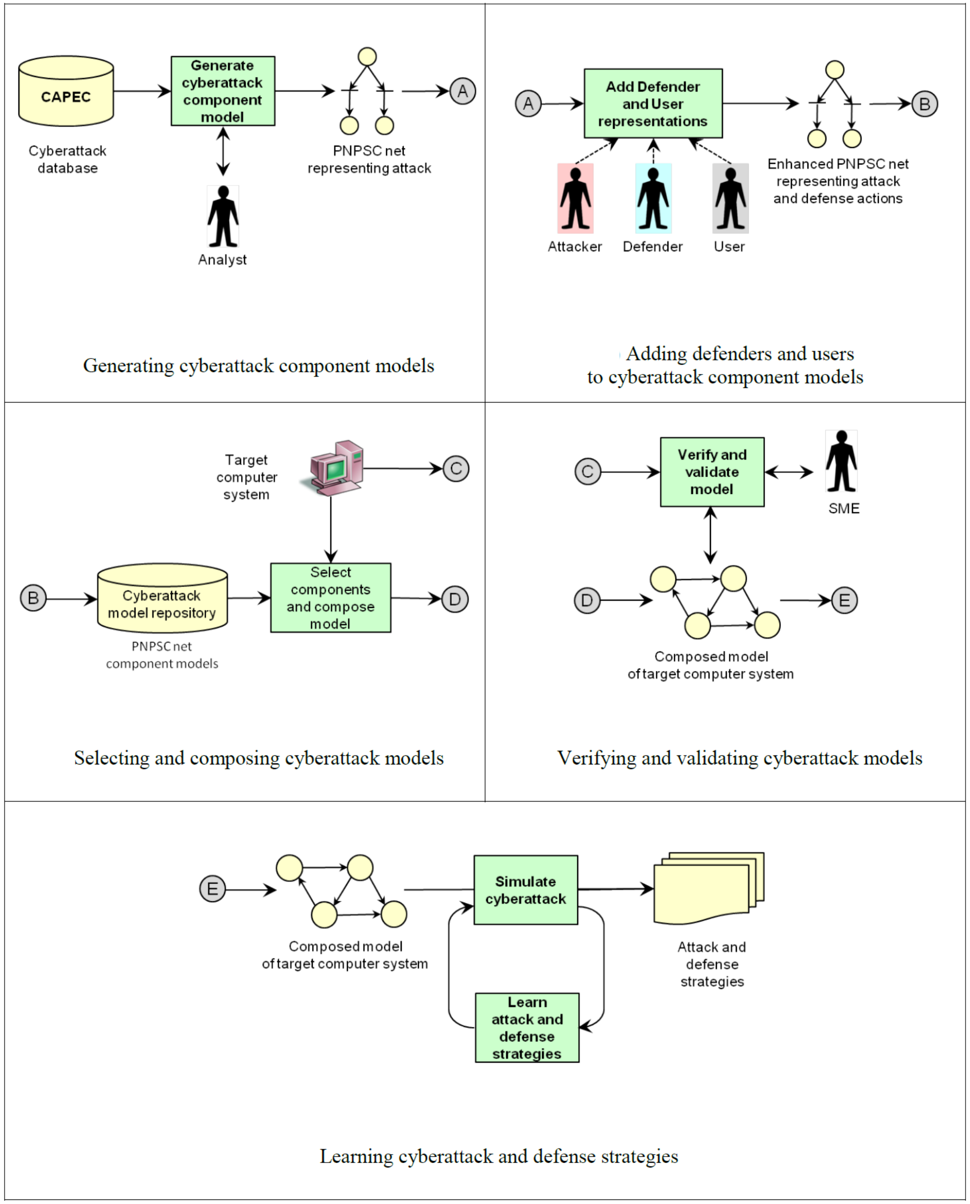
THE UNIVERSITY OF
ALABAMA IN HUNTSVILLE

Analyzing Cyber Attack Impacts and Defense Strategies Using Machine Learning

Daniel Colvett

Petri Nets with Players, Strategies, and Costs Overview

Research Background

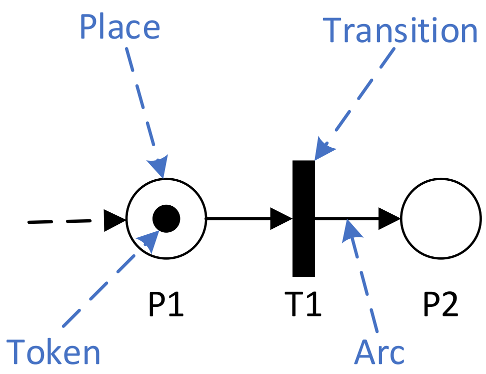


From Petty et al. [1]

[1] Petty, Mikel D, Phil M Showers, Tymaine S Whitaker, John A Bland, Walter Alan Cantrell, C Daniel Colvett, and Katia P Maxwell. 2019. “Modeling Cyberattacks with Extended Petri Nets: Research Program Update.” In *Proceedings of the 2019 AlaSim International Conference and Exposition. Huntsville, AL*, 11. Huntsville, AL.

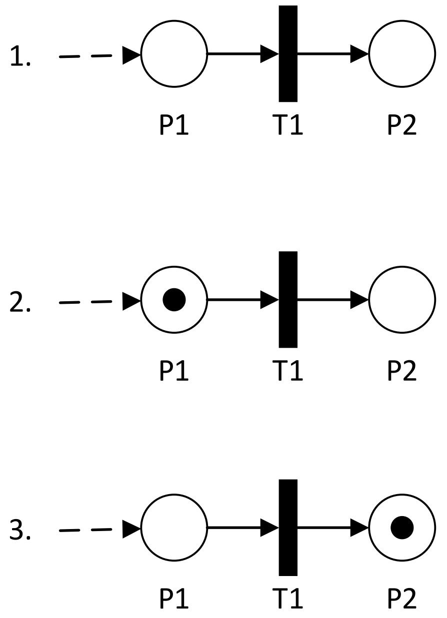
Petri Nets Overview

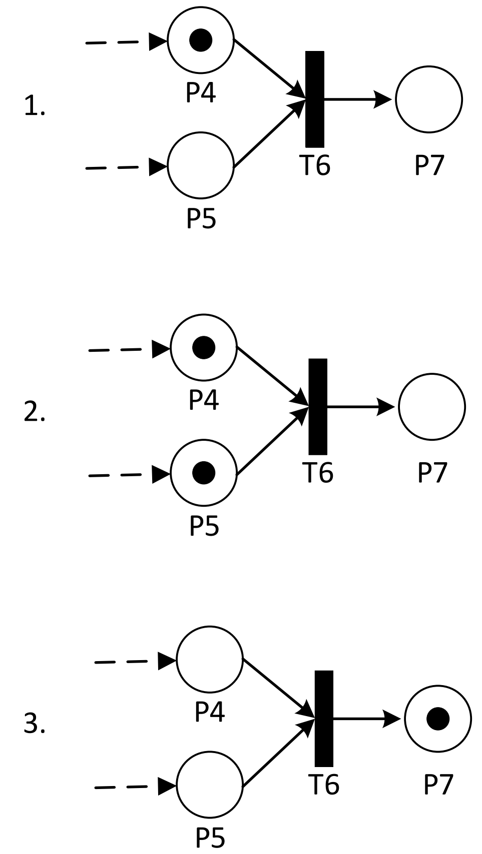
- Originally proposed by Carl Petri
  - 1962 dissertation [2]
  - Extended many times
- 6 Tuple Model
  - Places
  - Transitions
  - Arcs between places and transitions
  - Max tokens per place
  - Initial marking (Tokens)
  - Arc weights



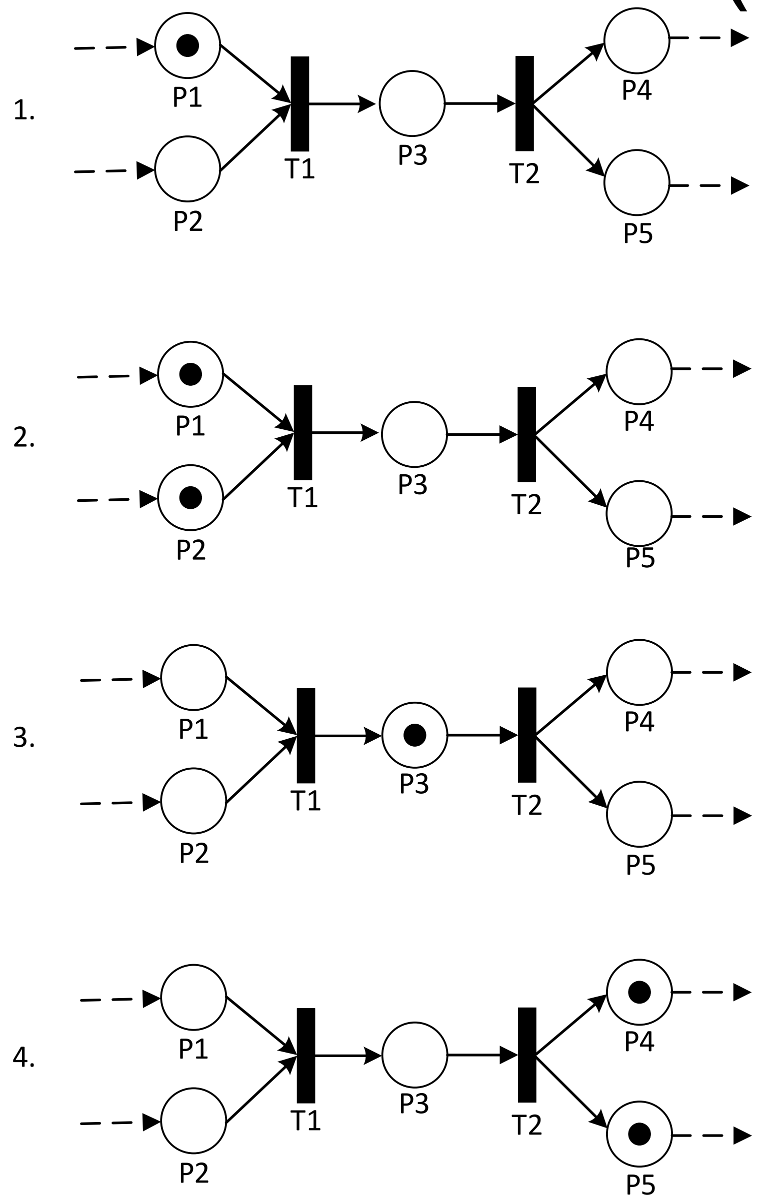
[2] C. A. Petri; Kommunikation mit Automaten, Ph.D. Thesis, Schriften des Rheinisch-Westfälischen Institutes für Instrumentelle Mathematik an der Universität Bonn Nr. 2, 1962.

