

APLICACIONES DE LA SIMULACIÓN DE EVENTOS DISCRETOS

RICARDO MARCELÍN JIMÉNEZ

* ***CONSTRUCCION***

* APLICACIONES

* TRABAJO FUTURO

INTRODUCCION

Se sabe de pocos estudios acerca del desempeño de los algoritmos distribuidos bajo diferentes ambientes de comunicaciones.

La simulación es una alternativa con la que puede conseguirse mayor precisión.

Usando nuestra herramienta, un desarrollador escribe su algoritmo en C++ y lo compila junto con las clases que componen la plataforma de simulación.

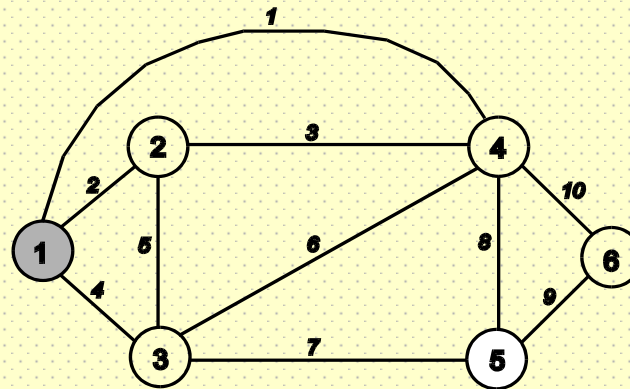
CARACTERÍSTICAS

- * Separa el algoritmo de su ambiente de comunicaciones
- * Permite simular eventos aleatorios
- * El usuario define los mensajes que se interambian
- * Soporta autómatas simples o compuestos
- * Ejecución simultanea de varios algoritmos

DECISIONES DE DISEÑO

1. utilizar un ambiente de simulación que ya existe (puede tomar mucho tiempo llegar a conocerlo suficientemente y, además, no ajustarse a nuestras necesidades)
2. diseñar un conjunto de macros adaptados a nuestro problema, para que trabajen encima de las mismas herramientas que en 1 (aún más largo)
3. construir la herramienta partiendo de cero, con nuestros requerimientos particulares en mente (se produce un código abierto, muy compacto, que puede enriquecerse con la aportación de la comunidad de interés)

EL MODELO SUBYACENTE

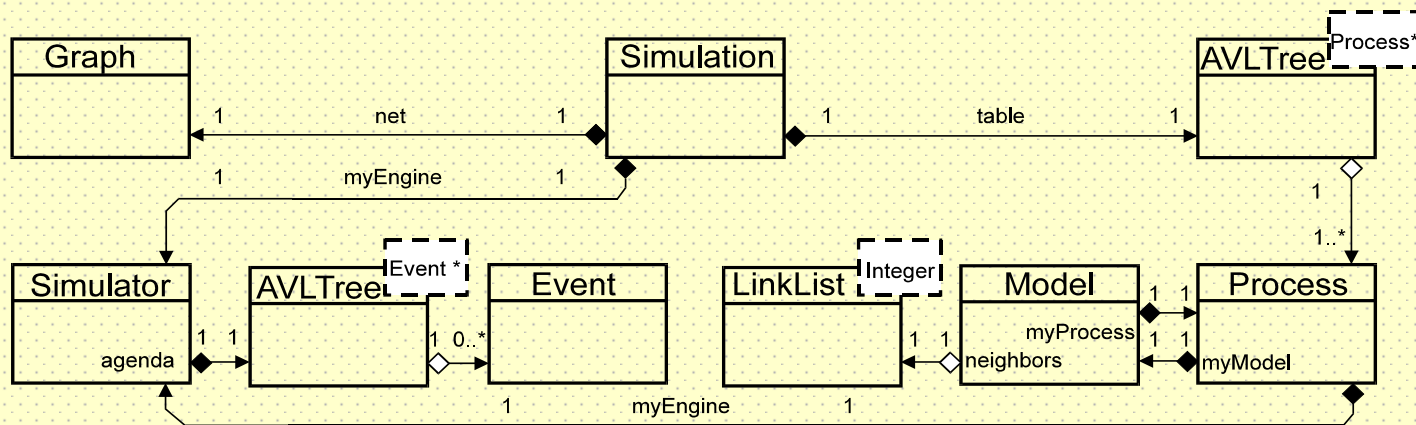


- Modelo de red asíncrona: una red de comunicaciones punto a punto, descrita por una "gráfica de comunicaciones"

