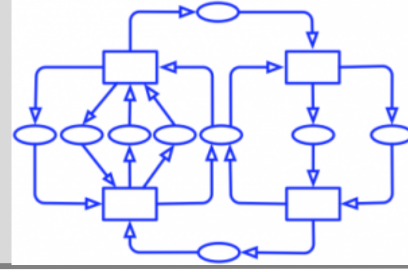
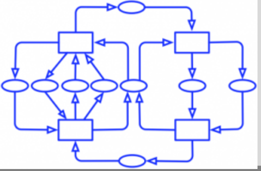


PNRD and iPNRD Integration Assisting Adaptive Control in Block World Domain

José Jean-Paul Zanlucchi de Souza Tavares
Gabriel de Almeida Souza



Summary

1 – Introduction

2 – PNRD and iPNRD Approach

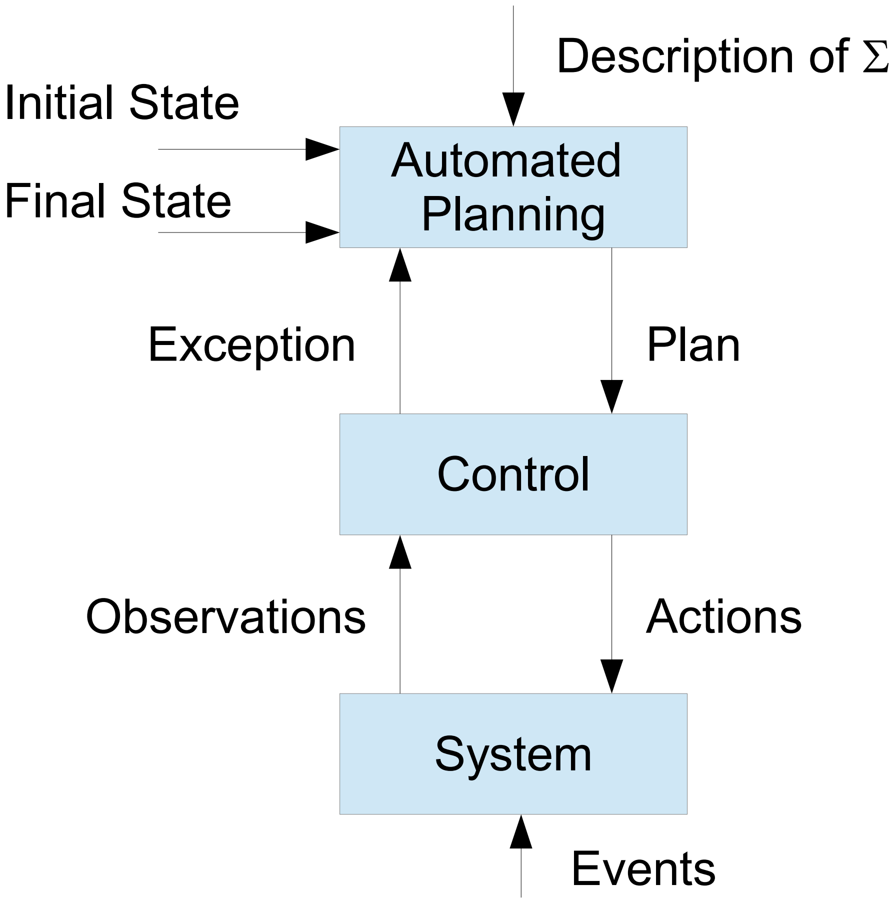
3 – PNRD and iPNRD Integration with Petri Space

4 – Robotic Adaptive Control Based on PNRD and iPNRD

5 – Conclusions and Further Works

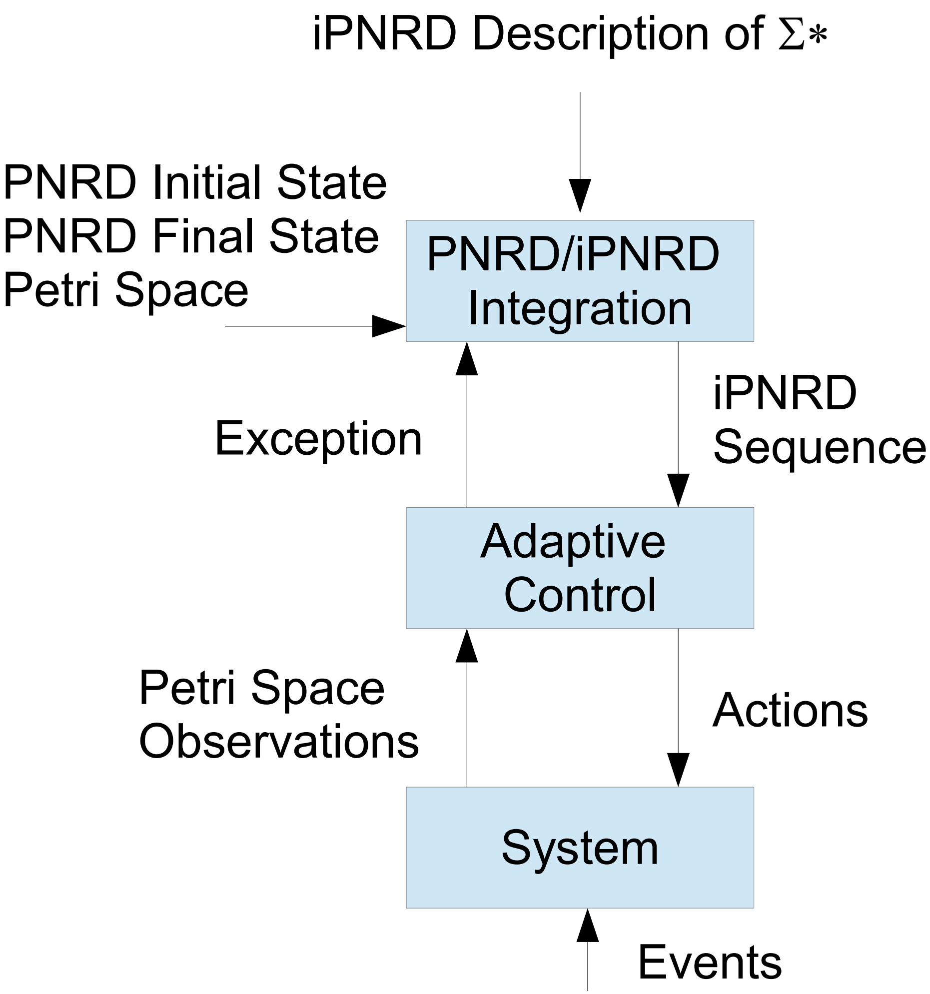
Automated Planning Conceptual Model

- Models are too abstract
- Researchers focus on improving performance of search algorithms more than on improving models' predictive capabilities
- Planning is formalized, whereas acting is not
- There is a clear gap between planning and performing, while the borderline between acting and performing is more blurred
- Total or partial observed
- Deterministic or non-deterministic
- Online process redesign and replanning are open issues



