de cada elemento dentro de los límites de su correspondiente componente relacional. Cada componente sólo podrá interactuar con el resto de la red a través de las variables externas, que tendrán un modo de funcionamiento multidireccional, es decir, relacional y no funcional.

Esta estrategia de diseño por componentes permitirá una mejor comprensión del modelo matemático de la red, al explicitar la correspondencia entre elementos físicos de la red y componentes relacionales del modelo, lo que facilitará en gran medida la depuración del sistema en la fase de desarrollo y su mantenimiento, modificación y posible ampliación en la fase de explotación.

Ejemplo de red de conducción de fluidos



Modelo lineal de una red de fluidos: dimensión espacial

En el análisis del modelo matemático para la planificación y optimización de una red de fluido vamos a distinguir entre sus dos dimensiones: espacial y temporal. La primera viene determinada por los elementos presentes en la red y su topología de interconexión. La segunda la impone el carácter multi-período que requiere la planificación, y va a exigir replicar instancias del modelo espacial enlazadas a través de variables de decisión que operan a modo de variables de estado.

Para la expresión de la dimensión espacial del modelo utilizaremos un planteamiento relacional (en el sentido expresado en la transparencia 9). Cada elemento i se modelará como una restricción sobre un conjunto de variables globales: .

$$\forall i \in elementos\ de\ la\ red$$
$$R_i(V_{i1}, V_{i2}, \ldots, V_{ip})$$

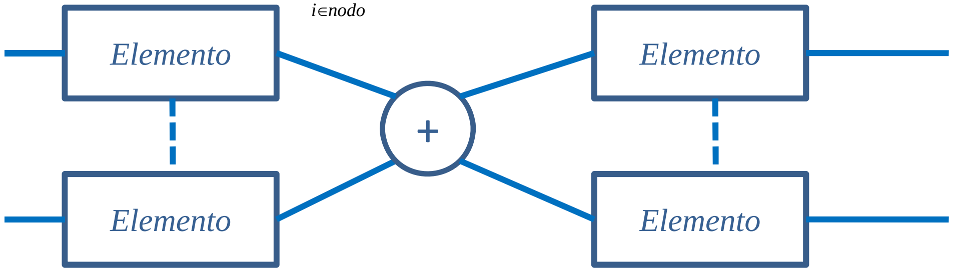
La interconexión de dos o más elementos se hará imponiendo unas restricciones a determinadas variables globales de los respectivos elementos. En el caso de los fluidos estas variables son la presión y el flujo.

La restricción sobre las presiones será la de igualdad:

$$\forall i, j \in nodo$$
$$Presión_i = Presión_j$$

La restricción sobre el flujo será la de conservación, es decir, de suma algebraica cero para todos los flujos que concurren en un punto de la red.

