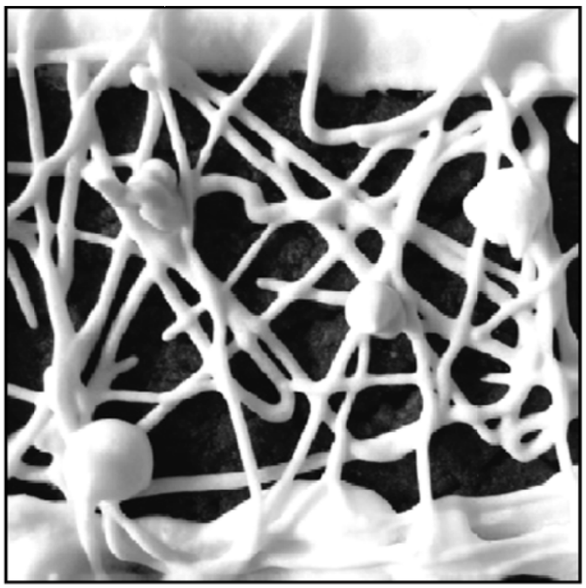
Concurrent Computing

Lecturers: Prof. Majid Mirmehdi (majid@cs.bris.ac.uk)
Dr. Tilo Burghardt (tilo@cs.bris.ac.uk)
Dr. Daniel Page (page@cs.bris.ac.uk)

Web: http://www.cs.bris.ac.uk/Teaching/Resources/COMS20001



LECTURE 13

INTRODUCTION TO PETRI NETS

Definition of Petri Nets

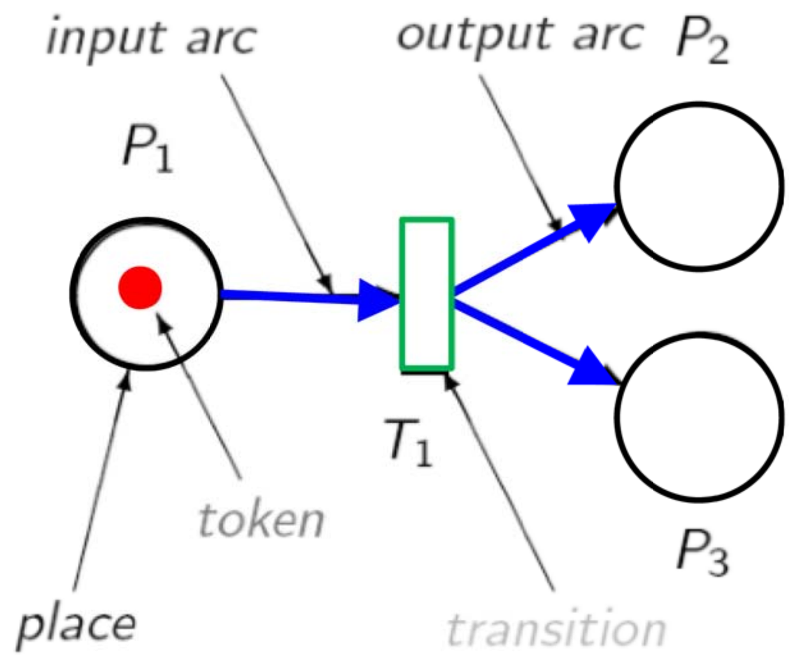
Carl Adam Petri

Formally, a Petri Net N is a 4-tupel describing an annotated, directed, bipartite graph:

$$N = \{P, T, A, M_0\}$$

where

- P is a finite set of **places**
- T is a finite set of **transitions**
- A is a finite set of **arcs** (arrows)
- M_0 is the initial token marking



