

# Rendezvous of Two Robots with Visible Bits

## ALGOSENSORS 2013

Giovanni Viglietta

School of Computer Science, Carleton University, Canada

Sophia Antipolis – September 6, 2013

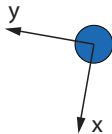
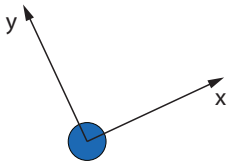
# The Rendezvous problem



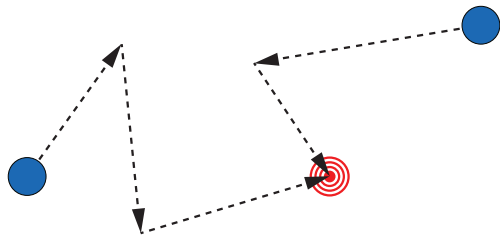
# The Rendezvous problem



# The Rendezvous problem



# The Rendezvous problem



Robots in the basic CORDA model have:

- Motorial capabilities  
(freely move in a Euclidean space)
- Sensorial capabilities  
(sense the positions of other robots)
- No explicit way of communicating

Robots in the basic CORDA model have:

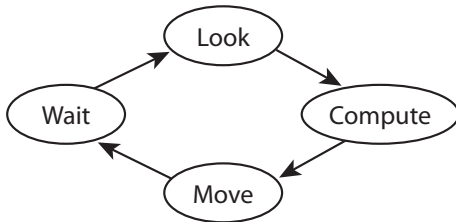
- Motorial capabilities  
(freely move in a Euclidean space)
- Sensorial capabilities  
(sense the positions of other robots)
- No explicit way of communicating

Additionally, they are:

- Homogeneous (all executing the same deterministic algorithm)
- Autonomous (no centralized control)
- Oblivious (no memory of past events)
- Anonymous (no unique identifiers)

# Synchronicity

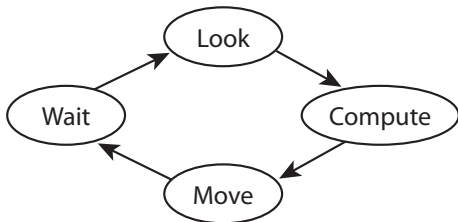
Robots' life cycle:





# Synchronicity

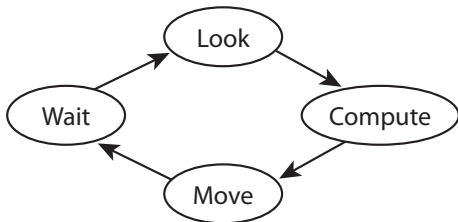
Robots' life cycle:



- *Fully Synchronous* model: there is a global clock. At each cycle, both robots are activated and execute it simultaneously.

# Synchronicity

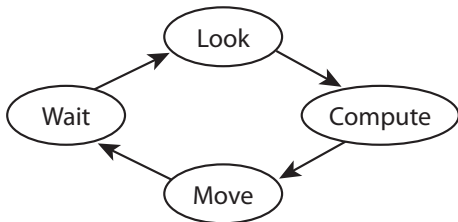
Robots' life cycle:



- *Fully Synchronous* model: there is a global clock. At each cycle, both robots are activated and execute it simultaneously.
- *Semi-Synchronous* model: there is a global clock but, at each cycle, a robot may be activated or not.

# Synchronicity

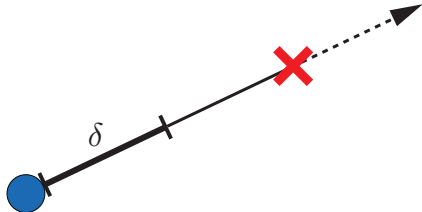
Robots' life cycle:



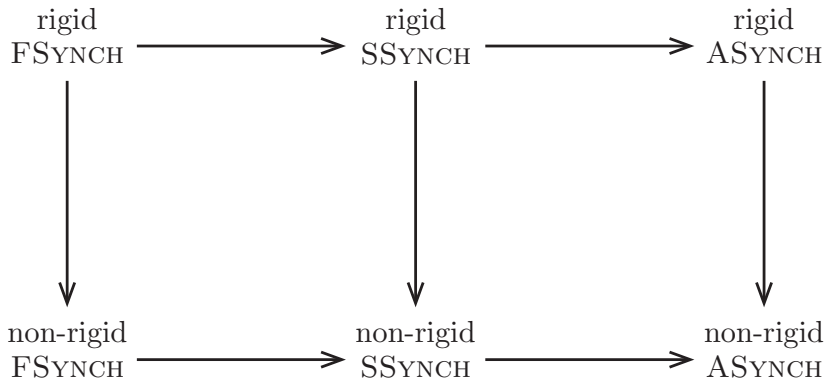
- *Fully Synchronous* model: there is a global clock. At each cycle, both robots are activated and execute it simultaneously.
- *Semi-Synchronous* model: there is a global clock but, at each cycle, a robot may be activated or not.
- *Asynchronous* model: there is no global clock. Each phase of a cycle may have any finite duration, and different robots' executions are completely independent.

# Non-rigid models

- An adversary can stop a robot before it reaches its destination.
- We assume that no robot can be stopped before it has moved by at least a constant  $\delta$  (unknown to the robots).



# Basic models



# Solvability in non-rigid FSYNCH



# Solvability in non-rigid FSYNCH



# Solvability in non-rigid FSYNCH





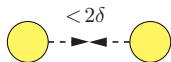
# Solvability in non-rigid FSYNCH



# Solvability in non-rigid FSYNC



# Solvability in non-rigid FSYNCH



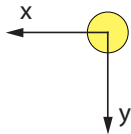
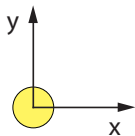
# Solvability in non-rigid FSYNCH



# Unsolvability in rigid SSYNCH



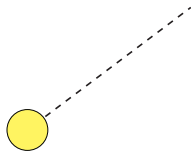
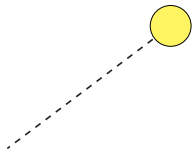
# Unsolvability in rigid SSYNCH



# Unsolvability in rigid SSYNCH

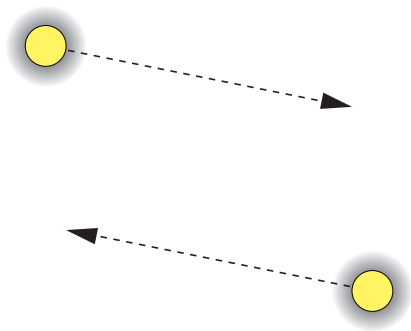


# Unsolvability in rigid SSYNCH

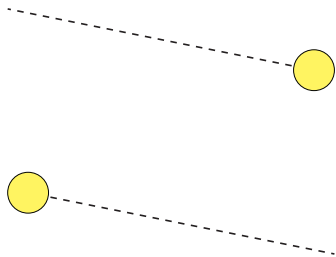




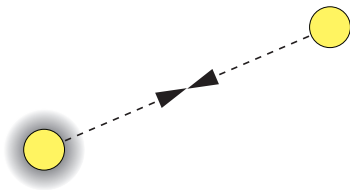
# Unsolvability in rigid SSYNCH



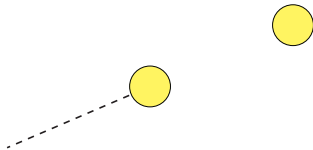
# Unsolvability in rigid SSYNCH



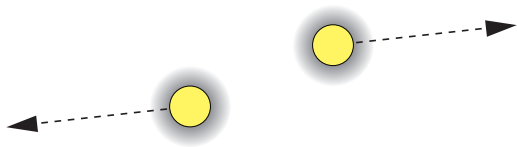
# Unsolvability in rigid SSYNCH



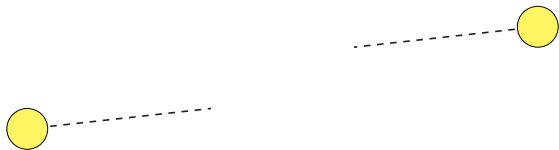
# Unsolvability in rigid SSYNCH



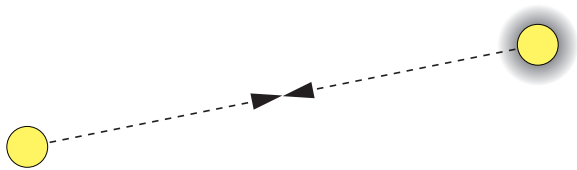
# Unsolvability in rigid SSYNCH



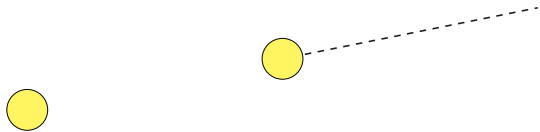
# Unsolvability in rigid SSYNCH



# Unsolvability in rigid SSYNCH



# Unsolvability in rigid SSYNCH





# Solvability of Rendezvous

- Solvable in non-rigid  $FSYNCH$
- Unsolvable in rigid  $SSYNCH$

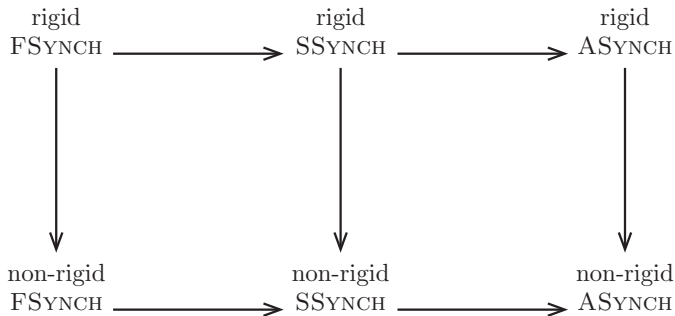
# Solvability of Rendezvous

- Solvable in non-rigid  $FSYNCH$
- Unsolvable in rigid  $SSYNCH$
- Rendezvous becomes solvable even in the non-rigid  $ASYNCH$  if each robot carries a “colored light” that can be seen by both robots and used in their computations.

