

A pair of purple, glowing chromosomes is shown against a black background. The chromosomes are arranged in a cross-like pattern, with two arms extending from a central point. The text "Algoritmos Genéticos" is overlaid on the chromosomes in a yellow, serif font, enclosed in a white rectangular box.

# Algoritmos Genéticos

# Introducción

- **Método de optimización Basado en la naturaleza, específicamente en la teoría genética de la selección natural. Inventado por Jhon Holland**
- **Metodología meta heurística de optimización, utilizada en la optimización de problemas no lineales combinatoriales**



## Aplicaciones de PM en Ing. Qca.



**LP**

$$\text{Min } f(x) = c^T \textcircled{x}$$

s.t.

$$A \cdot x \leq b$$

$$x \geq 0$$

**NLP**

$$\text{Min } f(x)$$

s.t.

$$h(z, w) = 0$$

$$g(z, w) \leq 0$$

$$z^L \leq z \leq z^U$$

**MILP**

$$\text{Min } f(x, y) = a^T \textcircled{x} + b^T \textcircled{y}$$

s.t.

$$A \cdot x + B \cdot y \leq b$$

$$x \geq 0$$

$$x^L \leq x \leq x^U$$

$$y \in \{0, 1\}^m$$

**MINLP**

$$\text{Min } \Phi(x, y) = c^T \cdot y + f(x)$$

s.t.

$$h(x) = 0$$

$$g(x) + B \cdot y \leq b$$

$$x^L \leq x \leq x^U$$

$$y \in \{0, 1\}^m$$

# Métodos de Optimización

## PROGRAMACION MATEMATICA

- Lineal entera
- Lineal Mixta
- Cuadrática
- No Lineal
- Binaria
- Mixta
- Mixta no lineal

## PROGRAMACION NO MATEMATICA

### HEURISTICOS

- Constructivo
- Descomposición
- Reducción
- Manipulación del modelo
- Búsqueda local

### METAHEURISTICOS

- Simulated Annealing
  - Tabu Search
  - GRASP
- Algoritmos Genéticos



# Aplicaciones de PM en Ing. Qca.



*Diseño de procesos*

*Síntesis de procesos*

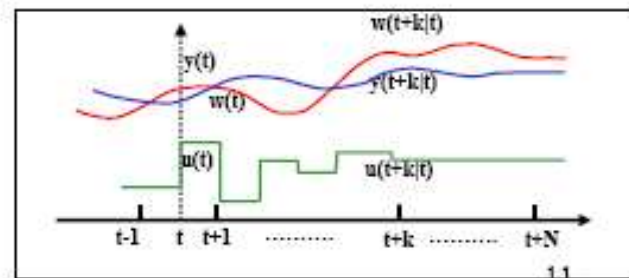
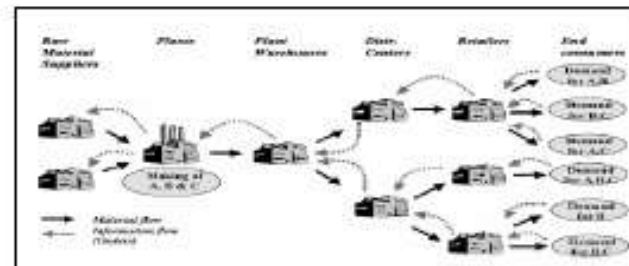
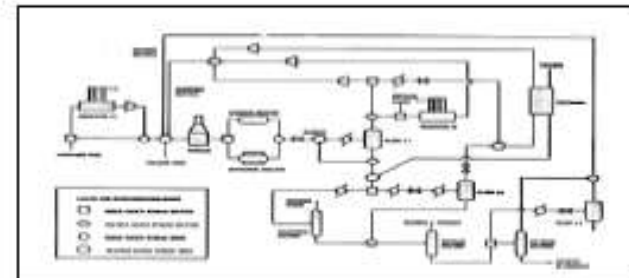
*Planeamiento de la producción*

*Scheduling operativo*

*Administr. cadenas de suministro*

*Control de procesos*

*Estimación de parámetros*





# **Métodos Meta Heurísticos**

**Métodos de optimización diseñados para resolver problemas combinatoriales, en los que los métodos heurísticos clásicos no son ni efectivos ni eficientes, utilizan distintos conceptos derivados de inteligencia artificial, mecanismos estadísticos y evolución biológica (Algoritmos Genéticos).**

# Características de la evolución

- Opera en los cromosomas en lugar de los individuos.
- La selección natural es el proceso donde los individuos mejor adaptados se reproducen mas frecuentemente.
- La evolución genética no tiene memoria, ya que se considera solo la información del periodo anterior.
- El proceso de reproducción se lleva acabo mediante la combinación de los cromosomas, donde pueden existir mutaciones que alteren al cromosoma.



# Componentes de un A.G.

- Representación cromosómica (codificación de la información como un string binario).
- Población inicial
- Medida de evaluación (fitness).
- Criterio de selección / eliminación de cromosomas.
- Operaciones de recombinación y mutación.
- Criterio de parada.

**PARTIDA** Genera población inicial con  $n$  cromosomas (soluciones factibles)

**[Fitness]** Evaluación del fitness  $f(x)$  de cada individuo  $x$  en la población

**[Nueva Población]** Crea nueva población en base a los siguientes pasos:

**[Selección]** Seleccione dos cromosomas de acuerdo a su fitness (el mejor fitness tiene mayor chance)

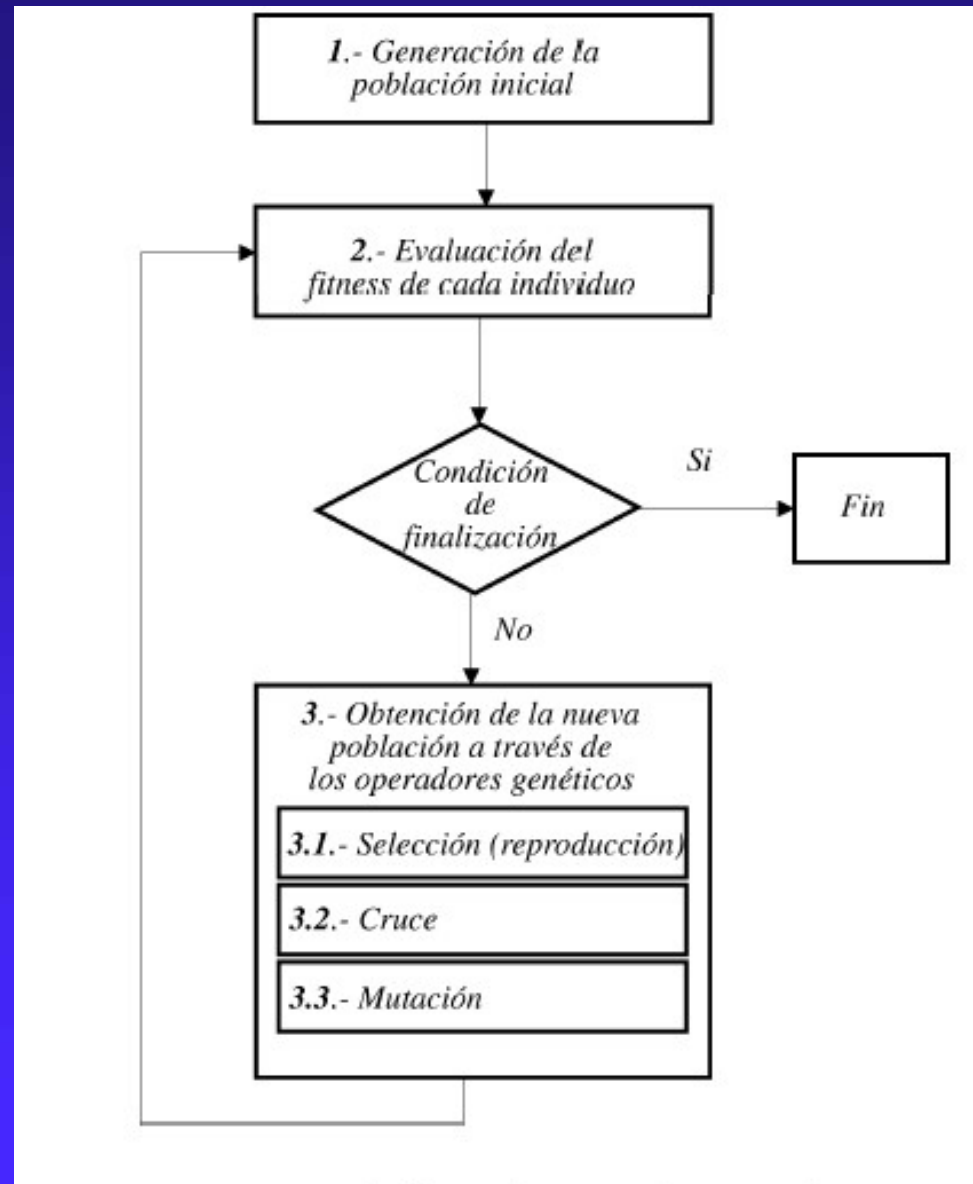
**[Cruzamiento]** Con la probabilidad de cruzamiento cruce a los parientes para formar uno nuevo (hijo). si no hay cruzamiento, se hacen copia de los parientes.

**[Mutación]** Con la probabilidad de mutación, mute a los hijos cambiando posiciones en los cromosomas. si no hay mutación, se hacen copia de los parientes.