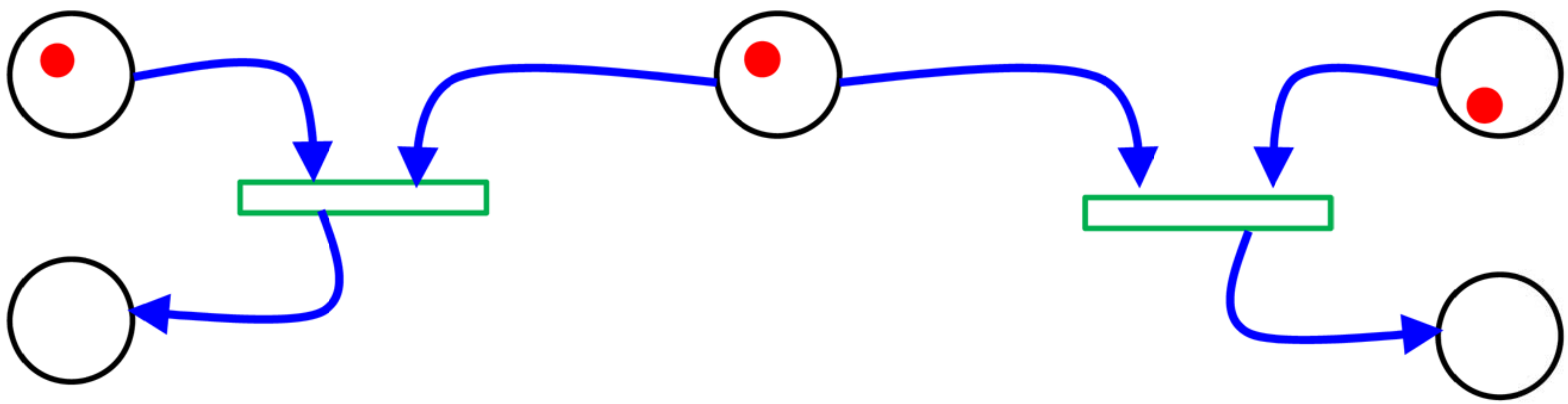
Elements of a Petri net

Basic Components of Petri Nets

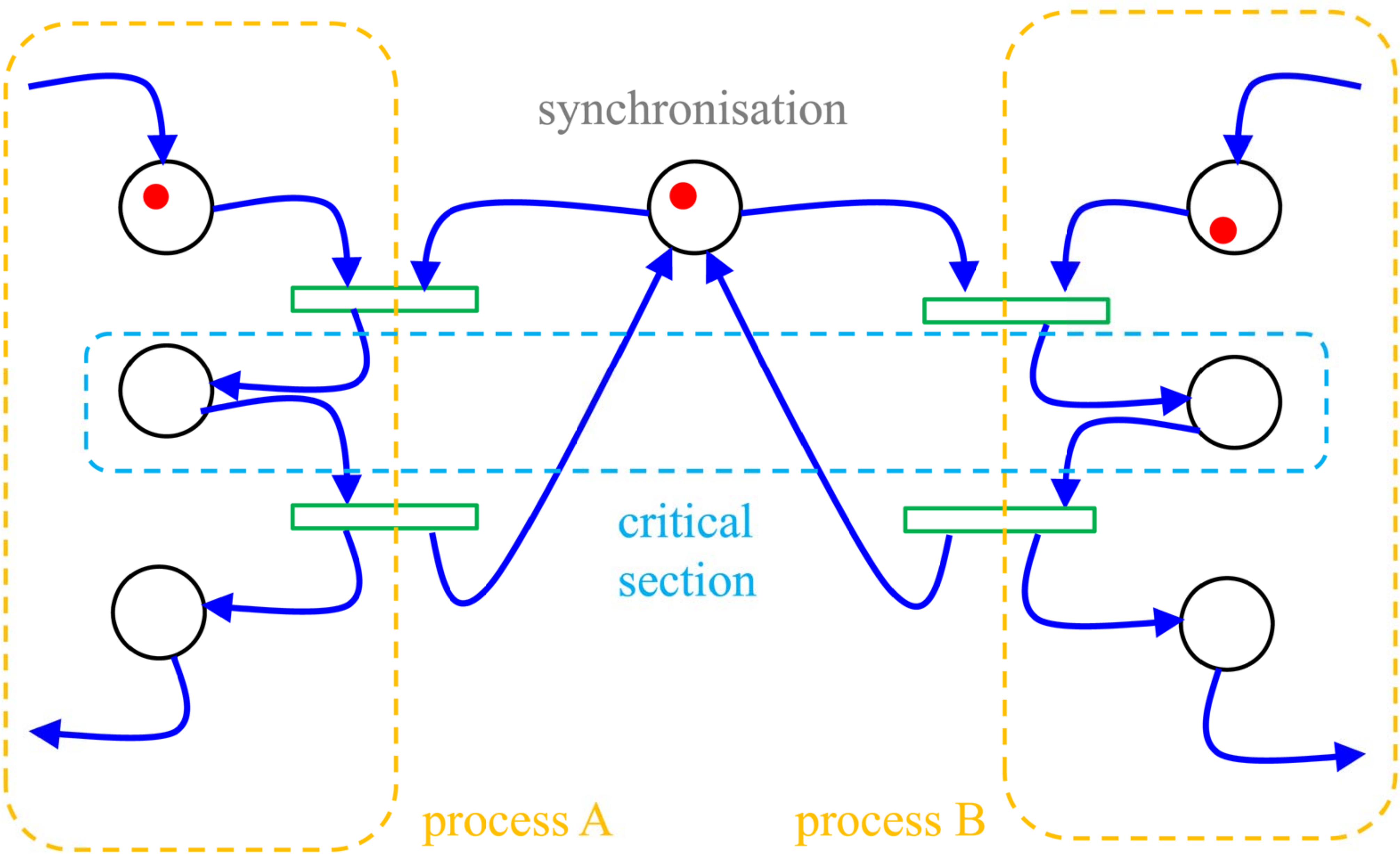
- **PLACES** … may contain a natural number of tokens;
(a distribution of tokens over the places of a net is called a *marking*)

