Petri Net Models of Adversarial Scenarios in Safety and Security

David H. Collins and Aparna V. Huzurbazar

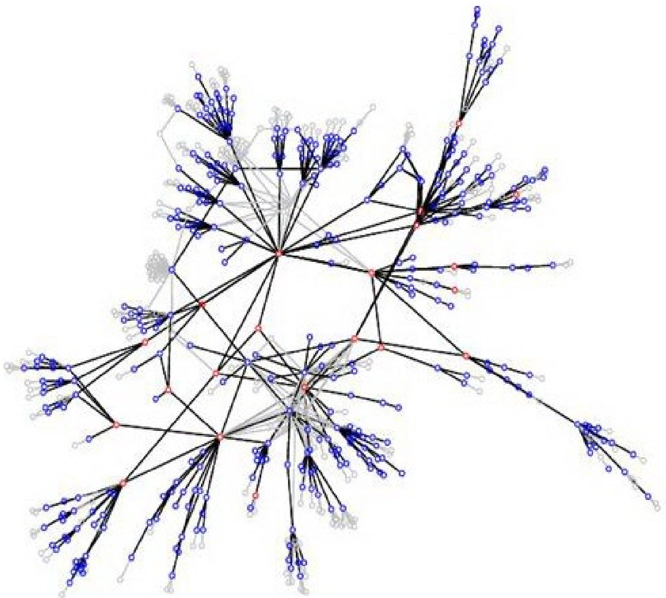
Statistical Sciences Group
Los Alamos National Laboratory

dcollins@lanl.gov
aparna@lanl.gov

Outline

- Adversarial scenarios with concurrent autonomous/interactive actors
- Modeling methodologies
  - Markov and semi-Markov processes (flowgraphs)
  - Stochastic game theory
  - Event graphs
  - Bayesian networks
- Overview of Petri nets
  - General concepts, history
  - Generalized stochastic Petri nets
  - Use as a scenario elicitation tool
  - Use as a simulation tool
- Simulation of Petri nets
- Application example

Adversarial systems and scenarios



- ***System***: a group of entities that interact to function as a whole
  - We view systems as ***evolving, concurrent streams of events and actions***
- ***Adversarial systems*** are characterized by conflict between parties with opposing goals
  - Actions of adversaries are concurrent and interdependent
  - Outcomes are known only probabilistically
- ***Scenario:*** a postulated sequence or development of events
- We look at tools for
  - Eliciting adversarial scenarios from subject matter experts
  - Quantitative modeling of scenario outcomes

Modeling methodologies

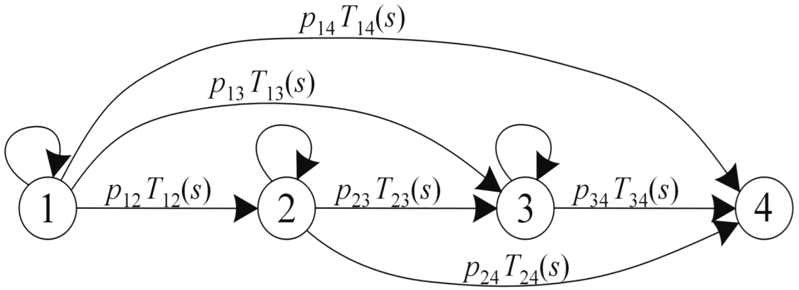
- State-space (e.g., Markov) models
- Event graphs
- Bayesian networks
- Stochastic game theory
  – Adds evolutionary/learning behavior to game theory
- Stochastic programming, evolutionary design, . . .

These methods (and others) assume sequential actions, serialized sample paths, or situations static in time.