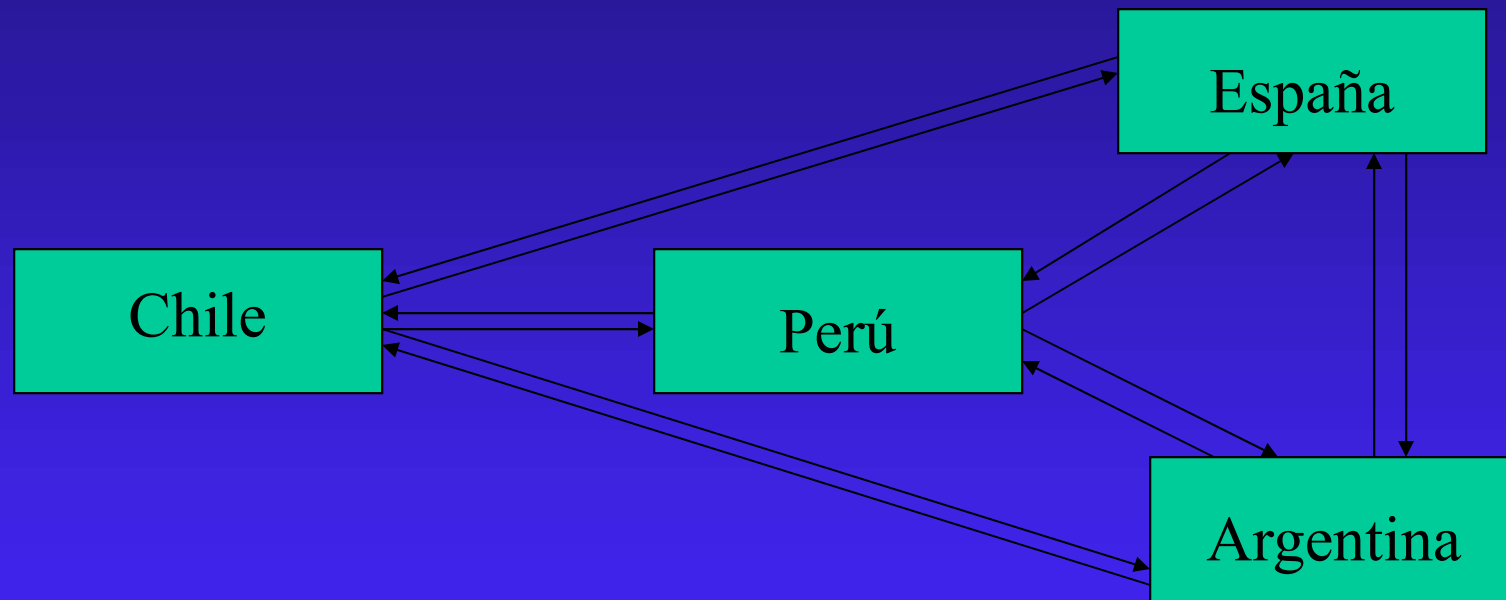
**[reemplazo]** Ponga a los hijos en la nueva población.

**[Nueva generación]** Reemplace a la nueva generación

**[Test]** Ver si se alcanza el numero maximo de generaciones. Si no volver a producir una nueva generación.

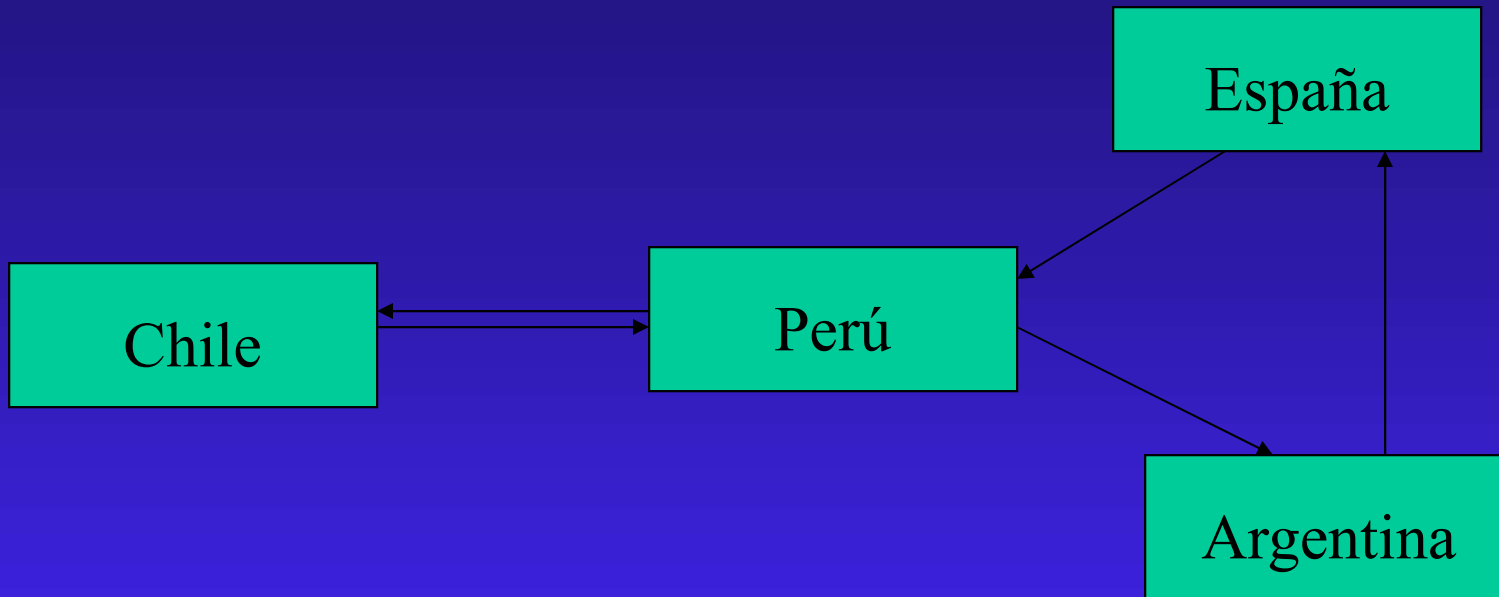


# Representación cromosómica



Problema de un turista , ¿cuál es la mejor ruta a seguir?

# Una posible elección



**Representación cromosómica binaria**

**[0 1 0 0 0 1 0 1 0 0 0 1 0 1 0]**

**Representación cromosómica numérica**

**[2 2 1 3 2]**

# Población inicial

- **Numero de individuos generado por el Algoritmo Genético para su evolución y reproducción.**

[1 1 0 2 3]

[2 0 1 4 4]

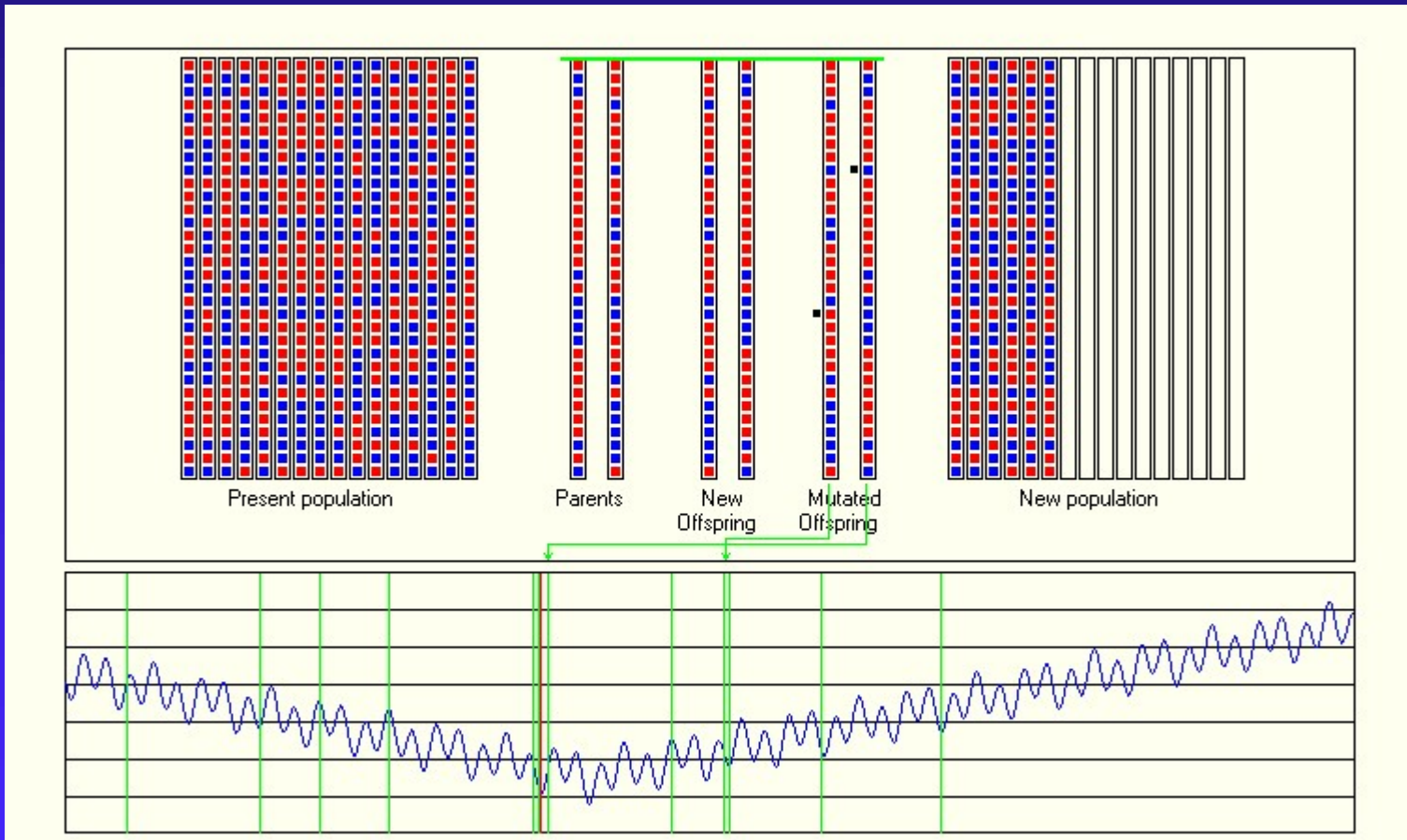
[1 2 1 4 1]

[2 0 2 1 4]

[3 3 4 3 2]

[2 4 0 1 1]

# Creación de la nueva generación



# Medida de evaluación

**Cada individuo al ser evaluado en la función objetivo de optimización genera una medida de su fitness que es la medida de la probabilidad de mantenerse en otras generaciones. La selección de los padres viene dada habitualmente por probabilidades.**