$$G(V,E), n=|V|, m=|E|$$

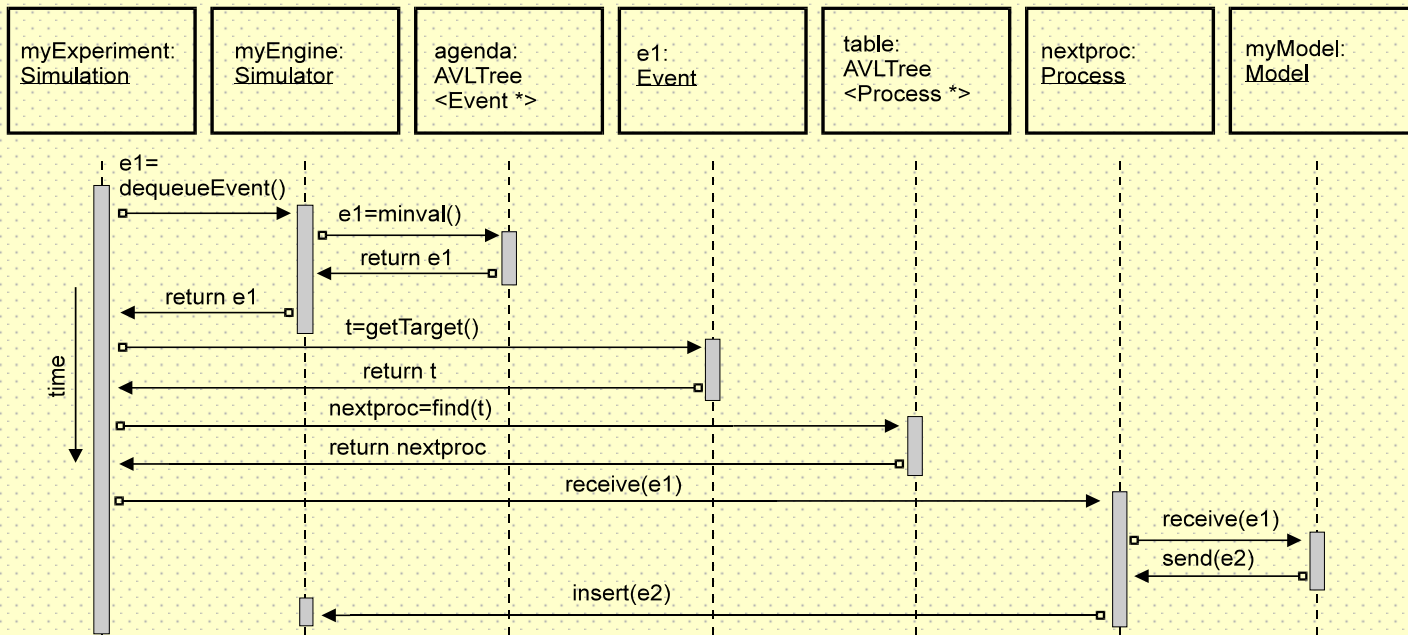
- V es el conjunto de nodos, i.e. los componentes activos de la red.
- E es el conjunto de enlaces, i.e. los canales bidireccionales que operan entre los nodos.
- A su vez, los nodos se modelan usando autómatas de entrada salida (I/O), que pueden reaccionar con tres tipos de eventos: de entrada, de salida e internos.

LA ARQUITECTURA (Diagrama de clases)



UN SIMULADOR DE EVENTOS DISCRETOS

LA ARQUITECTURA (Diagrama de secuencias)