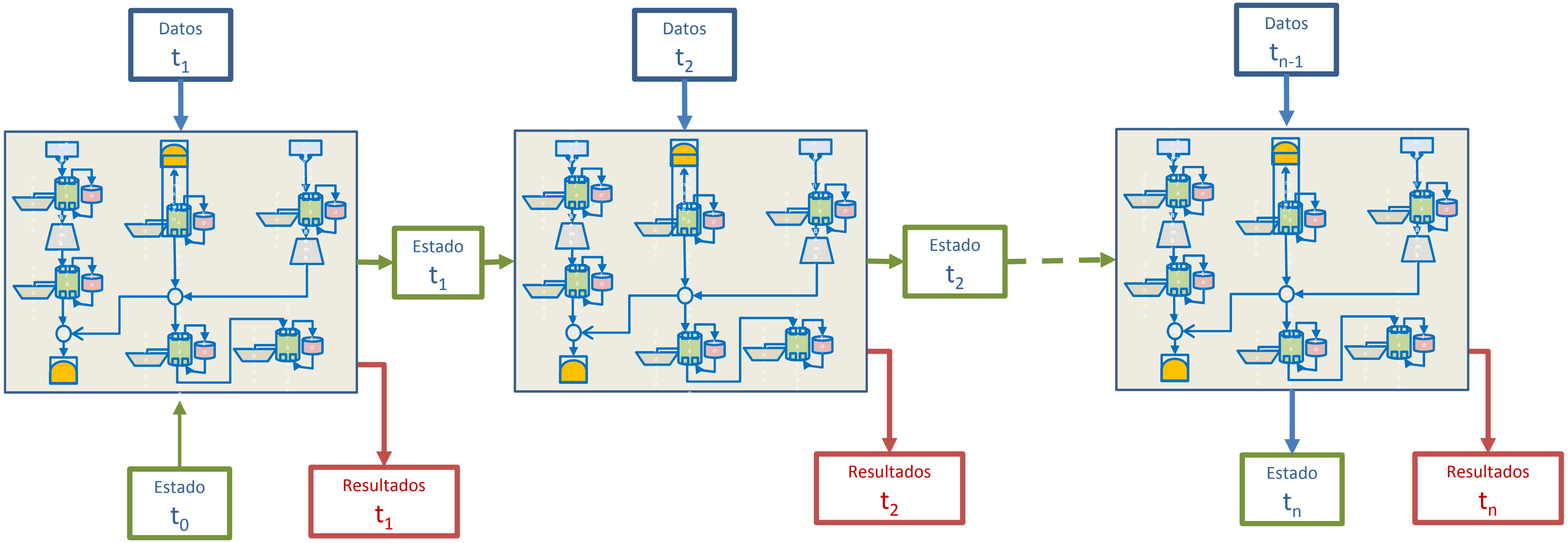
$$\sum_{i \in nodo} Flujo_i = 0$$



En el nivel de resolución que vamos a utilizar no vamos a modelar las presiones.

Modelo lineal de una red de fluidos: dimensión temporal

La dimensión temporal exige una repetición del conjunto de restricciones espaciales un número de veces igual al número de unidades de tiempo (períodos) que tenga el intervalo de optimización. La instancia espacial del modelo correspondiente al período de tiempo t recibirá los valores de estado del tiempo anterior (t-1) y los datos de entrada conocidos o estimados para el tiempo t:



El modelo matemático calculará los resultados de la planificación a lo largo del intervalo considerado (t_1, t_2, …, t_n) resolviendo el conjunto de restricciones y seleccionando la de mejor función objetivo:

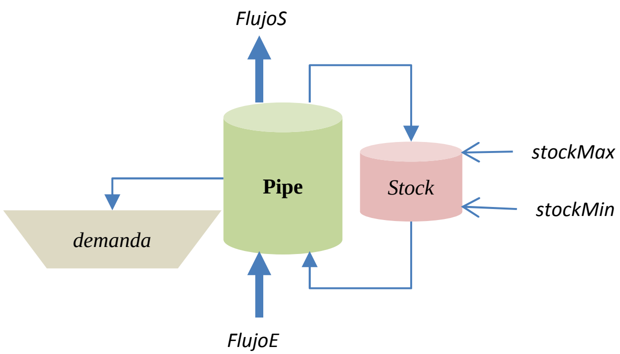
$$\forall t \in T\,(intervalo\ de\ optimización)$$
$$\forall i \in elementos\ de\ la\ red$$
$$R_i(V_{i1}, V_{i2}, \ldots, V_{ip}, Estado_{t-1}, Datos_t, t)$$

Modelo de tubería de conducción (pipe) 1

Las tuberías son las encargadas de conducir el fluido entre dos nodos de la red. El número de tuberías en toda la red dependerá del nivel de resolución que se haya fijado en el modelo. Cada tubería modela la parte de la red que suministra una cantidad de fluido demandada en una zona de la misma. Además, sirve de vehículo para transportar fluido hacia otras zonas.

Cada tubería tiene asociado un stock de fluido, que es la cantidad que almacena la tubería. Este stock está limitado por unos valores máximo y mínimo característicos de la tubería.

El modelo asociado a una tubería contemplará la posibilidad de que el fluido circule en las dos direcciones en períodos de tiempo diferentes manteniendo en ambas posibilidades los mismos datos. La expresión relacional del modelo será la siguiente:



Variables globales:

$FlujoE_t$: flujo de entrada

$FlujoS_t$: flujo de salida

$StockP_t$: cantidad de fluido que queda en la tuberia (estado)

Variables locales:

no existen en este modelo

Datos:

$stockMaxP$: cantidad máxima de fluido que puede quedar en la tubería

$stockMinP$: cantidad mínima de fluido que puede quedar en la tubería

$demanda_t$: cantidad de fluido que suministra la tubería a la demanda local

Modelo relacional:

$$\forall t \in T\,(\textit{intervalo de optimización})$$

$$R_{pipe}(FlujoS_t, FlujoE_t, StockP_t, datosP_t, estadoP_{t-1})$$

$$datosP_t \equiv (stockMaxP, stockMinP, estadoP_{t-1}, demanda_t)$$

$$estadoP_{t-1} \equiv stockP_{t-1}$$

Modelo de tubería de conducción (pipe) 2

La expresión relacional de la tubería la establece la ecuación de balance: el flujo de entrada (*FlujoE*$_t$) en un instante más el stock en el instante anterior (*Stock*$_{t-1}$) será igual al flujo de salida (*FlujoS*$_t$) más la demanda (*Demanda*$_t$) y más el stock que queda en la tubería (*Stock*$_t$) . Además habrá que añadir la restricción que limita el stock que puede quedar en la tubería. La restricción matemática será la siguiente:

$$\forall t \in T$$
$$FlujoS_t = FlujoE_t + Sock_{t-1} - Stock_t - demanda_t$$
$$Stock_{t_0} = stockInit$$
$$stockMin \le Stock_t \le stockMax$$