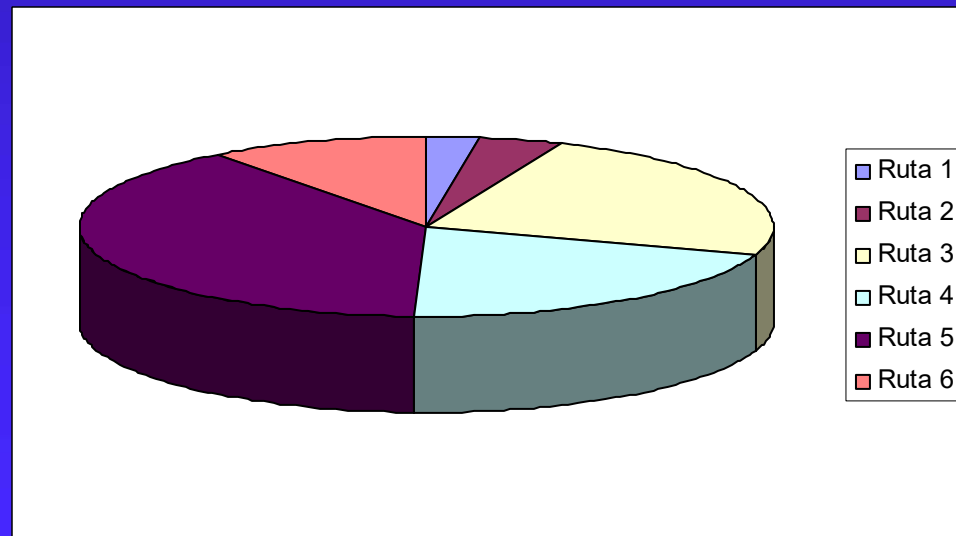


# Criterio de selección

Una vez obtenido el fitness de cada miembro de la población, se selecciona a los individuos que generaran a la próxima población, habitualmente utilizando el método de selección por Ruleta.

Selección por el procedimiento de ruleta

Fitness	Recorrido
0,1	Ruta 1
0,15	Ruta 2
0,9	Ruta 3
0,8	Ruta 4
1,5	Ruta 5
0,4	Ruta 6



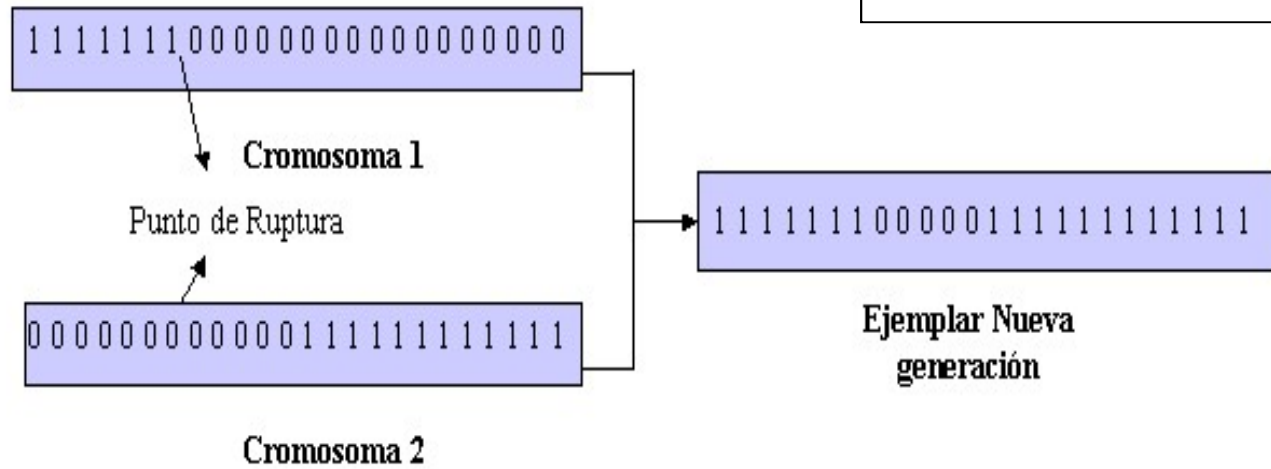
# Operación de recombinación y mutación

La recombinación se produce utilizando operadores de entrecruzamiento de los cuales los mas utilizados son

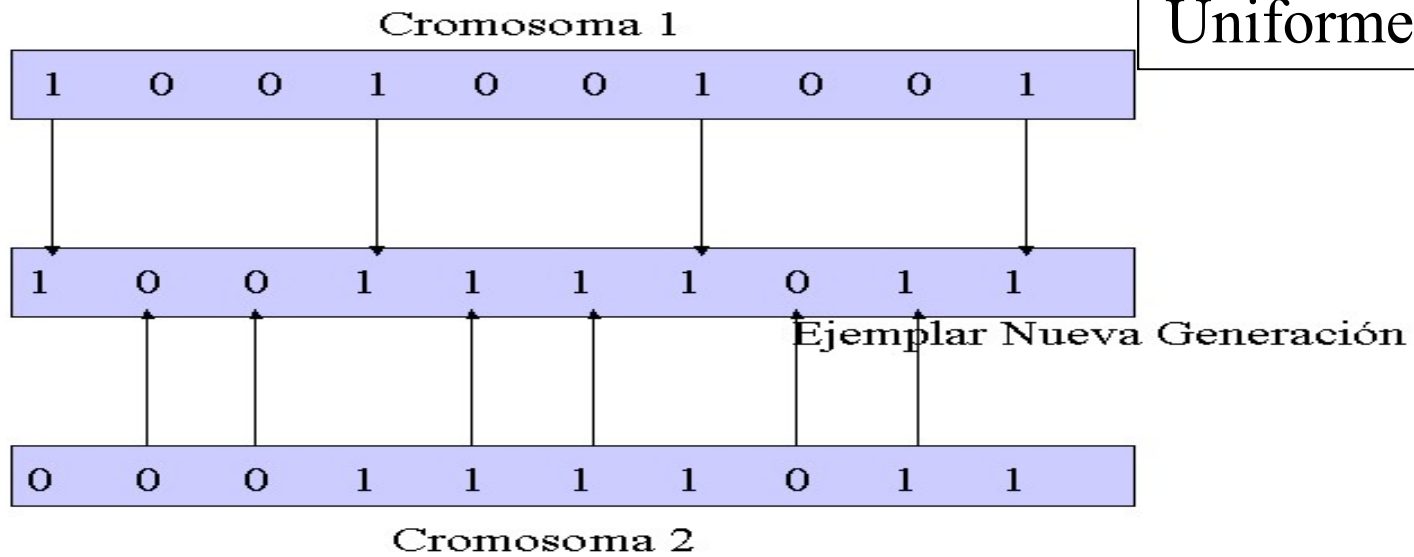
- De dos puntos
- De un punto
- Uniforme
- PMX, SEX (Combinación de los anteriores).

La mutación cambia uno de los valores de un bit de un cromosoma, introduciendo un factor de diversificación evitando la obtención de óptimos locales

## De Un Punto



## Uniforme



# Criterio de parada

Numero de generaciones recorridas

## Resultados obtenidos

- Múltiples soluciones una vez terminado el recorrido
- Se espera que en el infinito de generaciones la población final sea solo un individuo

# Programación en ambiente Matlab

```
bits=[3 4 4 4 4 4 4 ];  
vlb =[1 1 1 1 1 1 1 ];  
vub =[8 16 16 16 16 16 16 ];  
options(11:14)=[30 0.9 0.1 10];  
[y,bf]=genetic('Final4', [],options,vlb,vub,bits,Paramet  
ros);
```

# Programación en ambiente Matlab

**BITS:** Numero que puede tener cada string de un individuo, el numero indica la potencia de dos que puede tomar.

**VLB:** Valor mínimo que puede tomar un bit en un string

**VUB:** Valor máximo que puede tomar un bit en un string

# Parámetros del algoritmo

**OPTION:** Señala las opciones que uno ha considerado en el desarrollo del algoritmo genético, 11:14 indica las opciones que uno se está refiriendo.

- **Option 11** es el número de miembros en una población
- **Option 12** es la probabilidad de que ocurra crossover (reproducción)
- **Option 13** es la probabilidad de que ocurra mutación.
- **Option 14** es el criterio de parada, cuántas generaciones se recorrerán adicional a la primera.

# Programación en ambiente Matlab.

```
[y,bf]=genetic('Final4',[],options,vlb,vub,bits,Parametros);
```

**[y,bf]** indica lo que uno le pide al algoritmo genético y es el string opimo y bf el valor que genera en la función objetivo

```
genetic('Final4',[],options,vlb,vub,bits,Parametros);
```

es la llamada del A.G a la función final4 para que evalúe los individuos generados por el A.G y le la funcion entregue el fitness de cada individuo al A.G.