“Adversaries” versus “defenders” implies multiple concurrently evolving streams of events

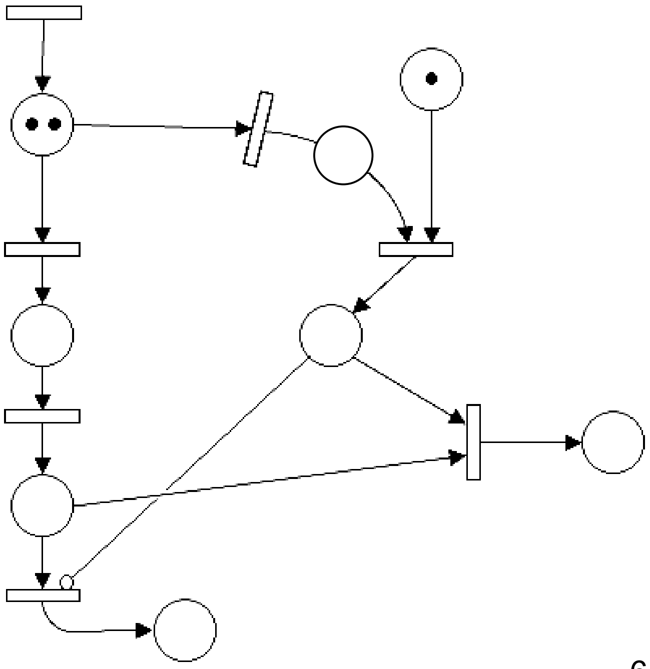
Markovian stochastic networks

- Multistate models for predicting time to occurrence of an event (Huzurbazar 2005, Collins, Warr, and Huzurbazar 2013)
- Markov process – present state (not history) determines the future
  - Discrete-time, discrete-state Markov chain
  - Continuous time Markov chain ($\Rightarrow$ exponential wait time distributions)
  - Semi-Markov process (arbitrary wait time distributions)
- Markov process extensions: lots of alternatives
  - Markov reward processes, Markov decision processes, hidden Markov, hidden semi-Markov models
  - *n*th-order Markov processes (richer dependency between successive states)
- State-space models can handle arbitrarily complex scenarios, but . . .
  - Cost is combinatorial explosion of states, loss of interpretability



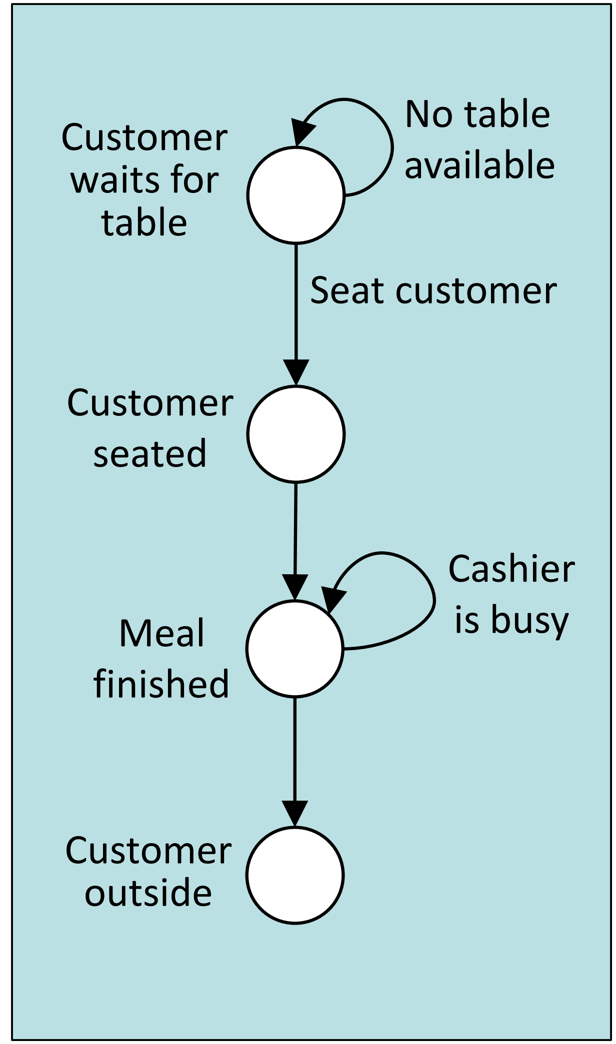
Petri Nets

- Developed by Carl Petri for analysis of parallel computer architectures (Petri 1962)
  - Based on a strong mathematical foundation (Peterson 1977, Reisig 1982)
- Add multiple entities, concurrency to state transition diagrams
  - Places
  - Transitions
  - Tokens (represent actors, passive entities, or event triggers)
- Twofold purpose
  - A visual communication aid to elicit models of system behavior
  - A tool for developing quantitative simulation models