Petri Nets Overview (cont.)



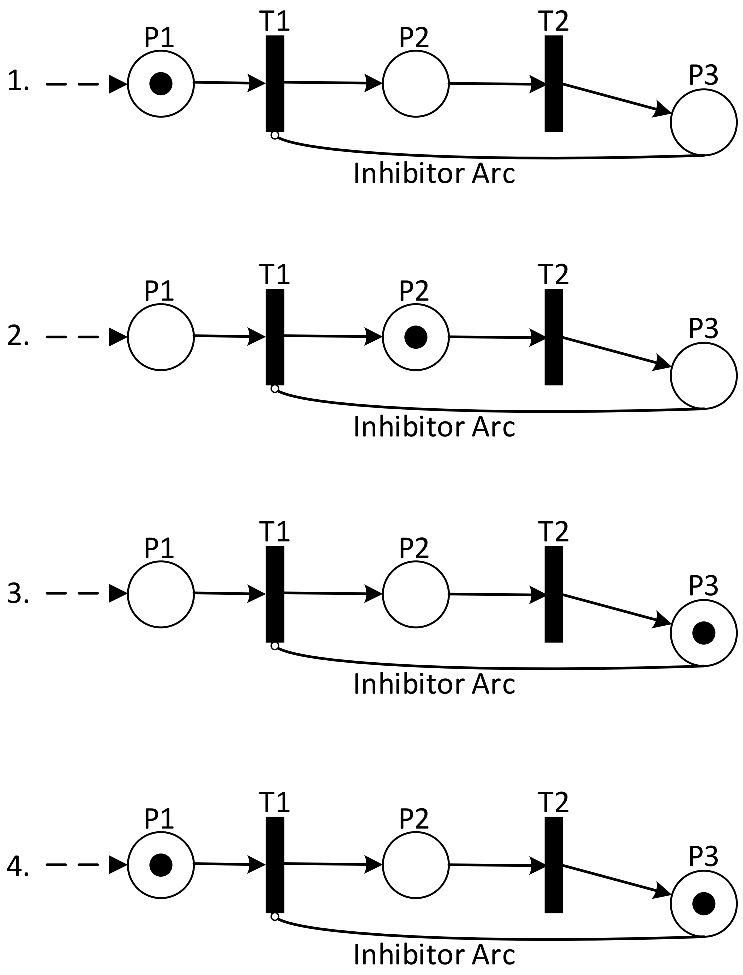


Petri Nets Overview (cont.)

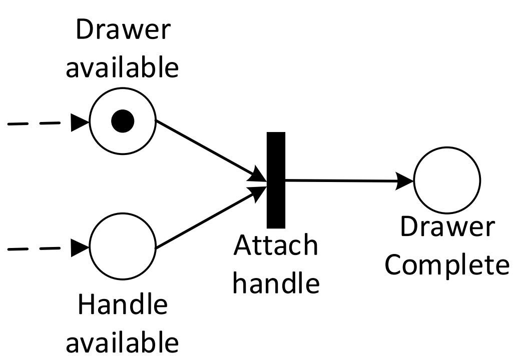


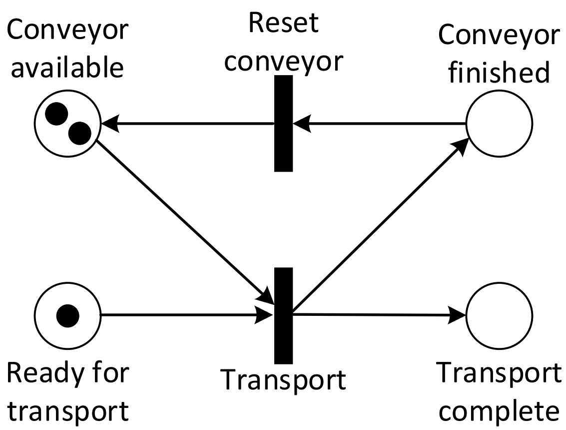
Petri Nets Overview (cont.)

- Inhibitor arcs prevent a transition from firing



Petri Nets Overview (cont.)





From Petty et al. [1]

[1] Petty, Mikel D, Phil M Showers, Tymaine S Whitaker, John A Bland, Walter Alan Cantrell, C Daniel Colvett, and Katia P Maxwell. 2019. "Modeling Cyberattacks with Extended Petri Nets: Research Program Update." In *Proceedings of the 2019 AlaSim International Conference and Exposition. Huntsville, AL*, 11. Huntsville, AL.

PNPSC Overview

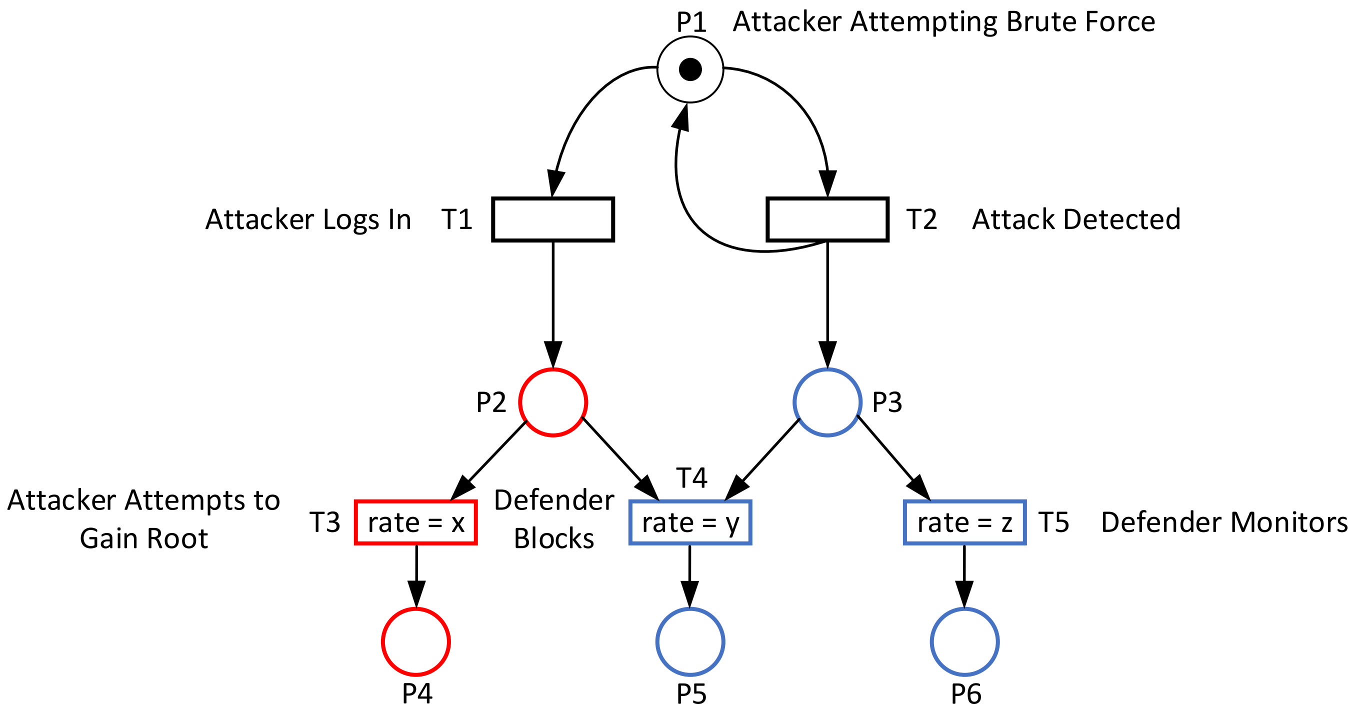
- Earlier work extended Petri nets to add players and strategies (PNPS) [3]

- Petri nets with players, strategies, and costs (PNPSC) extends the PNPS formalism [1]
  - Adds representation of the relative cost of actions taken
  - Resolves ambiguities in the original definitions

[1] Petty, Mikel D, Phil M Showers, Tymaine S Whitaker, John A Bland, Walter Alan Cantrell, C Daniel Colvett, and Katia P Maxwell. 2019. “Modeling Cyberattacks with Extended Petri Nets: Research Program Update.” In *Proceedings of the 2019 AlaSim International Conference and Exposition. Huntsville, AL*, 11. Huntsville, AL.

[3] Zakrzewska, Anita N., and Erik M. Ferragut. 2011. “Modeling Cyber Conflicts Using an Extended Petri Net Formalism.” In _2011 IEEE Symposium on Computational Intelligence in Cyber Security (CICS)_, 60–67. Paris, France: IEEE. https://doi.org/10.1109/CICYBS.2011.5949385.