Introduction

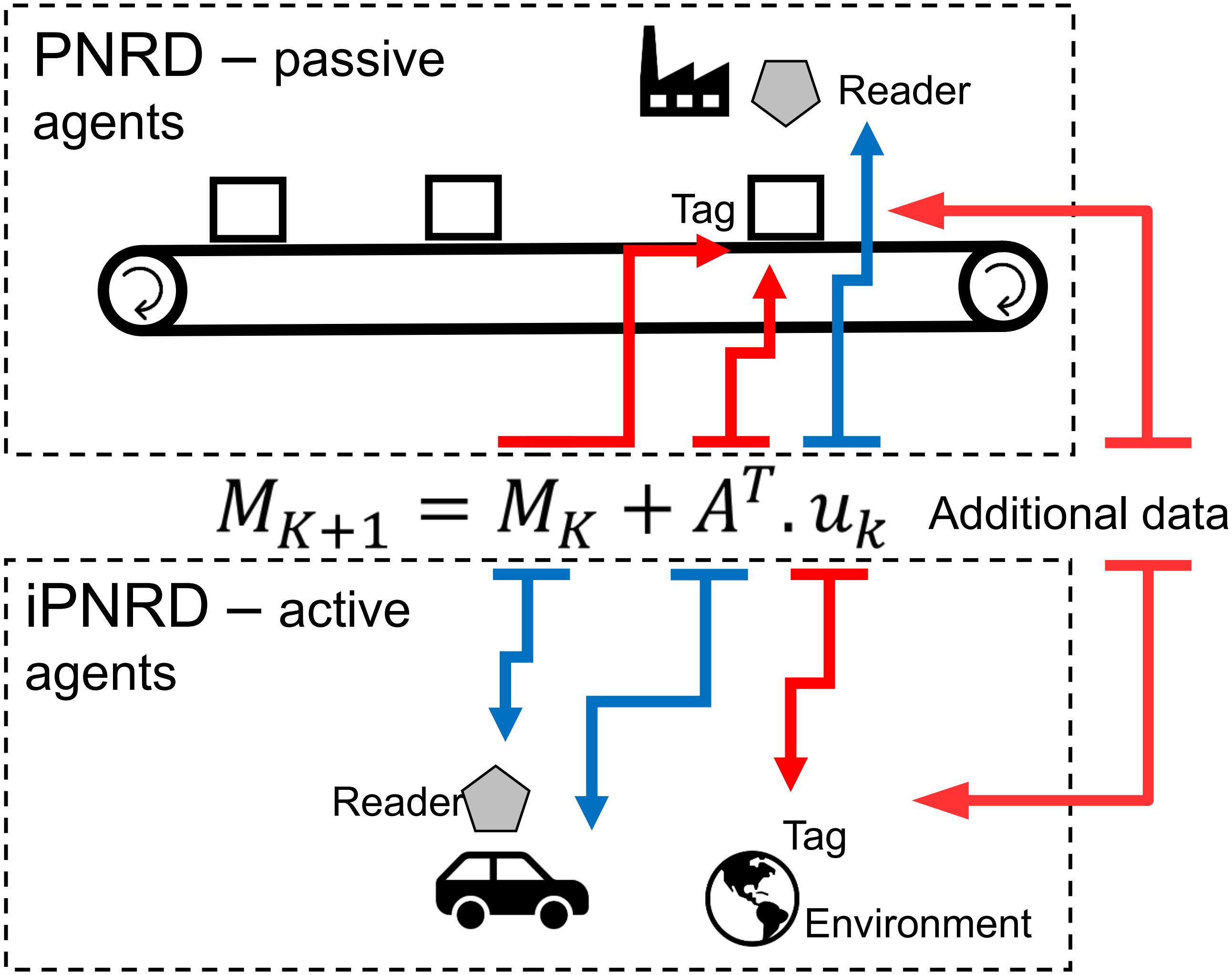
Proposition



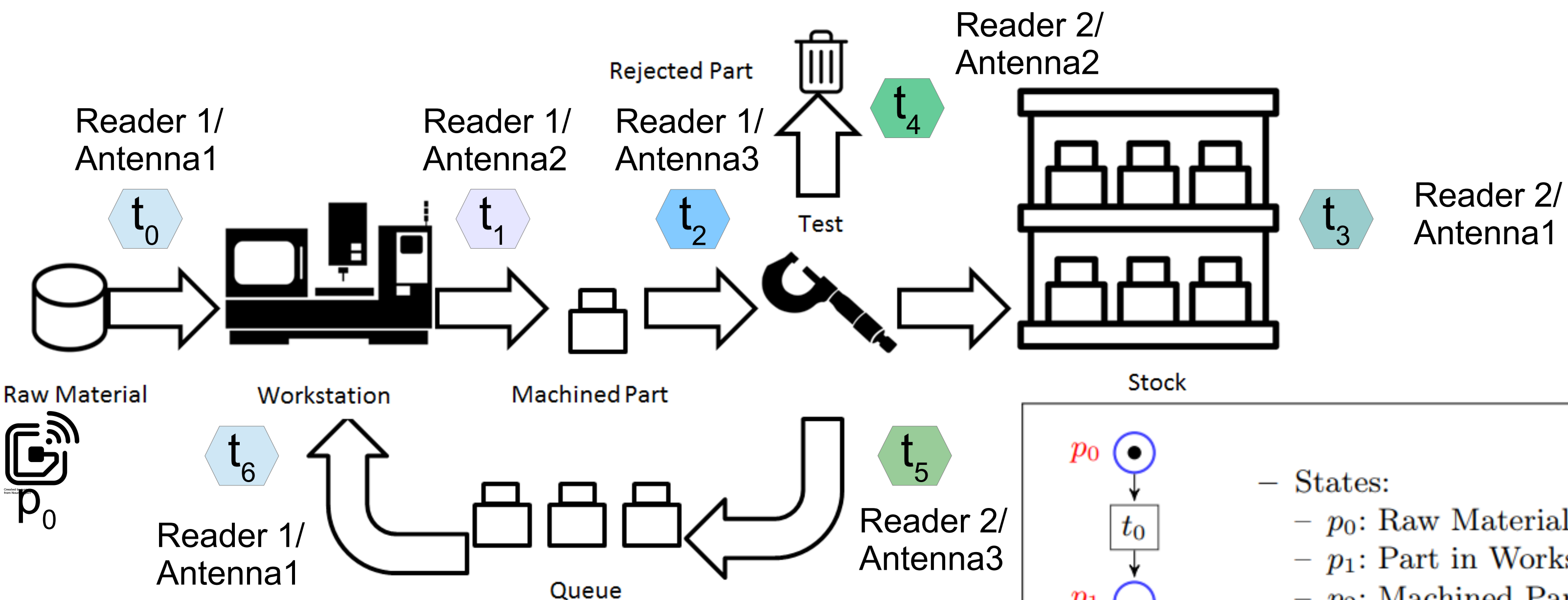
- iPNRD can describe discret event systems, as the example of 3 Blocks in Blocks World domain*
- PNRD is able to identify initial and final state
- This presentation shows an example of how is possible to generate an adative control based on PNRD/iPNRD integration
- Some issues request integration with physical positioning, thus a Petri Space observation is required
- Observations are able to detect non-deterministics events
- Systems can deal with RFID reader partial observation

PNRD and iPNRD

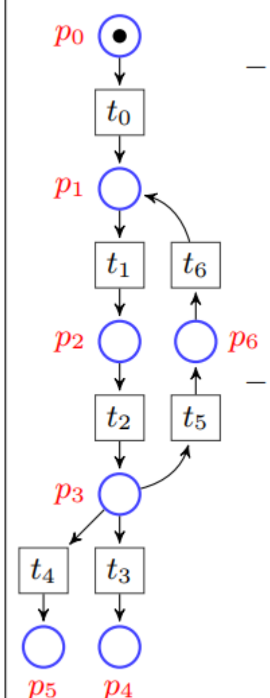
- Safe Petri Net
- Bridge between PN and RFID (Computational and Physical world)
- Only stand alone examples



PNRD Example

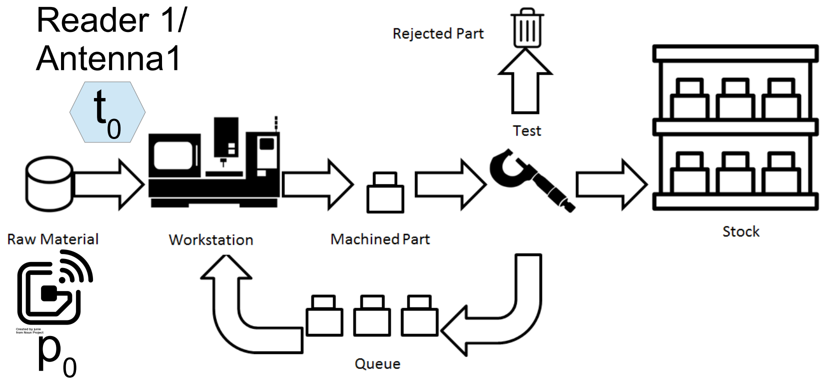


$$A^t = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$



- States:
  - p_0: Raw Material
  - p_1: Part in Workstation
  - p_2: Machined Part
  - p_3: Part in Test
  - p_4: Part in Stock
  - p_5: Rejected Part
  - p_6: Part in Queue
- Transitions and RFID readers:
  - t_0: Reader1/Antenna1
  - t_1: Reader1/Antenna2
  - t_2: Reader1/Antenna3
  - t_3: Reader2/Antenna1
  - t_4: Reader2/Antenna2
  - t_5: Reader2/Antenna3
  - t_6: Reader1/Antenna1

PNRD Example

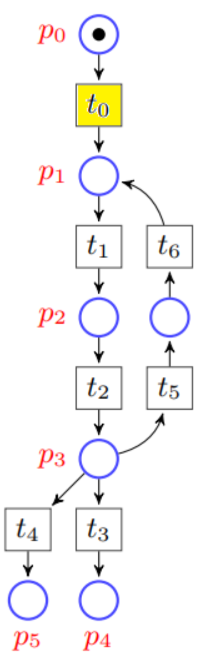


PNRD next state calculus

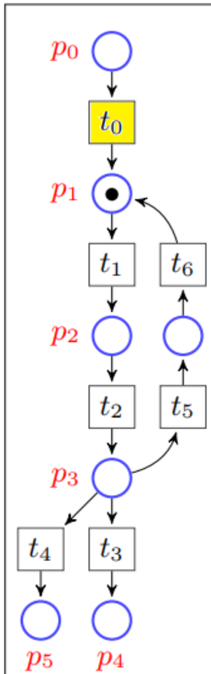
$$m_1 = m_0 + A^t.u_0$$

$$m_1 = p_0 + A^t.t_0$$

$$m_1 = p_1$$

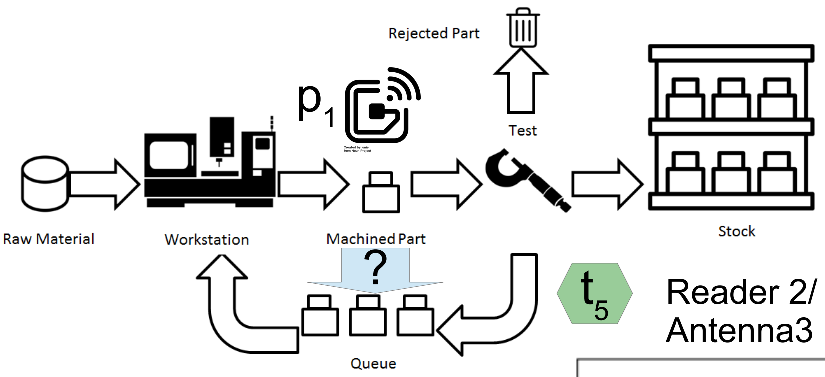


- States:
  - p_0: Raw Material
  - p_1: Part in Workstation
  - p_2: Machined Part
  - p_3: Part in Test
  - p_4: Part in Stock
  - p_5: Rejected Part
  - p_6: Part in Queue
- Transitions and RFID readers:
  - t_0: Reader1/Antenna1
  - t_1: Reader1/Antenna2
  - t_2: Reader1/Antenna3
  - t_3: Reader2/Antenna1
  - t_4: Reader2/Antenna2
  - t_5: Reader2/Antenna3
  - t_6: Reader1/Antenna1



- States:
  - p_0: Raw Material
  - p_1: Part in Workstation
  - p_2: Machined Part
  - p_3: Part in Test
  - p_4: Part in Stock
  - p_5: Rejected Part
  - p_6: Part in Queue
- Transitions and RFID readers:
  - t_0: Reader1/Antenna1
  - t_1: Reader1/Antenna2
  - t_2: Reader1/Antenna3
  - t_3: Reader2/Antenna1
  - t_4: Reader2/Antenna2
  - t_5: Reader2/Antenna3
  - t_6: Reader1/Antenna1

PNRD Example

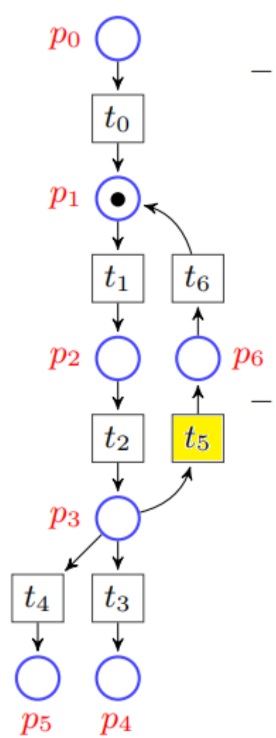


PNRD Exception

$m_2 = m_1 + A^t.u_1$

$m_1 = p_1 + A^t.t_5$

$m_1 = [0,1,0,-1,0,0,1]^t$ Exception!



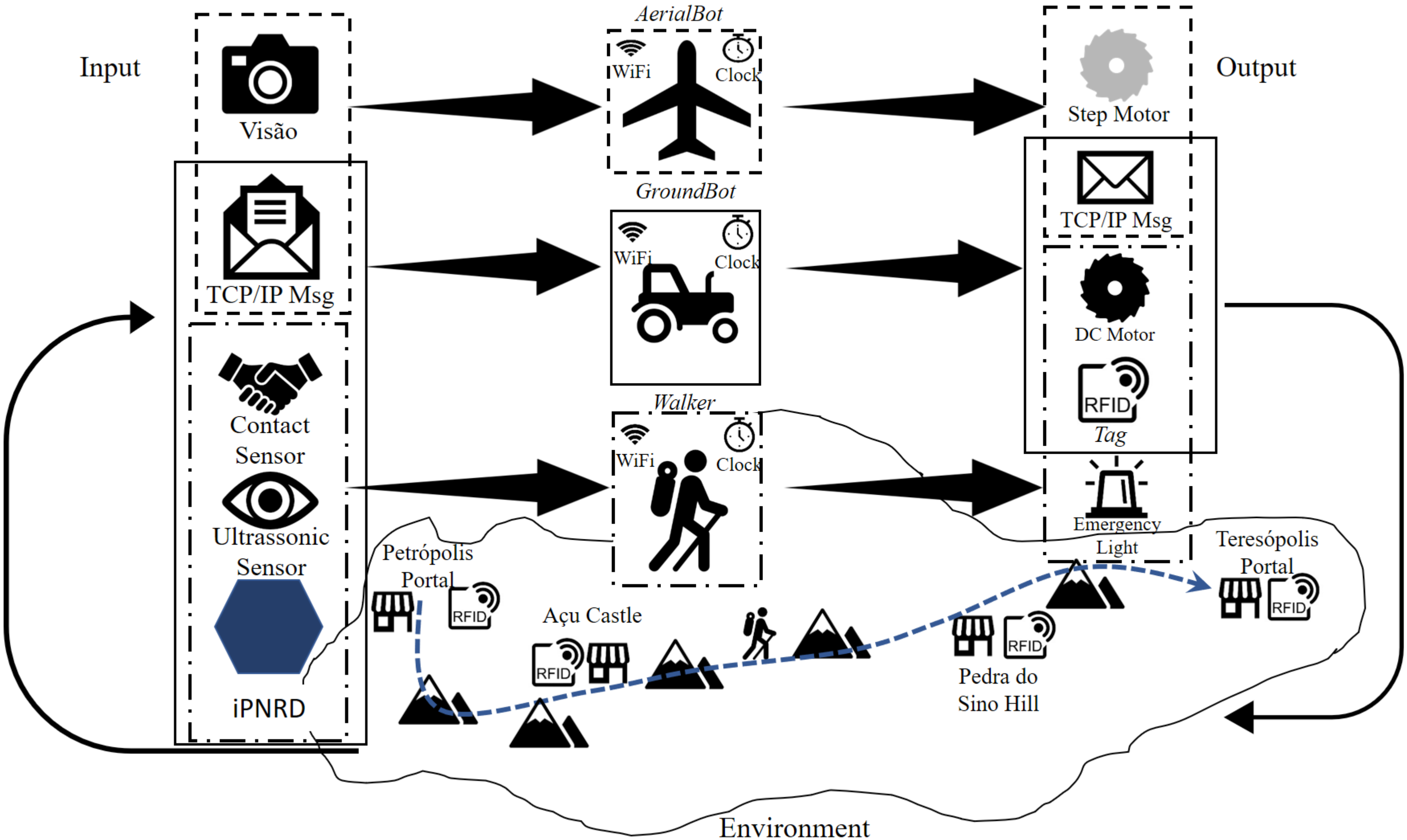
- States:
  - p_0: Raw Material
  - p_1: Part in Workstation
  - p_2: Machined Part
  - p_3: Part in Test
  - p_4: Part in Stock
  - p_5: Rejected Part
  - p_6: Part in Queue
- Transitions and RFID readers:
  - t_0: Reader1/Antenna1
  - t_1: Reader1/Antenna2
  - t_2: Reader1/Antenna3
  - t_3: Reader2/Antenna1
  - t_4: Reader2/Antenna2
  - t_5: Reader2/Antenna3
  - t_6: Reader1/Antenna1