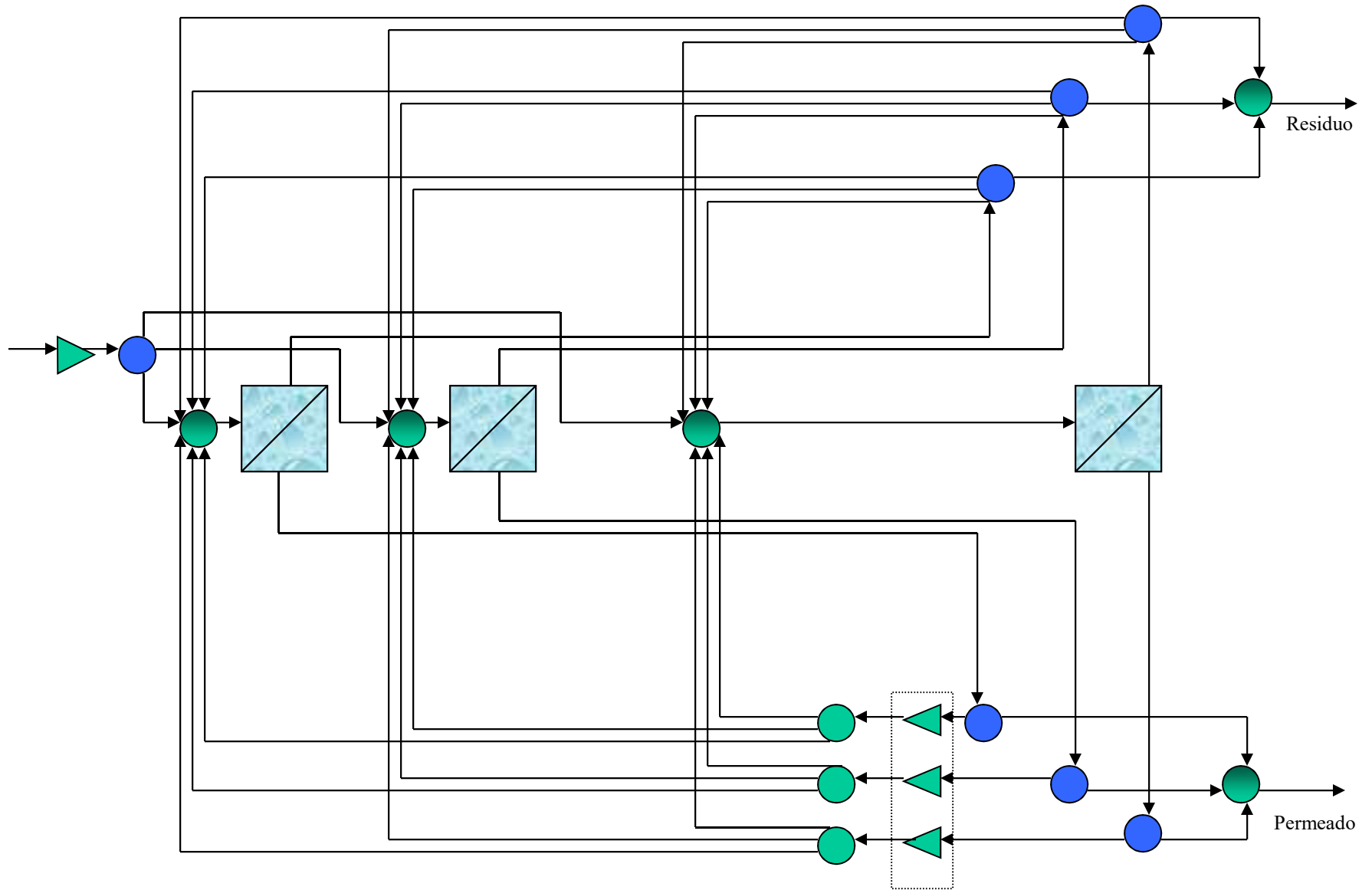
# Ejemplo

- **Optimización de la configuración de un sistema un sistema de osmosis inversa, dada una alimentación, y una función objetivo.**

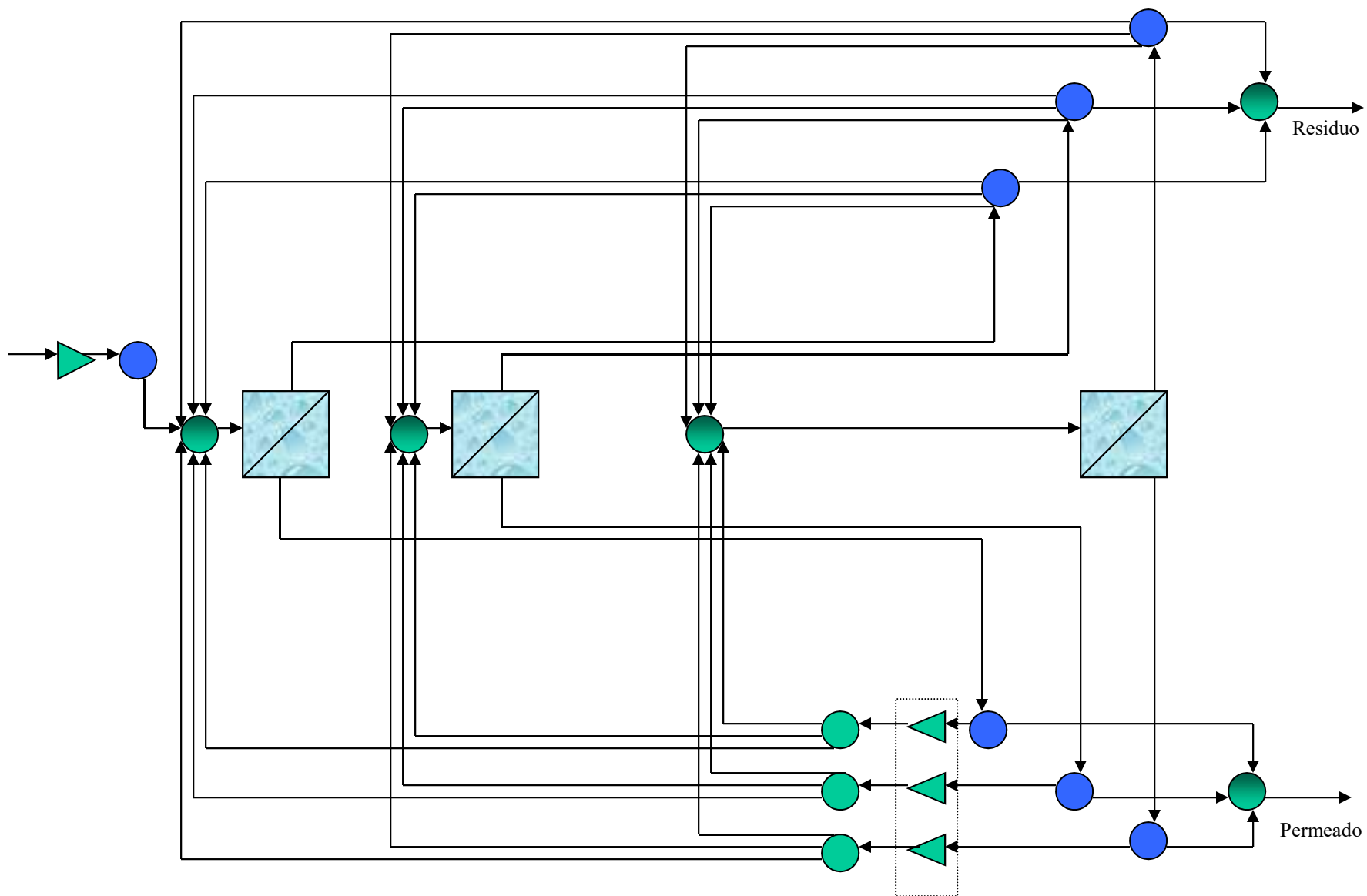
# La representación cromosómica

- Para la representación se considera la superestructura como generador de configuraciones, siendo las direcciones que toman las corrientes desde los spliter, el punto de partida para la representación cromosómica