OPL

```
StockP0 == stockInitP;
 forall(t in T)
 {
  FlujoPSt== FlujoPEt+StockPt-1-StockPt-demandaPt;
  StockPt <= stockMaxP;
  StockPt >= stockMinP;
 }
```

Para hacer pruebas vamos a completar el modelo de pipe en OPL con la definición de las variables de decisión y los datos de entrada necesarios:

```
 int Periodo = 3;
 range T = 1.. Periodo ;
 range T0 = 0.. Periodo ;

 float stockInitP = 0;
 float demandaP[T] = [10,20,15];
 float stockMaxP = 10;
 float stockMinP = 0;

 dvar float FlujoPE[T];
 dvar float FlujoPS[T];
 dvar float StockP[T0];

 maximize
 sum(t in T0) StockP[t];

subject to{
 StockP[0] == stockInitP;

 forall(t in T)
  {
   FlujoPS[t]== FlujoPE[t] + StockP[t-1] - StockP[t] -
demandaP[t];
   StockP[t] <= stockMaxP;
   StockP[t] >= stockMinP;
  }

  FlujoPE[1]==20;
  FlujoPE[2]==30;
  FlujoPE[3]==40;
}
```

Definimos el período de optimización T y T0 esta última para las variables iniciales que toman valor en el tiempo 0

Definimos unos datos para la tubería

Definimos las variables de decisión

En las pruebas vamos a suponer que utilizamos el criterio de stock mínimo

Fijamos externamente en la prueba el flujo de entrada en los tiempos 1, 2 y 3 de T

RESULTADO

```
StockP =  0  0  0  0 ;
FlujoPS =   10 10 25 ;
FlujoPE =   20 30 40 ;
```

Si cambiamos el criterio de óptimo para que el stock sea máximo:

```
maximize sum(t in T0) StockPt;
```

RESULTADO

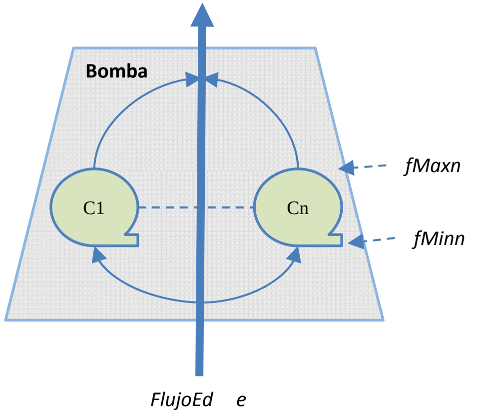
```
StockP =  0  10 10 10 ;
FlujoPS =    0 10 25 ;
FlujoPE =   20 30 40 ;
```

Modelo de bomba impulsora 1

Las bombas impulsoras se encargan de transmitir energía al fluido para vencer la resistencia de las tuberías. El modelo que vamos a plantear para una bomba también contempla la posibilidad de flujo bidireccional (como en las tuberías).

Una bomba podrá operar en varios puntos de funcionamiento activando un número determinado de turbos.

Cada punto de funcionamiento tiene asociado un coste (consumo de combustible).



Variables globales:

$FlujoB_t$: flujo que circula por la bomba

Variables locales:

$FlujoBP_t$: flujo en dirección positiva

$FlujoBN_t$: flujo en dirección negativa

$\forall k \in puntos\ de\ funcionamiento$

dp_{tk} : variables binarias asociadas al flujo positivo

dn_{tk} : variables binarias asociadas al flujo negativo

Datos:

$fMax_k$: flujo máximo asociado al punto de funcionamiento k

$fMin_k$: flujo mínimo asociado al punto de funcionamiento k

$consumo_k$: coste asociado al consumo de cada punto de operación

Modelo relacional:

$\forall t \in T$

$R_{bomba}(FlujoB_t, datosB_t, estadoB_{t-1})$

$datosB_t \equiv (\forall i \in bombas: fMax_k, fMin_k, consumo_k)$

$estadoP_{t-1} \equiv \Phi$

Modelo de bomba impulsora 2

La expresión de la relación $R_{bomba}(FlujoB_t, datosB_t, estadoB_{t-1})$ será la siguiente:

$$\forall t \in T$$

$$\sum_{k \in turbos} dp_{tk} * fMin_k \leq FlujoBP_t \leq \sum_{k \in turbos} dp_{tk} * fMax_k)$$