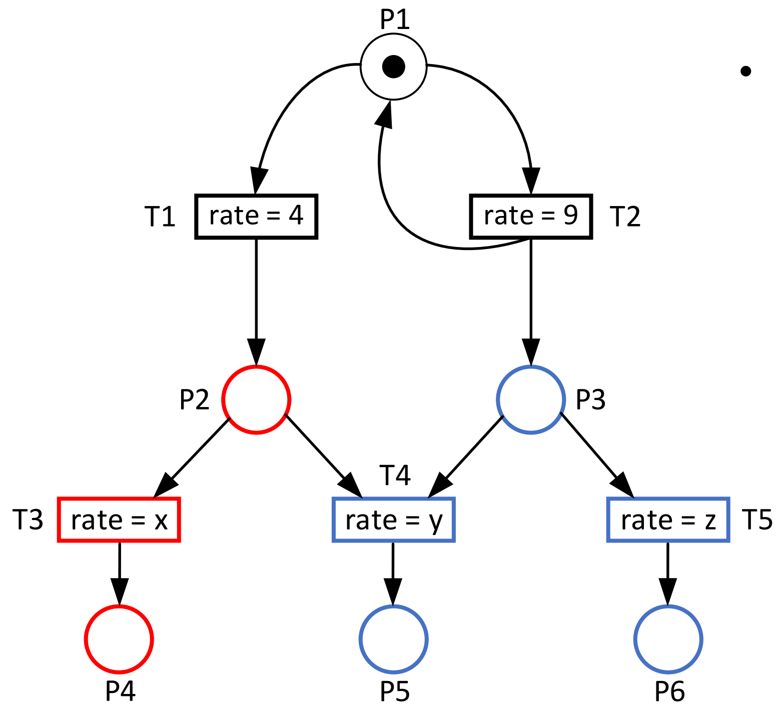
PNPSC Overview



PNPSC Overview - Rates

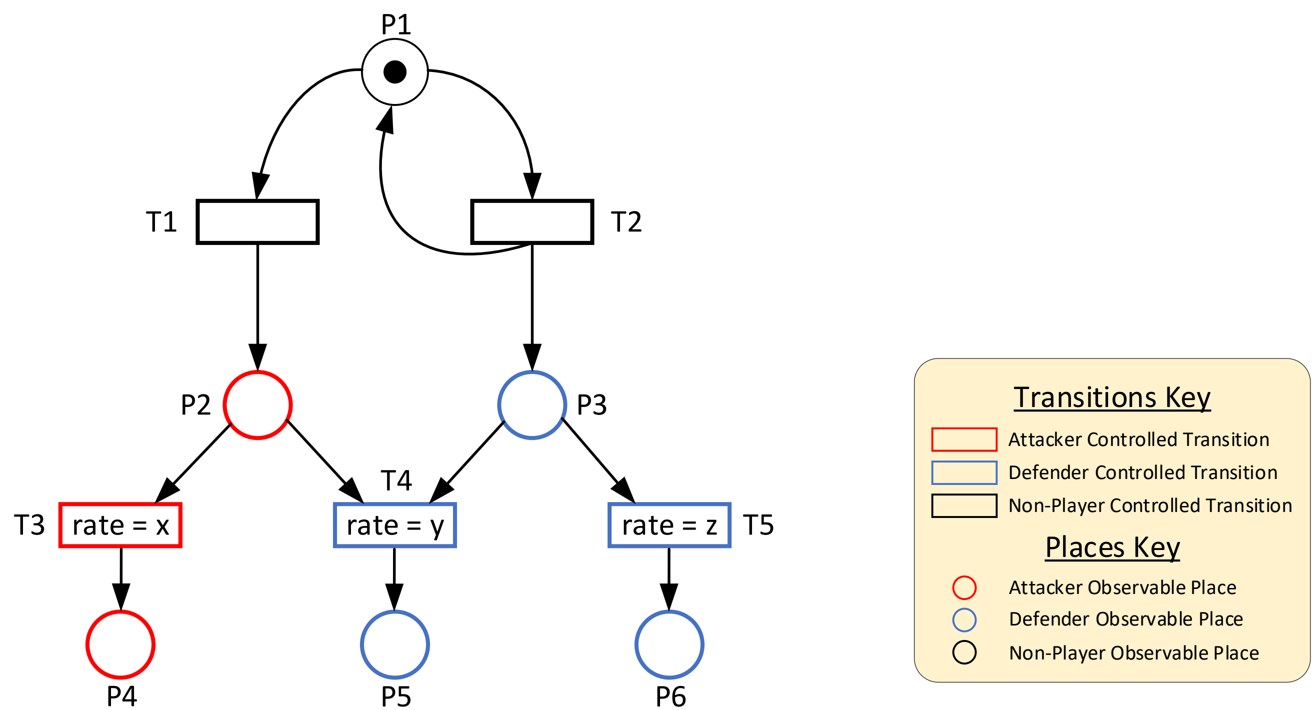
- T1 has rate 4
- T2 has rate 9
- T1 and T2 are both enabled at the same time

- Firing time is set for each enabled transition by using its rate
- Earliest scheduled is selected to fire
- Rates are relative
- Higher rate = more likely to fire
- Lower Rate = less likely to fire



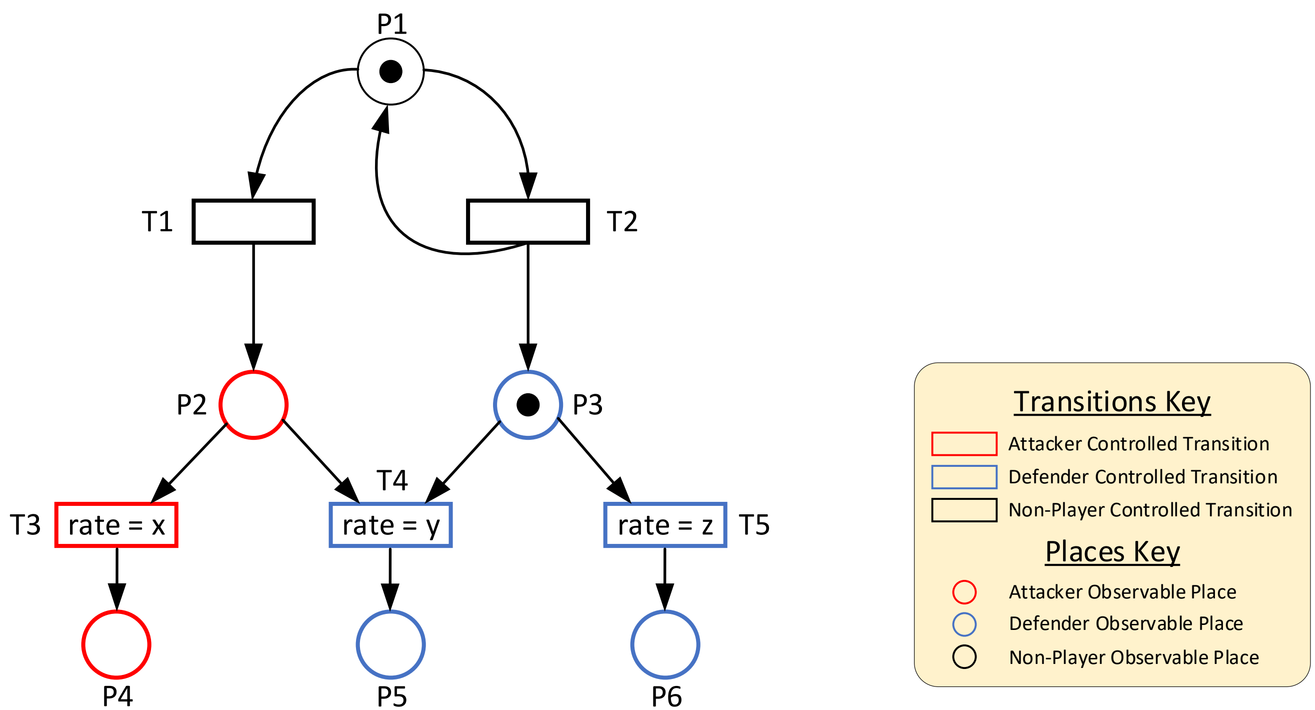
PNPSC Overview - Players

- PNPSC net can have one or more players
- Two or more players can have competing or cooperative goals
- Places can be player observable
- Transitions can be player controlled