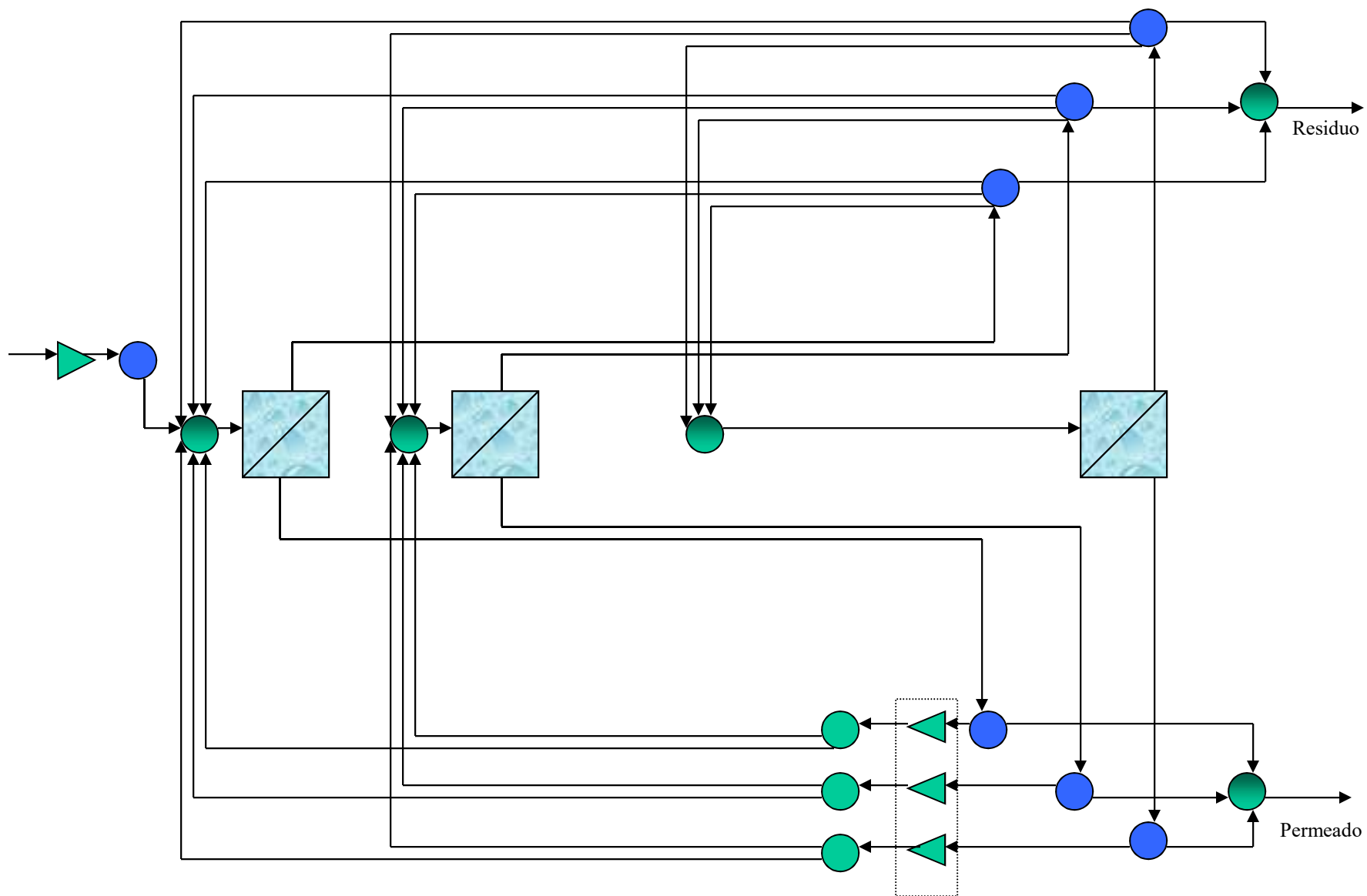
# *Super estructura*



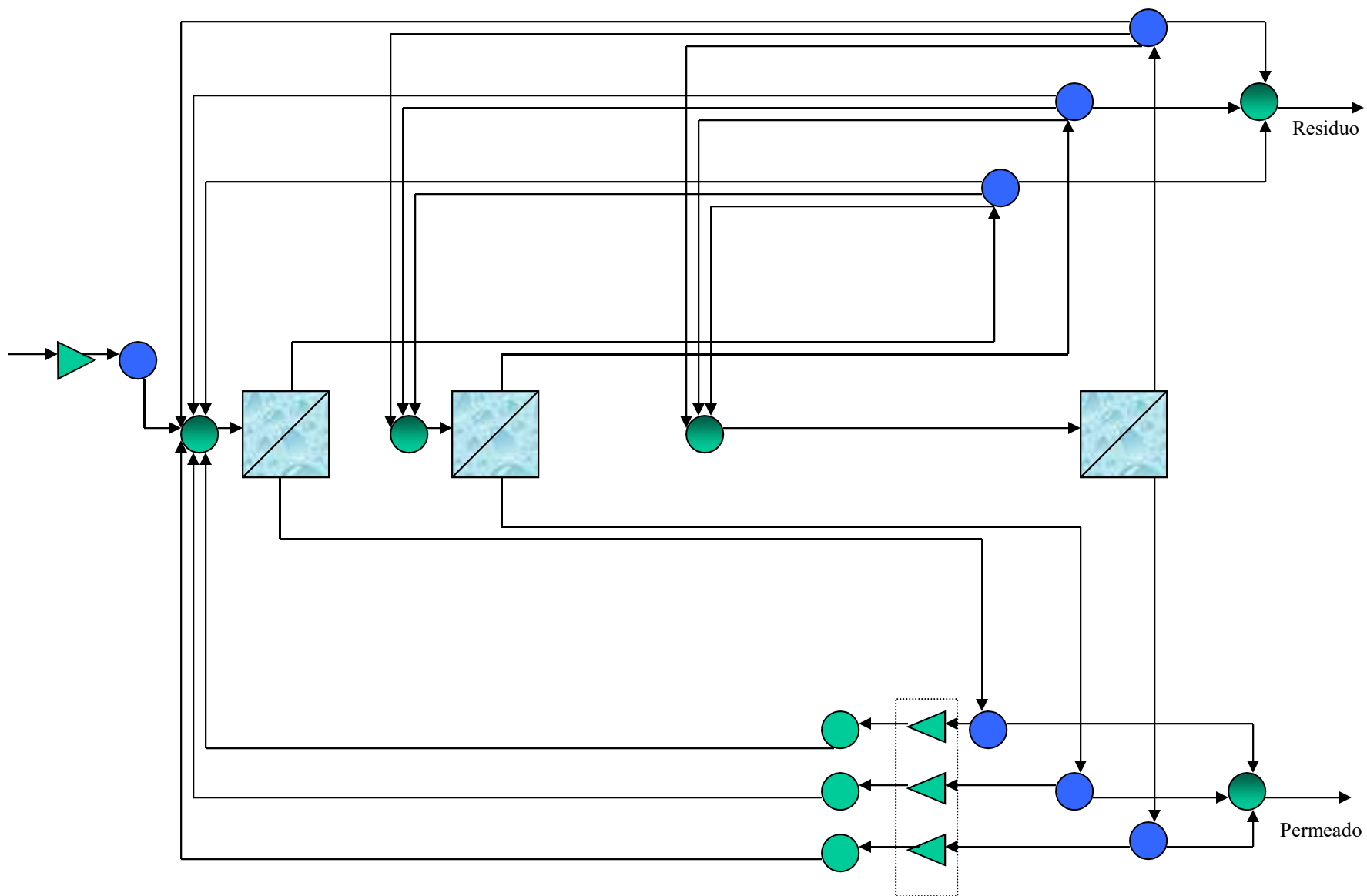
# Configuración en serie



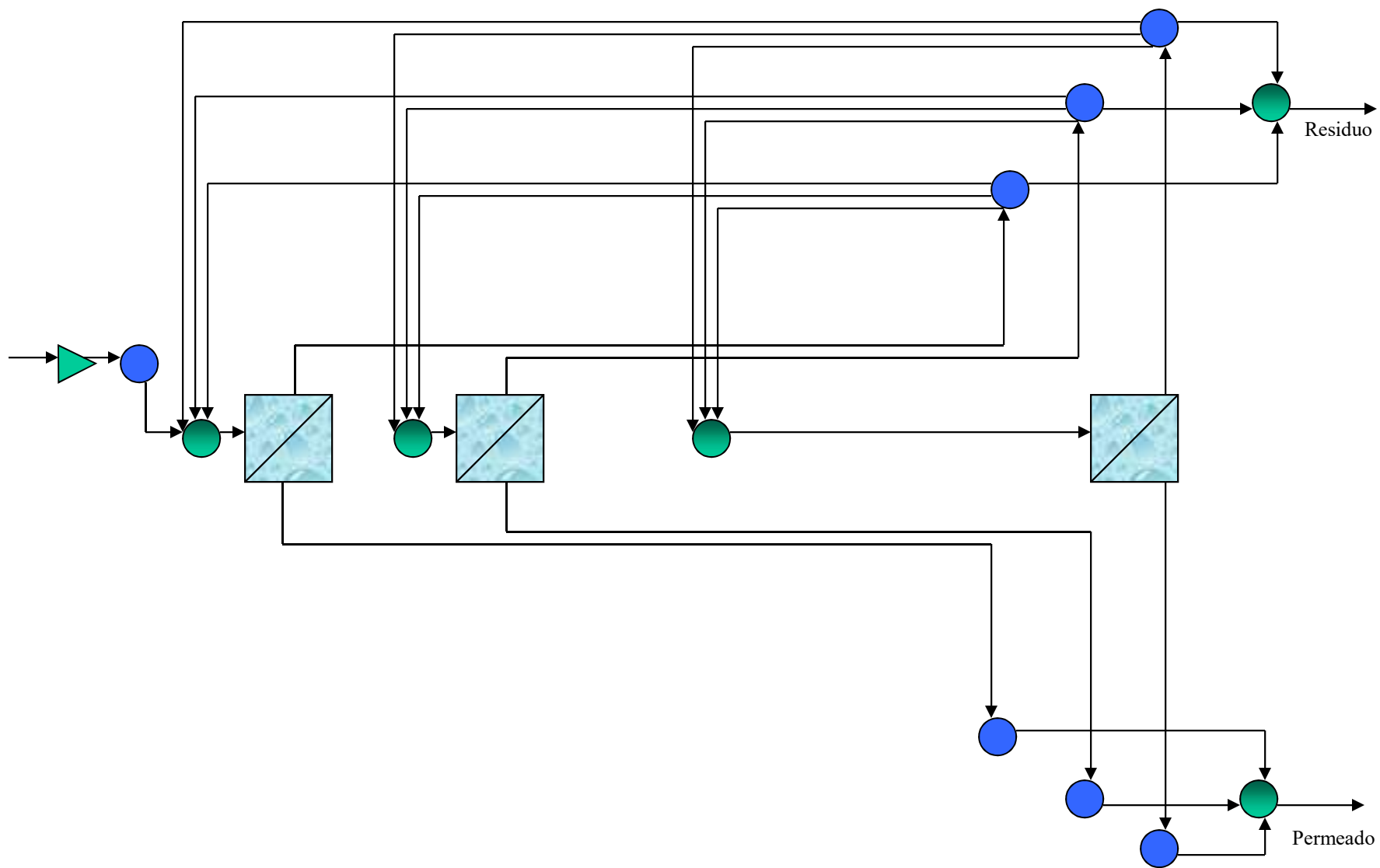
# Configuración en serie



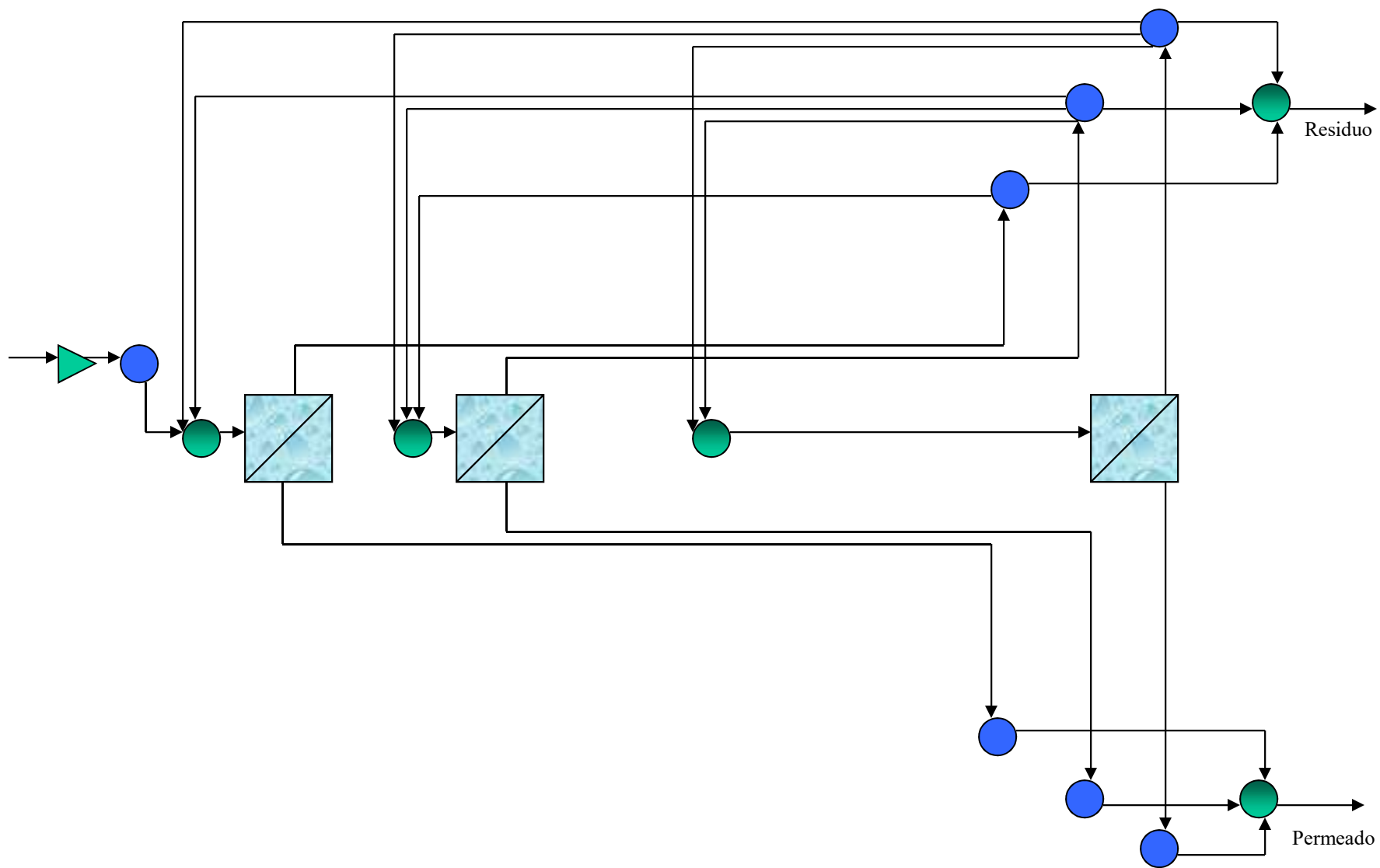
# Configuración en serie



# *Configuración en serie*

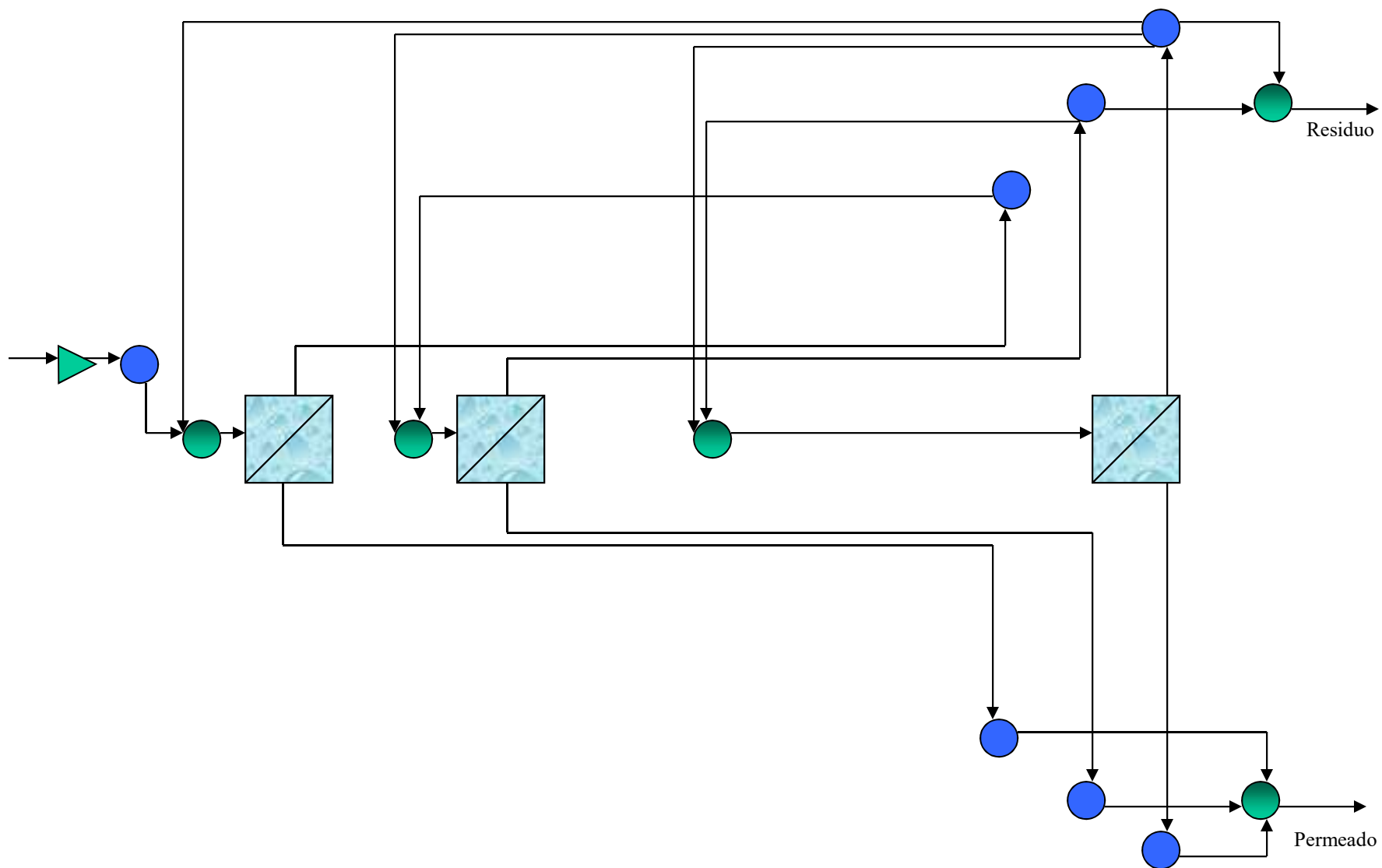