iPNRD Example

Petrópolis Teresópolis Trail

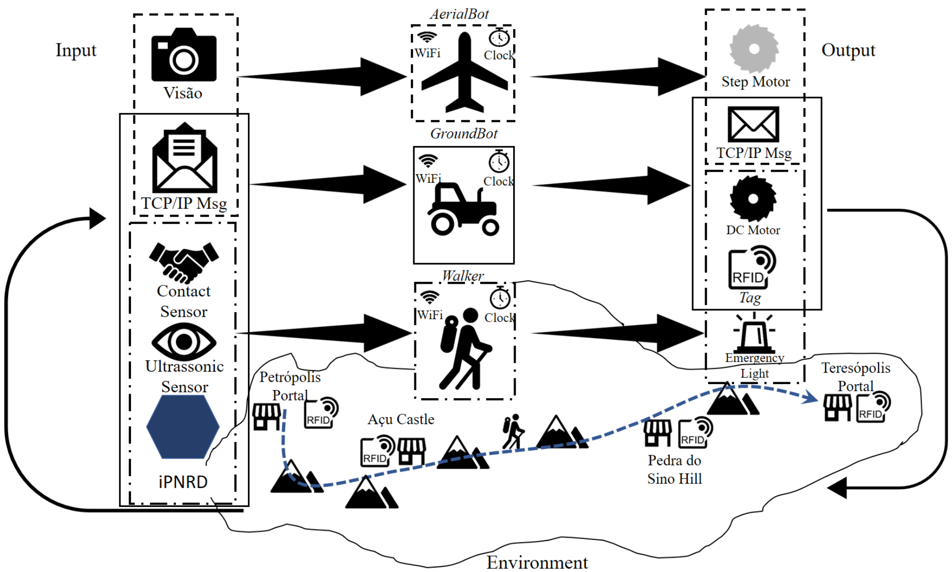
iPNRD Example

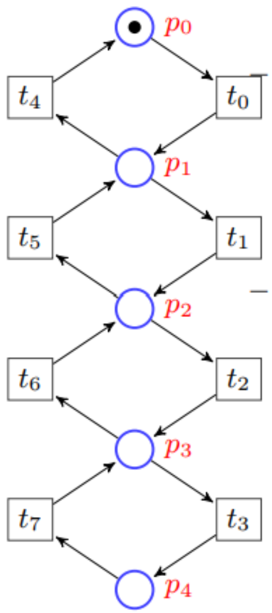
Search and Rescue Environment



iPNRD Example

Search and Rescue Environment

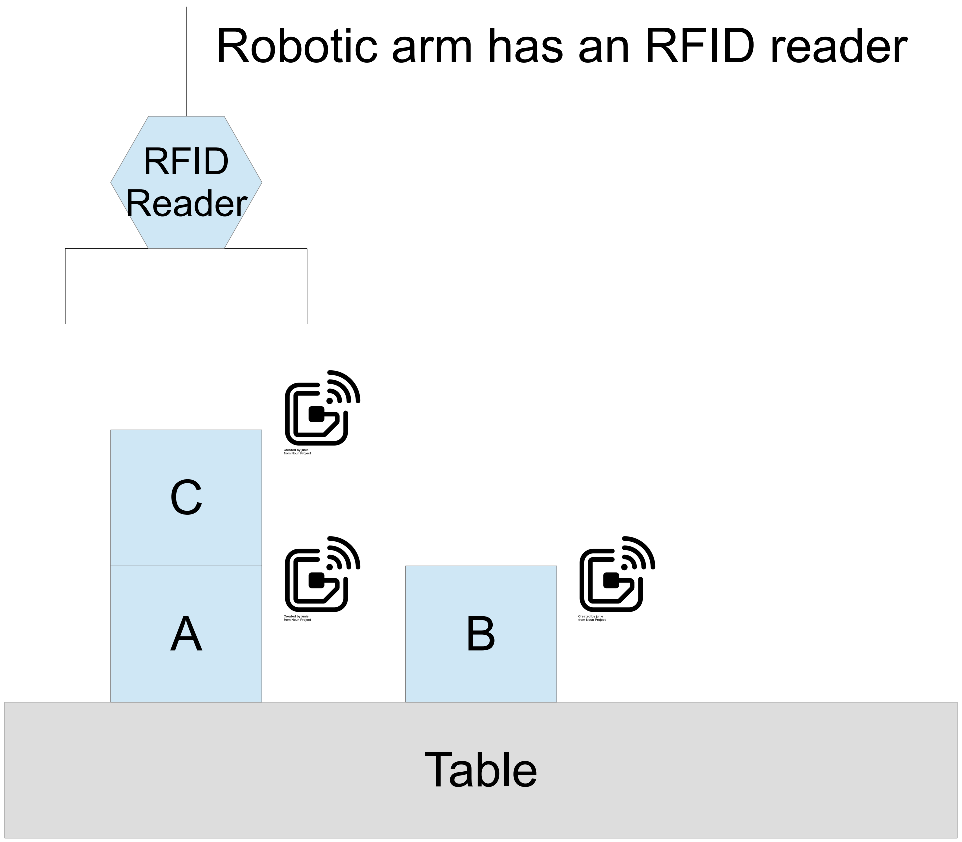




- States:
  - p_0: Walker at Petropolis Entrance
  - p_1: Walker at First Region
  - p_2: Walker at Second Region
  - p_3: Walker at Third region
  - p_4: Walker Teresopolis Entrance
- Transitions and tags:
  - t_0: Tag 1 at Petropolis Entrance
  - t_1: Tag 2 at "Castelo do Açu"
  - t_2: Tag 3 at "Pedra do Sino"
  - t_3: Tag 4 at Teresopolis Exit
  - t_4: Tag 1 at Petropolis Exit
  - t_5: Tag 2 at "Castelo do Açu"
  - t_6: Tag 3 at "Pedra do Sino"
  - t_7: Tag 4 at Teresopolis Entrance

$$A^t = \begin{bmatrix} -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \end{bmatrix}$$

Blocks Worlds of 3 Components Prerequisites

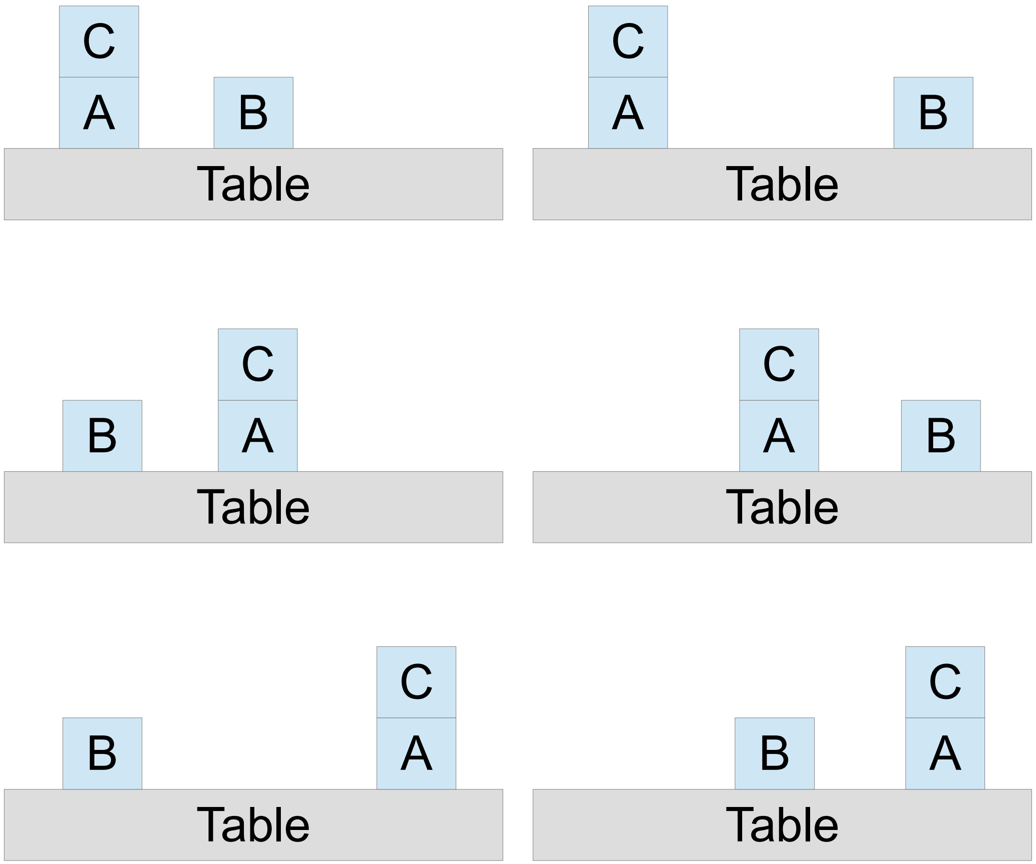


Each block is tagged in the top

- The goal is to move blocks from the initial state to the desired one
- Only one block may be moved at a time
- Only free blocks can be moved
- A physical model is needed to place the blocks and enable motion control => Petri Space
- The logical representation (PNRD) must be coherent whith the physical positioning (Petri Space)
- Notation:
  i. *XY* (*X* is over *Y*)

  ii. *X_Y* (*X* and *Y* is over the table in distinct columns)
- Optimal plan is NP-hard