UN SIMULADOR DE EVENTOS DISCRETOS

* CONSTRUCCION

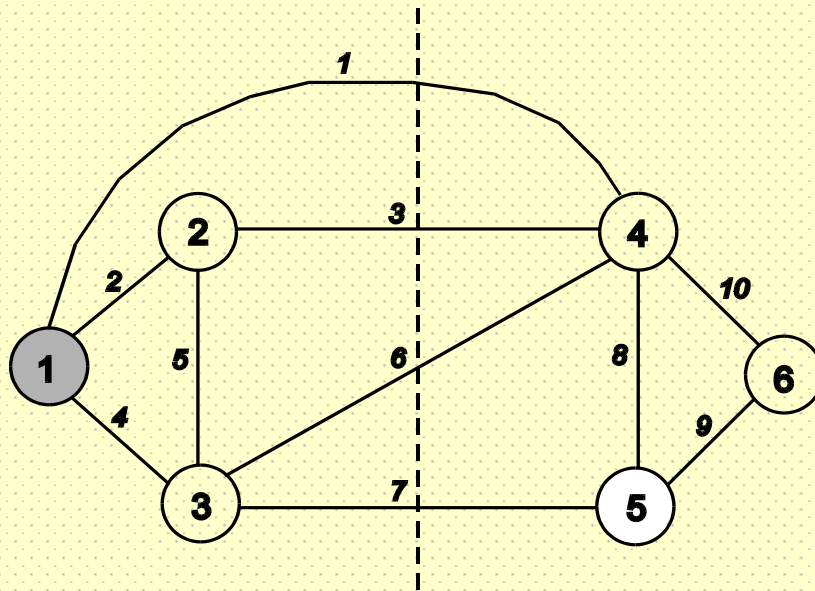
* ***APLICACIONES***

* TRABAJO FUTURO

¿QUÉ LLEVAMOS HECHO?

1. DFS, PI, PIF, Bellman-Ford, Dijkstra, Elección, AGM.
2. Problemas de optimización en computación distribuida.
Las soluciones heurísticas de tipo hormiga pueden ser una opción para construir sistemas distribuidos que puedan (re)configurarse de manera automática.
3. Problemas de enrutamiento en redes móviles.
La plataforma de simulación puede servir para estudiar redes de telecomunicaciones, algoritmos distribuidos, problemas de optimización, realidad virtual, etc.

EL PROBLEMA DE PARTICIÓN



- Modelo de red asíncrona
- Gráfica de comunicaciones

$$G(V,E), n=|V|, m=|E|$$

- G es ponderada, i.e.

$$\forall e \text{ in } E, \omega(e) > 0 \Rightarrow d(u,v).$$

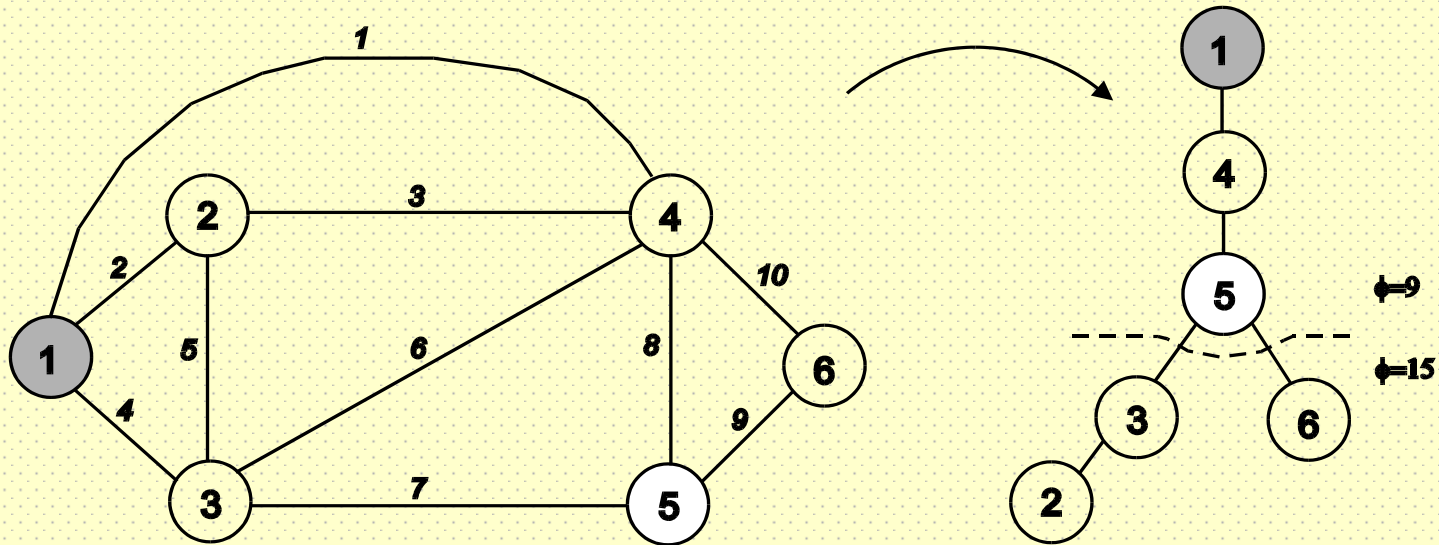
Suponiendo que $n=pN$,

Problema (SP): Encuentre una partición de V en subconjuntos V_1, V_2, \dots, V_p de tamaño N , tales que