# *Configuración en serie*

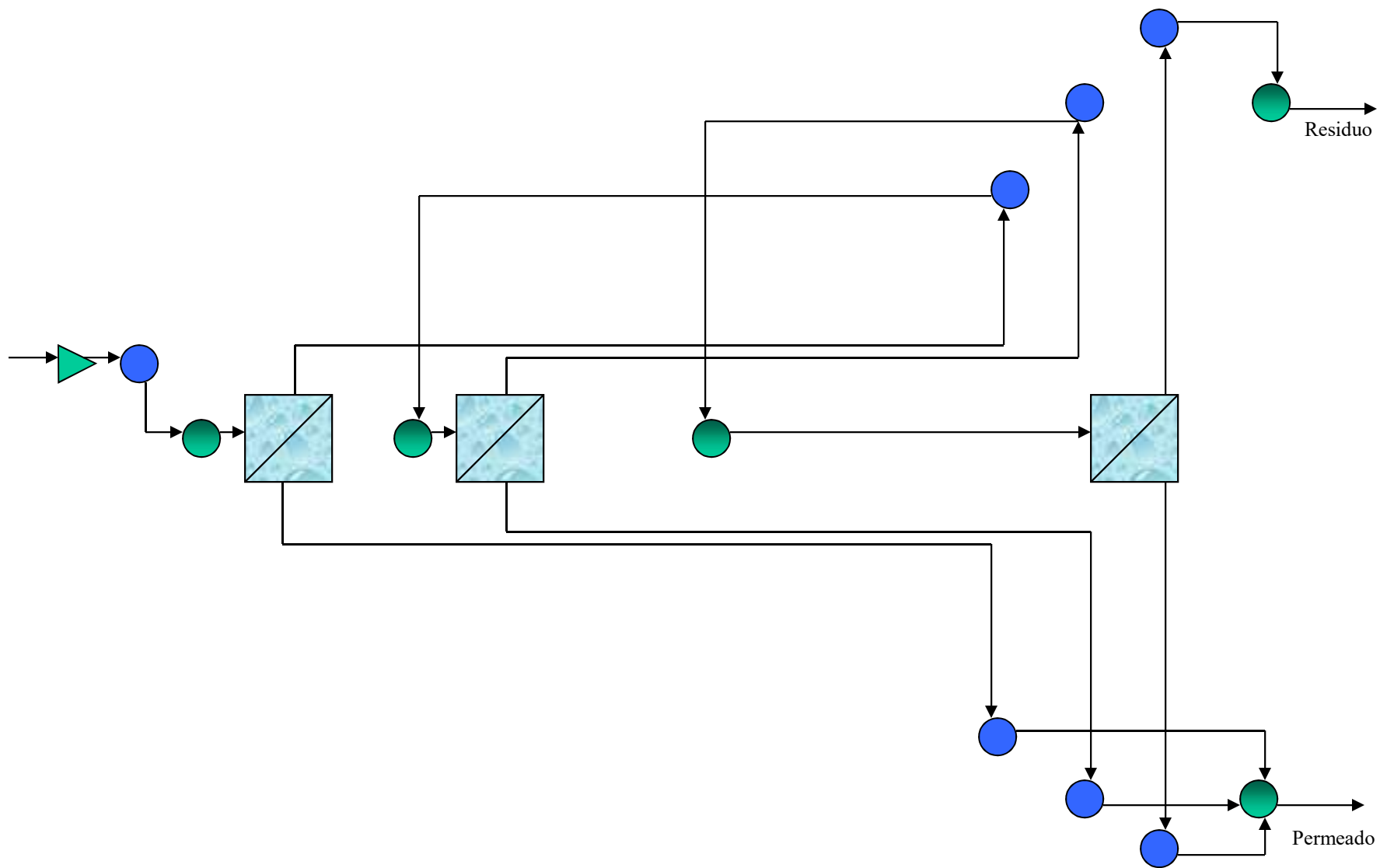




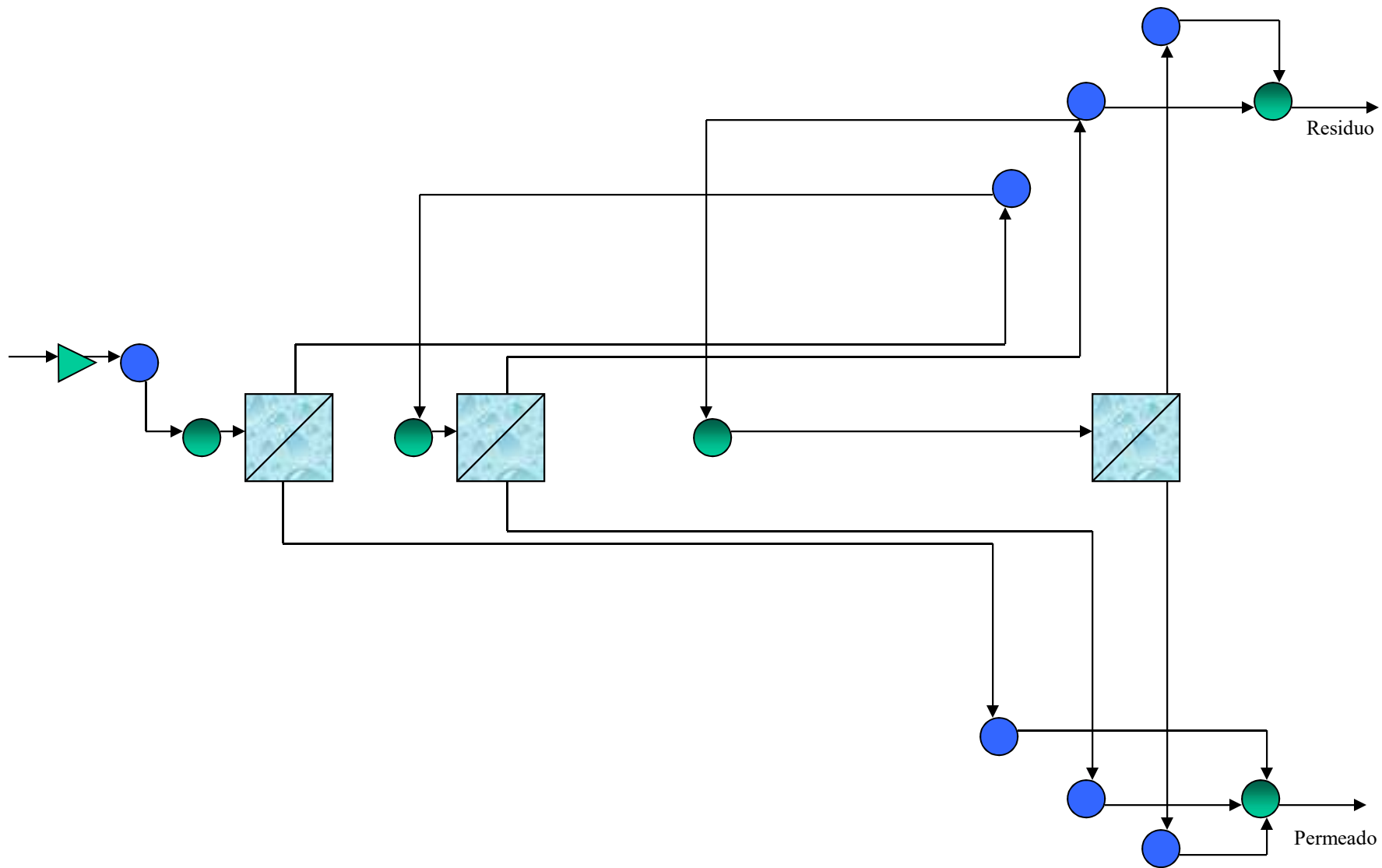
# *Configuración en serie*



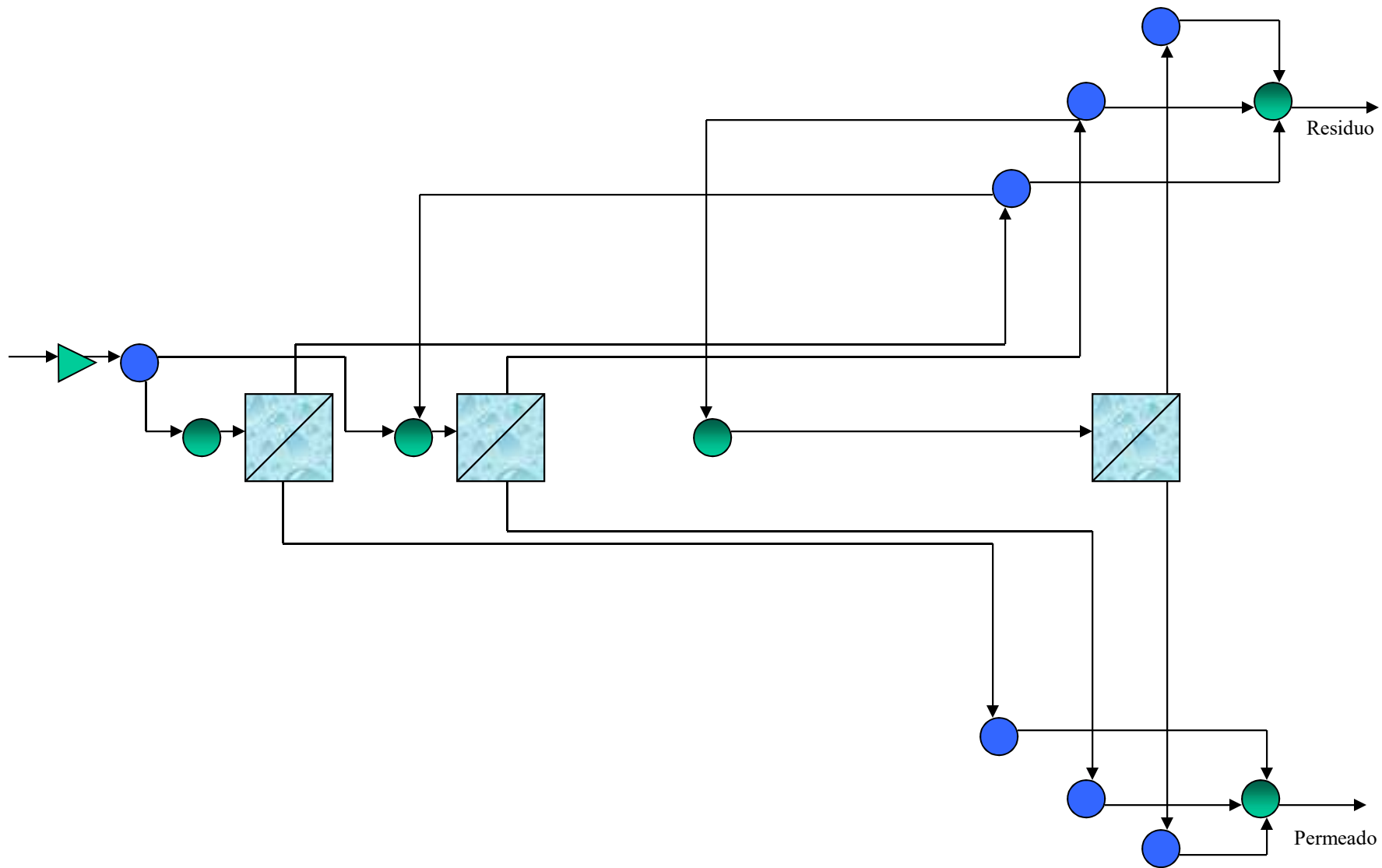
# *Configuración en serie*



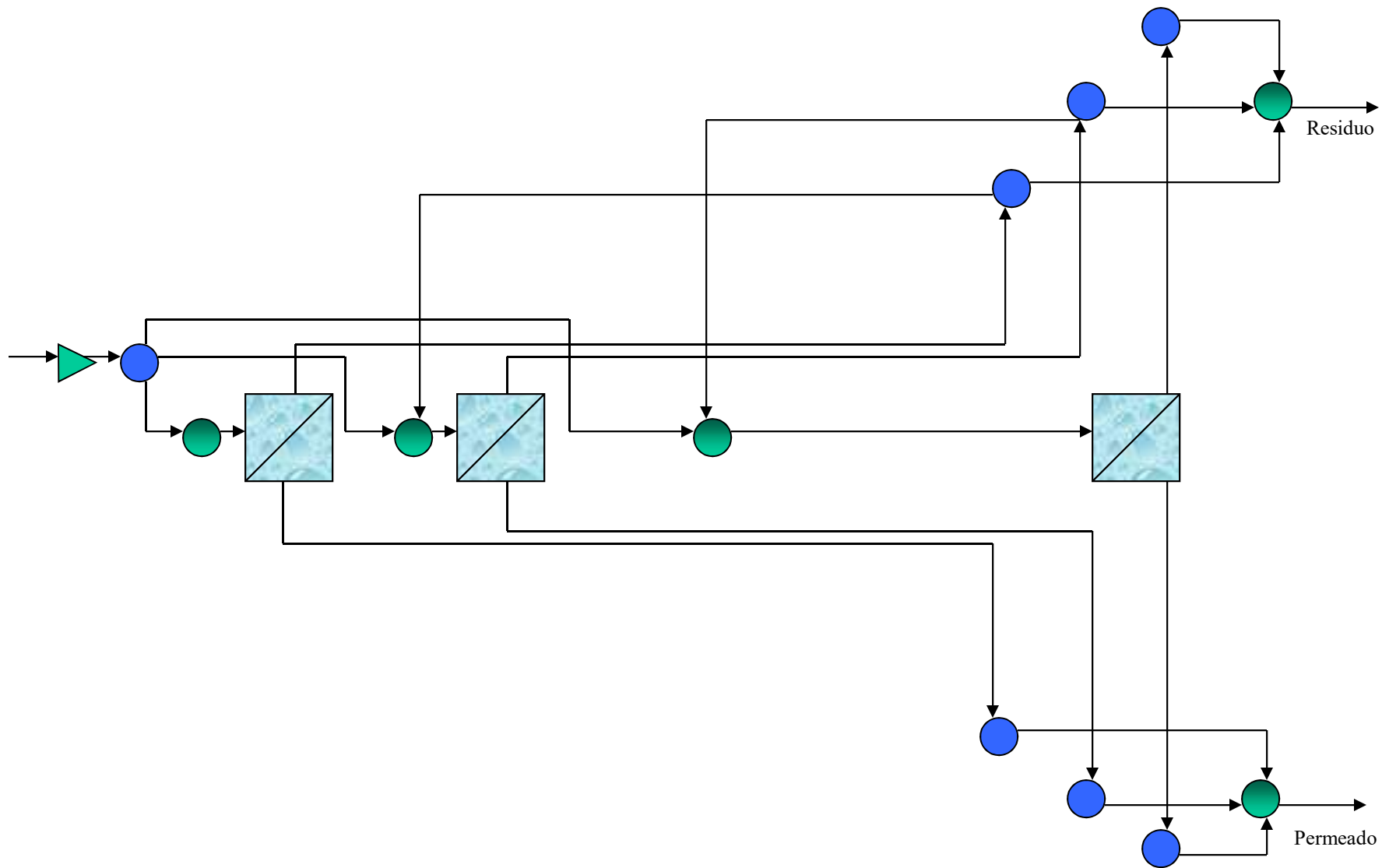
# *Configuración en paralelo*



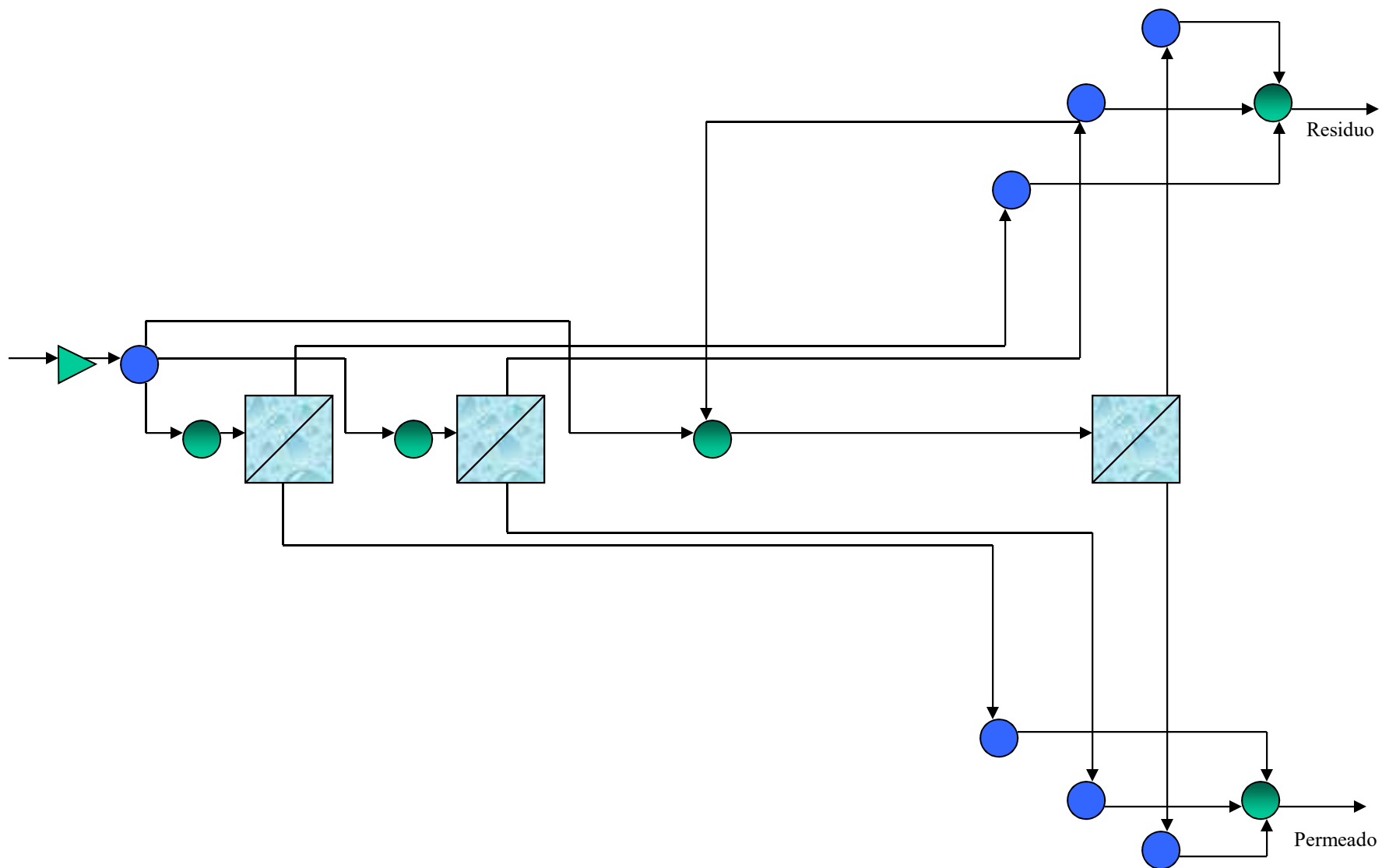