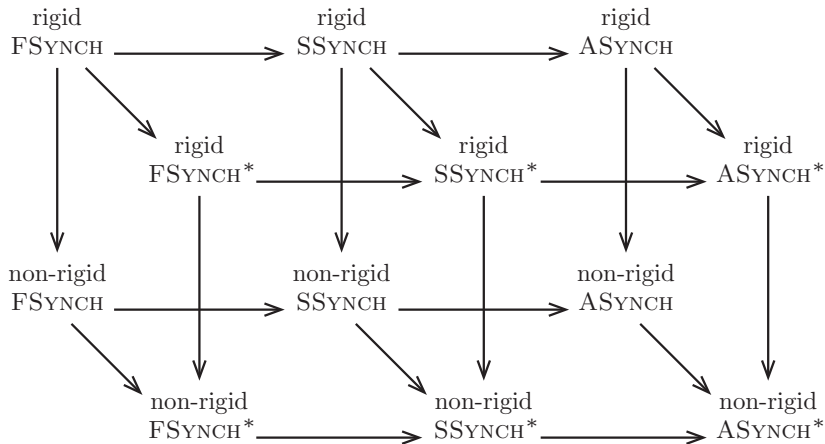
Theorem (Das, Flocchini, Prencipe, Santoro, Yamashita, 2012)

*Rendezvous is solvable in non-rigid  $ASYNCH$  if each robot carries a visible light with 4 possible states.*

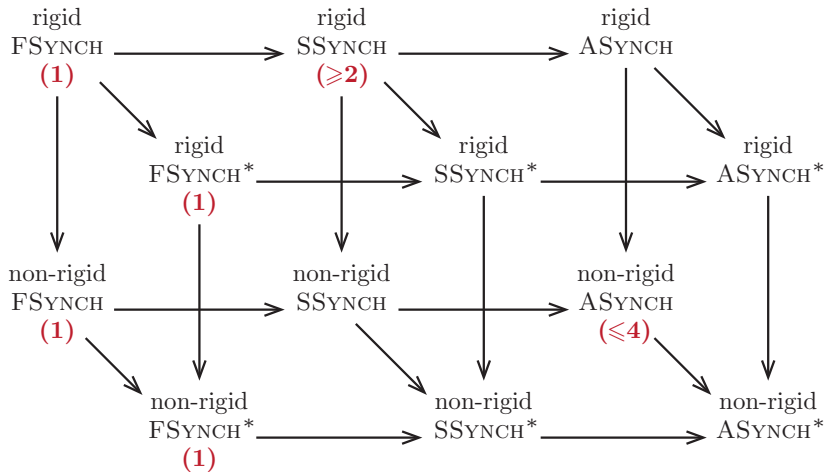
# Models with visible lights

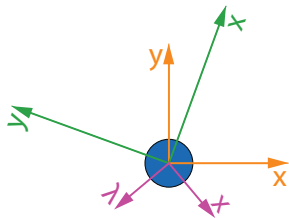


# Models with visible lights

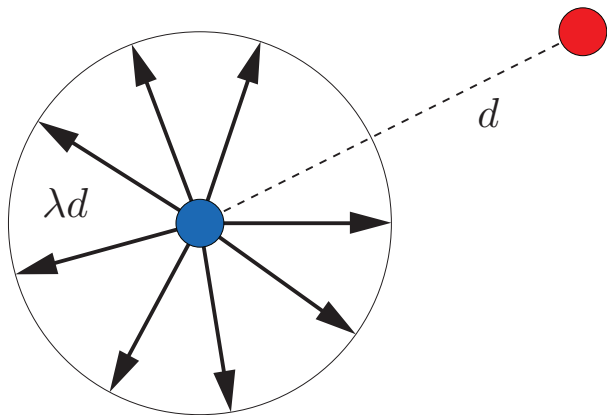


# Models with visible lights

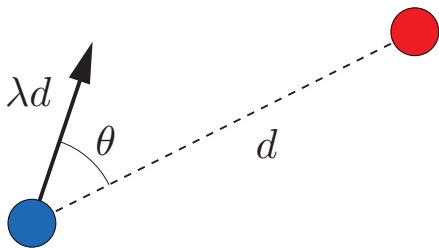




What if a robot's coordinate system is not self-consistent?

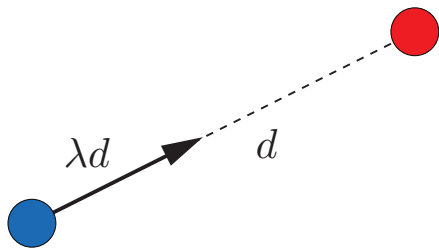


What if a robot's coordinate system is not self-consistent?

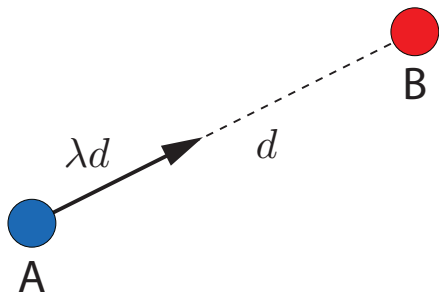


What if a robot's coordinate system is not self-consistent?

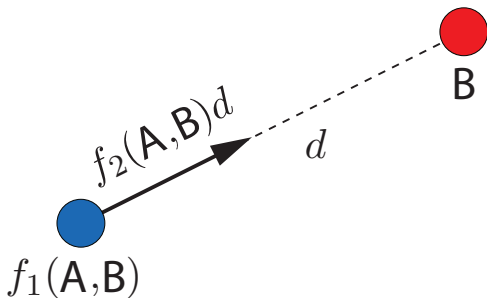




What if a robot's coordinate system is not self-consistent?

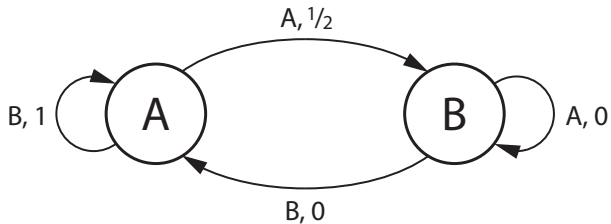


What if a robot's coordinate system is not self-consistent?

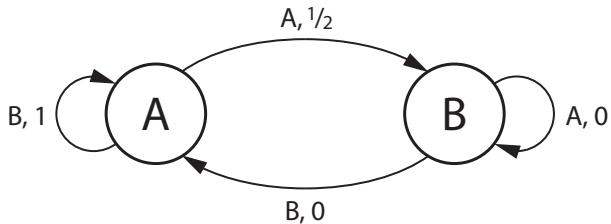


What if a robot's coordinate system is not self-consistent?