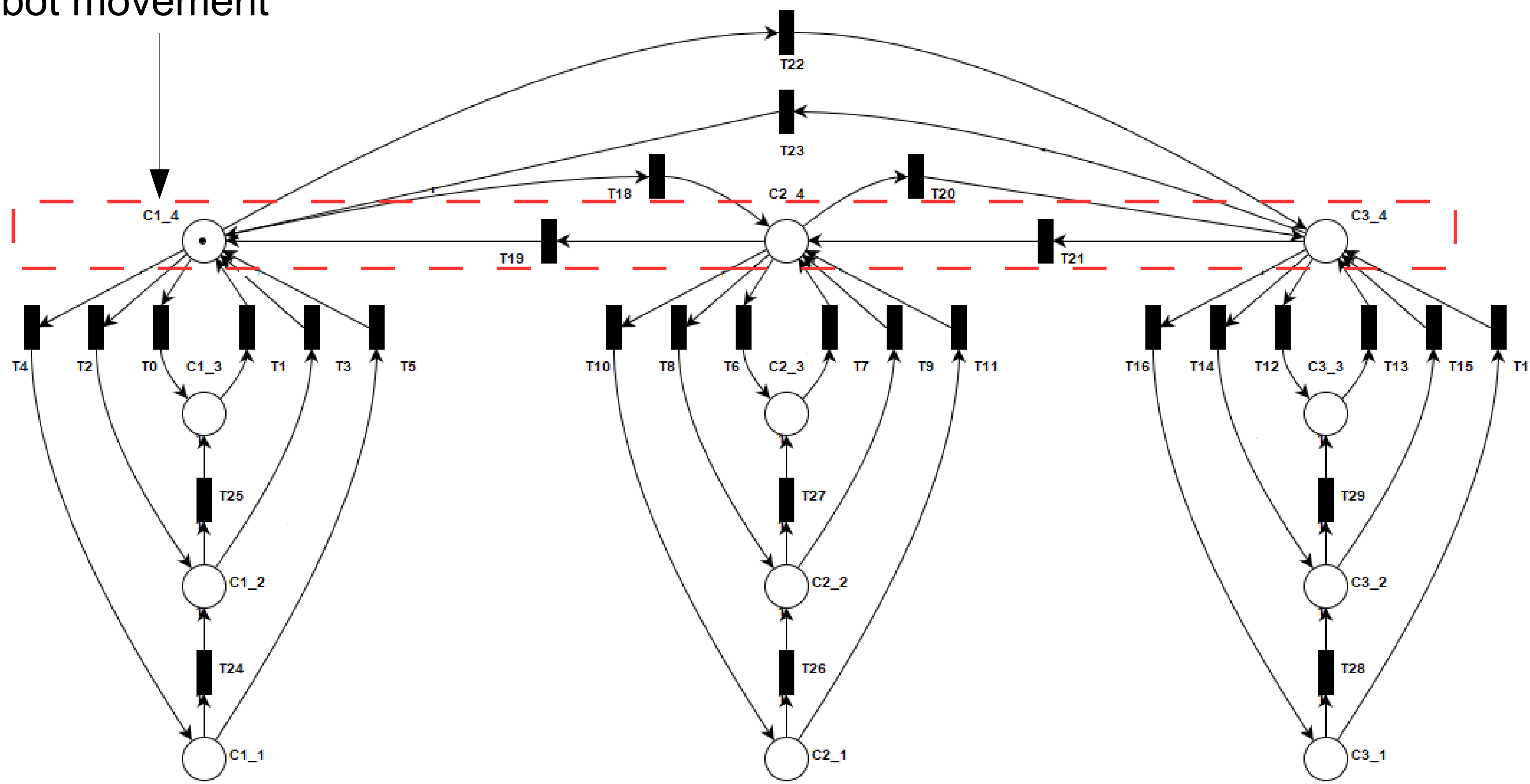
Blocks Worlds of 3 Components: Physical Ambiguity



- Example: *CA_B*
  i. There are 6 distincts physical configuration for 3 predefined columns
- *A_B_C* has 3! distincts column positioning while ACB has 3!/2!.
- There are 60 possible positions
- To deal with physical ambiguity, each block must be placed on the table in a predetermined column position
  i. This reduces the total number of physical position to 13, so PNRD/iPNRD is able to have full representation of this case

Blocks Worlds of 3 Components: Petri Space represents robot physical movement

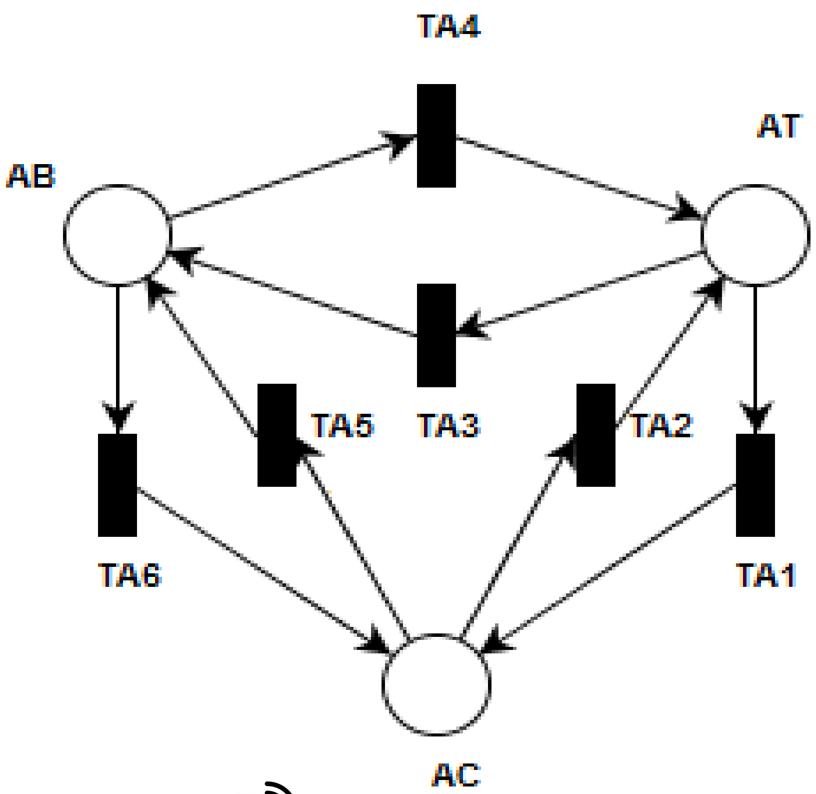
These places was included in order to avoid colision with another block during robot movement



Each place is related to a predefined position in the 3 predefined columns
Each transition is related to a specific movement (point-to-point)

PNRD & iPNRD Integration

Blocks Worlds of 3 Components: PNRD Model for Block *A*



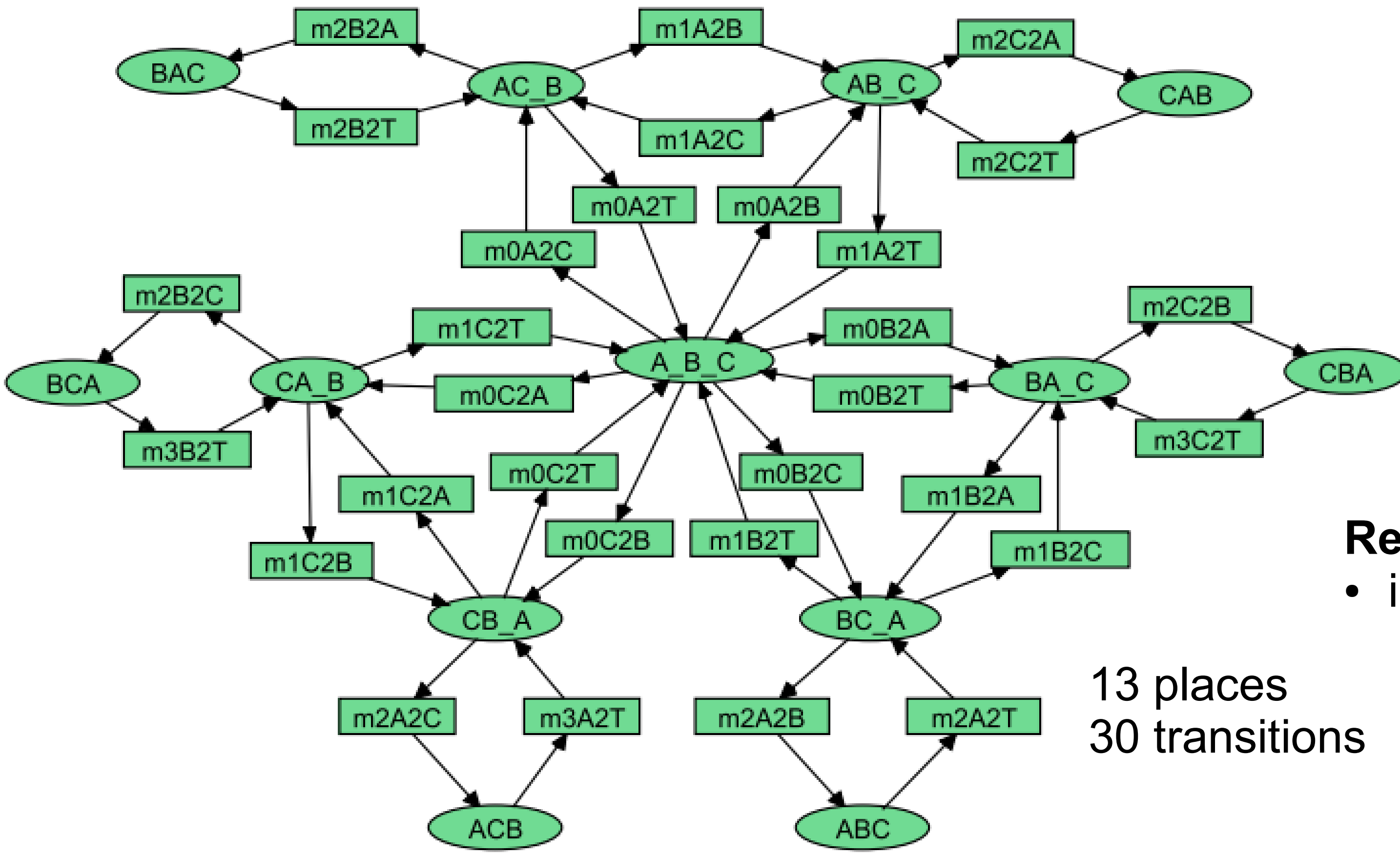
Block

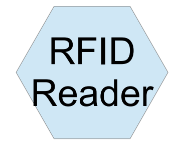
Tag Data:
- PNRD A^t model
- Block Initial Marking
- Block Final Marking

- This representation is simplified and coherent with robotic arm movement in the Petri Space
- The PNRD model identifies if *A* is over the Table (*AT*), or on top of another block (*AB* or *AC*)
- There are no reference regarding robotic arm claw in this model
- PNRD additional data stores the final state
- Another block PNRD model is similar

PNRD & iPNRD Integration

Blocks Worlds of 3 Components: iPNRD Model





Reader Data:

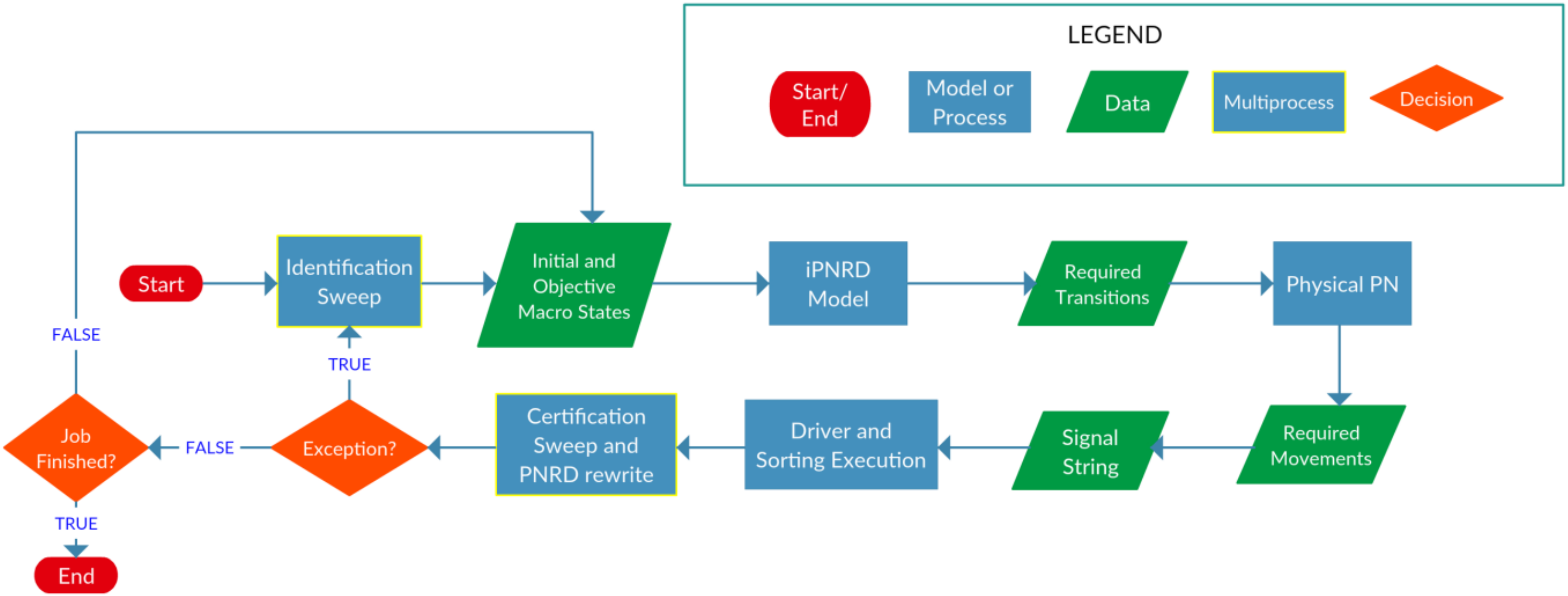
- iPNRD A^t model

13 places
30 transitions

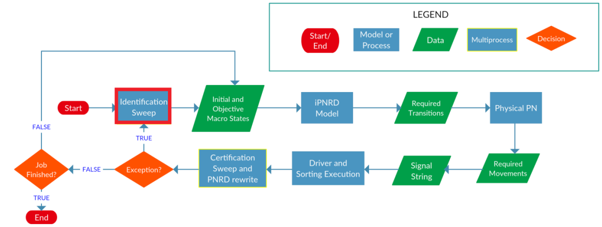
miX2Y means movement number *i* of block *X* to position *Y* (on table *T* or another block)

Robotic Adaptive Control

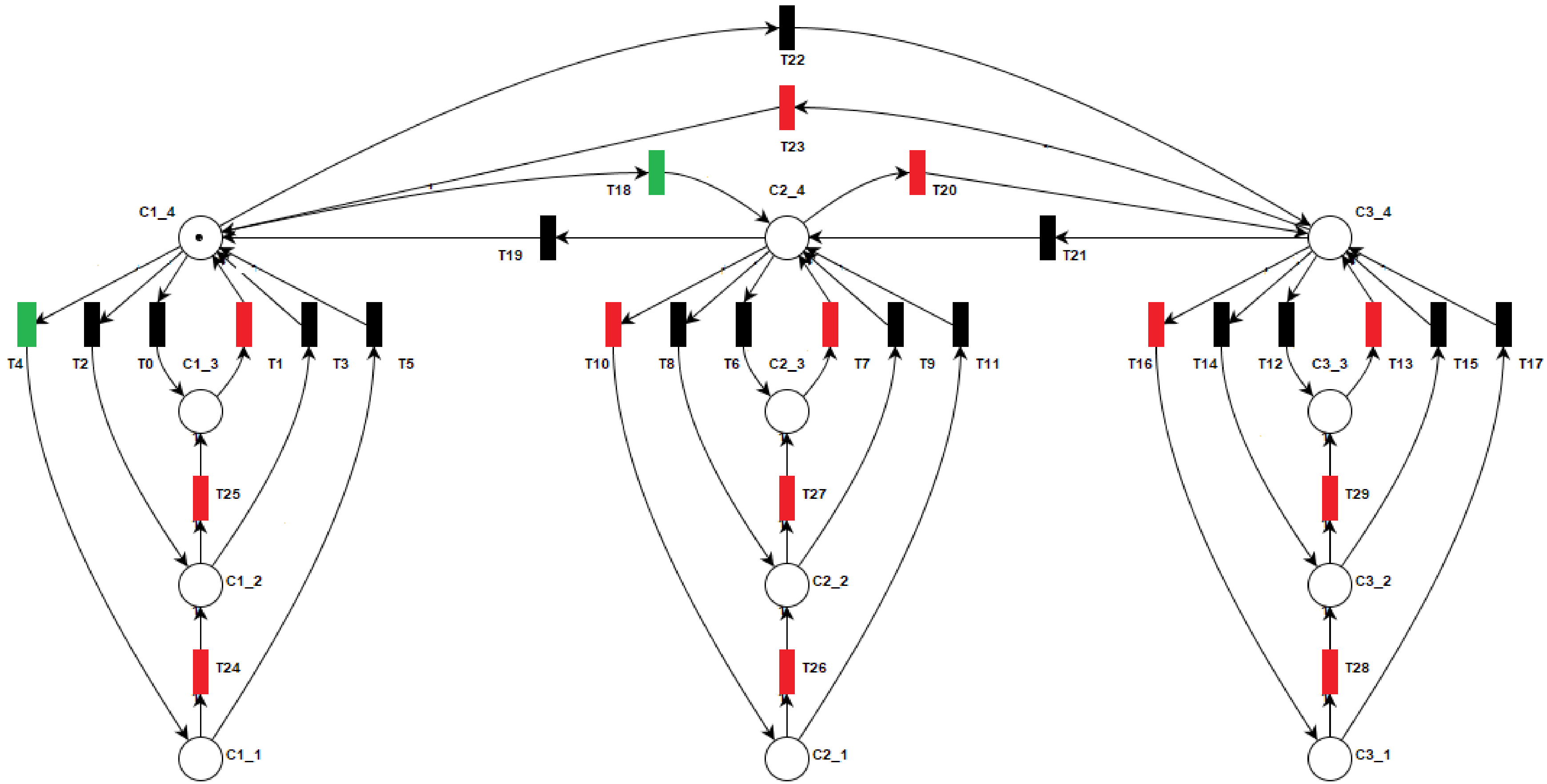
Main Process



Robotic Adaptive Control



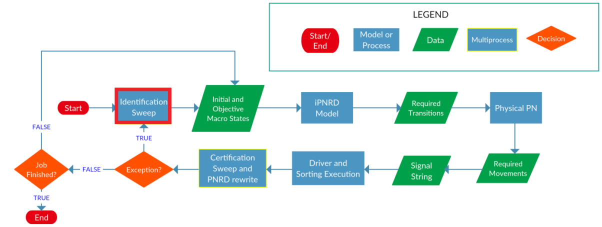
Fixed Initial Identification Sweep



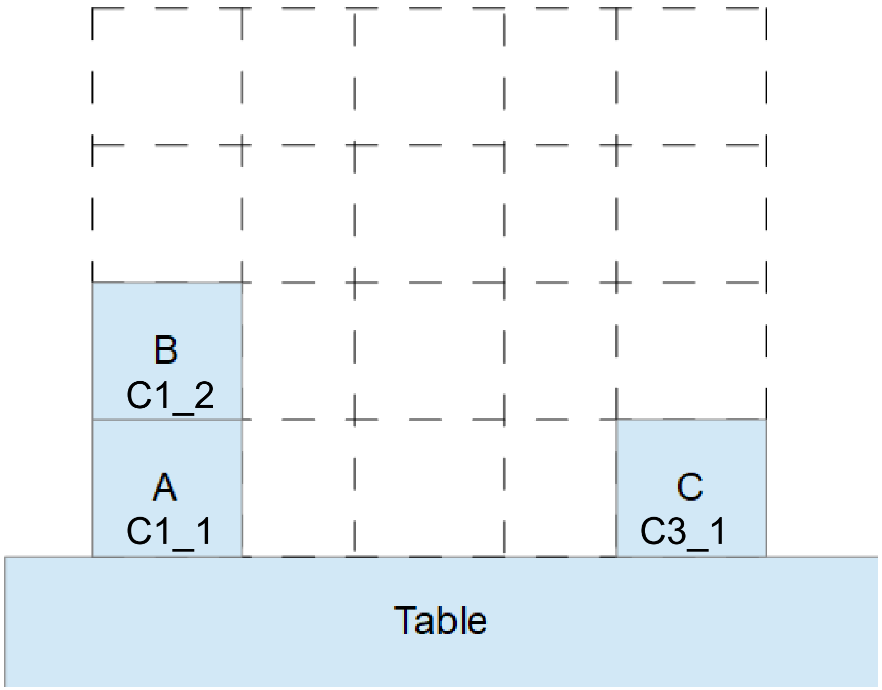
T4 → T24 → T25 → T1 → T18 → T10 → T26 → T27 → T7 → T20 → T16 → T28 → T29 → T13 → T23

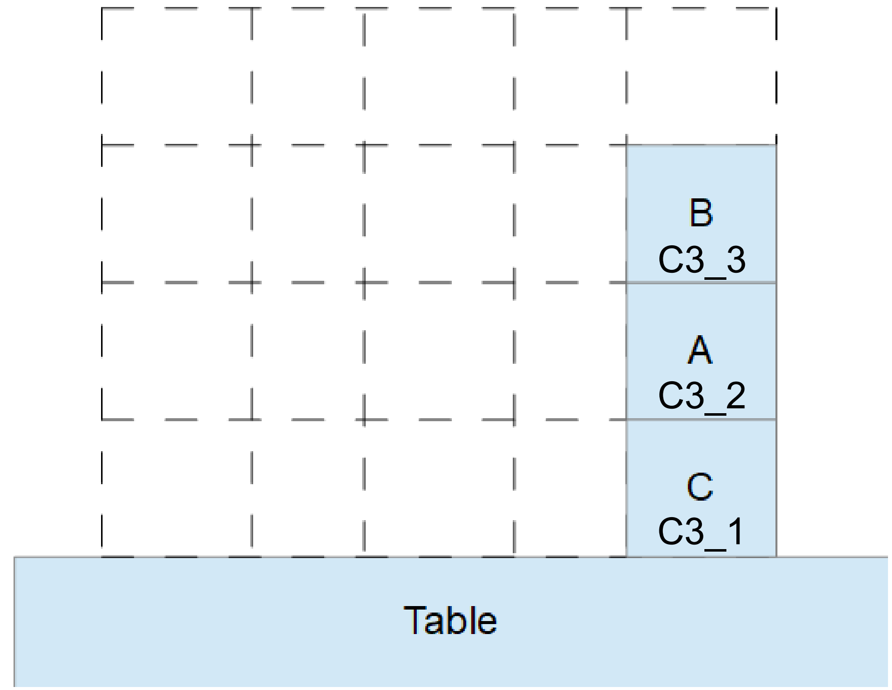
Can be optimized!