- **TRANSITIONS** … model activity, they may *fire* whenever enabled (i.e. a token is present at all input arcs)

- **ARCS** … are directed and run from a place to a transition or vice versa (never between places or between transitions)



Example Petri Net: Binary Semaphore

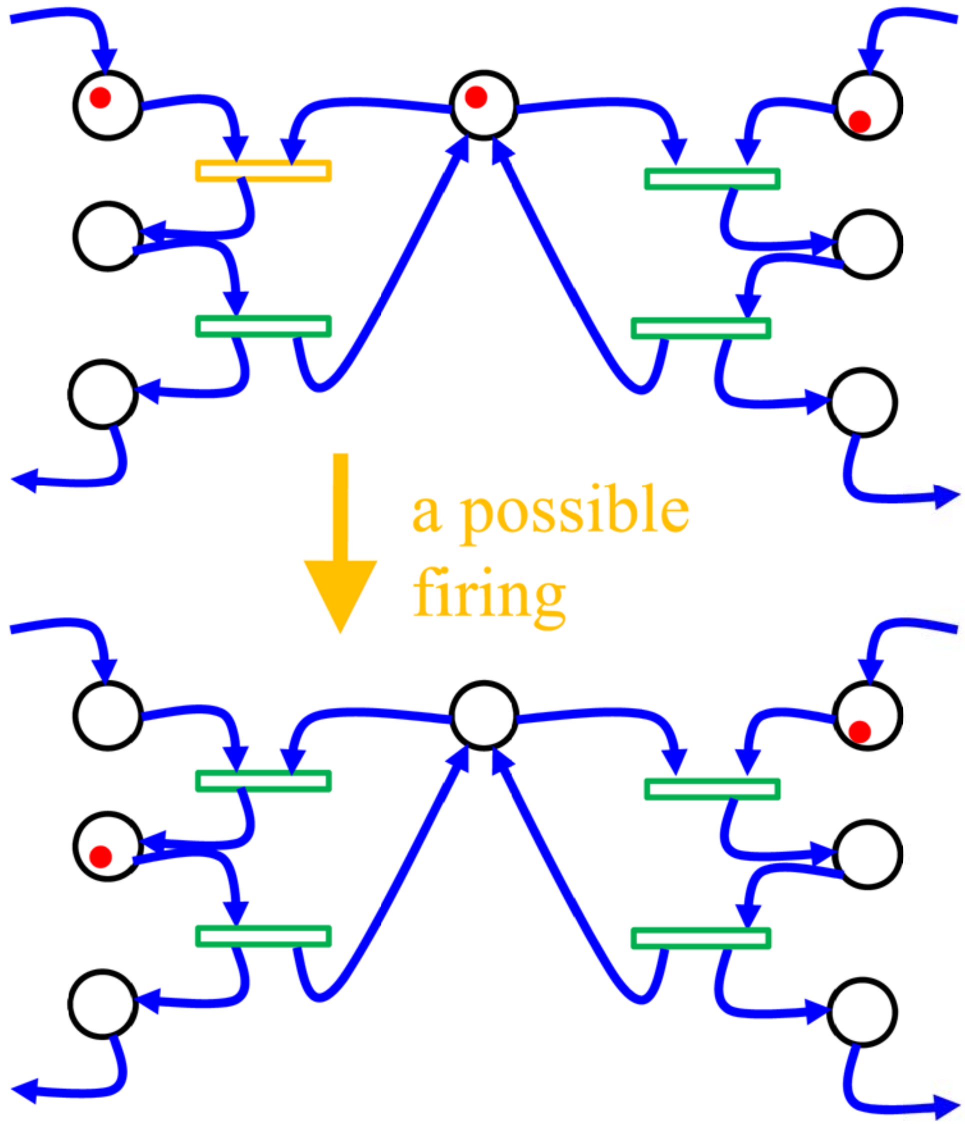


Dynamics and Non-Determinism

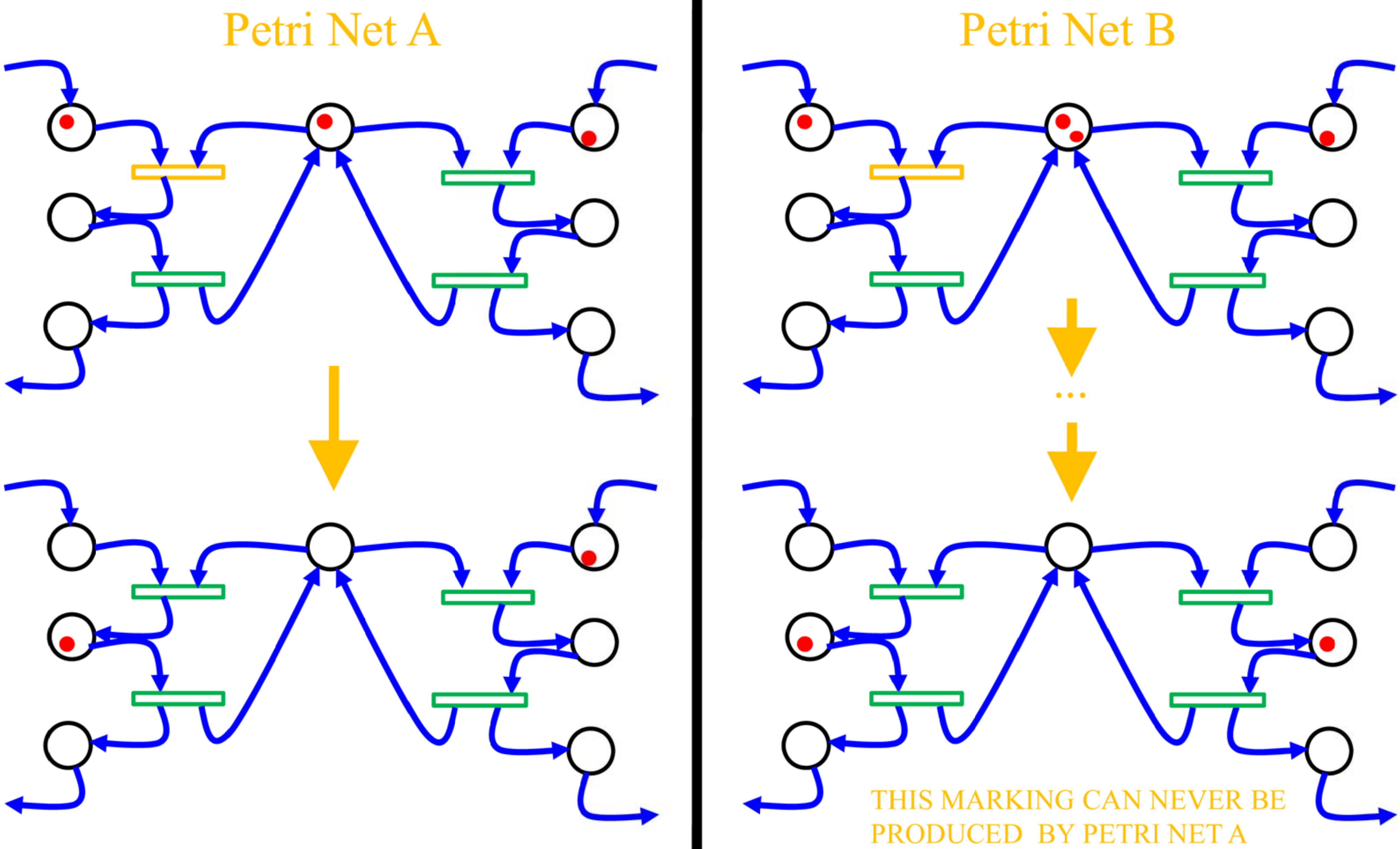
- **FIRING** …when a transition fires (atomic), it consumes *all* input tokens, and places tokens at the end of *all* output arcs.