# *Configuración en paralelo*



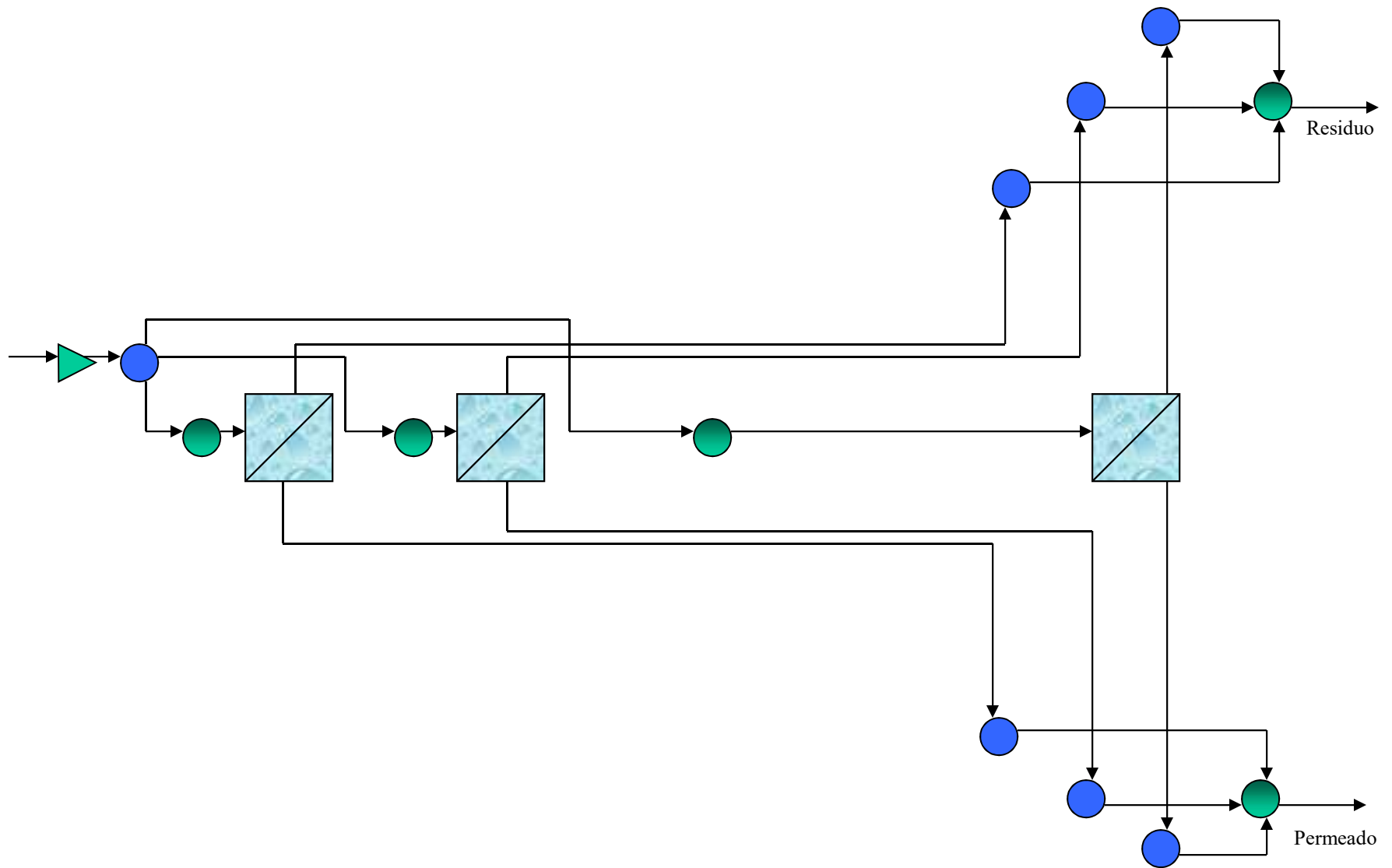
# *Configuración en paralelo*



# *Configuración en paralelo*

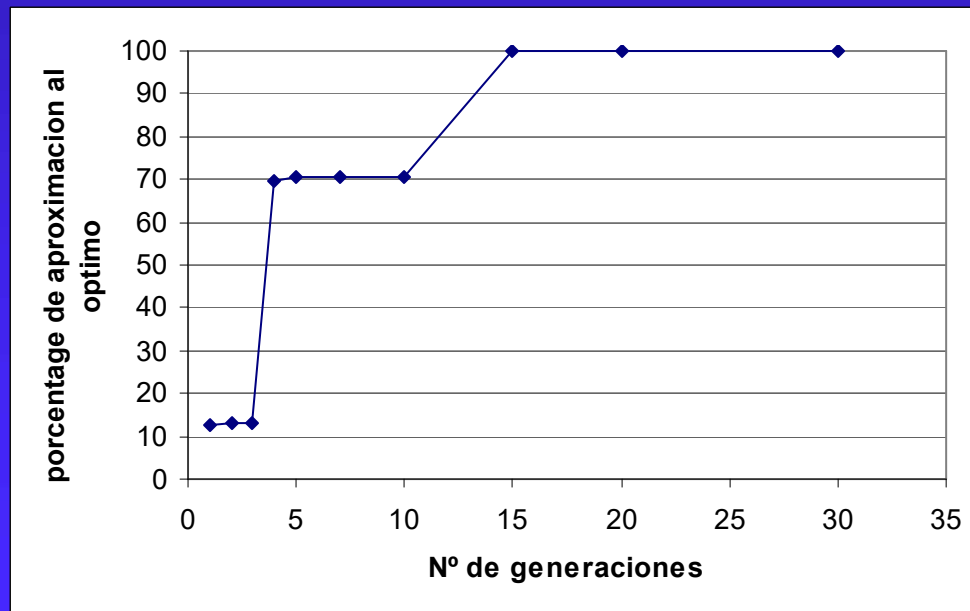


# *Configuración en paralelo*



# Evolución con respecto al optimo

Al evaluar el algoritmo genético con una alimentación dada, la configuración optima encontrada en función de el numero de generaciones puede ser representada por la siguiente figura







Introduction to genetic algorithms with Java applets.url

<http://cs.felk.cvut.cz/~xobitko/ga/>



Genetic and Evolutionary Algorithm Toolbox - GEATbx Main page.url

<http://www.geatbx.com/>