$$\sum_{i=1}^p \sum_{u \in V_i} \sum_{v \in V_i - \{u\}} d(u,v)$$

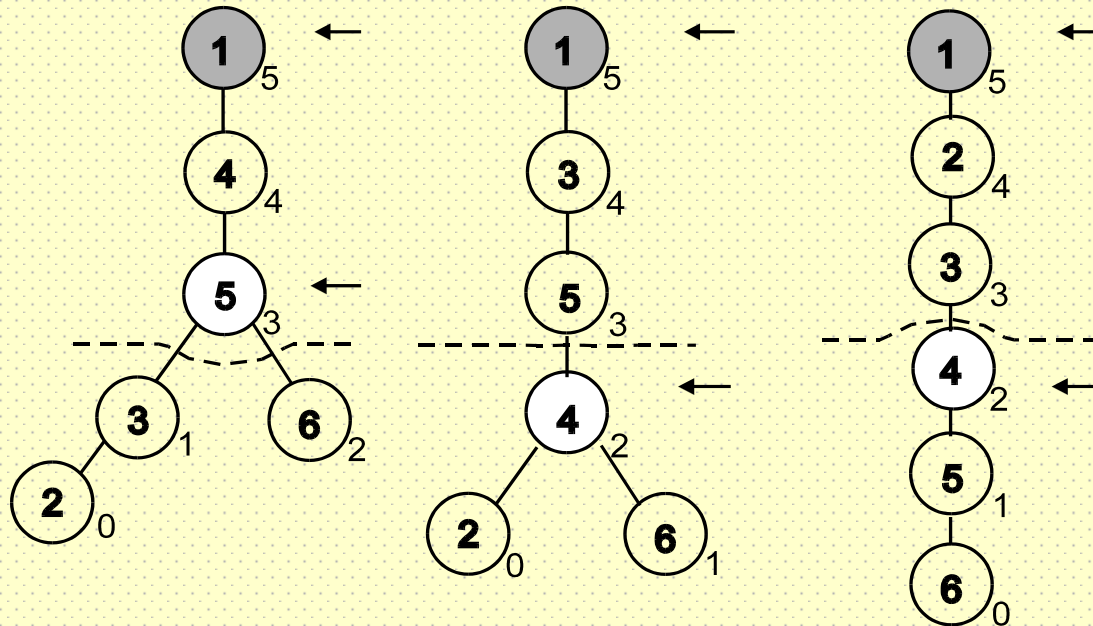
es mínima. SP es NP-duro para $N > 2$

EL BLOQUE DE CONSTRUCCIÓN



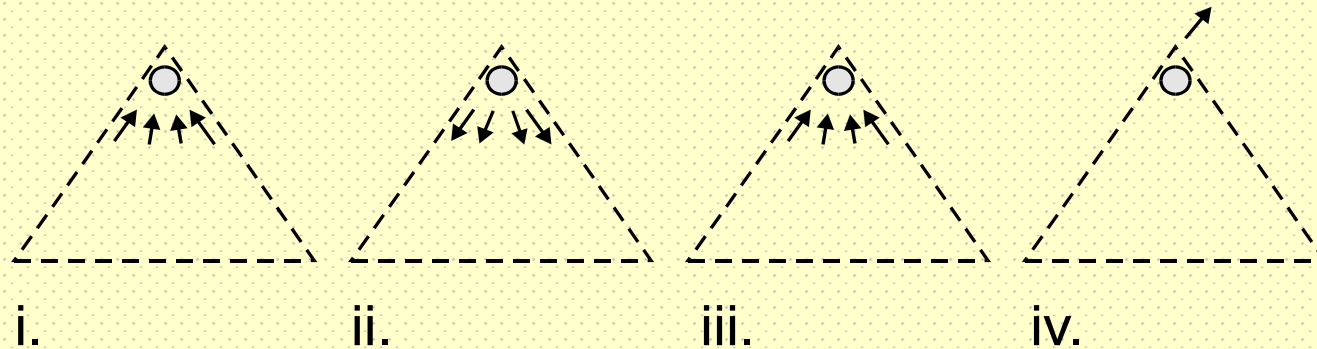
- Algoritmo distribuido DFS.
- Z hormigas = Z agentes DFS distribuidos.