Limita el flujo de sentido positivo entre los valores máximo y mínimo que imponen el punto de funcionamiento

$$\sum_{k \in turbos} dn_{tk} * (-fMin_k) \geq FlujoBN_t \geq \sum_{k \in turbos} dn_{tk} * (-fMax_k)$$

Limita el flujo de sentido negativo entre los valores máximo y mínimo que imponen el punto de funcionamiento

$$\sum_{k \in turbos} dp_{tk} + \sum_{k \in turbos} dn_{tk} \leq 1$$

Obliga a que exista un único punto de funcionamiento en un único sentido

$$FlujoB_t = FlujoBP_t + FlujoBN_t$$

El flujo que circula por la bomba es la suma del positivo y el negativo (pero uno de ellos tiene que ser necesariamente cero por la restricción anterior)

$$Minimize \sum_{t \in T, k \in turbos} (dp_{tk} + dn_{tk}) * consumo_k$$

Se utilizará el punto de funcionamiento que minimice el consumo (coste)

Y su codificación en OPL:

OPL

```
forall(t in T)
 {
 sum(k in turbos)dp[t,k]*fMin[k] <= FlujoBP[t];
 sum(k in turbos)dp[t,k]*fMax[k] >= FlujoBP[t];
 sum(k in turbos)dn[t,k]*(-fMin[k]) >= FlujoBN[t];
 sum(k in turbos)dn[t,k]*(-fMax[k]) <= FlujoBN[t];
 sum(k in turbos)(dp[t,k]+dn[t,k])<=1;
 FlujoB[t] == FlujoBP[t]+FlujoBN[t];
 }
```

```
minimize sum(t in T, tu in turbos)
(dp[t,tu]+dn[t,tu])*Consumo[tu];
```

Modelo de bomba impulsora 3

Para hacer pruebas vamos a completar el modelo OPL con la definición de las variables de decisión y los datos de entrada necesarios, igual que hicimos para las tuberías:

```
int Periodo = 3;
range T = 1.. Periodo ;

range turbos = 1..3;
float fMin[turbos] = [0, 0,0];
float fMax[turbos] = [15, 25,35];
float Consumo[turbos] = [10, 20,30];

dvar int dp[T,turbos] in 0..1;
dvar int dn[T,turbos] in 0..1;

dvar float FlujoBP[T];
dvar float FlujoBN[T];
dvar float FlujoB[T];

minimize sum(t in T, tu in
turbos)(dp[t,tu]+dn[t,tu])*Consumo[tu];

subject to{
forall(t in T)
 {
 sum(k in turbos)dp[t,k]*fMin[k] <= FlujoBP[t];
 sum(k in turbos)dp[t,k]*fMax[k] >= FlujoBP[t];

 sum(k in turbos)dn[t,k]*(-fMin[k]) >= FlujoBN[t];
 sum(k in turbos)dn[t,k]*(-fMax[k]) <= FlujoBN[t];

 sum(k in turbos)(dp[t,k]+dn[t,k])<=1;

 FlujoB[t] == FlujoBP[t]+FlujoBN[t];
 }
  FlujoB[1]==11;
  FlujoB[2]==20;
  FlujoB[3]==30;
 }
```

Definimos el período de optimización T . T0 no hace falta ya que la bomba no tiene estado

Definimos unos datos para la bomba

Definimos las variables de decisión

Utilizamos el criterio de coste mínimomínimo

Fijamos externamente en la prueba el flujo de entrada en los tiempos 1, 2 y 3 de T

```
// solution (optimal) with objective 60
dp = [[1 0 0]
      [0 1 0]
      [0 0 1]];
dn = [[0 0 0]
      [0 0 0]
      [0 0 0]];
FlujoBP = [11 20 30];
FlujoBN = [ 0  0  0];
FlujoB =  [11 20 30];
```

RESULTADO

```
Si cambiamos el valor
del flujo:

 FlujoB1== 11;
 FlujoB2==-20;
 FlujoB3==-30;
```

```
// solution (optimal) with objective 60
dp = [[1 0 0]
      [0 0 0]
      [0 0 0]];
dn = [[0 0 0]
      [0 1 0]
      [0 0 1]];
FlujoBP = [11   0   0];
FlujoBN = [ 0 -20 -30];
FlujoB =  [11 -20 -30];
```

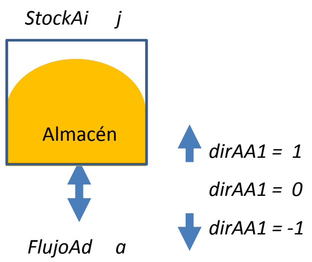
RESULTADO

Modelo de almacenamiento

Los almacenamientos regulan el funcionamiento de la red, permitiendo durante unos períodos acumular fluido en su interior, y extrayéndolo en otros.