Robotic Adaptive Control



Example of Identification Sweep Result



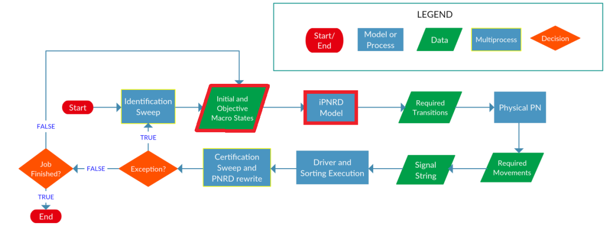


Petri Space: {C1_1,C1_2,C3_1}
PNRD initial marking set {*AT*,*BA*,*CT*}

PNRD final marking set {*AC*,*BA*,*CT*}
(Tag additional data)
Expected Petri Space: {C3_2,C3_3,C3_1}

Robotic Adaptive Control

Petri Space, PNRD, and iPNRD Place Relationship



	Petri Space Block A	Petri Space Block B	Petri Space Block C	*PNRD* A	*PNRD* B	*PNRD* C	*iPNRD* State
	$C1\_1$	$C2\_1$	$C3\_1$	AT	BT	CT	A_B_C
Initial Marking	$C1\_1$	$C1\_2$	$C3\_1$	AT	BA	CT	BA_C
	$C1\_1$	$C2\_1$	$C1\_2$	AT	BT	CA	CA_B
	$C1\_1$	$C1\_2$	$C1\_3$	AT	BA	CB	CBA
	$C1\_1$	$C1\_3$	$C1\_2$	AT	BC	CA	BCA
	$C2\_2$	$C2\_1$	$C3\_1$	AB	BT	CT	AB_C
	$C1\_1$	$C2\_1$	$C2\_2$	AT	BT	CB	CB_A
	$C2\_2$	$C2\_1$	$C2\_3$	AB	BT	CA	CAB
	$C2\_3$	$C2\_1$	$C2\_2$	AC	BT	CB	ACB
	$C3\_2$	$C2\_1$	$C3\_1$	AC	BT	CT	AC_B
	$C1\_1$	$C3\_2$	$C3\_1$	AT	BC	CT	BC_A
Final Marking	$C3\_2$	$C3\_3$	$C3\_1$	AC	BA	CT	BAC
	$C3\_3$	$C3\_2$	$C3\_1$	AB	BC	CT	ABC

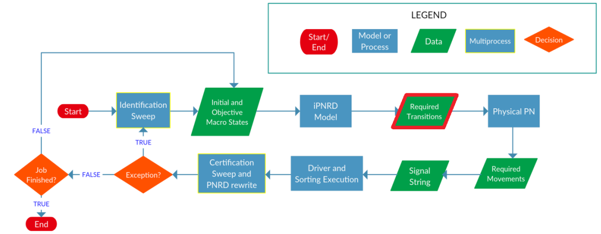
iPNRD Initial Marking

iPNRD Final Marking

The set of PNRD marking (initial and final) and the corresponding set of Petri Space identify the marking of the iPNRD (initial and final)

Robotic Adaptive Control

Based on iPNRD states a Breadth-First Search (BFS) algorithm defines iPNRD transitions
The time and space complexity of this process is $O(T.P)$
where T = number of transitions and P = number of places



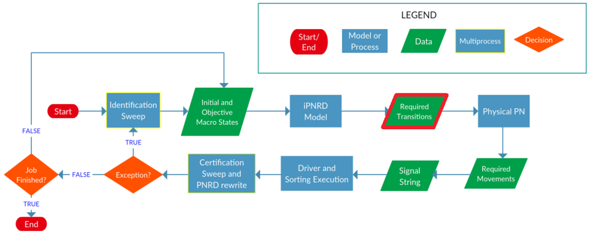
iPNRD State	iPNRD Transition	m0 B2A	m0 B2T	m0 B2C	m1 B2T	m0 A2B	m1 A2T	m0 A2C	m0 A2T	m0 C2A	m1 C2T	m0 C2B	m0 C2T	m1 B2A	m1 B2C	m1 A2C	m1 A2B	m1 C2A	m1 C2B	m2 C2B	m3 C2T	m2 A2B	m2 A2T	m2 C2A	m2 C2T	m2 B2A	m2 B2T	m2 B2C	m3 B2T	m2 A2C	m3 A2T
A_B_C		-1	1					-1	1	-1	1	-1	1																		
BA_C			-1											-1	1					-1	1										
BC_A				1	-1									1	-1							-1	1								
AB_C						1	-1									-1	1							-1	1						
AC_B								1																		-1	1				
CA_B										1	-1							-1	1									-1	1		
CB_A												1	-1					1	-1											-1	1
CBA																				1	-1										
ABC																						1	-1								
CAB																								1	-1						
BAC																										1	-1				
BCA																												1	-1		
ACB																														1	-1

Initial
Intermediate
Intermediate
Final

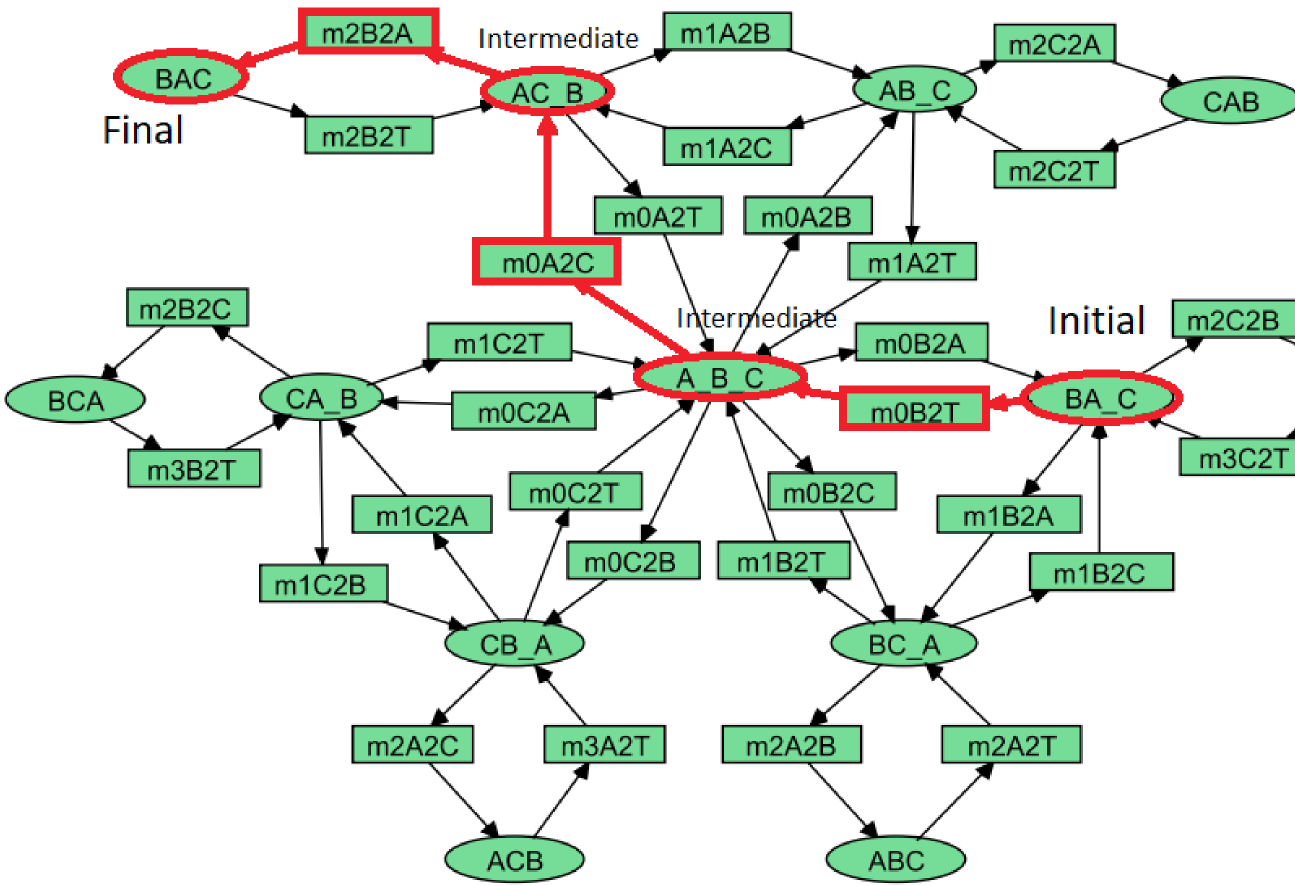
iPNRD A^t is a non labelled matrix
Each column index is related to a specific transition/movement.

Can be optimized!

Robotic Adaptive Control



iPNRD Transition sequence: [m0B2T, m0A2C, m2B2A]

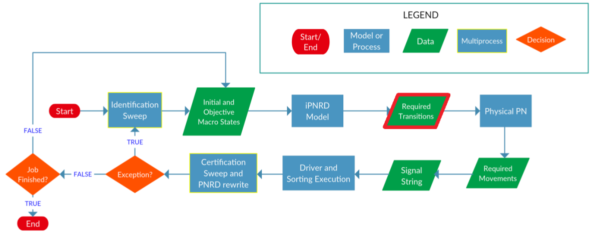


This path refers to a single Block's movement, it means, robot claw is closed with one Block

There is a lack of empty arm movement (opened claw)

So, there is a need to connect Petri Space final position of one stage to the initial position of next stage!

Robotic Adaptive Control



iPNRD and Petri Space Transition Relationship

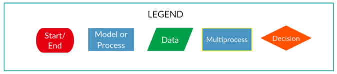
For eah iPNRD transition there are a correspondent Petri Space transition set. This relationship can be optimized!

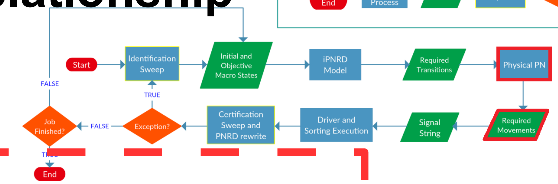
iPNRD Transition	Petri space Transition set	iPNRD Transition	Petri space Transition set	iPNRD Transition	Petri space Transition set
2 *m0A2C*	*T5+T22+T14*	*m0C2B*	*T17+T21+T8*	*m2C2A*	*T17+T21+T6*
m0A2T	*T15+T23+T4*	*m1C2T*	*T9+T20+T16*	*m2C2T*	*T7+T20+T16*
m0A2B	*T5+T18+T8*	*m1A2C*	*T9+T20+T14*	*m2C2B*	*T17+T23+T0*
m1A2T	*T9+T19+T4*	*m1A2B*	*T15+T21+T8*	*m3C2T*	*T1+T22+T16*
m0B2A	*T11+T19+T2*	*m1B2A*	*T15+T23+T2*	*m2A2B*	*T5+T22+T12*
1 *m0B2T*	*T3+T18+T10*	*m1B2C*	*T3+T22+T14*	*m2A2T*	*T13+T23+T4*
m0B2C	*T11+T20+T14*	*m1C2B*	*T3+T18+T8*	*m2A2C*	*T5+T18+T6*
m1B2T	*T15+T21+T10*	*m1C2A*	*T9+T19+T2*	*m3A2T*	*T7+T19+T4*
m0C2A	*T17+T22+T2*	3 *m2B2A*	*T11+T20+T12*	*m2B2C*	*T11+T19+T0*
m0C2T	*T3+T22+T16*	*m2B2T*	*T13+T21+T10*	*m3B2T*	*T1+T18+T10*

Only represent movements when the robot moves a Block with closed claw!