- **MULTIPLE TOKENS** … may be present anywhere in the net (even at the same place);

-**NON-DETERMINISM** … execution of Petri nets is non-deterministic: when multiple transitions are enabled at the same time, any one of them may fire;



Initial Marking is Part of a Net's Behaviour



Types of Analysis of Petri Nets

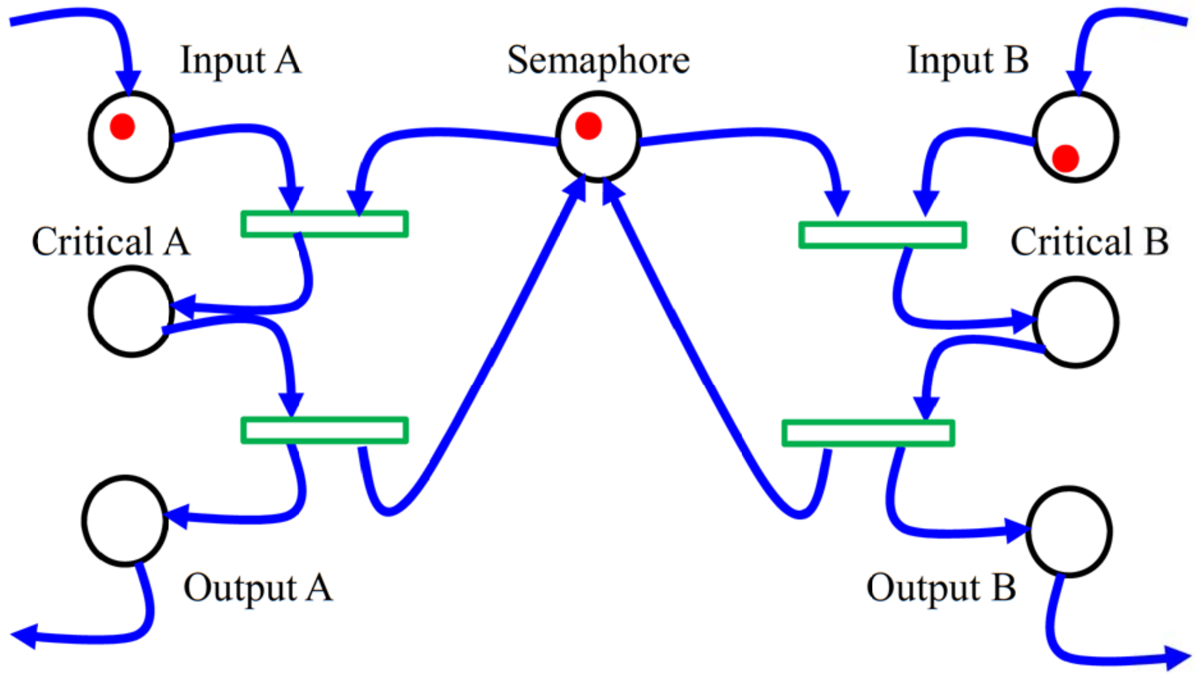
- **Analysis Goal:** demonstrate (In)correct Behavior
- **Problem in General:** too many possibilities to test for full (in)correctness proof

- **Dynamic Analysis ('Playing the Token Game')**
  – executes specification to reveal properties
  – requires executable specifications
  – an experiment characterizes a single behaviour

- **Static Analysis**
  – examines specification structure to reveal properties
  – useful in the absence of execution semantics, but also where execution would be impractical

Example Task for Dynamic Analysis

Is it possible for both processes to be in their critical sections at the same time in the same marking?