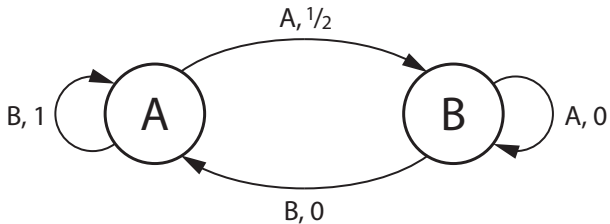
# Algorithm for non-rigid SSYNCH\*



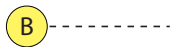
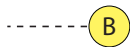
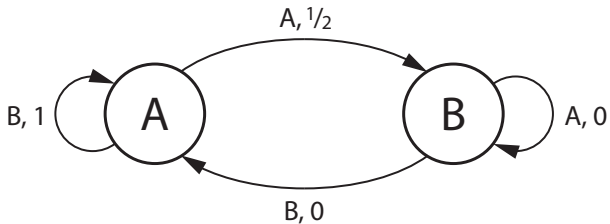
# Algorithm for non-rigid SSYNCH\*



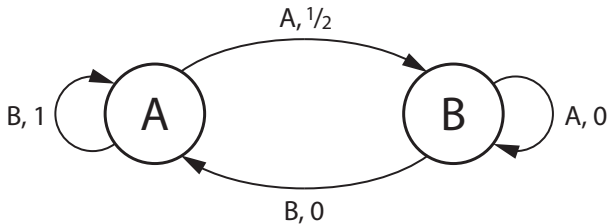
# Algorithm for non-rigid SSYNCH\*



# Algorithm for non-rigid SSYNCH\*

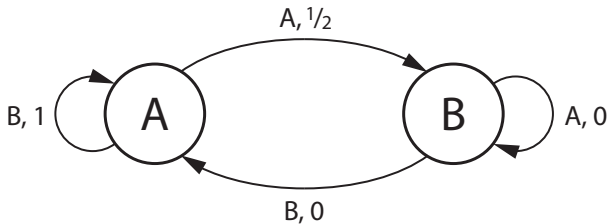


# Algorithm for non-rigid SSYNCH\*

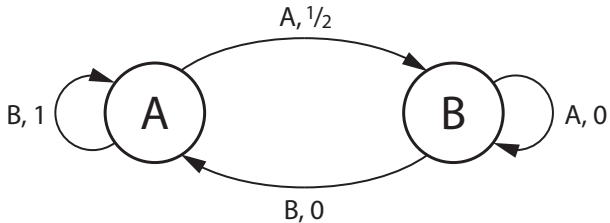




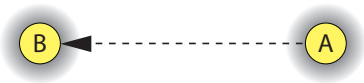
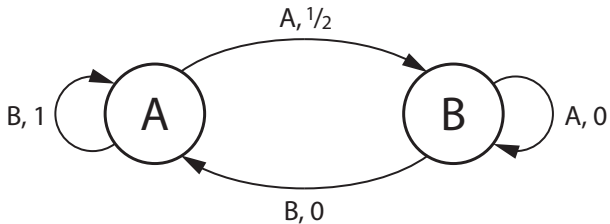
# Algorithm for non-rigid SSYNCH\*



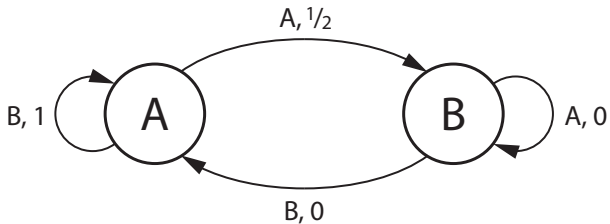
# Algorithm for non-rigid SSYNCH\*



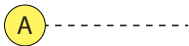
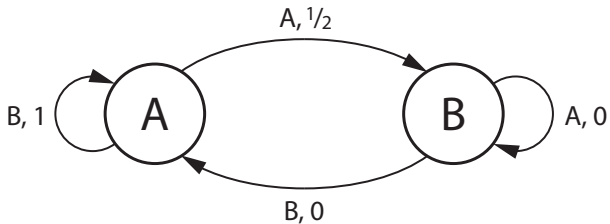
# Algorithm for non-rigid SSYNCH\*



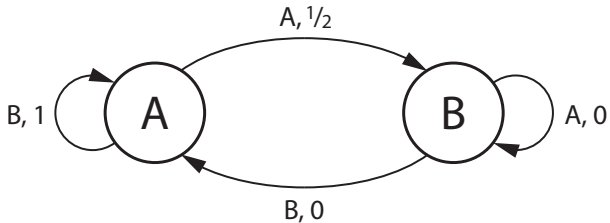
# Algorithm for non-rigid SSYNCH\*



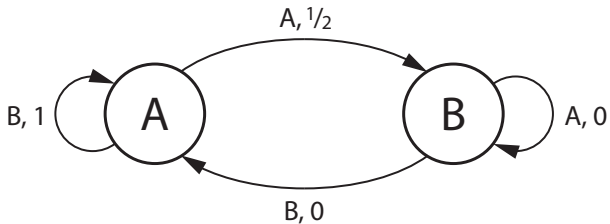
# Algorithm for non-rigid SSYNCH\*



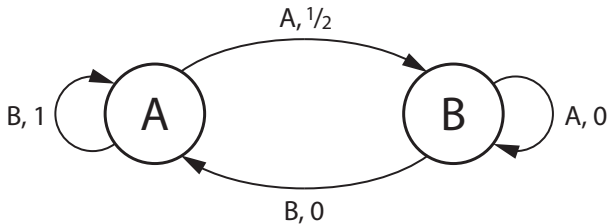
# Algorithm for rigid ASYNCH



# Algorithm for rigid ASYNCH

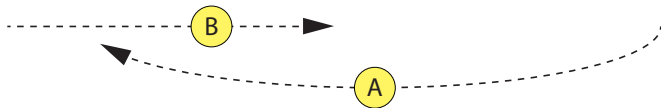
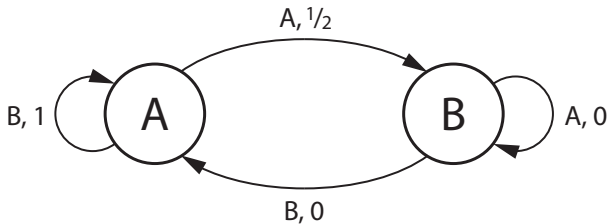


# Algorithm for rigid ASYNCH

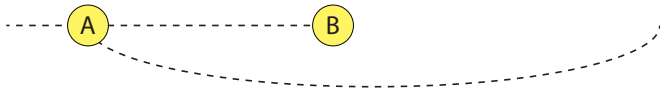
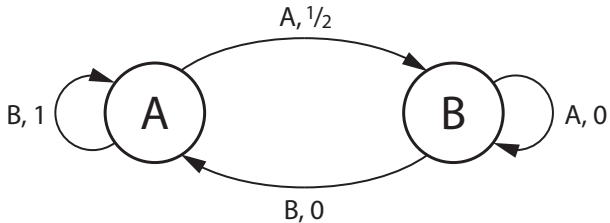




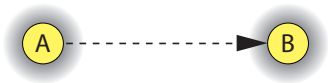
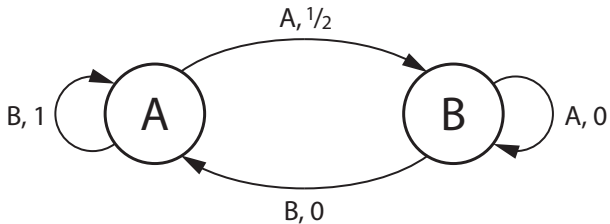
# Algorithm for rigid ASYNCH



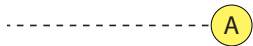
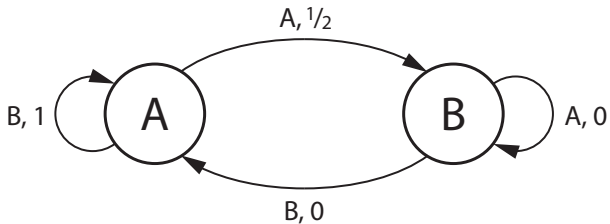
# Algorithm for rigid ASYNCH



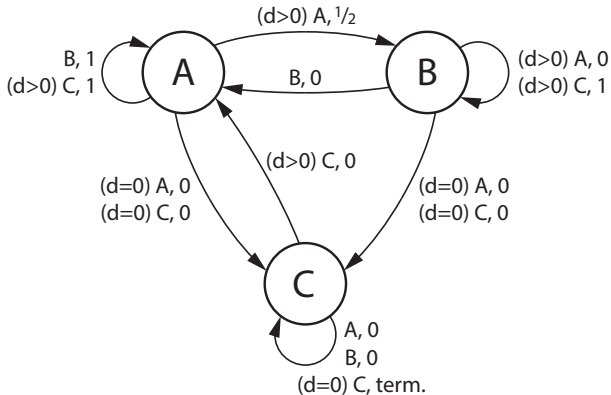
# Algorithm for rigid ASYNCH



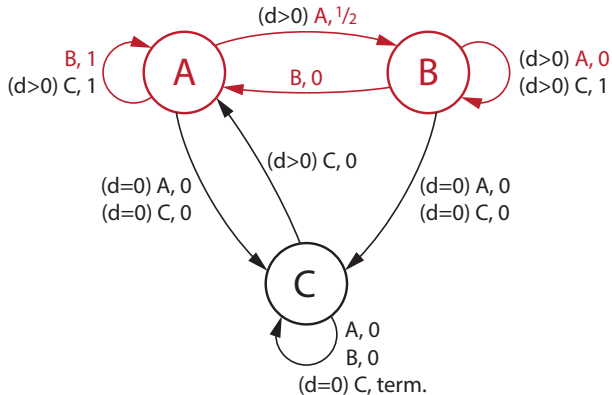
# Algorithm for rigid ASYNCH



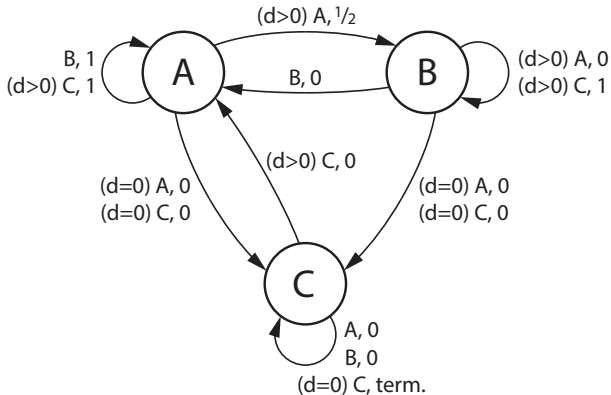
# Algorithm for non-rigid ASYNCH\* with termination