El modo de funcionamiento en situaciones reales puede ser complejo, modulando la política de almacenamiento y extracción en función de necesidades actuales o futuras (previsión) de la demanda y los costes de la materia prima.

En nuestro caso el funcionamiento va a venir regulado por un vector $dirAT$ cuyas componentes indicarán con los valores 1, -1 y 0 si hay inyección, extracción o ninguna de las dos, respectivamente. La cantidad a inyectar o extraer la supondremos constante en todos los períodos del intervalo de planificación. La expresión relacional del modelo será la siguiente:



$$StockA[0] = stockInitA$$
$$\forall t \in T$$
$$StockA[t] = StockA[t-1] + FlujoA[t] * dirA[t]$$

OPL

```
int cuota = 20;
int dirAT=1     1     1;
float stockInitAalmacenes= 0;

StockA0 == stockInitA;
forall(t in T)
{
 StockAt==StockAt-1 + FlujoAt;
 FlujoAt== cuota*dirAT;
}
```

Variables globales:

$FlujoA_t$: flujo que entra/sale del almacén

$StockA_t$: cantidad acumulada en el período t

Variables locales:

ϕ

Datos:

$dirA_t$: indica con 1, -1 y 0 el sentido del amacenamiento en t

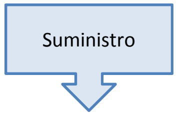
$cuota$: cantidad almacenada o extraida

Modelo relacional:

$$\forall t \in T$$
$$R_{almacén}(FlujoA_t, dirA_t, datosA, estadoA_{t-1})$$
$$datosA_t \equiv cuota$$
$$estadoA_{t-1} \equiv StockA_{t-1}$$

Modelo de suministro

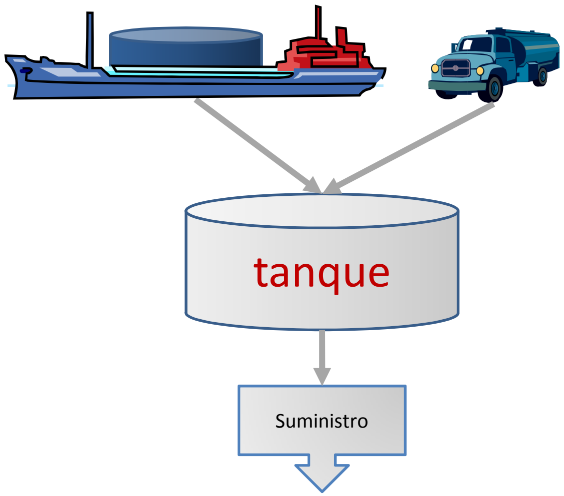
El modelo de la planta que suministra la materia prima lo vamos a simplificar al máximo en este ejemplo. Lo representaremos por una variable de decisión que puede adoptar un valor comprendido entre 0 y 600 en cualquier período del intervalo de planificación. Al no existir más que una planta de suministro tampoco asignaremos coste a la materia prima:



suministroPt

```
forall(t in T)0 <= suministroPt <=600;
```

Dependiendo del tipo de fluido considerado podemos extender el modelo de suministro incorporando un tanque de almacenamiento alimentado por vehículos marinos o terrestres. Se pueden establecer ventanas de descarga para estos vehículos y diferentes alternativas para las cantidades que transportan y el coste de su materia prima.