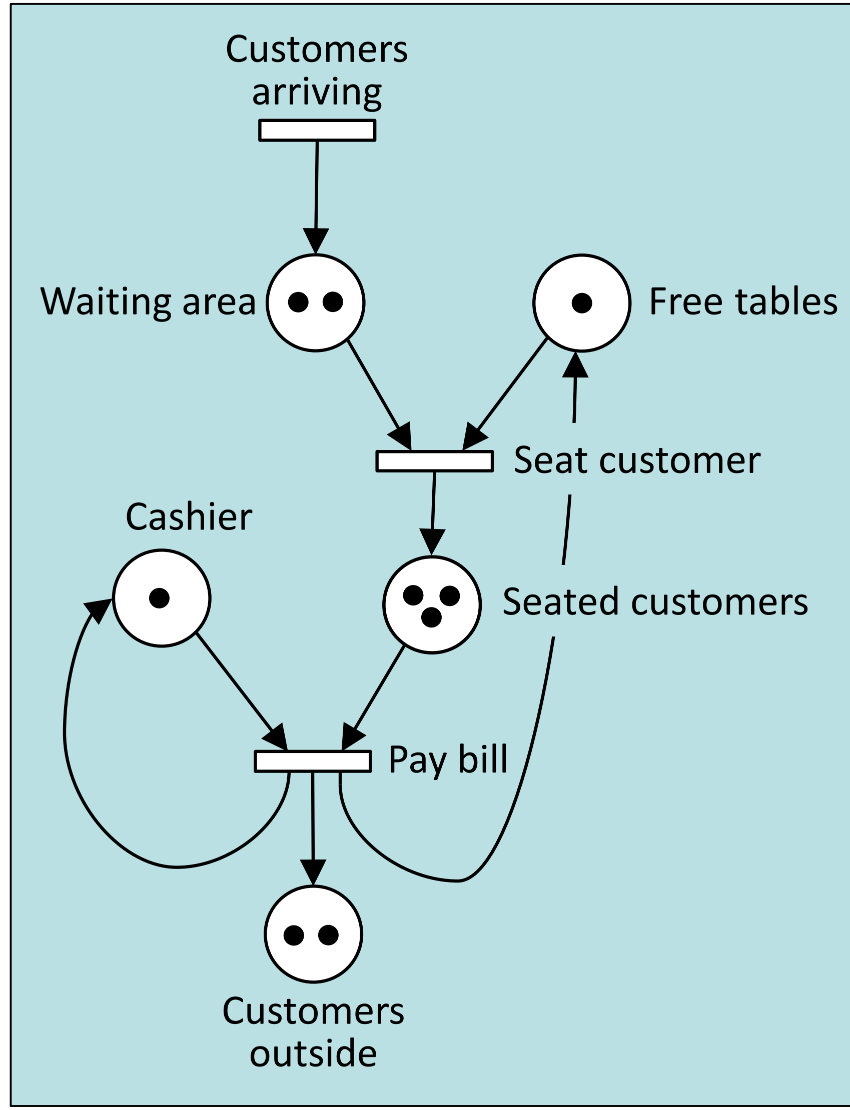
Example: State transition model of restaurant service

- A state transition model represents the time history of a *single actor* (customer)
- Interaction between actors can only be modeled indirectly (e.g., “No table available,” “Cashier busy”)
- Multiple actors can only be incorporated by proliferating states (e.g., states for “Customer waits for one other,” “Customer waits for two others,” etc.)
- Deadlock or inability to reach a given state may occur due to factors that are not modeled by the state graph
- A more powerful representation is needed for modeling adversarial scenarios



Petri net model of restaurant service

- *Tokens* represent concurrent actors or other entities occupying *places*
- *Transitions* fire when each input place has a token, and can generate multiple output tokens
- Multiple interacting actors can be represented in an obvious way
- Conflict and cooperation can easily be represented
- Mathematical formalism allows determination of deadlocks, reachable states, etc.



Extensions to the basic Petri net model

- Timed nets – deterministic transition times
- Stochastic nets
  - Explicit probabilities for non-deterministic transitions
  - Probabilistic transition times – Markov (exponential) or semi-Markov (arbitrary distribution)
- Logic extensions, e.g., inhibitory arcs
- Transitions that generate multiple tokens
- Hierarchical decomposition of nets

Example scenario: Storage locker break-in

Intruders penetrate a chain-link fence. With probability $p = 0.9$, a silent alarm is transmitted to a security company.