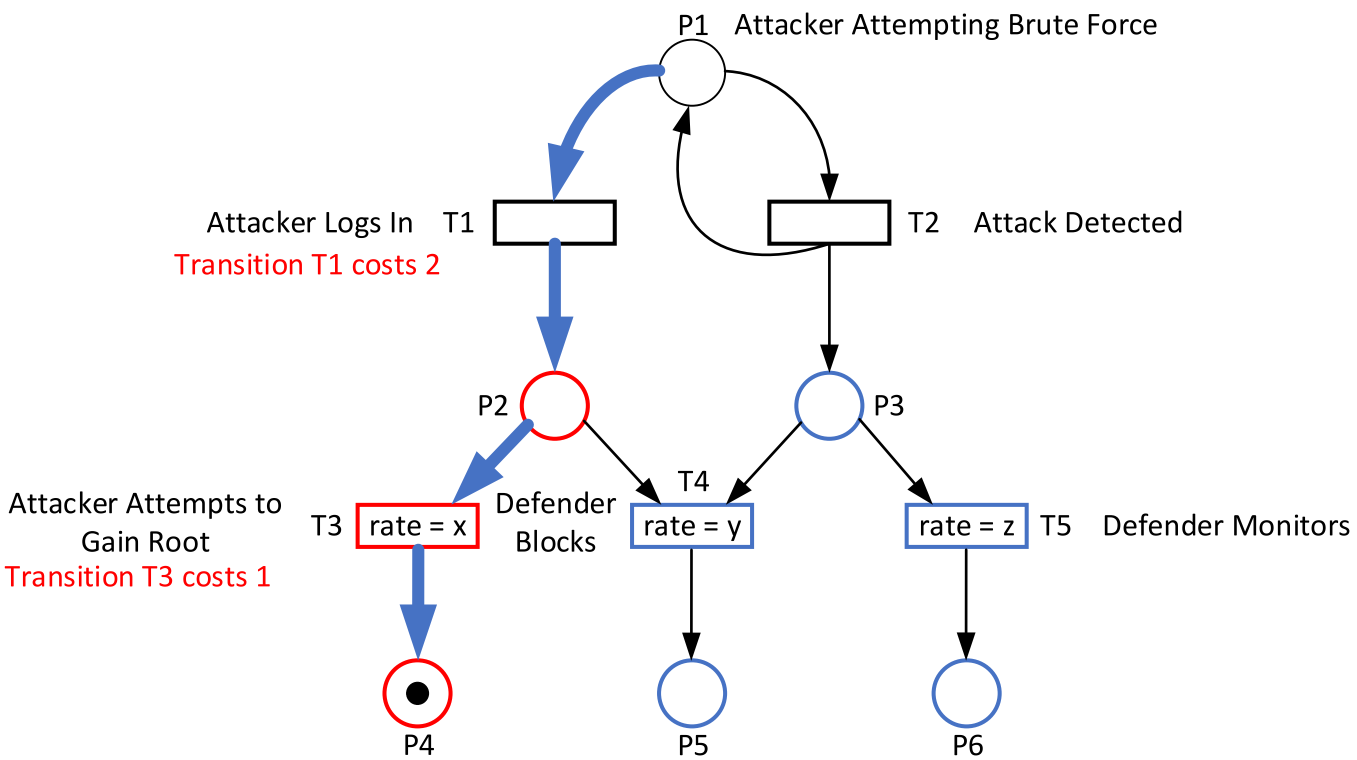
PNPSC Overview - Strategies

- Defender can observe P3 is marked
- If Defender’s goal is to block the attack (P5 marked), then their strategy would be to increase T4’s rate and lower T3’s rate



PNPSC Overview – Fire Costs

- Costs are added for transition firing
- T1 and T3 have a costs of 2 and 1 respectively
- Total costs = 3



PNPSC Overview – Change Costs

- Cost based on changing player controlled rates
- Rates of player-controlled transition have associated cost to change
- Cost can be the summation of the change
- Example
  - If there are 3 Player Controlled transitions, then changing fire rate costs:
  - {0,0,0} -> {4,0,4} would have a cost of 8
  - {1,2,3} -> {2,2,4} would have a cost of 2

PNPSC Formalism

A PNPSC is 14-tuple formally defined as $PNPSC = (P, T, W, M_0, B, L, G, \Theta, O, F, \Omega, \Gamma, C, D)$, where

1) P, T, W, M_0, B, L; as defined for a standard Petri net
2) $G = \{g_1, g_2, \ldots\}$; finite, non-empty set of players
3) $\Theta = (T_0, T_1, T_2, \ldots, T_{|G|})$; partition of transition set T into $|G| + 1$ subsets such that $\Theta = T_0 \cup T_1 \cup T_2 \cup \ldots \cup T_{|G|}$ and $T_j \cap T_k = \varnothing$ for $0 \leq j, k \leq |G|$ and $j \neq k$; T_i = set of transitions controlled by player g_i for $1 \leq i \leq |G|$ and T_0 = set of stochastic transitions not controlled by any player
4) $O = (O_1, O_2, \ldots, O_|G|)$; collection of $|G|$ subsets of place set P, i.e, $O_i \subseteq P$ for $1 \leq i \leq |O|$; O_i is the subset of place set P observable by player g_i
5) $F : T_0 \rightarrow \mathbb{R}^+$; fixed firing rates for non-player-controlled transitions
6) $\Omega : (T - T_0) \rightarrow (\mathbb{R}^+ \times \mathbb{R}^+)$; initial and maximum firing rates for player-controlled transitions
7) $\Gamma : (\Gamma_1, \Gamma_2, \ldots, \Gamma_{|G|}$; collection of functions $\Gamma_i : M^*_{Oi} \rightarrow \mathbb{R}^{+|T_i|}$ where each Γ_i is a mapping from the possible markings of player g_i's observable places to the desired firing rates for each of player g_i's controlled transitions
8) C = (C_{fire}, C_{change}); where C_{fire}:$(T \rightarrow \mathbb{R}^+)$ is the cost for firing a transition and C_{change}:$(T \times \mathbb{R}^+) \rightarrow \mathbb{R}^+$ is the cost for changing the rate of a transition by $\delta \in \mathbb{R}^+$
9) $D : T \rightarrow \wp(G)$; players that incur a cost for a fired or changed transition

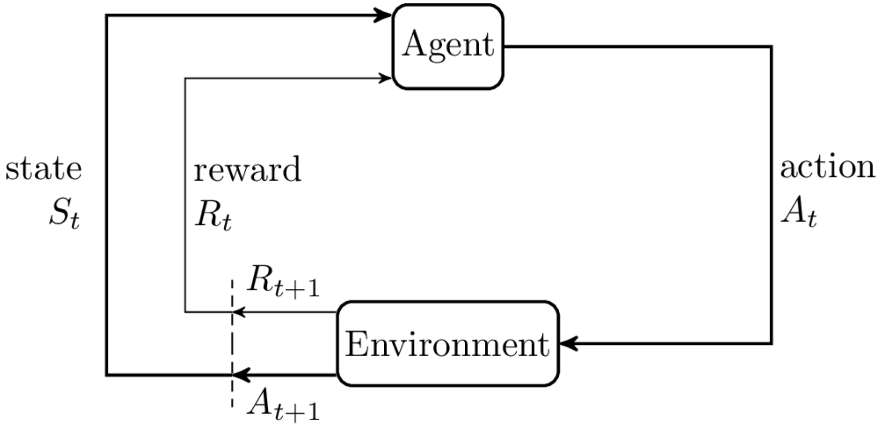
From Petty et al. [1]

[1] Petty, Mikel D, Phil M Showers, Tymaine S Whitaker, John A Bland, Walter Alan Cantrell, C Daniel Colvett, and Katia P Maxwell. 2019. "Modeling Cyberattacks with Extended Petri Nets: Research Program Update." In *Proceedings of the 2019 AlaSim International Conference and Exposition. Huntsville, AL*, 11. Huntsville, AL.

Machine Learning Overview

Machine Learning Overview

- Agent - learner
- Environment – everything outside of the agent
- Action – what an agent can change to impact the environment
- State – Representation of the current environment
- Reward – Consequence of Action

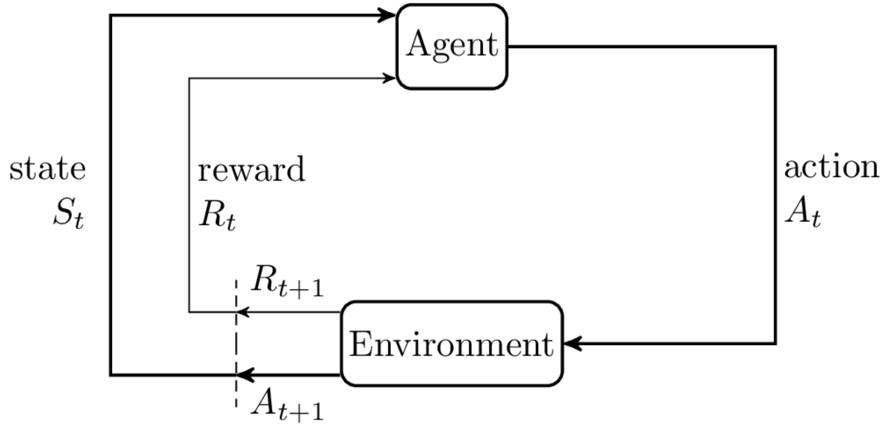


From Sutton and Barto [4]

[4] Sutton, Richard S., and Andrew G. Barto. 2018. _Reinforcement Learning: An Introduction_. 2nd ed. Adaptive Computation and Machine Learning Series. Cambridge, Massachusetts: The MIT Press.

Machine Learning Overview (cont.)

- Reinforcement learning can be applied to PNPSC nets [5].
  - Agent - Player
  - Environment – PNPSC net
  - Action – changing fire rates of the player controlled transitions
  - State – the player observable marking of the PNPSC net
  - Reward – Final state achieved (positive if successful, negative if not)



From Sutton and Barto [4]

[4] Sutton, Richard S., and Andrew G. Barto. 2018. _Reinforcement Learning: An Introduction_. 2nd ed. Adaptive Computation and Machine Learning Series. Cambridge, Massachusetts: The MIT Press.

[5] Bland, John A., Mikel D. Petty, Tymaine S. Whitaker, Katia P. Maxwell, and Walter Alan Cantrell. 2020. “Machine Learning Cyberattack and Defense Strategies.” _Computers & Security_ 92 (May): 101738. https://doi.org/10.1016/j.cose.2020.101738.

THE UNIVERSITY OF ALABAMA IN HUNTSVILLE

Machine Learning Overview (cont.)

- ε-Greedy Technique used

- Set ε–value to set algorithm's probability of taking nongreedy action

- Nongreedy actions are exploratory

From Sutton and Barto [4]

[4] Sutton, Richard S., and Andrew G. Barto. 2018. _Reinforcement Learning: An Introduction_. 2nd ed. Adaptive Computation and Machine Learning Series. Cambridge, Massachusetts: The MIT Press.