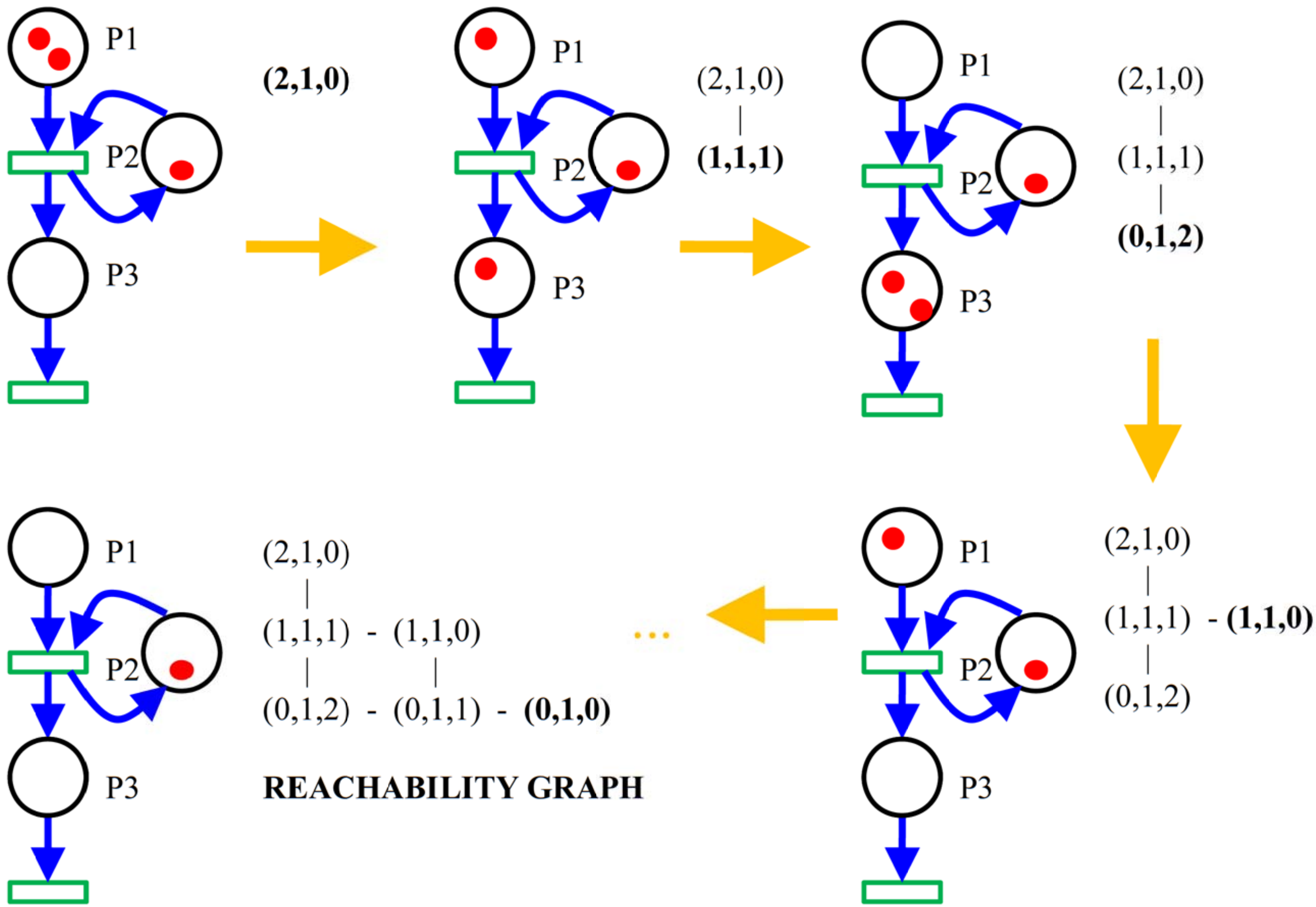


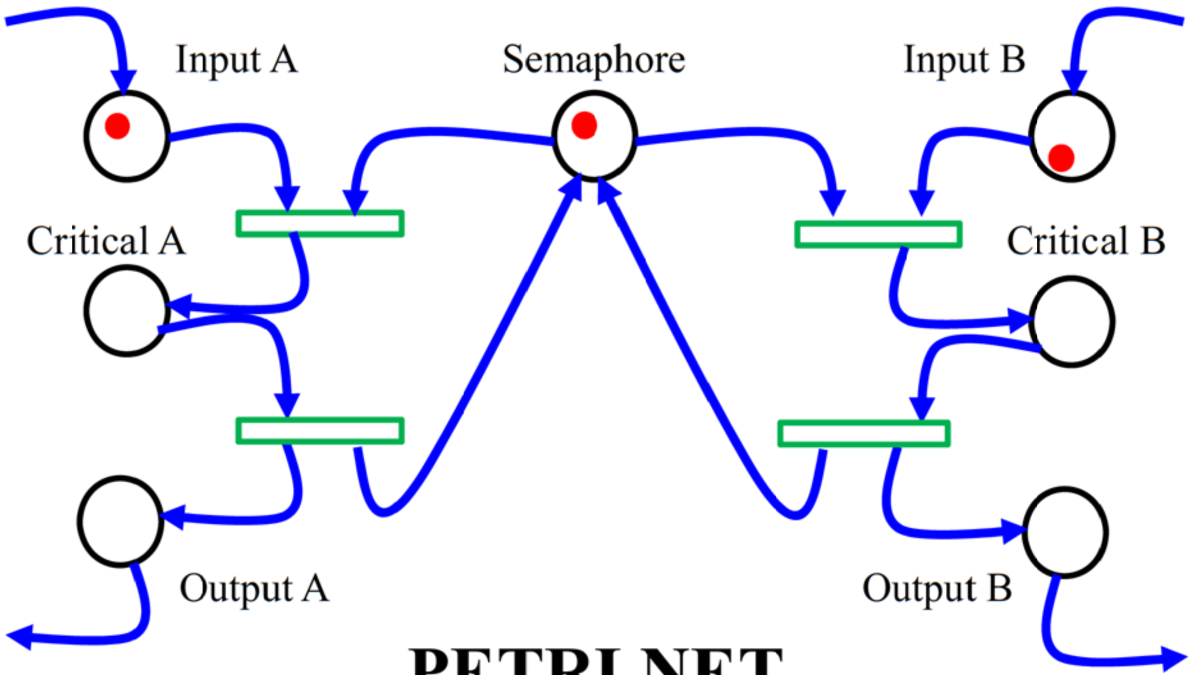
…that is, is the following a valid marking?

M = (InputA , CriticalA , OutputA, Semaphore,
InputB , CriticalB , OutputB) = (?,1,?,?,?,1,?)