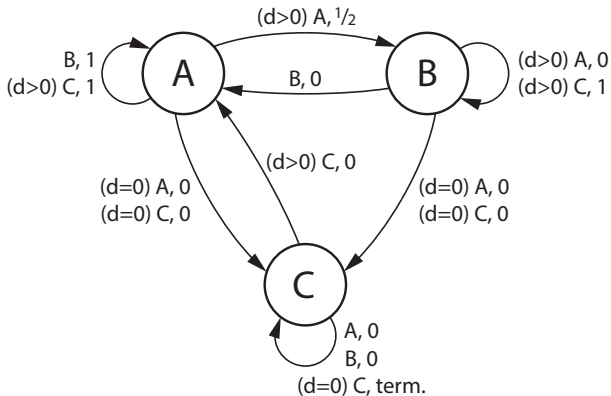
# Algorithm for non-rigid ASYNCH\* with termination



# Algorithm for non-rigid ASYNCH\* with termination



# Algorithm for non-rigid ASYNCH\* with termination

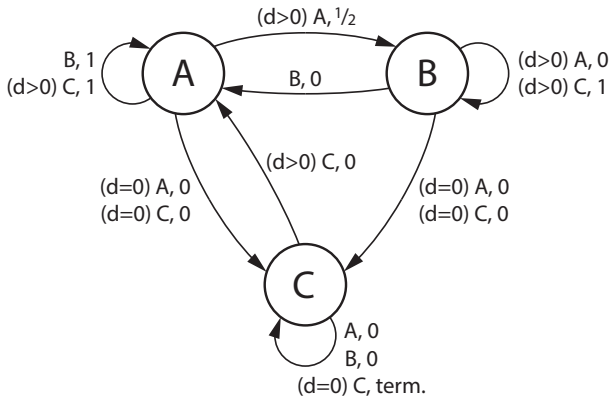


A

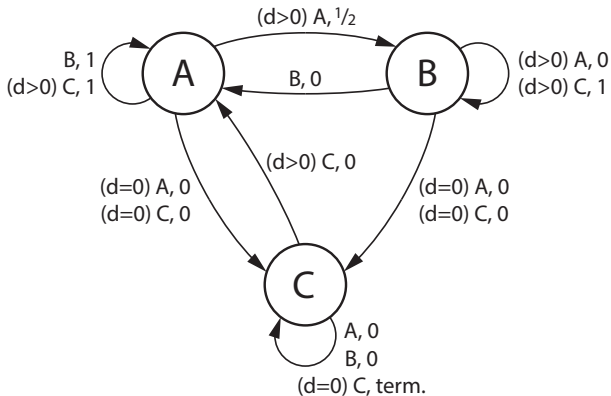
A



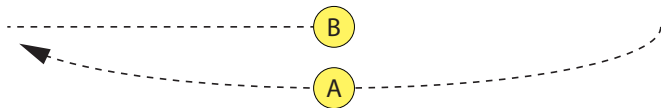
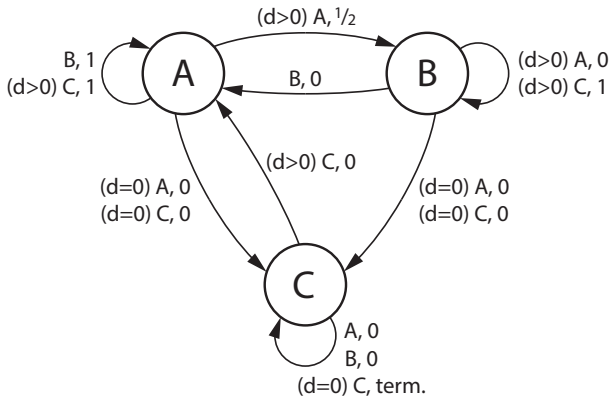
# Algorithm for non-rigid ASYNCH\* with termination



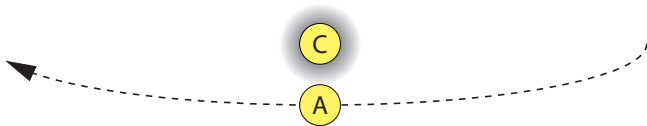
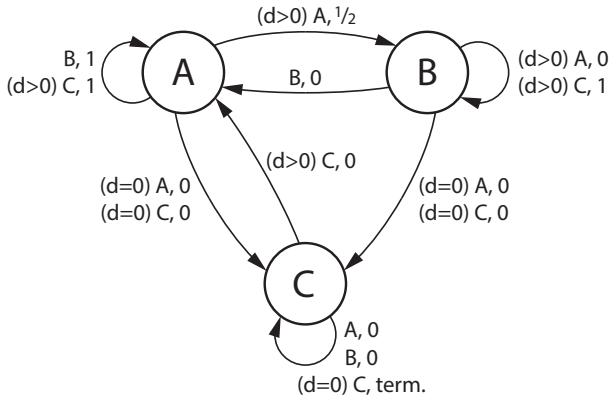
# Algorithm for non-rigid ASYNCH\* with termination



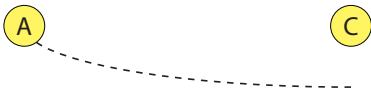
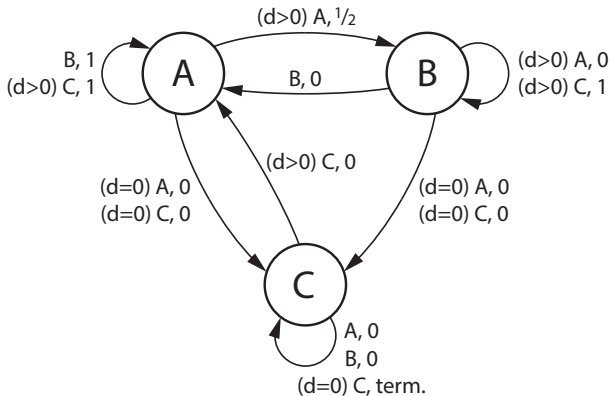
# Algorithm for non-rigid ASYNCH\* with termination



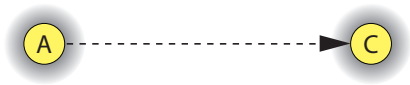
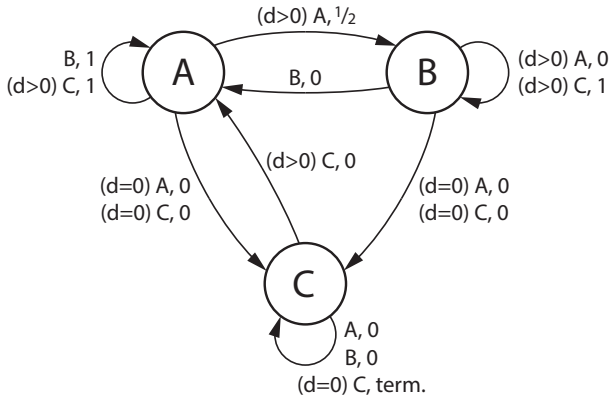
# Algorithm for non-rigid ASYNCH\* with termination



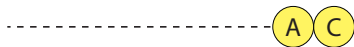
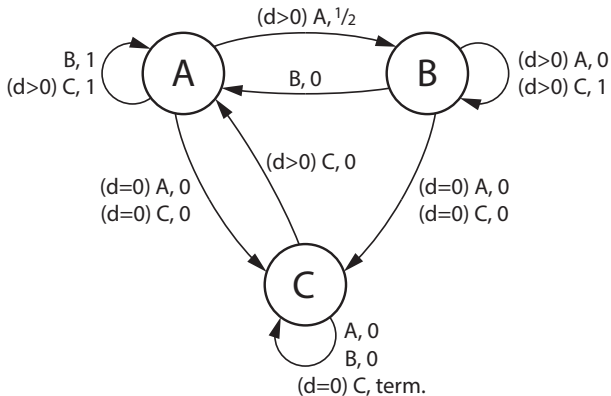
# Algorithm for non-rigid ASYNCH\* with termination