Enhancements to Previous Work

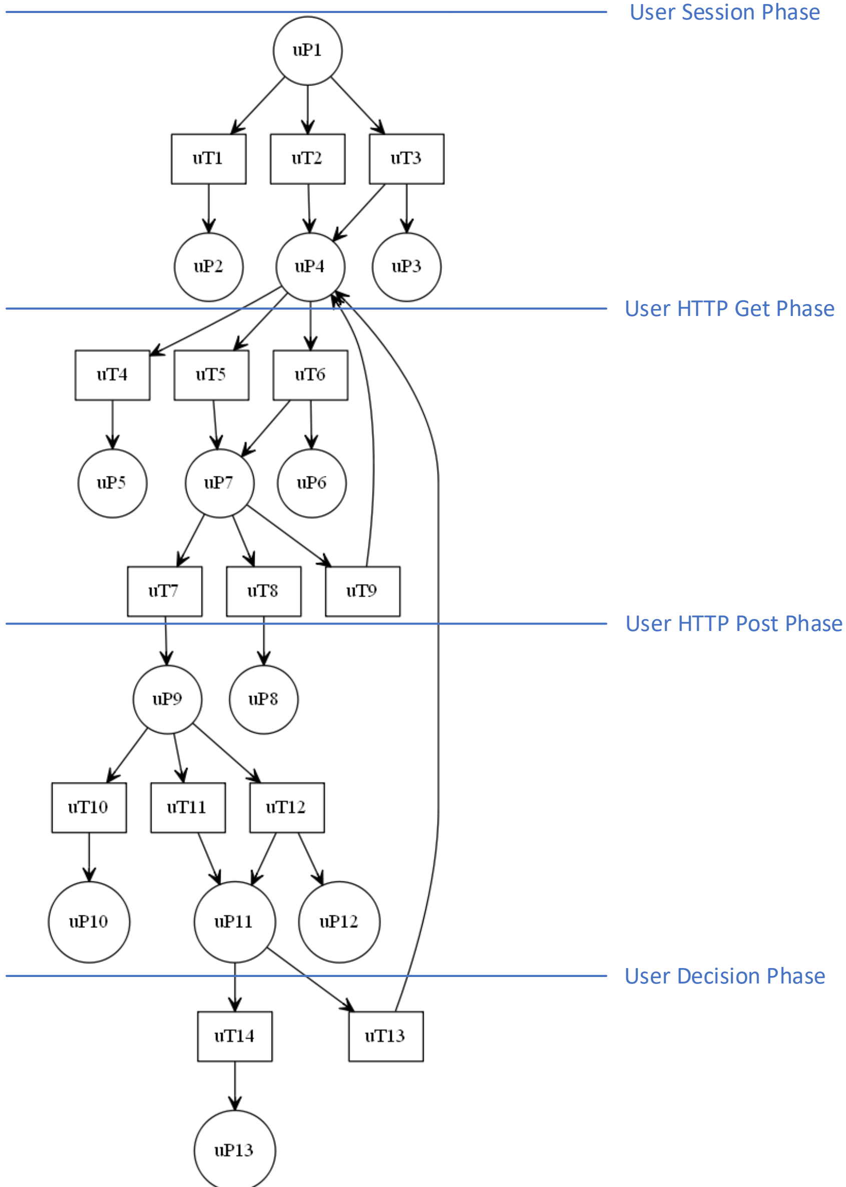
Enhancements

- Previous work was using machine learning and PNPSC nets [5], but has since been enhanced
  - Initial PNPSC simulator was limited to possible rates and number of players because of state space explosion issues. This was corrected by creating a PNPSC simulator using a database management system [6]
  - Representation for the computer system user was not present.

[5] Bland, John A., Mikel D. Petty, Tymaine S. Whitaker, Katia P. Maxwell, and Walter Alan Cantrell. 2020. “Machine Learning Cyberattack and Defense Strategies.” _Computers & Security_ 92 (May): 101738. https://doi.org/10.1016/j.cose.2020.101738.

[6] C. D. Colvett, M. D. Petty, J. A. Bland and K. R. Baker, "Simulating Cyberattacks with a Petri Net Discrete Event Simulator," _2019 International Conference on Computational Science and Computational Intelligence (CSCI)_, 2019, pp. 67-71, doi: 10.1109/CSCI49370.2019.00018.

Enhancements (cont.)



Place Description	
Place Name	**Description**
uP1	User initiate connection to application
uP2	User was incorrectly blocked from creating a session
uP3	User was allowed to create session, but flagged as possible attack
uP4	User begins HTTP/HTTPS GET request for information
uP5	User's HTTP/HTTPS GET request for information is incorrectly blocked as an attack
uP6	User's HTTP/HTTPS GET request for information allowed, but flagged as possible attack
uP7	User completes HTTP/HTTPS GET request for information. User decides if additional actions required
uP8	User closes application session
uP9	User begins HTTP/HTTPS POST request to send data to server
uP10	User's HTTP/HTTPS POST request to send data is incorrectly blocked as an attack
uP11	User completes HTTP/HTTPS POST request to send data to server. User decides if additional actions required
uP12	User's HTTP/HTTPS POST request to send data, but flagged as possible attack
uP13	User closes application session

Transition Description	
Transition Name	**Description**
uT1	Defender blocks User from initiating session
uT2	Defender allows User to initiate session
uT3	Defender allows User to session, but flags as possible attack
uT4	Defender incorrectly blocks User HTTP/HTTPS GET request for data
uT5	Defender allows User HTTP/HTTPS GET request for data
uT6	Defender allows User HTTP/HTTPS GET request for data, but flags as possible attack
uT7	User needs to perform HTTP/HTTPS POST request to send data to server
uT8	User has completed all necessary actions, closes application session
uT9	User needs to perform additional HTTP/HTTPS GET request for information
uT10	Defender incorrectly blocks User HTTP/HTTPS POST request to send data to server
uT11	Defender allows User HTTP/HTTPS GET POST request to send data to server
uT12	Defender allows User HTTP/HTTPS POST request to send data, but flags as possible attack
uT13	User needs to perform additional HTTP/HTTPS GET request for information
uT14	User has completed all necessary actions, closes application session

uT1, uT4, uT10 - Defender Blocks User

uT3, uT6, uT12 - Defender Flags User Request

Cross-Site Scripting Model (CAPEC 63)

MITRE CAPEC Database

- Cross-Site Scripting was chosen as it consistently ranks high in the Open Web Application Security Project (OWASP) Top Ten security vulnerabilities

- The MITRE Common Attack Pattern Enumeration Classification (CAPEC) database was used as the baseline for describing the attack. Cross-Site Scripting has a CAPEC ID of 63 [7].

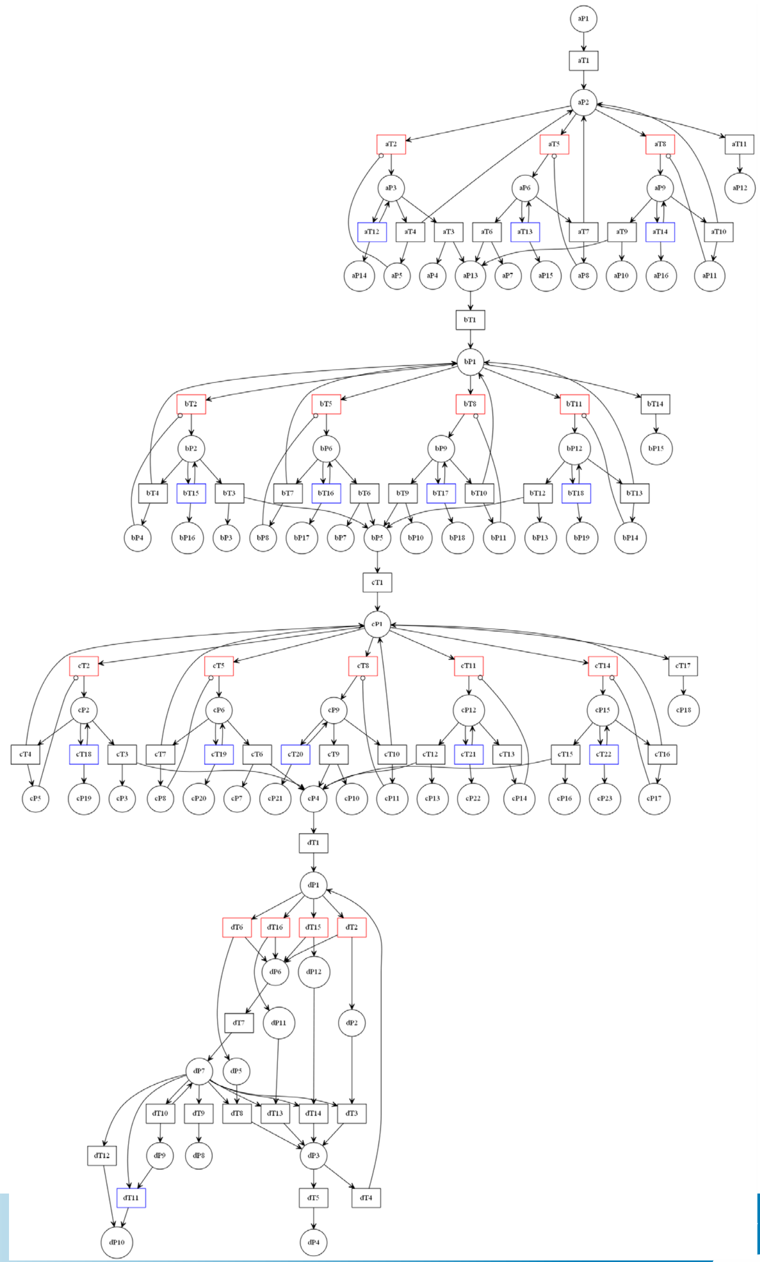
- For full details on the Cross-Site Scripting PNPSC nets, see [5]. To see details on how the net was validated, see [8].

[5] Bland, John A., Mikel D. Petty, Tymaine S. Whitaker, Katia P. Maxwell, and Walter Alan Cantrell. 2020. "Machine Learning Cyberattack and Defense Strategies." _Computers & Security_ 92 (May): 101738. https://doi.org/10.1016/j.cose.2020.101738.

[7] The MITRE Corporation, "Common Attack Pattern Enumeration and Classification". https://capec.mitre.org/, October 2, 2021

[8] Cantrell, Walter A, Katia P Mayfield, Mikel D Petty, Tymaine S Whitaker, and John A Bland. 2018. "Structured Face Validation of Extended Petri Nets for Modeling Cyberattacks." In Proceedings of the 2017 AlaSim International Conference and Exposition, Huntsville, AL.