Ejemplo de red de transporte de fluidos

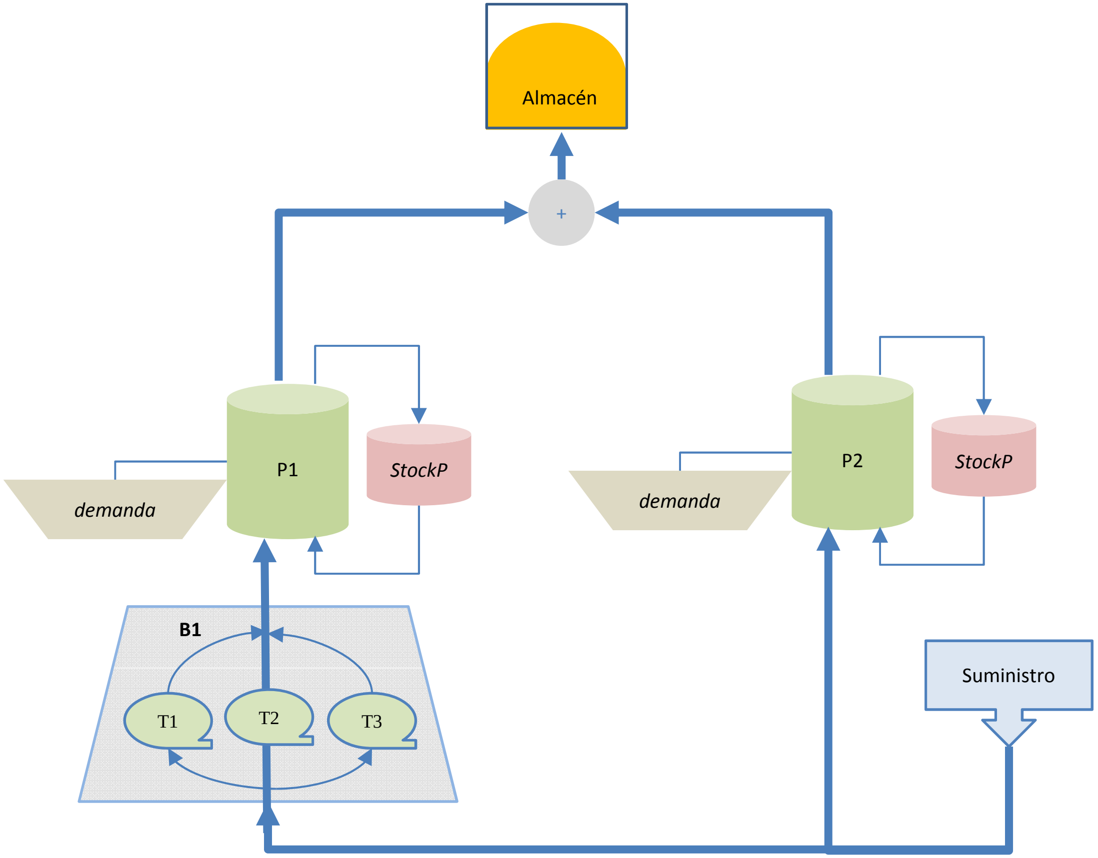
En este ejemplo vamos a conectar diferentes componentes para formar una red con la topología que aparece en la siguiente figura.

El criterio de óptimo que vamos a utilizar para planificar el funcionamiento de la red durante el intervalo de operación será el de satisfacer las dos demandas locales siguiendo los recorridos que minimicen el coste.

Al existir un único suministro, el criterio quedará reducido a satisfacer la demanda minimizando el uso de bombas de impulsión.

DATOS DE LA RED

```
T = 3;

BOMBA BIDIRECCIONAL
Número de puntos de funcionamiento = 3
Punto funcionamiento = 1      2      3
Flujo mínimo     =      10.1   20.2   30.3
Flujo máximo     =      15.1   25.2   35.3
Consumo =               15.1   25.2   35.3

ALMACENAMIENTO BIDIRECCIONAL
t  =           1      2      3
dirA =         1      1      1
Stock inicial = 0

PIPE BIDIRECCIONAL
Número de tuberías = 2
Tubería          1    2
Stock inicial =  0    0
Stock mínimo =   0    0
Stock máximo =  10   10

Tubería      1  2     1  2     1  2
Demanda =   40 30    30 15    10 20
t  =         1        2        3

PLANTA DE SUMINISTRO
Suministro mínimo = 0
Suministro máximo = 600
```



Ejemplo de red de transporte de fluidos: modelo OPL 1

Par poder utilizar el mismo modelo OPL de un componente de la red en más de una instancia (como es el caso de los dos pipes que aparecen en este ejemplo), introduciremos un nuevo índice asociado a cada instancia cuyo valor será el nombre o número asociado al la correspondiente instancia. Para ello tendremos que modificar ligeramente los modelos OPL analizados anteriormente:

Declaración y asignación de datos

```
int Periodo = 3;
range T = 1.. Periodo ;
range T0 = 0.. Periodo ;

// MODELO DE BOMBA BIDIRECCIONAL: DATOS
int numTurbos = 3;
 {string} bombas = {"B1"};
 range turbos = 1..numTurbos;
 float fMin[turbos] = [10.1, 20.2,30.3];
 float fMax[turbos] = [15.1, 25.2,35.3];
 float consumo[turbos] = [15.1, 25.2,35.3];

// MODELO DE ALMACENAMIENTO BIDIRECCIONAL: DATOS
{string} almacenes = {"A1"};
 int cuota = 20;
 int dirA[T]=[1,1,1];
 float stockInitA[almacenes]= [0];

// MODELO DE PIPE BIDIRECCIONAL: DATOS
 {string} pipes = {"P1","P2"};
 float stockInitP[pipes]= [0,0];
 float stockMinP[pipes]= [0,0];
 float stockMaxP[pipes]= [10,10];
 float demandaP[T,pipes]= [[40,30],[30,15],[10,20]];

// MODELO SIMPLIFICADO DE PLANTA DE SUMINISTRO: DATOS
 {string} plantas = {"PL1"};
 forall(t in T, p in plantas)0 <= suministroP[t] <=600;
```