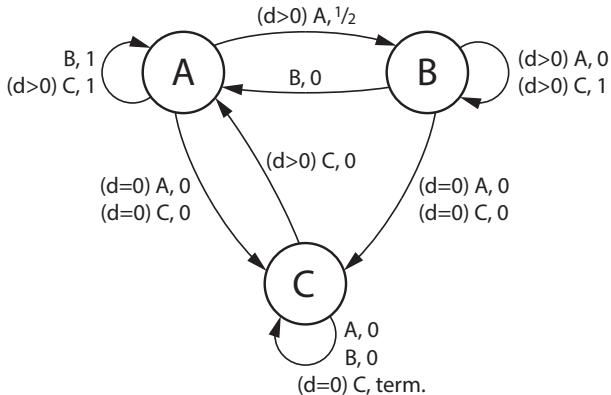
# Algorithm for non-rigid ASYNCH\* with termination



# Algorithm for non-rigid ASYNCH\* with termination

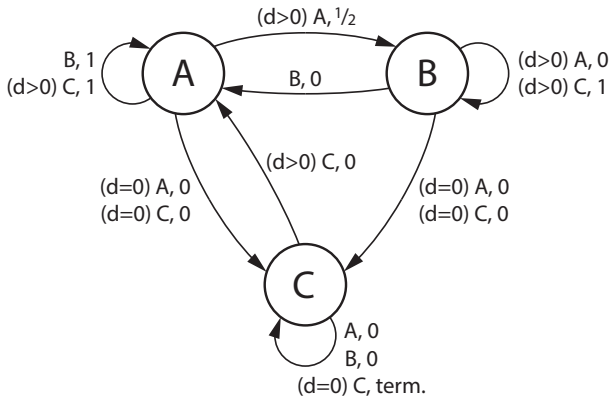


# Algorithm for non-rigid ASYNCH\* with termination



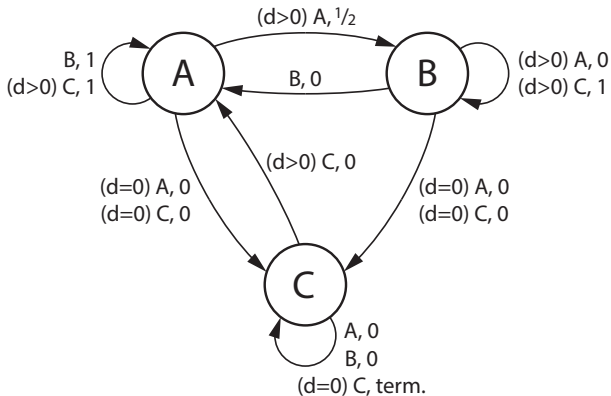


# Algorithm for non-rigid ASYNCH\* with termination



TC

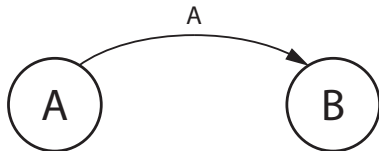
# Algorithm for non-rigid ASYNCH\* with termination



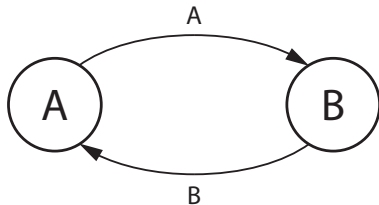
# Unsolvability in rigid $ASYNCH^*$ with two states



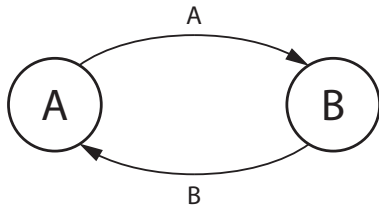
# Unsolvability in rigid $ASYNCH^*$ with two states



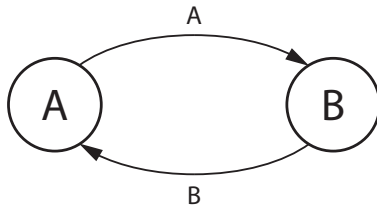
# Unsolvability in rigid $ASYNCH^*$ with two states



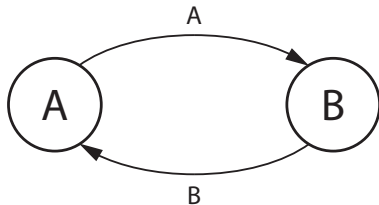
# Unsolvability in rigid $ASYNCH^*$ with two states



# Unsolvability in rigid $ASYNCH^*$ with two states

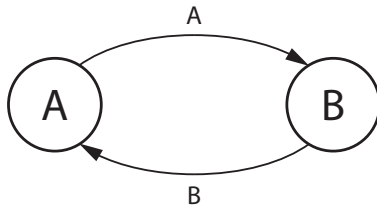


# Unsolvability in rigid $ASYNCH^*$ with two states

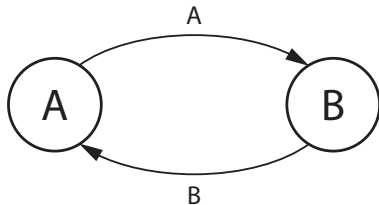




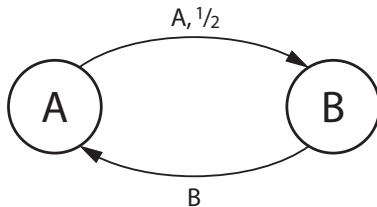
# Unsolvability in rigid $ASYNCH^*$ with two states



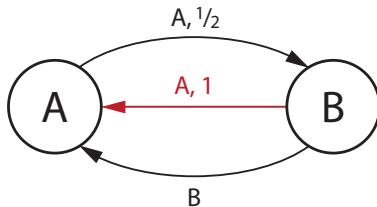
# Unsolvability in rigid $ASYNCH^*$ with two states



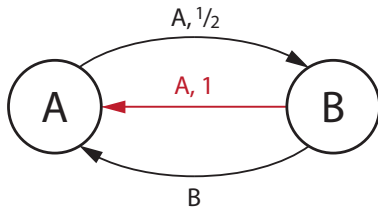
# Unsolvability in rigid $ASYNCH^*$ with two states



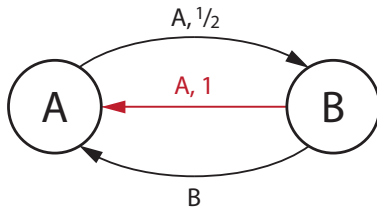
# Unsolvability in rigid $ASYNCH^*$ with two states



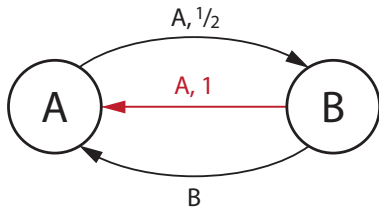
# Unsolvability in rigid $ASYNCH^*$ with two states



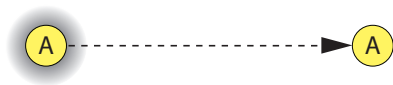
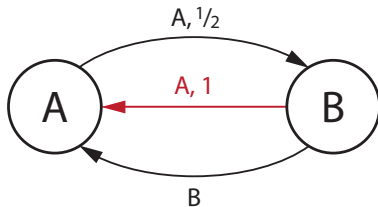
# Unsolvability in rigid $ASYNCH^*$ with two states



# Unsolvability in rigid $ASYNCH^*$ with two states

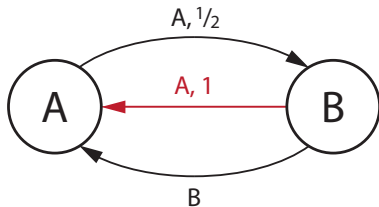


# Unsolvability in rigid $ASYNCH^*$ with two states

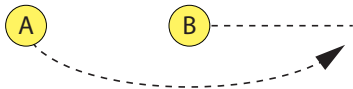
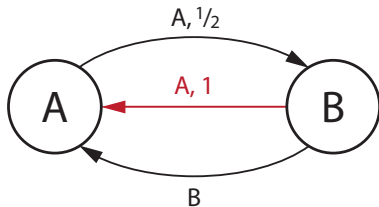




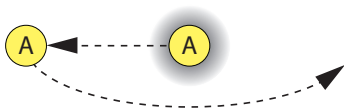
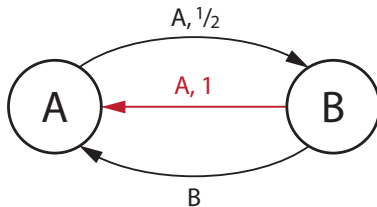
# Unsolvability in rigid $ASYNCH^*$ with two states