Analysis Task: evaluate *all* markings and check for above pattern…

Example: Reachability Graph Construction

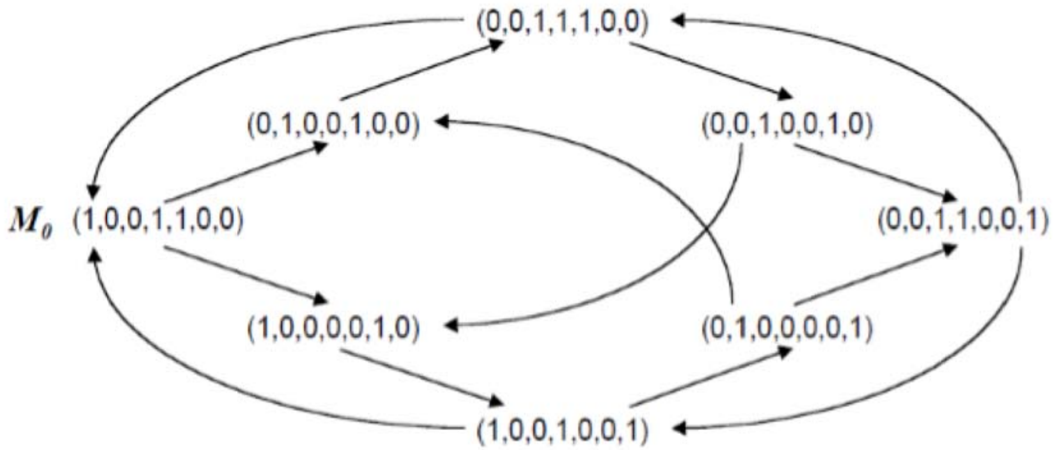




PETRI NET

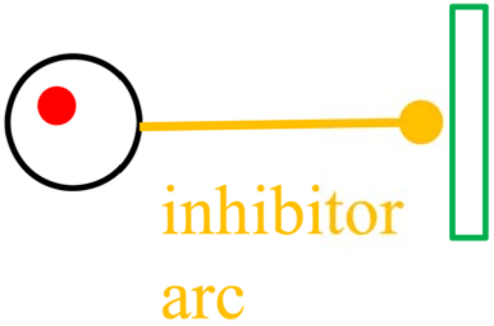
M_0 = (InputA , CriticalA , OutputA, Semaphore, InputB , CriticalB , OutputB) = (1,0,0,1,1,0,0)

…it is not possible for both processes to be in their critical sections at the same time…there is no marking (?,1,?,?,?,1,?)…

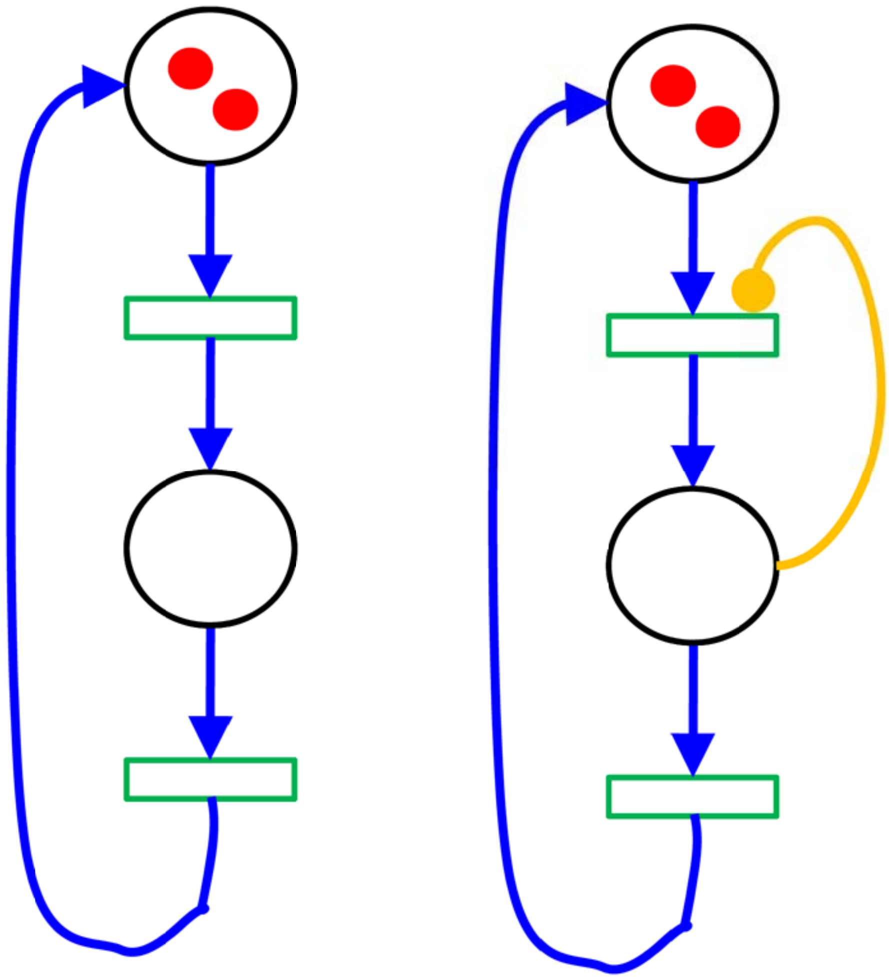


REACHABILITY GRAPH

Inhibitor Arcs



- an inhibitor arc from a place to a transition is used to indicate that token presence at the place disables the linked transition



How does the behaviour of the above Petri Nets change by introduction of an inhibitor arc?

Petri Nets vs CSP

- **PROCESS STATES** … map to reachable markings, i.e. the nodes of the reachability graph

- **TRACES** … can be compared to the information contained in paths through the reachability graph starting at M_0

- **FAILURES** … are paths and markings with the inability to perform any transitions (due to lack of enabled transitions)

- **LIVELOCK** … despite deadlock-freedom, ‘important’ places are never reached by tokens