Robotic Adaptive Control

Petri Space Places and iPNRD Transition Relationship





Petri Space Places		
C1_4	C2_4	C3_4
C1_3	C2_3	C3_3
C1_2	C2_2	C3_2
C1_1	C2_1	C3_1

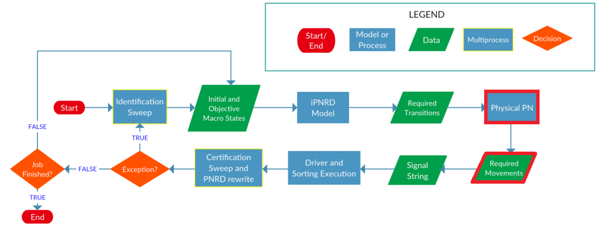
iPNRD Transition sequence:
[m0B2T, m0A2C, m2B2A]

iPNRD Transition	Initial Petri Space	Final Petri Space
m0 B2A	C2_1	C1_2
m0 B2T	C1_2	C2_1
m0 B2C	C2_1	C3_2
m1 B2T	C3_2	C2_1
m0 A2B	C1_1	C2_2
m1 A2T	C2_2	C1_1
m0 A2C	C1_1	C3_2
m0 A2T	C3_2	C1_1
m0 C2A	C3_1	C1_2
m1 C2T	C1_2	C3_1
m0 C2B	C3_1	C2_2
m0 C2T	C2_2	C3_1
m1 B2A	C2_2	C3_2
m1 B2C	C3_2	C2_2
m1 A2C	C1_2	C3_2
m1 A2B	C3_2	C1_2
m1 C2A	C1_2	C2_2
m1 C2B	C2_2	C1_2
m2 C2B	C3_1	C1_3
m3 C2T	C1_3	C3_1
m2 A2B	C1_1	C3_3
m2 A2T	C3_3	C1_1
m2 C2A	C3_1	C2_3
m2 C2T	C2_3	C3_1
m2 B2A	C2_1	C3_3
m2 B2T	C3_3	C2_1
m2 B2C	C2_1	C1_3
m3 B2T	C1_3	C2_1
m2 A2C	C1_1	C2_3
m3 A2T	C2_3	C1_1

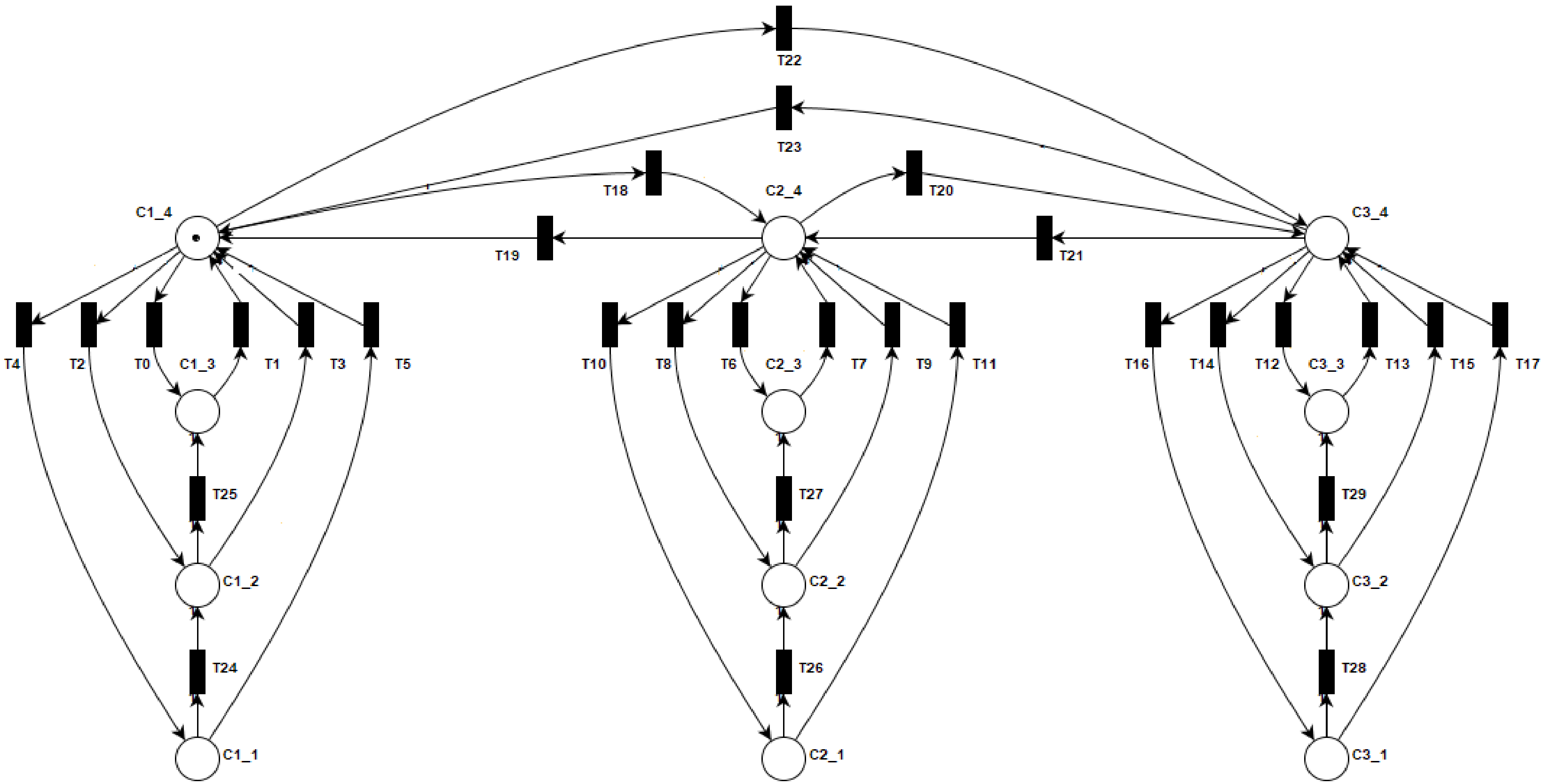
1 2 3

Robot Movement is based on the following Petri Space Macro Places:
[C1_4 → C1_2 → C2_1 → C1_1 → C3_2 → C2_1 → C3_3 → C1_4]
→ Movement with opened claw/ → Movement with closed claw

Robotic Adaptive Control



Petri State transition sequence with initial, intermediates and final places

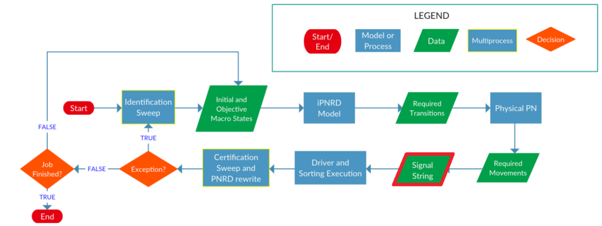


[C1_4 → **T2** → C1_2 → **T3+T18+T10** → C2_1 → **T11+T19+T4** → C1_1 → **T5+T22+T14** → C3_2 → **T15+T21+T10** → C2_1 → **T11+T20+T12** → C3_3 → **T13+T23** → C1_4]

→ Movement with opened claw/ → Movement with closed claw

Robotic Adaptive Control

For each Petri State transition a specific string is associated
This string comands the robot actuators



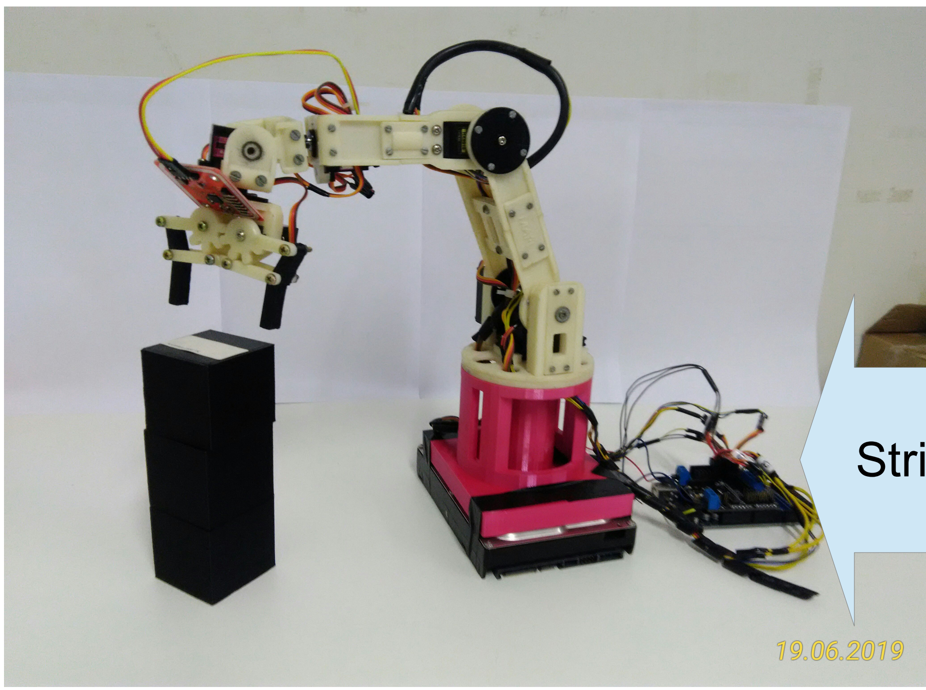
[C1_4 → **T2** → C1_2 → **T3+T18+T10** → C2_1 → **T11+T19+T4** → C1_1 → **T5+T22+T14** → C3_2 → **T15+T21+T10** → C2_1 → **T11+T20+T12** → C3_3 → **T13+T23** → C1_4]

→ Movement with opened claw/ → Movement with closed claw

T_x

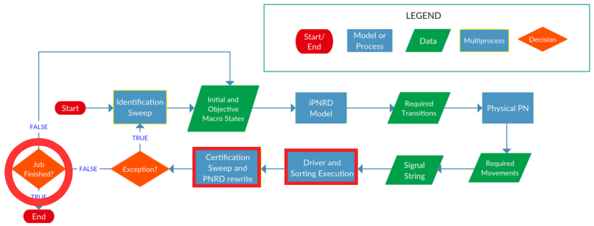
G0M8G1M9M7M5M3M1G0M0M2M4M6M8M10G1M11M9M7M5M3G0M2M4M6

YWIXJHFDBWACEGIKXLJHFDWCEGZ



String

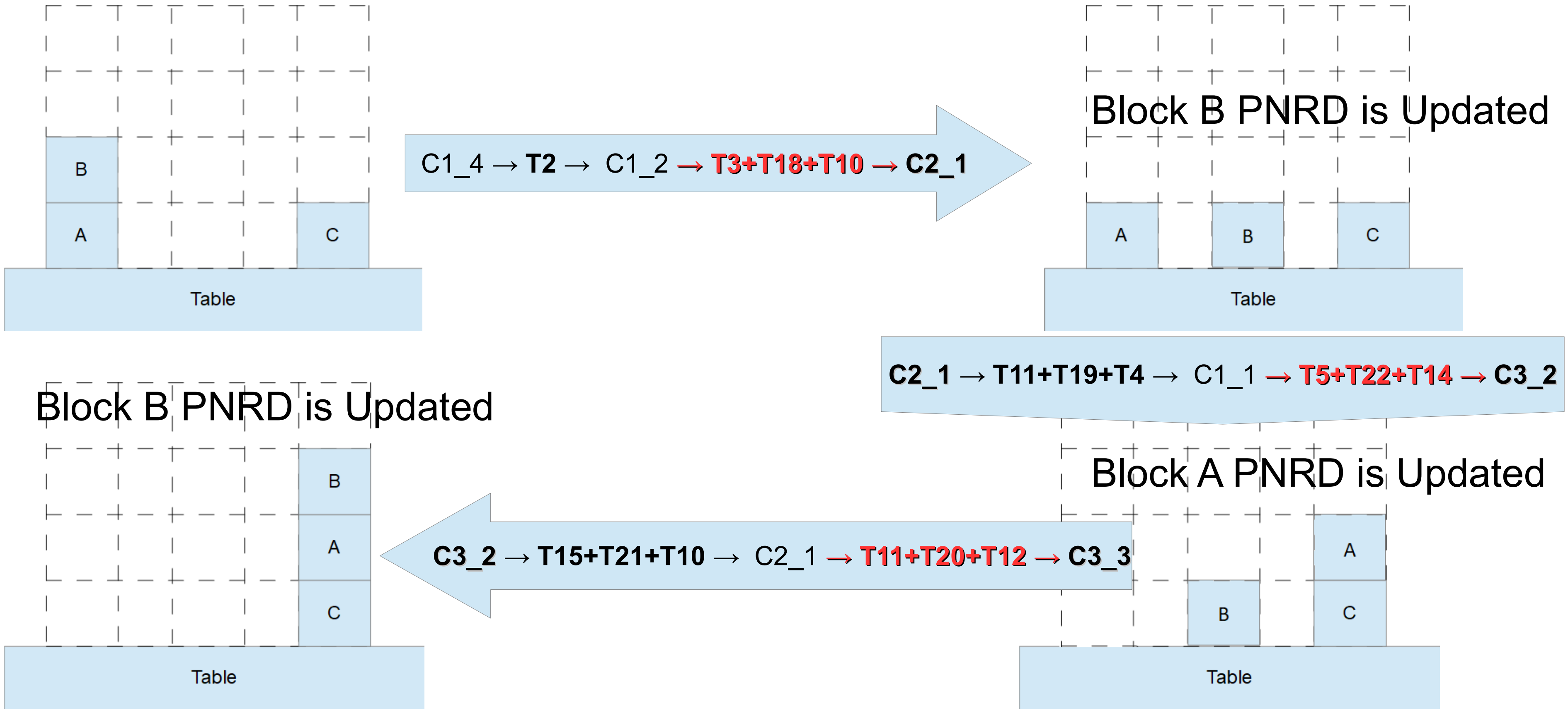
Robotic Adaptive Control



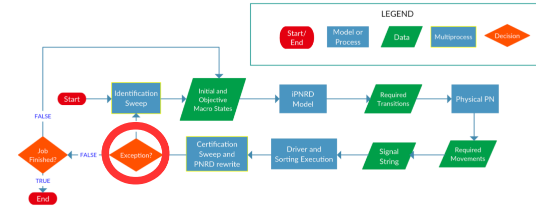
Finishing each macro transition movement, Petri Space position is cheched, and, if it is coherent, PNRD is updated until the final positioning