LA JORNADA DE UN NIDO ARTIFICIAL (pasos 1&2)



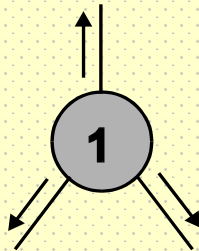
- Z hormigas se liberan desde el nido. Cada una desarrolla un recorrido aleatorio sobre G .
- La hormiga z , $z=1,\dots,Z$, visita cada nodo y lo asigna al subconjunto V_{zi} , $i=1,\dots,p$, en construcción. Cuando $|V_{zi}|=N$, comienza un nuevo subconjunto.

LA JORNADA ... (paso 3)



- i. el concentrador reúne la lista de los nodos en V_i .
- ii. el concentrador difunde esta lista dentro de la partición que administra.
- iii. cada nodo en V_i envía su candidato a diámetro.
- iv. el concentrador determina ϕ y lo envía hacia el nido.

LA JORNADA ... (pasos 4&5)

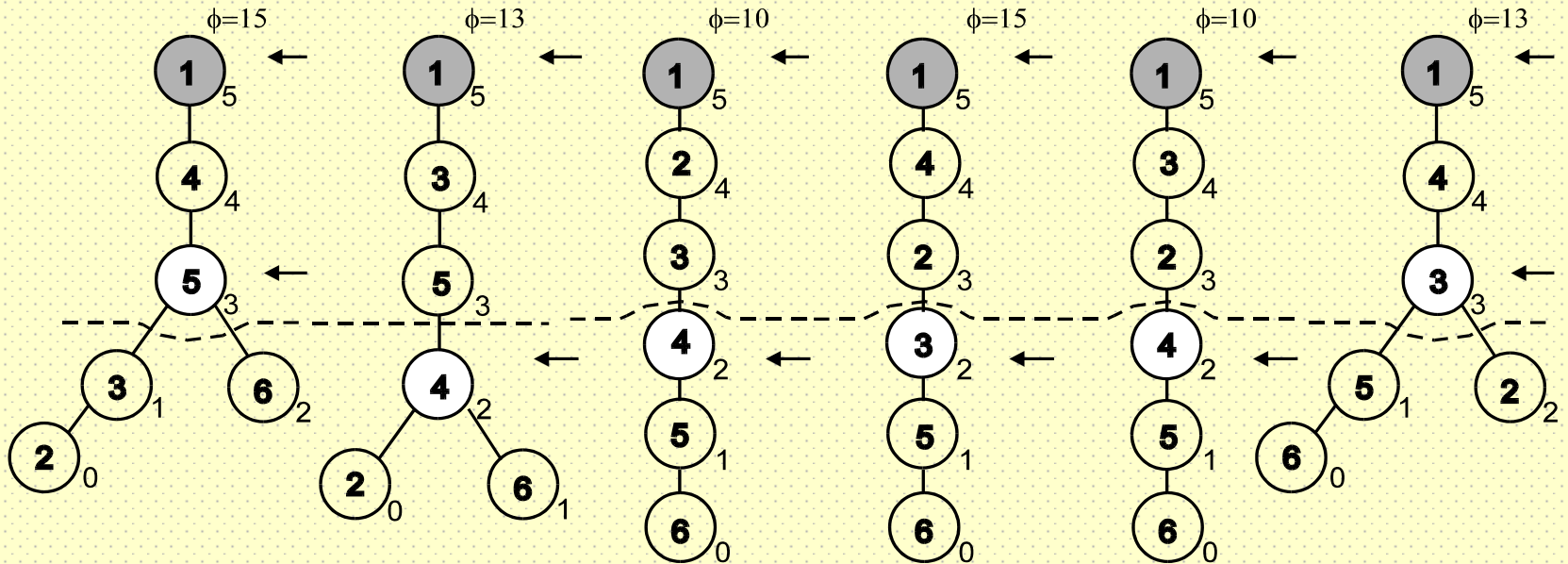


Cuando se han recibido todos los valores ϕ 's, el nido selecciona al mayor de ellos que caracteriza a la solución construida por la hormiga z.