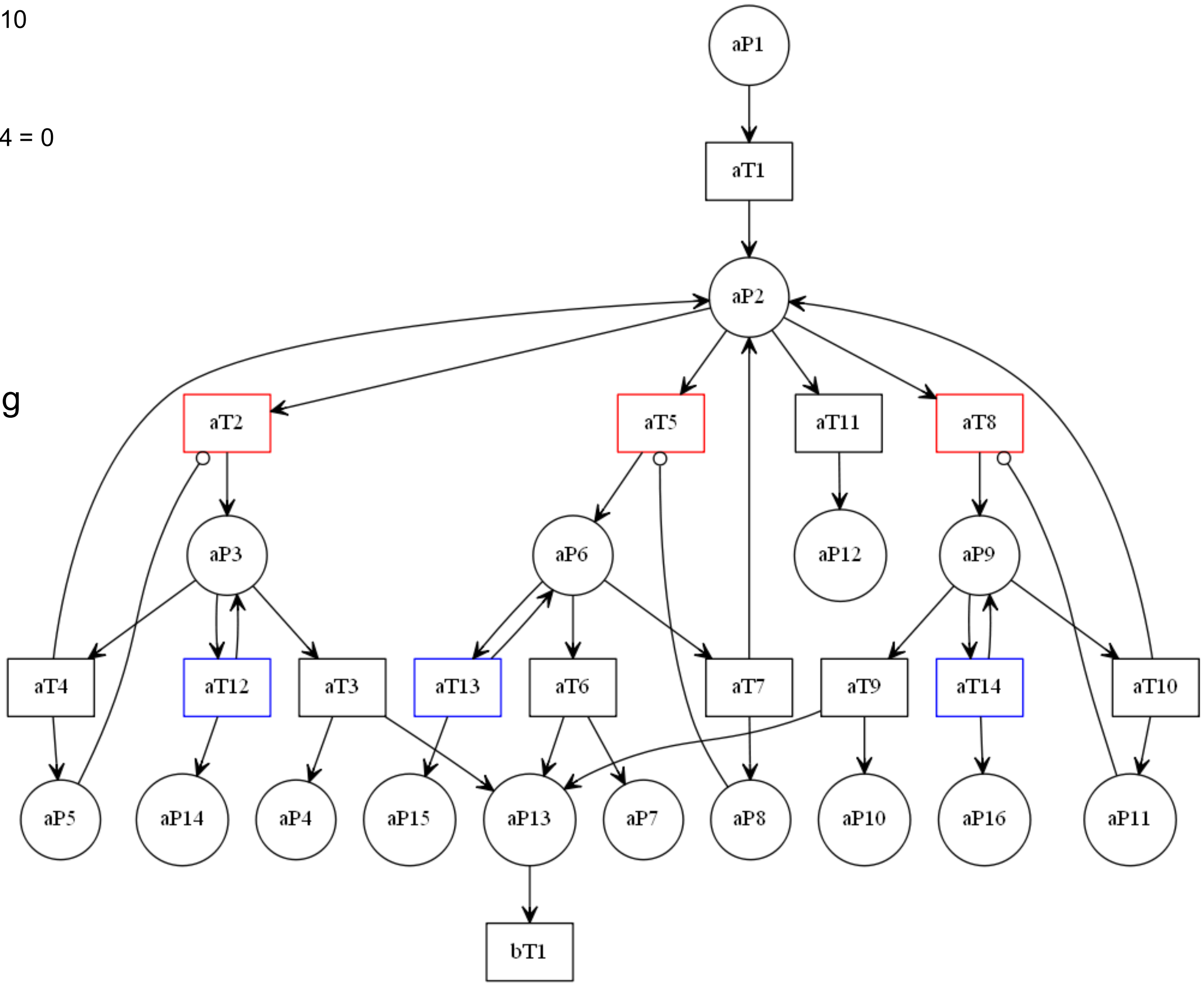
Cross Site Scripting – Full Net



Cross Site Scripting – Explore Phase

- When aP2 is marked:
  - ---Attacker---
  - Sets Rates – aT2 = 0, aT5 = 10, aT8 = 10
  - Average Reward = 7.19
  - ---Defender---
  - Sets Rates – aT12 = 10, aT13 = 0, aT14 = 0
  - Average Reward = -19.75

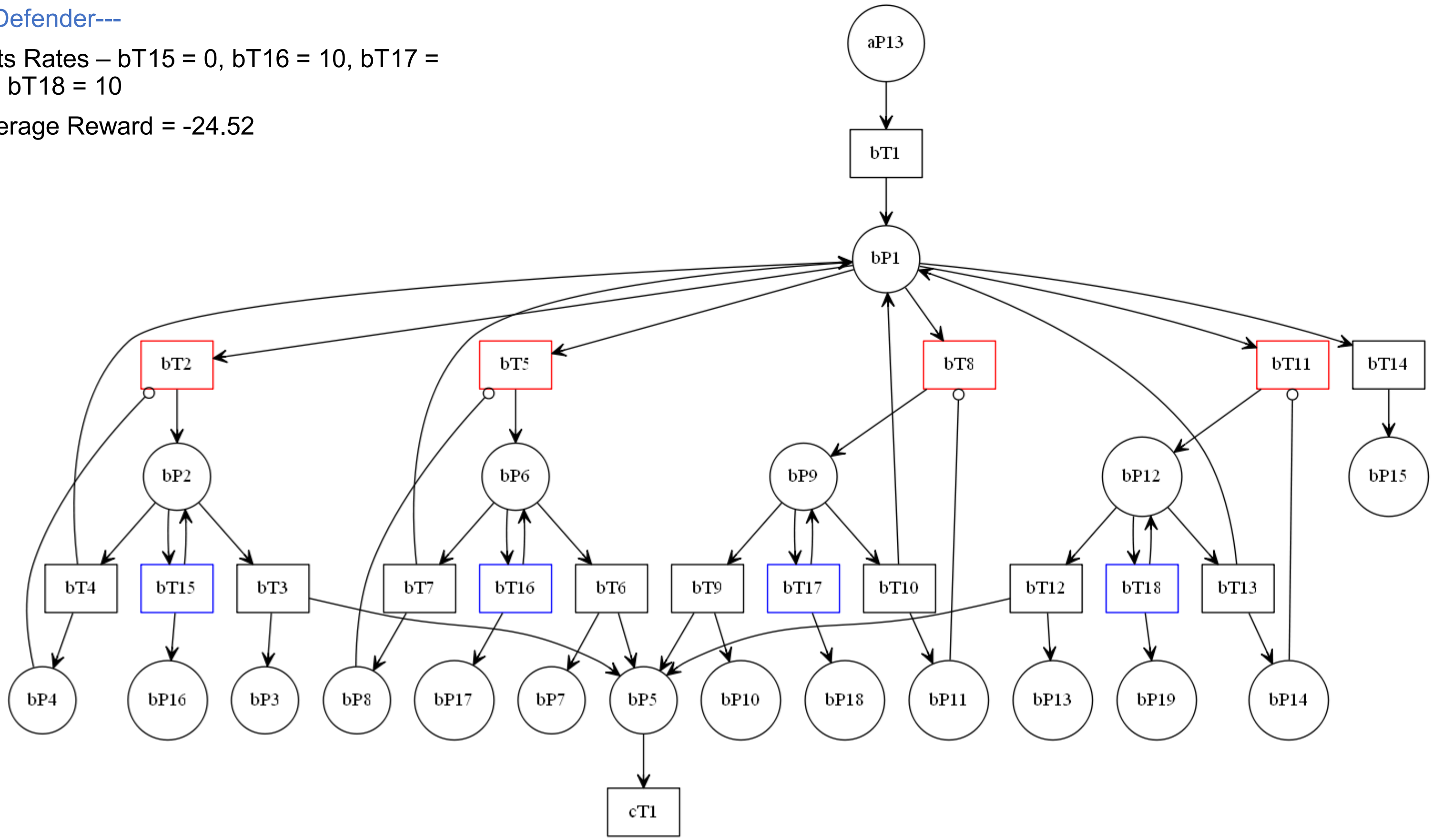
- aT2 – Spider website
- aT5 – Proxy tool to find all links
- aT8 – Manual Brute force browsing



Cross Site Scripting – Experiment Phase

- When aP4 and bP1 are marked:
  - ---Attacker---
  - Sets Rates – bT2 = 0, bT5 = 10, bT8 = 0, bT11 = 10
  - Average Reward = -25.03
- When aP14 and bP1 are marked:
  - ---Defender---
  - Sets Rates – bT15 = 0, bT16 = 10, bT17 = 10, bT18 = 10
  - Average Reward = -24.52