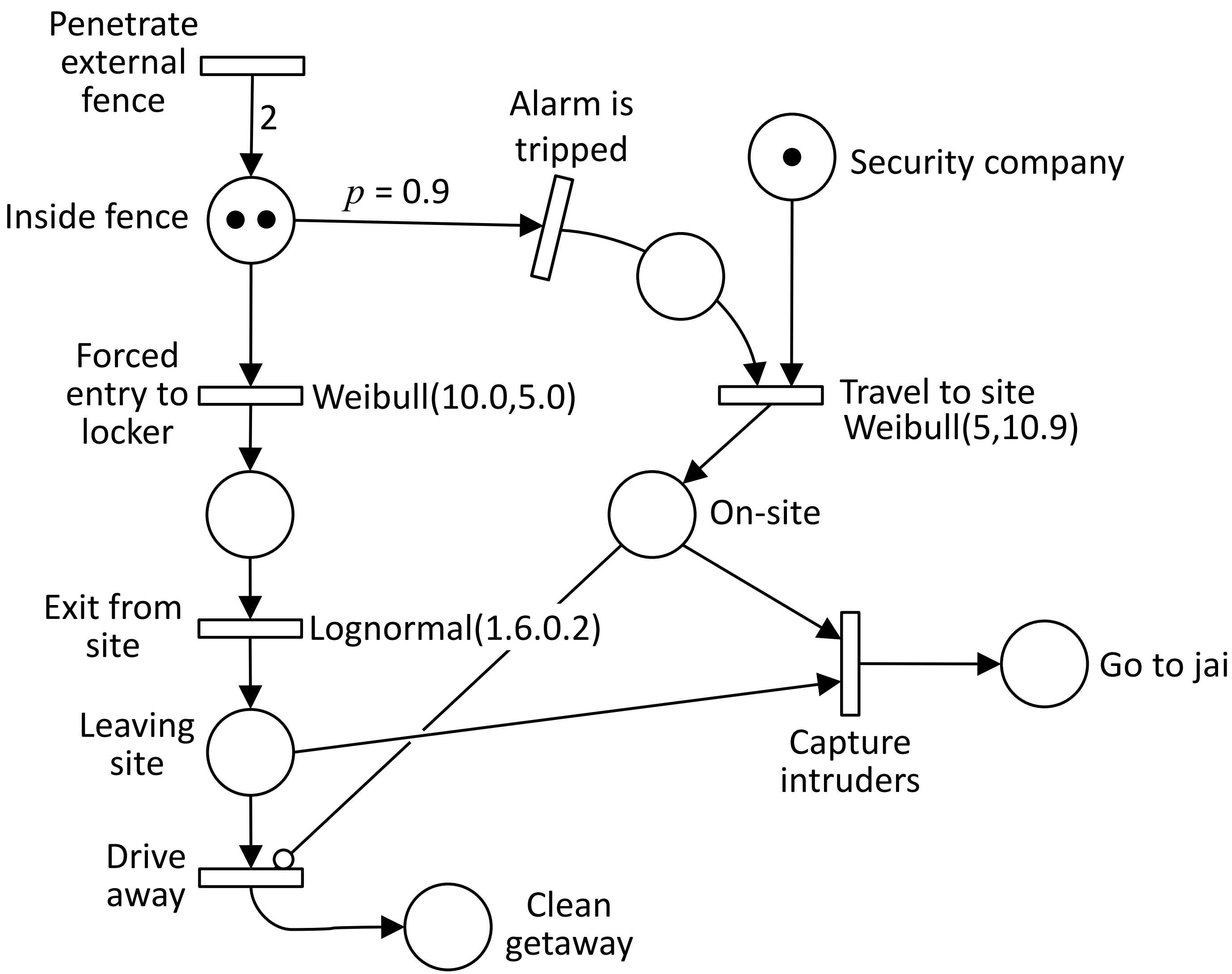
Patrol is dispatched from the security company; travel time varies depending on the location of the nearest patrol car that is able to respond (not on another call). Travel time $T \sim$ Weibull(5, 10.9).