Cuando todas las hormigas han concluido, se cierra el ciclo. El nido difunde el mensaje de recompensa para premiar a los recorridos que mejoraron la última solución conocida.

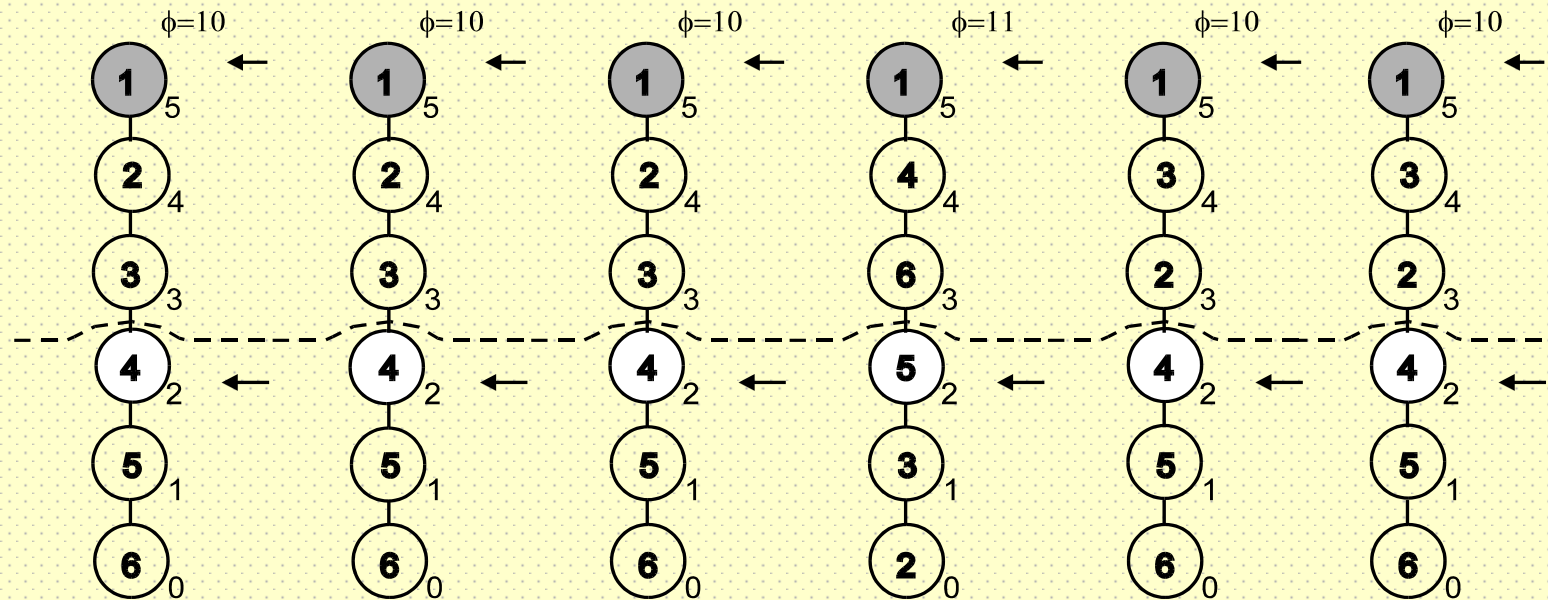
los pasos 1 a 4 se repiten durante un número fijo de ciclos o hasta alcanzar un criterio de convergencia.