Declaración de las variables de decisión

```
// MODELO DE BOMBA BIDIRECCIONAL: VARIABLES DE DECISION
 dvar int dp[T,bombas,turbos] in 0..1;
 dvar int dn[T,bombas,turbos] in 0..1;

 dvar float FlujoBP[T,bombas];
 dvar float FlujoBN[T,bombas];
 dvar float+ FlujoB[T,bombas];


// MODELO DE ALMACENAMIENTO BIDIRECCIONAL: VARIABLES DE DECISION
 dvar float FlujoA[T,almacenes];
 dvar float+ StockA[T0,almacenes];

// MODELO DE PIPE BIDIRECCIONAL: VARIABLES DE DECISION
 dvar float FlujoPE[T,pipes];
 dvar float FlujoPS[T,pipes];
 dvar float StockP[T0,pipes];

// MODELO DE LA PLANTA DE SUMINISTRO: VARIABLES DE DECISION
 dvar float+ suministroP[T,plantas]in 0..400;
```

Declaración de la función de costo

```
minimize

// MODELO DE BOMBA BIDIRECCIONAL: FUNCION DE COSTE
sum(i in T,j in bombas, k in turbos)(dp[i,j,k]+dn[i,j,k])*consumo[k];
```

Ejemplo de red de transporte de fluidos: modelo OPL 2

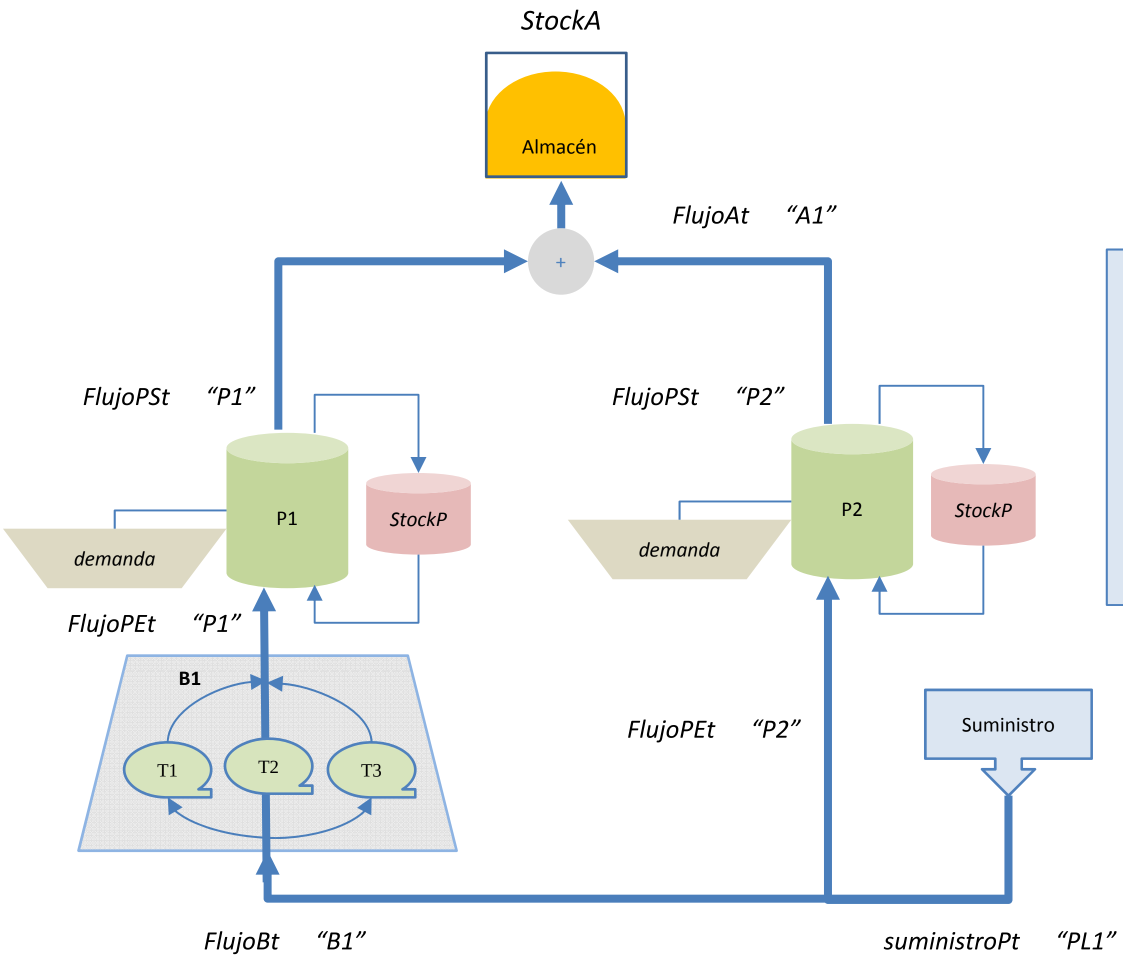
Expresión de las restricciones

```
subject to
{
// MODELO DE BOMBA BIDIRECCIONAL: RESTRICCIONES
forall(t in T, j in bombas)
 {
 sum(k in turbos)dp[t,j,k]*fMin[k] <= FlujoBP[t,j];
 sum(k in turbos)dp[t,j,k]*fMax[k] >= FlujoBP[t,j];
 sum(k in turbos)dn[t,j,k]*(-fMin[k]) >= FlujoBN[t,j];
 sum(k in turbos)dn[t,j,k]*(-fMax[k]) <= FlujoBN[t,j];
 sum(k in turbos)(dp[t,j,k]+dn[t,j,k])<=1;
 FlujoB[t,j] == FlujoBP[t,j]+FlujoBN[t,j];
 }
// MODELO DE ALMACENAMIENTO BIDIRECCIONAL: RESTRICCIONES
forall(j in almacenes)StockA[0,j] == stockInitA[j];
forall(t in T, j in almacenes)
 {
 StockA[t,j]==StockA[t-1,j] + FlujoA[t,j];
 FlujoA[t,j]== cuota*dirA[t];
 }
// MODELO DE PIPE BIDIRECCIONAL: RESTRICCIONES
forall(i in pipes)StockP[0,i] == stockInitP[i];

forall(t in T, j in pipes)
 {
 FlujoPS[t,j]== FlujoPE[t,j] + StockP[t-1,j] - StockP[t,j] - demandaP[t,j];
 StockP[t,j] <= stockMaxP[j];