Intruders forcibly open the storage locker. Time taken for this varies depending on the type of lock, etc. $T \sim$ Weibull(10, 5).

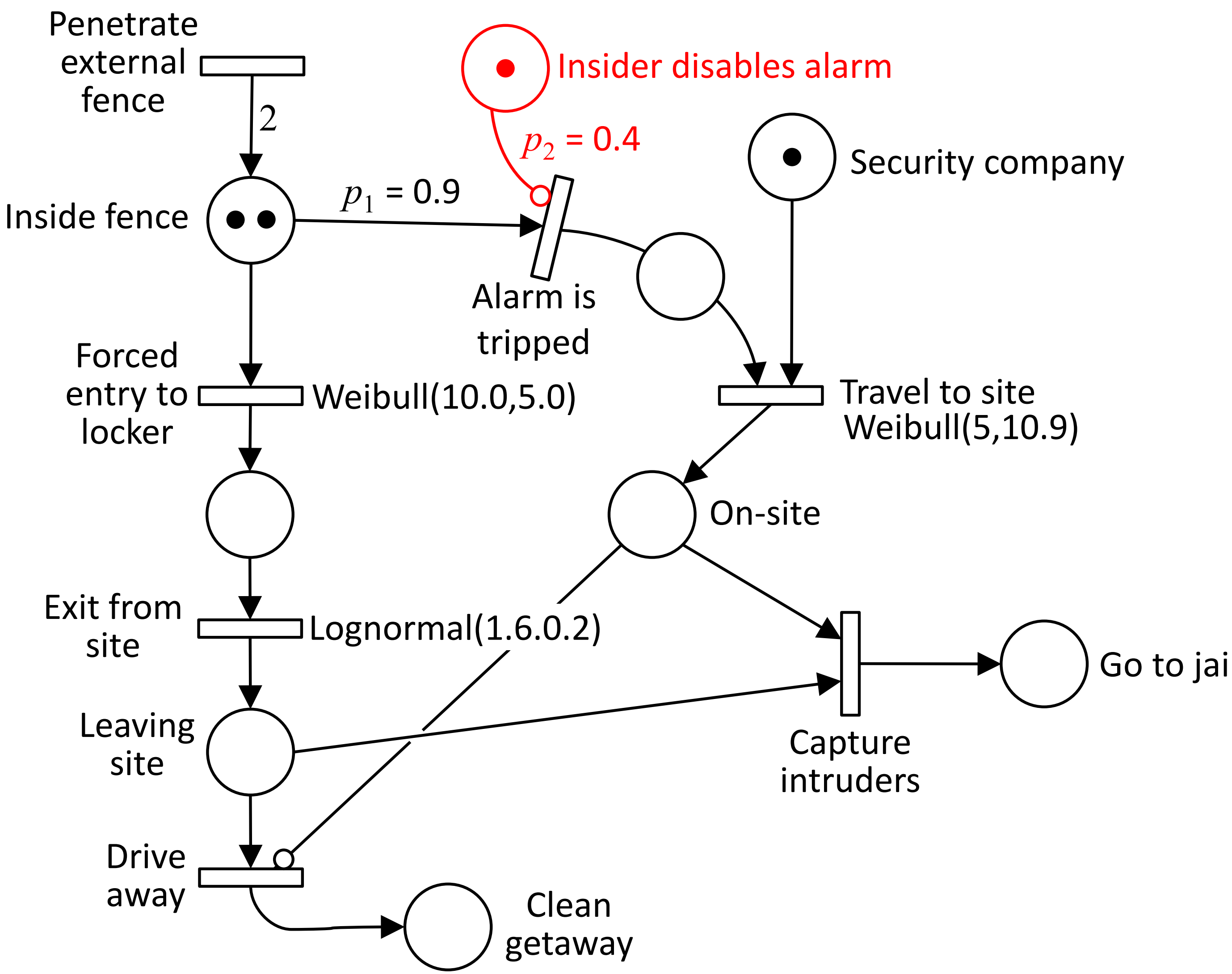
Intruders remove locker contents and exit. This is also a stochastic variable, distributed $T \sim$ Lognormal(1.6, 0.2).