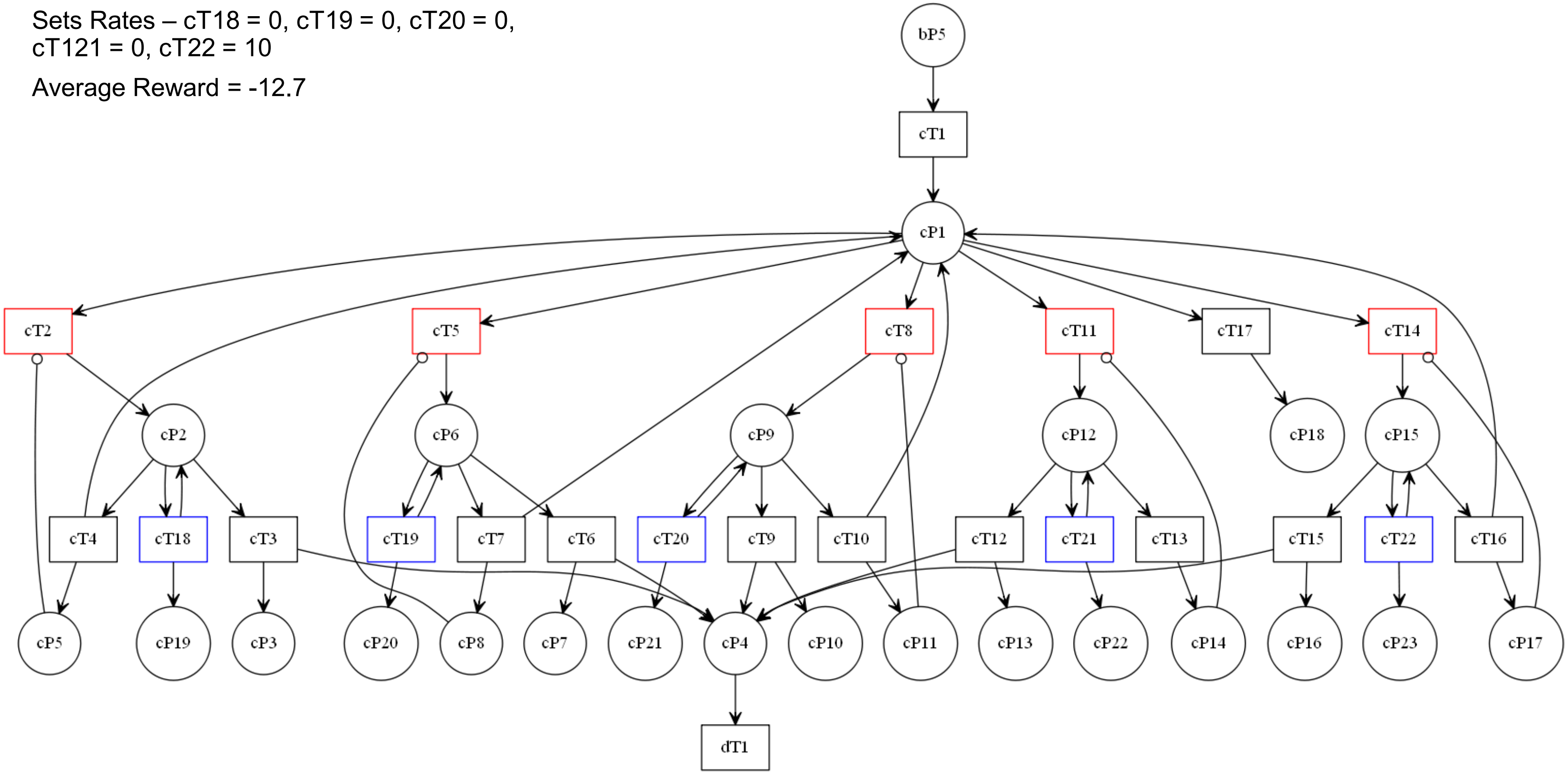
- bT2 – Probes XSS strings to known URLS
- bT5 – Proxy tool to record results
- bT8 – Probs XSS strings in UI entry fields
- bT11 – XSS injection scripts into resources



Cross Site Scripting – Exploit Phase

- When aP4, bP3 and cP1 are marked:
  - ---Attacker---
  - Sets Rates – cT2 = 0, cT5 = 0, cT8 = 10, cT11 = 10, cT14 = 0
  - Average Reward = 40
- When aP14, bP18 and cP1 are marked:
  - ---Defender---
  - Sets Rates – cT18 = 0, cT19 = 0, cT20 = 0, cT121 = 0, cT22 = 10
  - Average Reward = -12.7

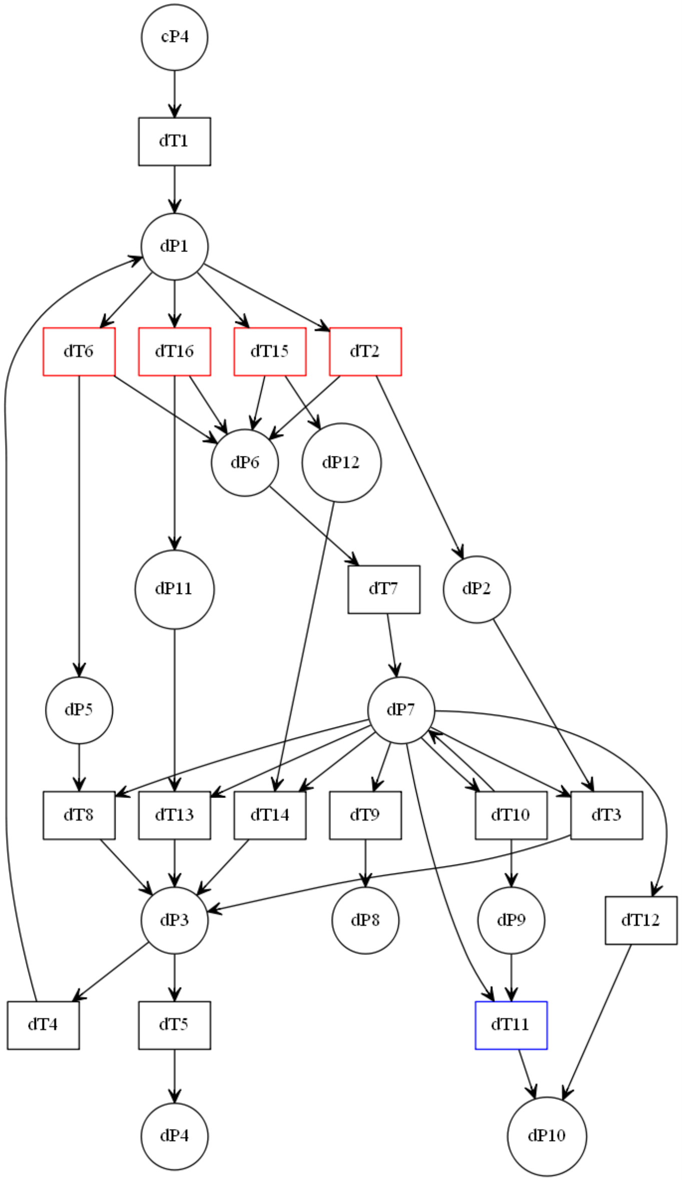
- cT2 – Load victim's browser with script to get information
- cT5 – Cause victim's browser to run command
- cT8 – Load victim's browser with script to perform actions
- cT11 – Load victim's browser with script to request other web sites
- cT14 – Load victim's browser with false information



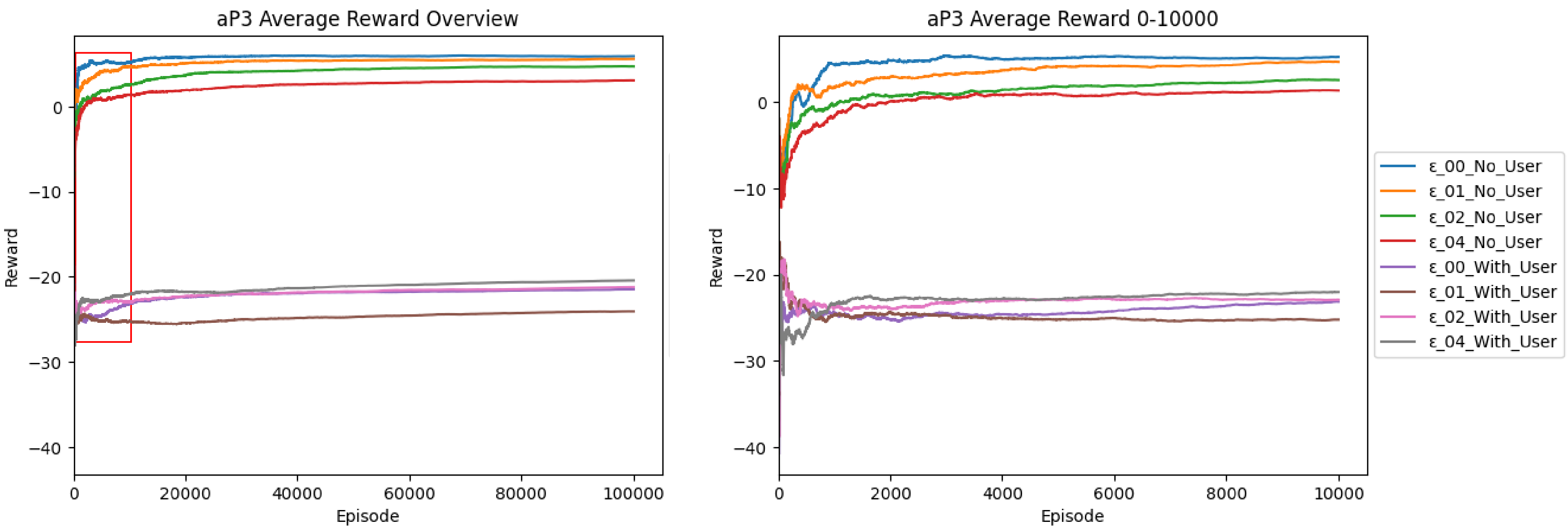
Cross Site Scripting – Goals Phase

- When aP4, bP7, cP3 and dP1 are marked:
  - ---Attacker---
  - Sets Rates – dT6 = 10, dT16 = 10 dT15 = 10, dT2 = 10
  - Average Reward = 45
- When aP14, bP18, cP23 and dP1 are marked:
  - ---Defender---
  - Sets Rates – dT11 = 10
  - Average Reward = 10

- dT2 – Read application
- dT6 – Gain privileges
- dT15 – Execute unauthorized code
- dT16 – Modify application



Defender Average Reward



- Average reward of 3.06 for ε-0.04 no user
- Average reward of -20.42 for ε-0.04 with user