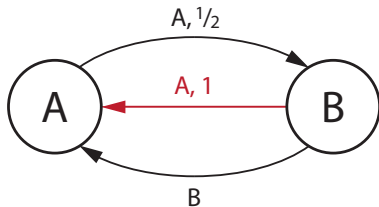
# Unsolvability in rigid $ASYNCH^*$ with two states



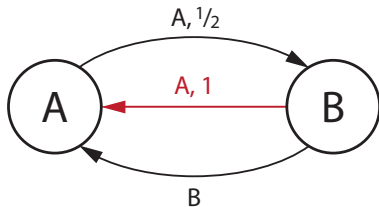
# Unsolvability in rigid $ASYNCH^*$ with two states



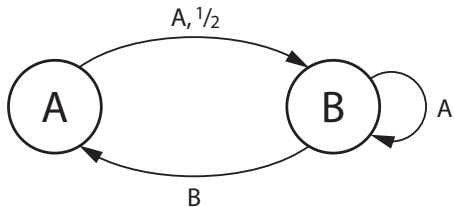
# Unsolvability in rigid $ASYNCH^*$ with two states



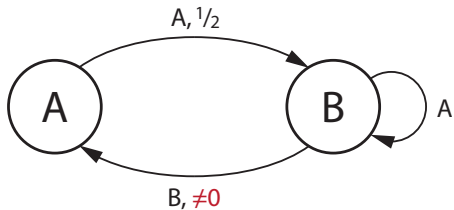
# Unsolvability in rigid $ASYNCH^*$ with two states



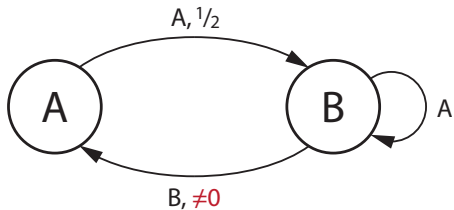
# Unsolvability in rigid $ASYNCH^*$ with two states



# Unsolvability in rigid $ASYNCH^*$ with two states

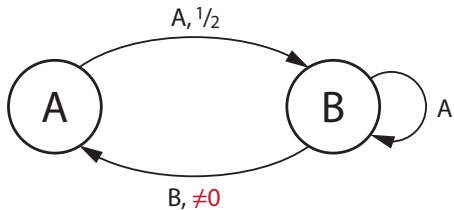


# Unsolvability in rigid $ASYNCH^*$ with two states

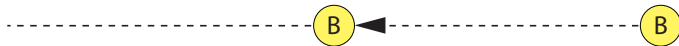
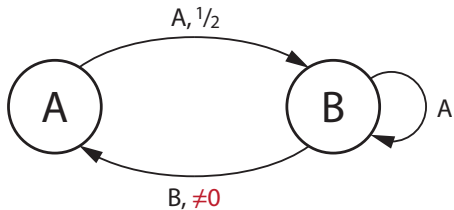




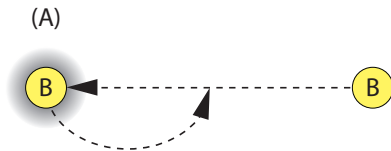
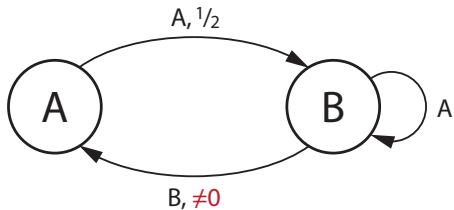
# Unsolvability in rigid $ASYNCH^*$ with two states



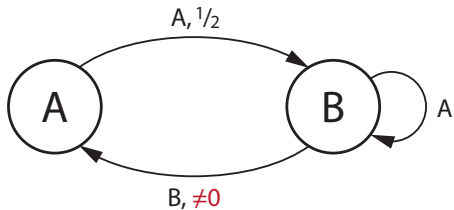
# Unsolvability in rigid $ASYNCH^*$ with two states



# Unsolvability in rigid $ASYNCH^*$ with two states



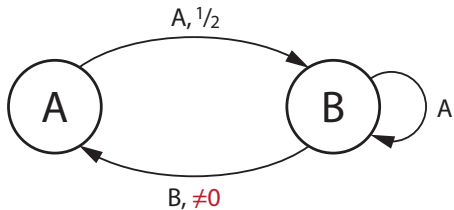
# Unsolvability in rigid $ASYNCH^*$ with two states



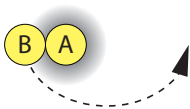
(A)



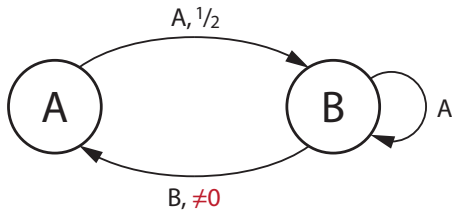
# Unsolvability in rigid $ASYNCH^*$ with two states



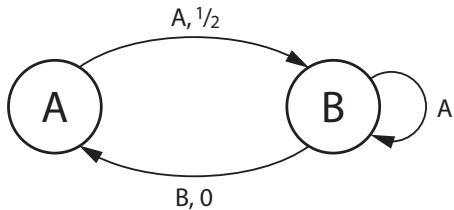
(A)



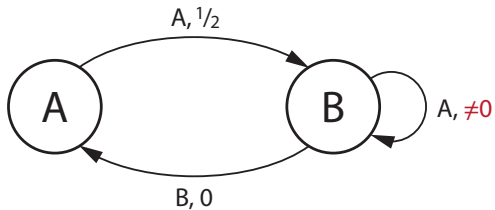
# Unsolvability in rigid $ASYNCH^*$ with two states



# Unsolvability in rigid $ASYNCH^*$ with two states

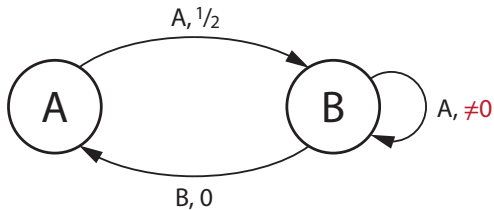


# Unsolvability in rigid $ASYNCH^*$ with two states

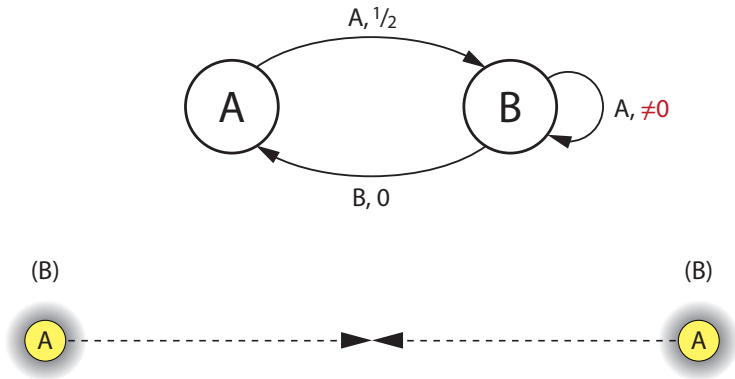




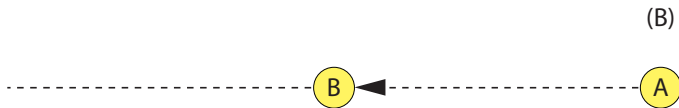
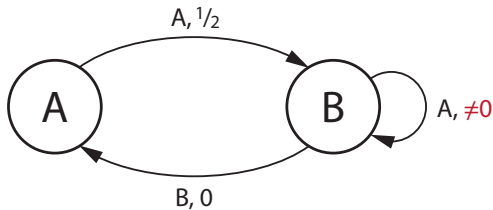
# Unsolvability in rigid $ASYNCH^*$ with two states



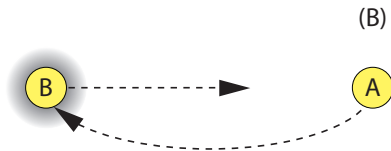
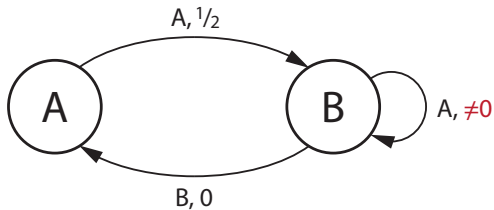
# Unsolvability in rigid $ASYNCH^*$ with two states



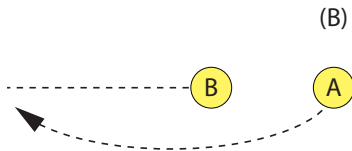
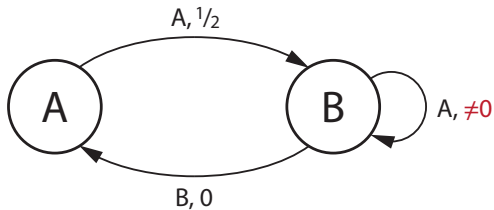
# Unsolvability in rigid $ASYNCH^*$ with two states