USANDO NUESTRO SIMULADOR



UN SIMULADOR DE EVENTOS DISCRETOS

USANDO NUESTRO SIMULADOR



UN SIMULADOR DE EVENTOS DISCRETOS

* CONSTRUCCION

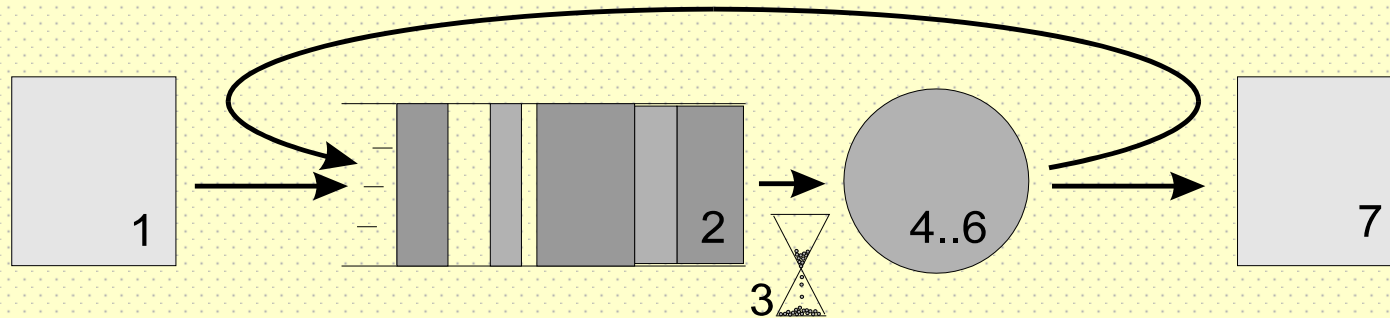
* APLICACIONES

* ***TRABAJO FUTURO***

¿QUÉ SIGUE?

1. Construir una interfaz gráfica de calidad.
2. Desarrollar la versión distribuida del simulador.
3. Convencer a la gente de las posibilidades del trabajo en equipo.

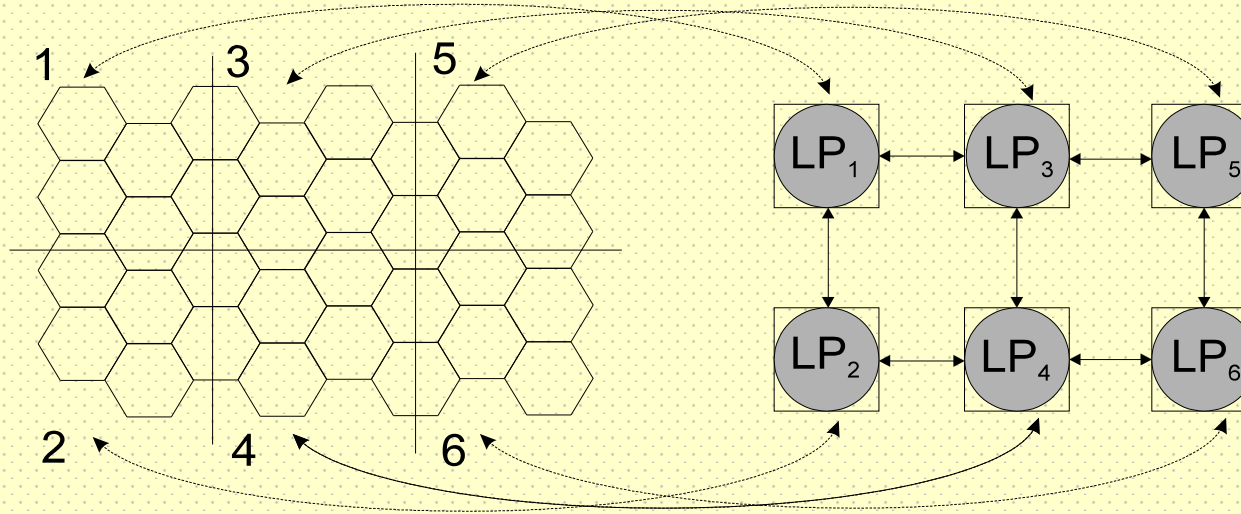
LIMITACIONES DE LA SED CENTRALIZADA



1. Rutinas de inicialización y entradas
2. Calendarizador
3. Reloj
4. Variables de estado
5. Rutinas para manejo de eventos
6. Rutina principal
7. Rutinas de salida y registro

Limitaciones: simular un experimento de gran tamaño (donde intervienen muchos componentes) puede tomar varias horas, ¡o días!

ALTERNATIVA



Repartir la carga de trabajo sobre una red de computadoras

TEMAS DE INVESTIGACIÓN

- Faltan modelos generales para predecir el escenario óptimo de una simulación.
- Estudiar nuevas técnicas de sincronización entre componentes de procesamiento.
- ***Mecanismos para tolerar degradación en las capacidades de los componentes, incluyendo las fallas de paro.***

GRACIAS