Lecture 3

Modelling program/operating system operation using PN's. Derivation of a state based model of wait and signal.

What are we modelling?

- Programs encode a set of states and the transitions between them.
- Petri-Nets can be used to improve our understanding of systems prior to program development

OR

- Petri-Nets can be used to help us to understand what an existing program does.

Methodology

- In both cases we must first identify the what the important program states are. This helps us to form a clear understanding of what the program does, or will do.
- Second we determine what actions or conditions trigger changes between these states.
- Tokens may represent the flow of data, or the movement of other objects in which we are interested.

Example problem

- Suppose we wish to model the way in which wait and signal manage processes.
- In this situation the states are the states of a process, and the tokens represent processes.
- Arcs show how processes pass between the process states used by signal and wait.
- We will model the case of a simple semaphore with value 1 or 0.

Problems

- Suspended processes are not distinguished.
- Signal does not awake a suspended process.
- We need to be able to determine when there are no suspended processes and respond accordingly.
- For this we need to define a new type of arc.

What are our states?

Waiting Process

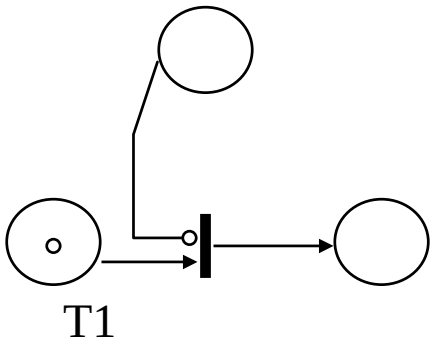
$\{P_n\}$
Wait(S);
$\{P_{n+1}\}$

Signalling Process

$\{P_k\}$
Signal(S);
$\{P_{k+1}\}$

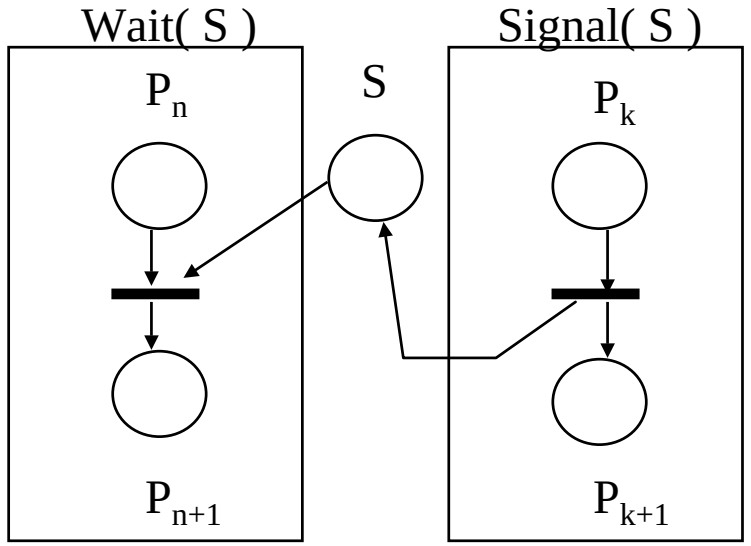
Inhibitory arcs

- Allow us to enable transitions when a place is empty.
- Eg.

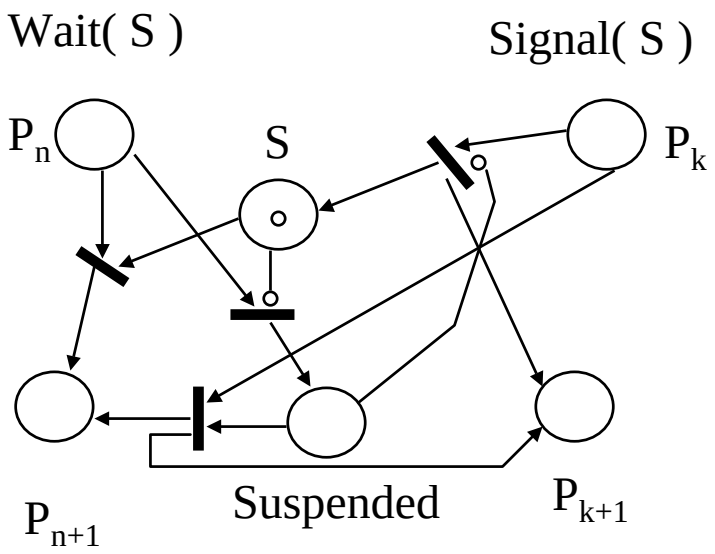


- T1 is enabled.

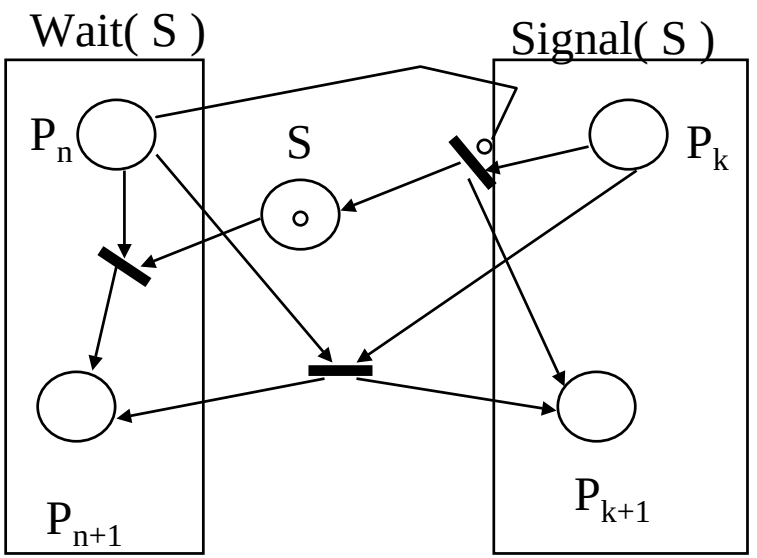
Initial model



Final model



Wait and Signal revisited



Interpretation

- # tokens in S is the value of the semaphore.
- Other tokens represent processes.
- Places represent the possible process states.
- Transitions model the execution of the program.
- This is the best sort of net as it is closely related to the structure of the problem being modelled.

Comments

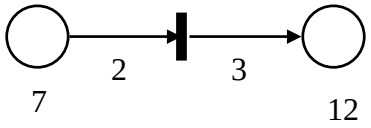
- We still cannot distinguish between processes about to perform a Wait and stalled processes awaiting a Signal.
- Stalled processes are not given priority when a Signal occurs.

Solution

- Introduce an additional state which represents suspended processes.

Extensions

- Introduce
  - Bounded places.
  - Weighted arcs.
- Eg.



- New Firing Rule: A transition is enabled if each source place has as many tokens as the weight of the related arc, and all the destination places have room for the tokens which will be generated

Related techniques

- Place transition nets.
- Predicate transition nets
  - coloured tokens
  - coloured arcs
  - boolean guards on transitions
- Numerical Petri-Nets
- Problem: Analysis can become very complex.