Defender Strategies Comparison

- For the same Petri net marking, the defender would choose a different solution 80% of the time if the computer system user were present
  - Example: If the marking at the start of the exploit phase was bP17 = 1, then:
  - Defender strategy no computer system user: [cT18 = 0, cT19 = 0, cT20 = 0, cT21 = 10, cT22 = 10]
  - Defender strategy with computer system user: [cT18 = 0, cT19 = 0, cT20 = 0, cT21 = 0, cT22 = 0]

Results Discussion

- All defender scenarios showed improvement over time.
- Most variation occurs in the first 10,000 episodes
- The computer system user impacts the defender
  - Lower average reward
  - Strategies change – especially if a marking was seen more than 30 times

Future Work - UAV Model

UAV OV1 State Model

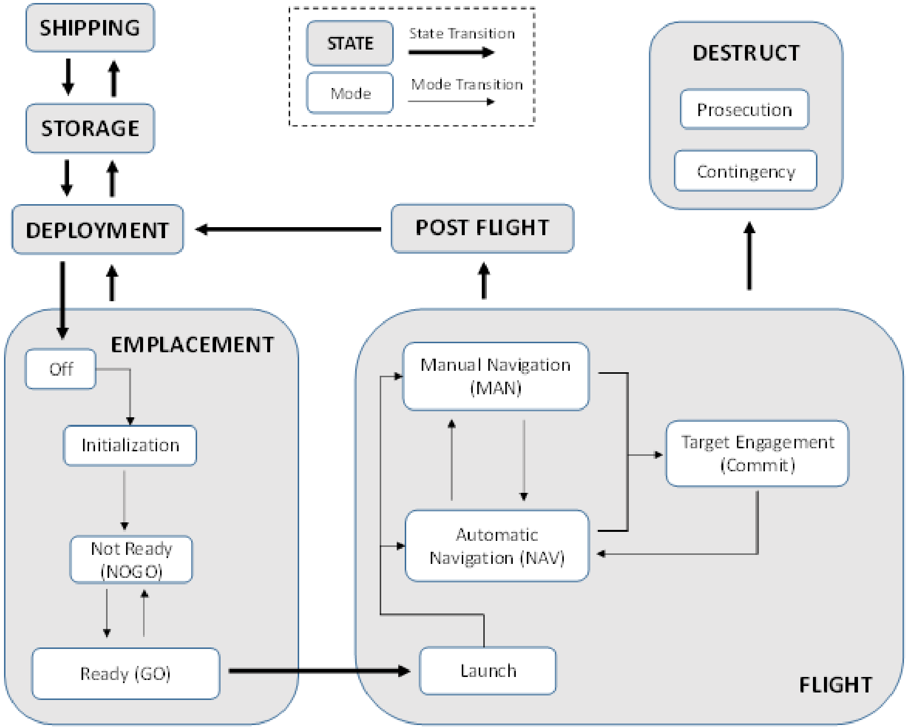
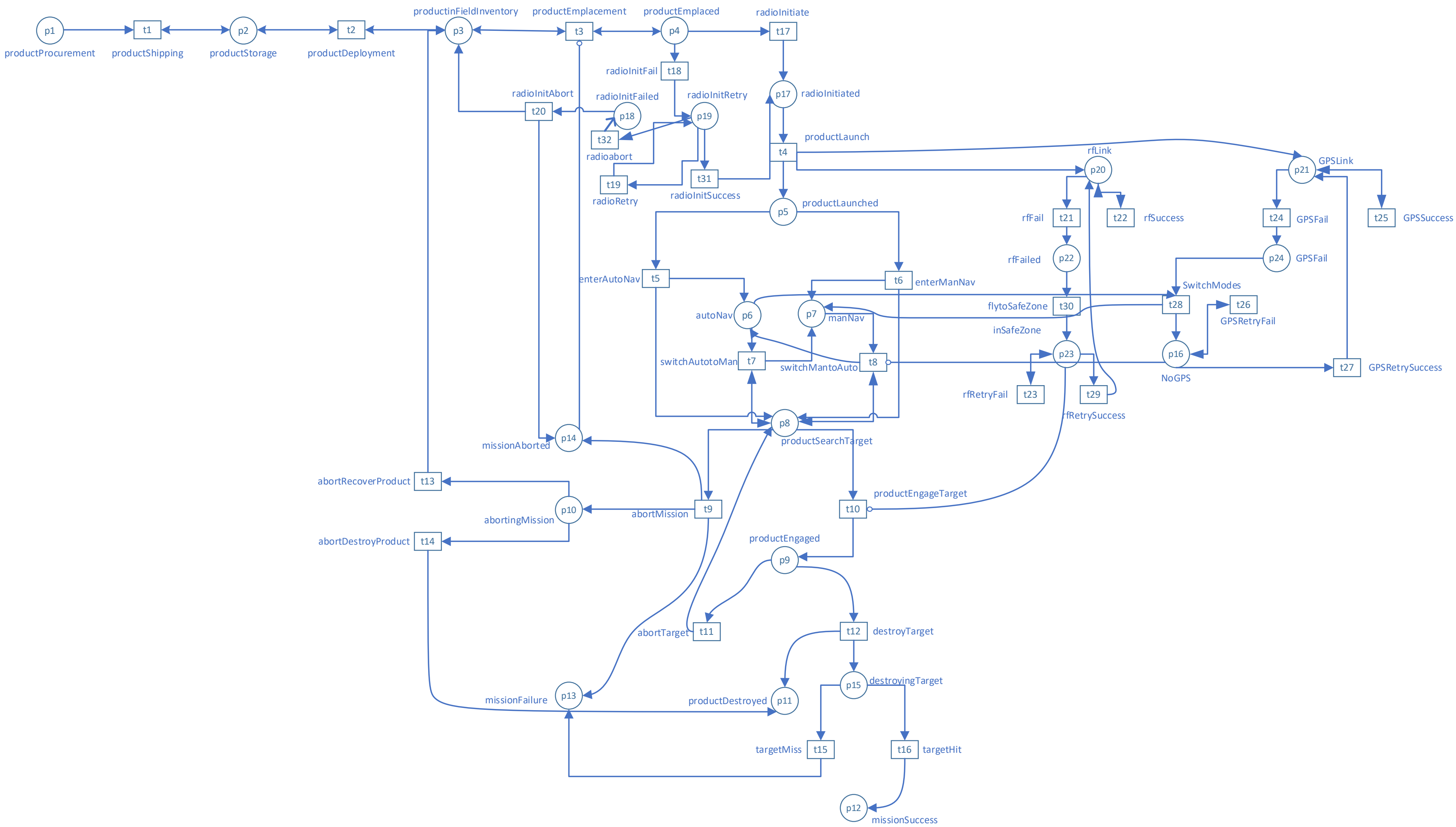


Figure 1. System States and Modes

- Can we go from a high level state model to PNPSC net?

THE UNIVERSITY OF ALABAMA IN HUNTSVILLE

UAV PNPSC Net



UAV PNPSC Net (cont.)

- With PNPSC net, can analyze different defense solutions to determine:
  - Impact to successfully completing mission
  - Defense solutions effectiveness against different attackers
  - Time duration before system recovers from attack

Summary

Summary

- Modeling with PNPSC provides opportunities to utilize machine learning to improve design solutions
- Reinforcement learning allows learning within a PNPSC model without training sets
- Using a database mitigates the state space explosion compute issue at the expense of increased run time
- The inclusion of the computer system user impacts the defender's strategy

Summary (Cont.)

- You can answer questions such as
  - Does implementing a defense solutions increase the chance to detect an attack or significantly impact the users
  - Does increasing my skillset as an attacker make the most sense
  - Does changing my defense solution increase my chance of accomplishing the mission
  - How long does it take for my system to recover from an attack

References

[1] Petty, Mikel D, Phil M Showers, Tymaine S Whitaker, John A Bland, Walter Alan Cantrell, C Daniel Colvett, and Katia P Maxwell. 2019. “Modeling Cyberattacks with Extended Petri Nets: Research Program Update.” In _Proceedings of the 2019 AlaSim International Conference and Exposition. Huntsville, AL_, 11. Huntsville, AL.

[2] C. A. Petri; Kommunikation mit Automaten, Ph.D. Thesis, Schriften des Rheinisch-Westfälischen Institutes für Instrumentelle Mathematik an der Universität Bonn Nr. 2, 1962.

[3] Zakrzewska, Anita N., and Erik M. Ferragut. 2011. “Modeling Cyber Conflicts Using an Extended Petri Net Formalism.” In _2011 IEEE Symposium on Computational Intelligence in Cyber Security (CICS)_, 60–67. Paris, France: IEEE. https://doi.org/10.1109/CICYBS.2011.5949385.

[4] Sutton, Richard S., and Andrew G. Barto. 2018. _Reinforcement Learning: An Introduction_. 2nd ed. Adaptive Computation and Machine Learning Series. Cambridge, Massachusetts: The MIT Press.

[5] Bland, John A., Mikel D. Petty, Tymaine S. Whitaker, Katia P. Maxwell, and Walter Alan Cantrell. 2020. “Machine Learning Cyberattack and Defense Strategies.” _Computers & Security_ 92 (May): 101738. https://doi.org/10.1016/j.cose.2020.101738.

[6] C. D. Colvett, M. D. Petty, J. A. Bland and K. R. Baker, "Simulating Cyberattacks with a Petri Net Discrete Event Simulator," _2019 International Conference on Computational Science and Computational Intelligence (CSCI)_, 2019, pp. 67-71, doi: 10.1109/CSCI49370.2019.00018.

[7] The MITRE Corporation, “Common Attack Pattern Enumeration and Classification”. https://capec.mitre.org/, October 2, 2021

[8] Cantrell, Walter A, Katia P Mayfield, Mikel D Petty, Tymaine S Whitaker, and John A Bland. 2018. “Structured Face Validation of Extended Petri Nets for Modeling Cyberattacks.” In _Proceedings of the 2017 AlaSim International Conference and Exposition, Huntsville, AL.