 StockP[t,j] >= stockMinP[j];
 }
 forall(t in T)FlujoPE[t,"P2"] <=62;

// MODELO DE LA PLANTA DE SUMINISTRO: RESTRICCIONES
 forall(t in T, p in plantas)0 <= suministroP[t,p] <=600;

// MODELO DE LA RED: RESTRICCIONES
forall(t in T)
 {
  FlujoPE[t,"P1"]== FlujoB[t, "B1"];
  FlujoPS[t,"P1"]+FlujoPS[t,"P2"]== 20*dirA[t];
  FlujoB[t,"B1"]+FlujoPE[t,"P2"]== suministroP[t,"PL1"];
 }
}
```

Hemos limitado la capacidad del flujo que permite el pipe 2 a 62. En general se deberían haber contemplado estos límites en el modelo de componente y con datos parametrizables.

Estas restricciones establecen la suma algebraica cero de todos los flujos que concurren en los nodos de interconexión de la red , que para el ejemplo son tres.

Ejemplo de red de transporte de fluidos: resultados




```
FlujoB = [[35.3] [0] [0]];
StockA = [[0] [20][40][60]];
FlujoA = [[20][20][20]];
StockP = [[0 0] [7.3 0] [0 0] [0 0]];
FlujoPS = [[-12 32] [-22.7 42.7][-10 30]];
FlujoPE = [[35.3 62][0 57.7] [0 50]];
suministroP = [[97.3] [57.7][50]];
```

Proyecto

El proyecto consiste en

1. Especificar una red logística junto con los datos asociados.
2. Elaborar el modelo matemático de la red.
3. Implementar y depurar el modelo en OPL.
4. Obtener e interpretar los resultados

Sugerencias de proyecto:

1. Red logística de transporte guiada por la demanda introduciendo algunas de las ampliaciones citadas en la transparencia 5.
2. Red logística de transporte guiada por la oferta del tipo planteado en las transparencias 6, 7, 8, 9 y 10 que responda a una aplicación real.
3. Red de conducción de fluidos con topología análoga a la de la transparencia 11 e introduciendo un tanque en las plantas de suministros.
4. Cualquier otra red logística de complejidad análoga a las anteriores que el alumno tenga interés en desarrollar.