[C1_4 → **T2** → C1_2 **→ T3+T18+T10 → C2_1** → **T11+T19+T4** → C1_1 **→ T5+T22+T14 → C3_2** → **T15+T21+T10** → C2_1 **→ T11+T20+T12 → C3_3** → **T13+T23** → C1_4]

→ Movement with opened claw/ → Movement with closed claw



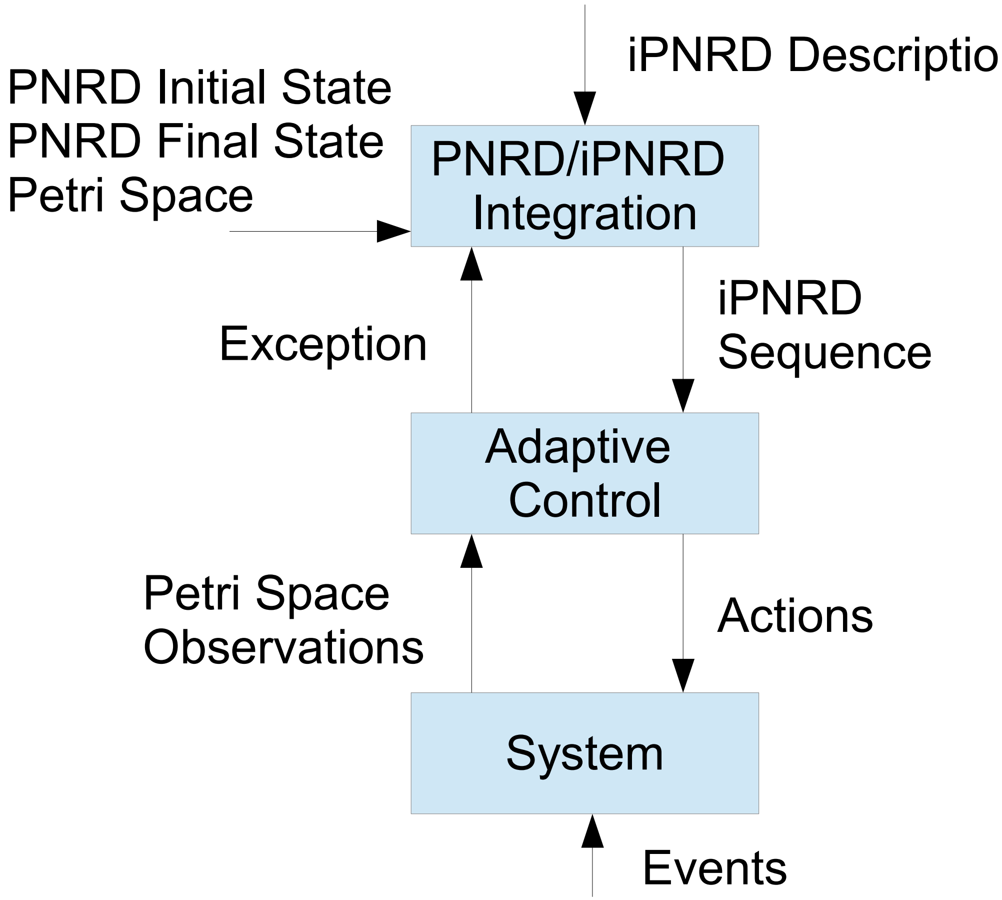
PNRD/ iPNRD Exceptions



- If, and only if, the Petri state set is distinct from PNRD state set during the identification sweep; this means a(some) block(s) was(were) moved from its(their) initial position
  - Petri Space set has priority over PNRD state set, consequently the PNRD states must be updated following the physical layout
- Constraints in this exception arise from the restrictions of the physical world and the proposed Blocks World restrictions
  - There are 13 different logical dispositions in the iPNRD and PNRD although there are 27 distinct PNRD states set, feasible or not
- As the mapping is a bijective function with 3 PNRD inputs, having 3 possible values each, most permutations are invalid having no real transfer to an iPNRD state
- The same occurs on the physical layer to iPNRD or even PNRD
  - Mathematically there are many configurations, but physical model is restricted only to possible configurations
- PNRD final state set can be unfeasible. For example: PNRD final state set = {*AB*, *BA*, *CA*} is impossible to be reached
- Any free block can be deliberately moved during robot movement

Conclusions

This proposition is valid for this example

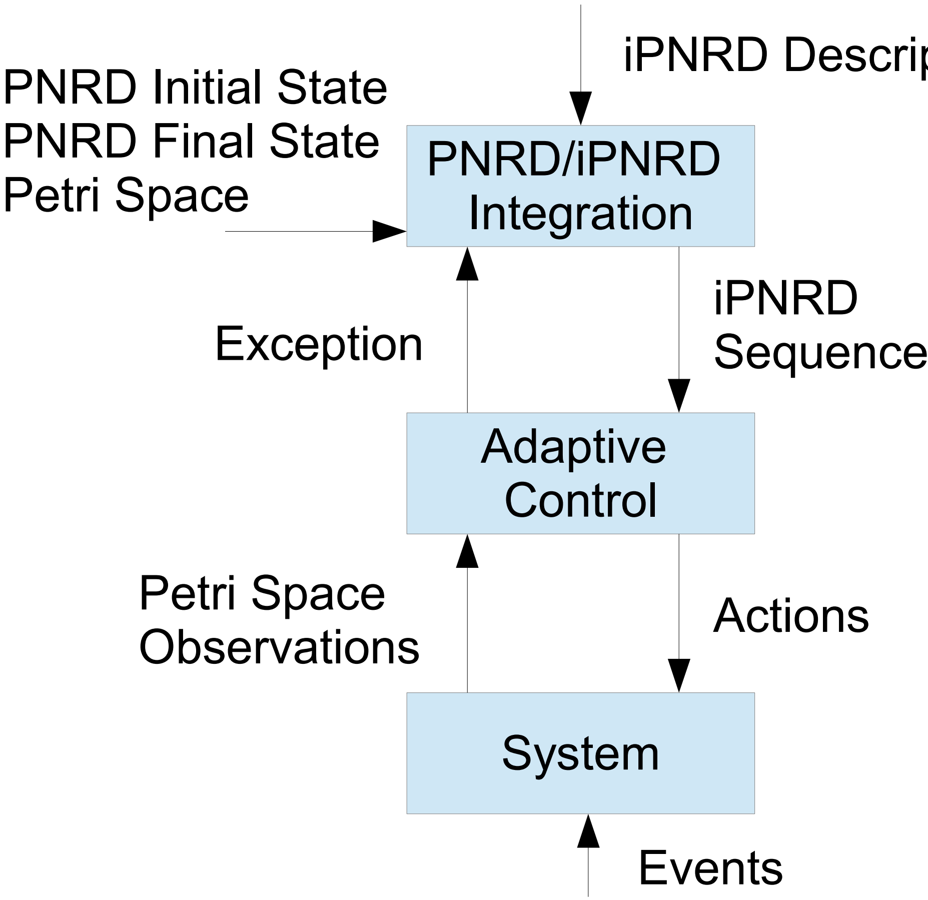


This PNRD/iPNRD integration changes original IPNRD next state calculus!

- This paper presented how PNRD and iPNRD integration can be applied in an adaptive control
- For more complex system (as example 10 blocks) iPNRD model is not able to represent system, and it must be simplified based on automated planning result
- Dealing with automatic model generators and bigger system this problem gets harder, and it must be further investigated
- The adequate fit for positions and trajectory are essential, with uncertainty and precision levels inside a certain tolerance

Conclusions

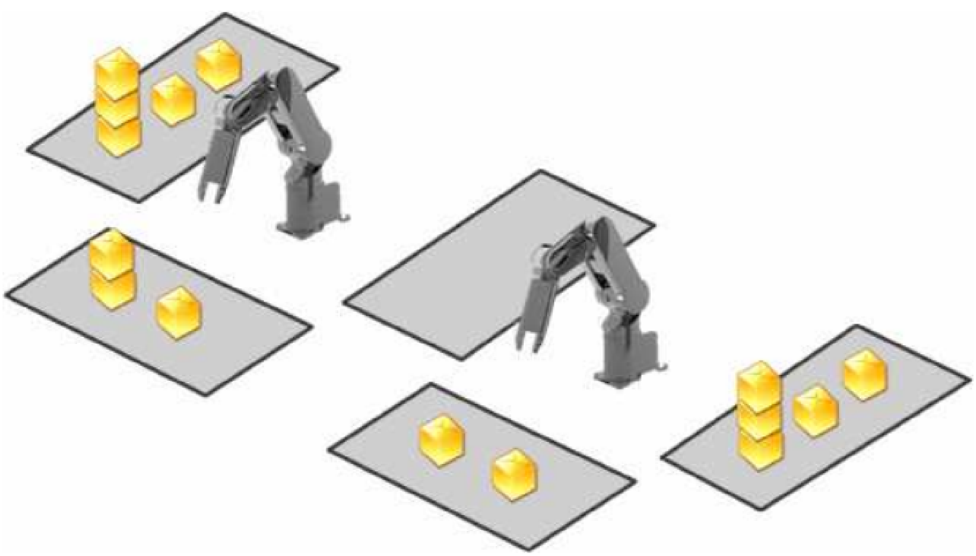
This proposition is valid for this example



- Partial observability is another issue, because it makes exception detection much harder and limits the sweeps average speed
- By including more sensors, partial observability issues could be greatly reduced
- Building auxiliary systems that make exceptions harder to happen or faster to detect could increase system performance
- Adding other sensors the data could be compared to whatever additional information lies inside the PNRD
- Vision systems could make the methods more flexible

Conclusions

Further works



- The RFID system generates a complementary data structure locally, and, in case of network issues, it could assist robot movement independently, acting as Edge Computing application.
  - This aspect must be studied straight forward
- Other means for improvement are making the machines and objectives more flexible and having multiple agents
- A more powerful sensory system coupled with tools for inference and statistical analysis could provide useful data for concluding about the exceptions causes

Acknowledgements

This paper is dedicated to Prof. José Reinaldo Silva from Politechnic School of University of São Paulo

Acknowledgements

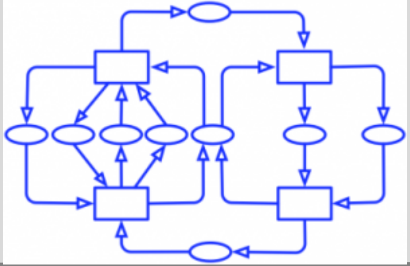
CAPES

UFU

CNPq

FEMEC

PÓS-GRADUAÇÃO
ENGENHARIA MECÂNICA | UFU



PNRD and iPNRD Integration Assisting Adaptive Control in Block World Domain

José Jean-Paul Zanlucchi de Souza Tavares
jean.tavares@ufu.br
+55(34)3239-4003

Gabriel de Almeida Souza