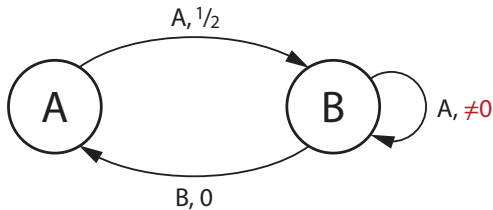
# Unsolvability in rigid $ASYNCH^*$ with two states



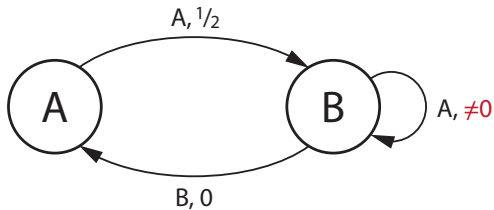
# Unsolvability in rigid $ASYNCH^*$ with two states



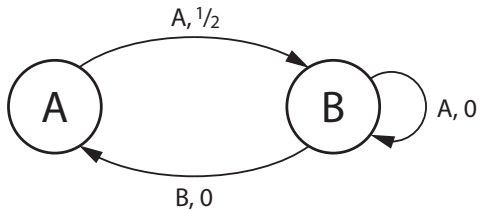
# Unsolvability in rigid $ASYNCH^*$ with two states



# Unsolvability in rigid $ASYNCH^*$ with two states

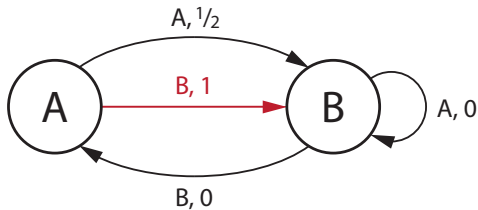


# Unsolvability in rigid $ASYNCH^*$ with two states

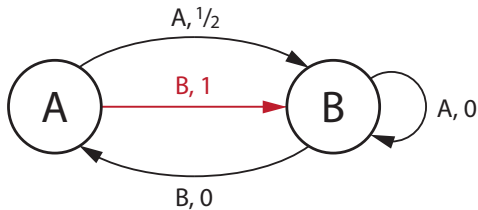




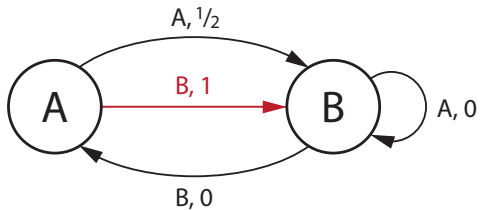
# Unsolvability in rigid $ASYNCH^*$ with two states



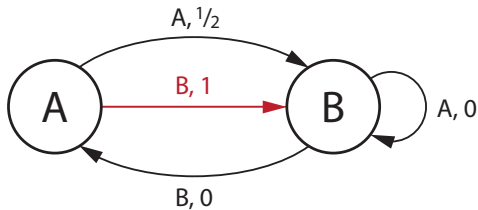
# Unsolvability in rigid $ASYNCH^*$ with two states



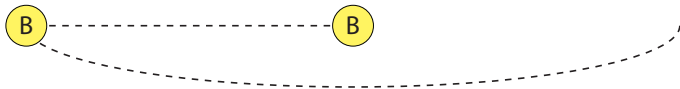
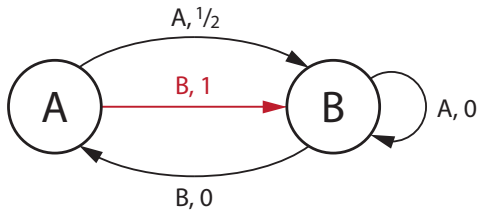
# Unsolvability in rigid $ASYNCH^*$ with two states



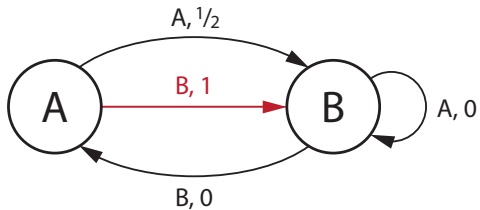
# Unsolvability in rigid $ASYNCH^*$ with two states



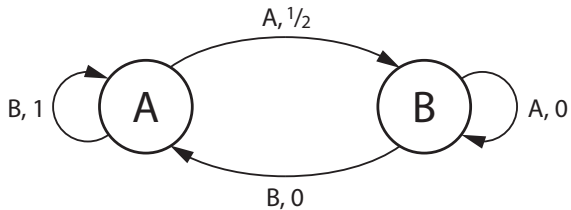
# Unsolvability in rigid $ASYNCH^*$ with two states



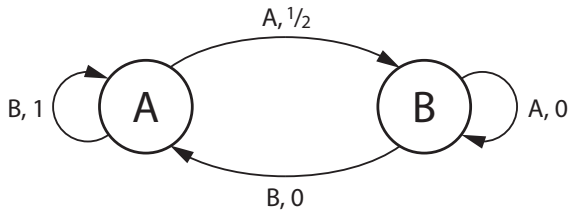
# Unsolvability in rigid $ASYNCH^*$ with two states



# Unsolvability in rigid $ASYNCH^*$ with two states

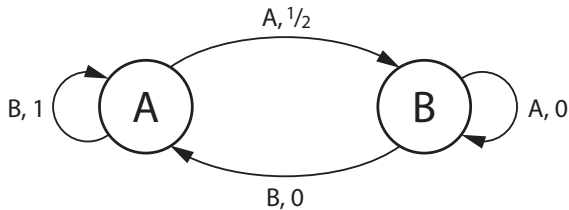


# Unsolvability in rigid $ASYNCH^*$ with two states





# Unsolvability in rigid $ASYNCH^*$ with two states



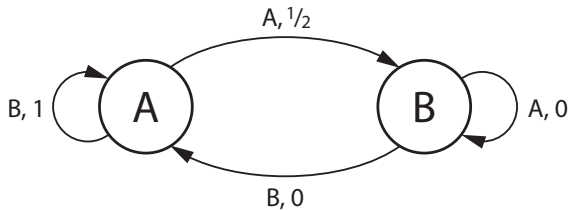
(A)



(A)



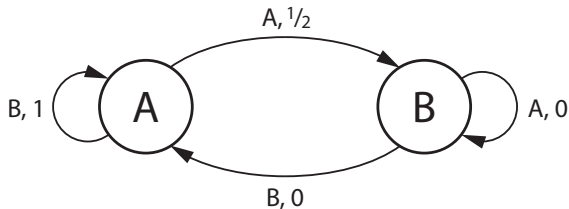
# Unsolvability in rigid $ASYNCH^*$ with two states



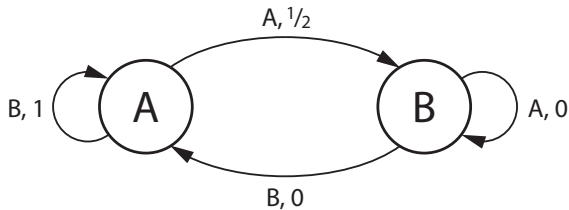
(A)



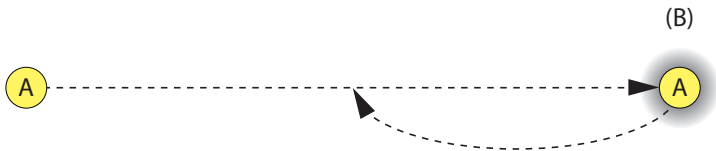
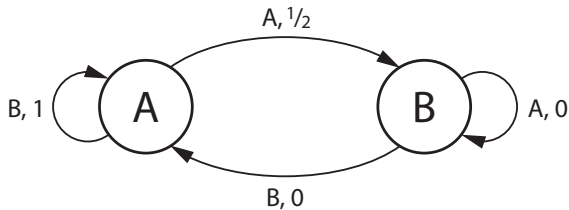
# Unsolvability in rigid $ASYNCH^*$ with two states



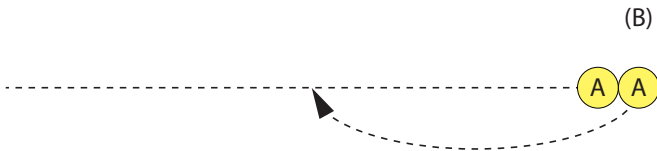
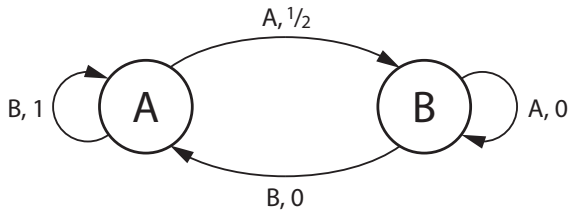
# Unsolvability in rigid $ASYNCH^*$ with two states



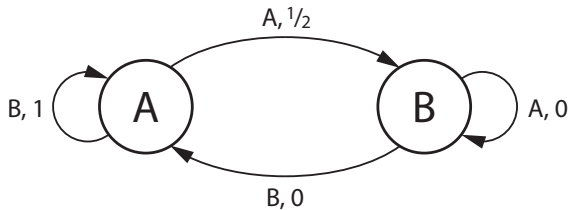
# Unsolvability in rigid $ASYNCH^*$ with two states



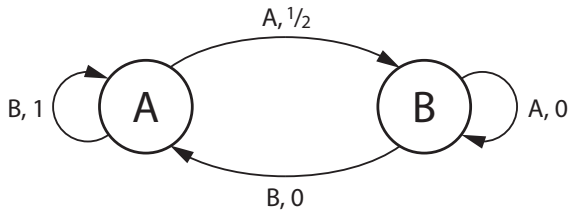
# Unsolvability in rigid $ASYNCH^*$ with two states



# Unsolvability in rigid $ASYNCH^*$ with two states



# Unsolvability in rigid $ASYNCH^*$ with two states

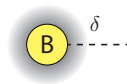
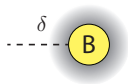




# Unsolvability in non-rigid ASYNCH with two states



# Unsolvability in non-rigid ASYNCH with two states



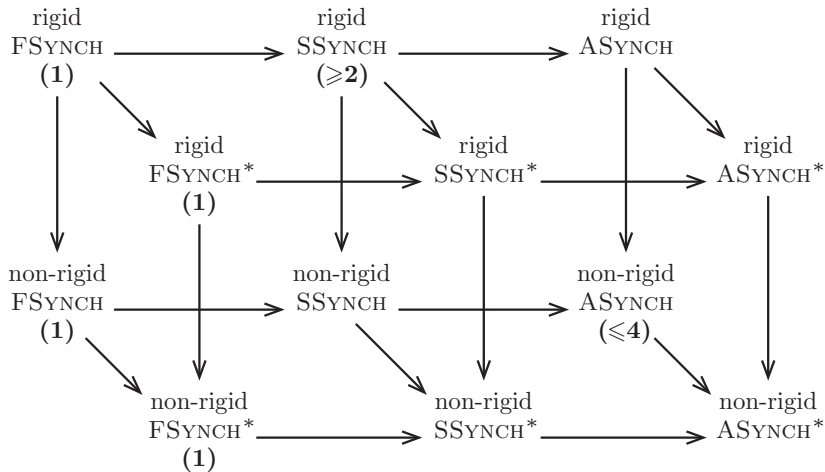
# Unsolvability in non-rigid ASYNCH with two states



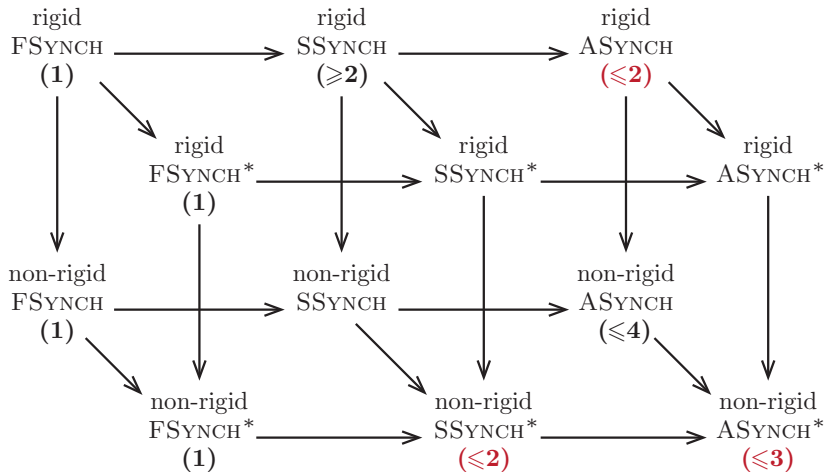
# Unsolvability in non-rigid ASYNCH with two states



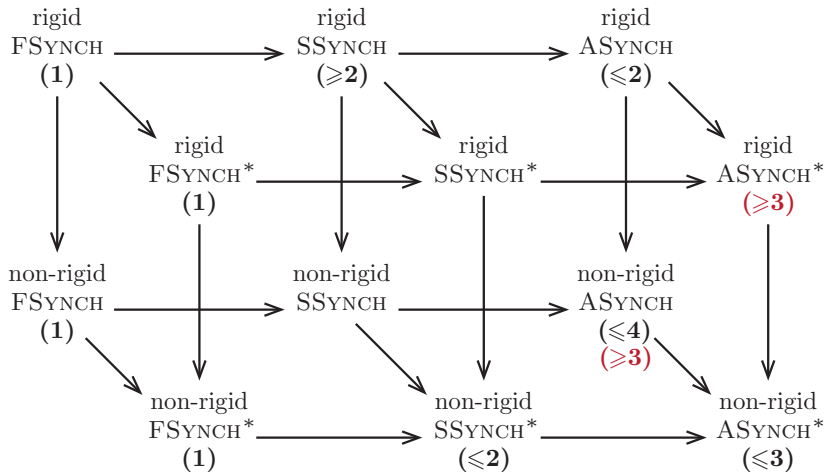
# Conclusion



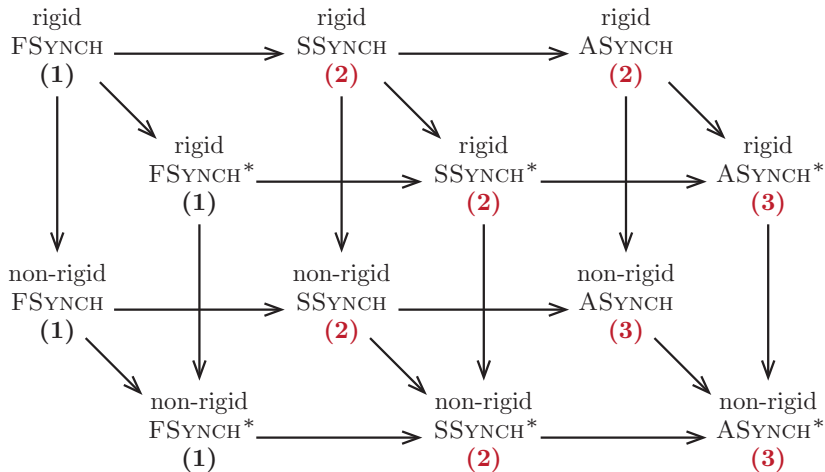
# Conclusion



# Conclusion

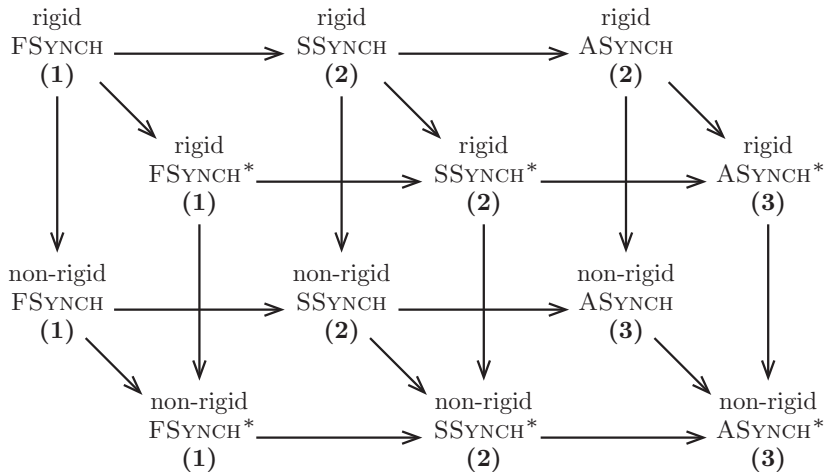


# Conclusion









# Conclusion



# References

-  S. Das, P. Flocchini, G. Prencipe, N. Santoro, M. Yamashita.  
The power of lights: synchronizing asynchronous robots using visible bits.  
In *Proceedings of ICDCS 2012*, pp. 506–515.
-  P. Flocchini, G. Prencipe, N. Santoro.  
Distributed computing by oblivious mobile robots.  
*Morgan & Claypool*, 2012.
-  P. Flocchini, N. Santoro, G. Viglietta, M. Yamashita.  
Rendezvous of two robots with constant memory.  
To appear in *Proceedings of SIROCCO 2013*.
-  I. Suzuki, M. Yamashita.  
Distributed anonymous mobile robots: formation of geometric patterns.  
*SIAM Journal on Computing*, vol. 28, pp. 1347–1363, 1999.