Concurrently, if the alarm was tripped the security patrol has been traveling to the scene. If the patrol arrives before the intruders exit, they are captured. If not, they make a clean getaway.

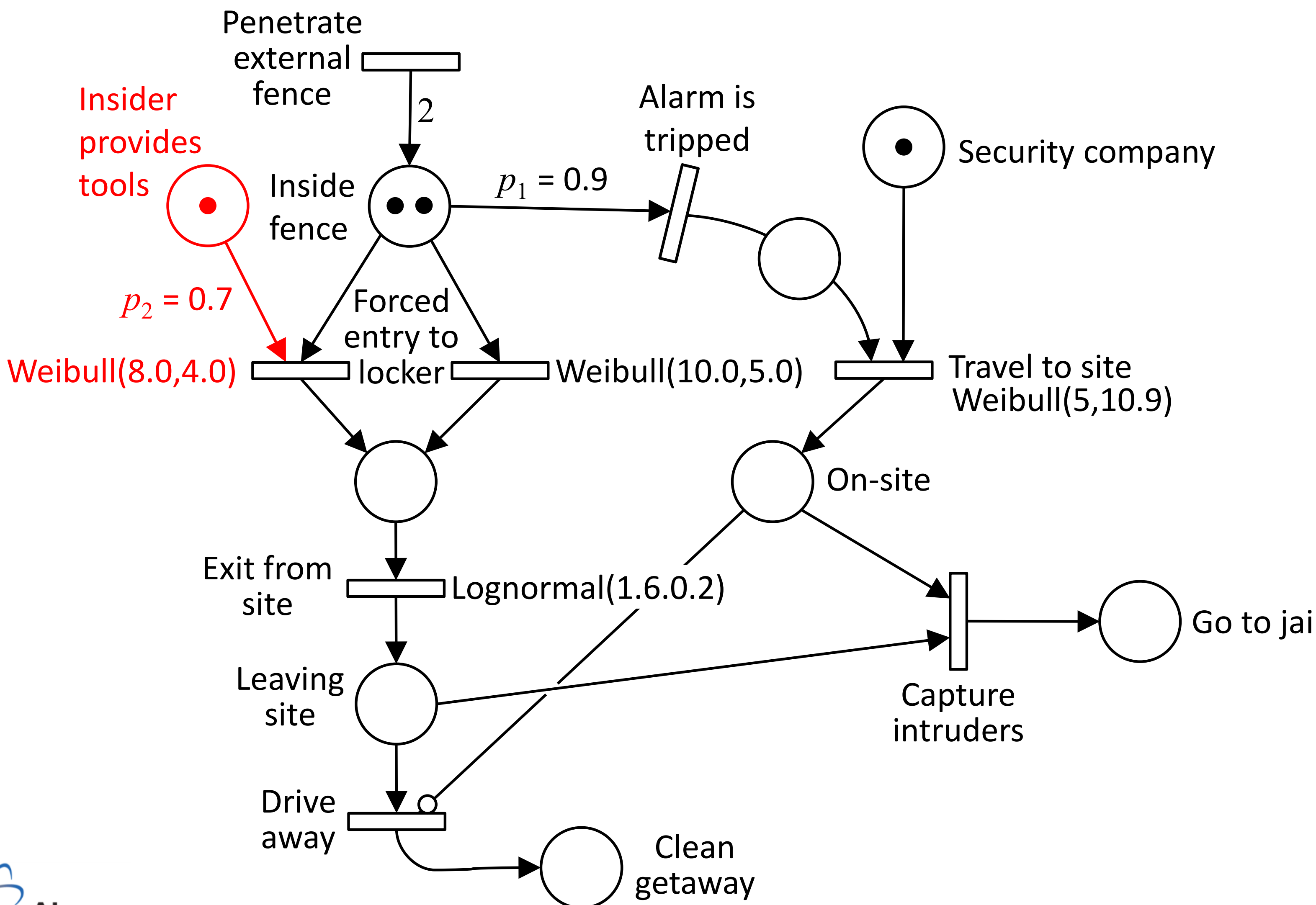
Petri net model for storage locker break-in



Storage locker break-in (insider threat case 1)



Storage locker break-in (insider threat case 2)

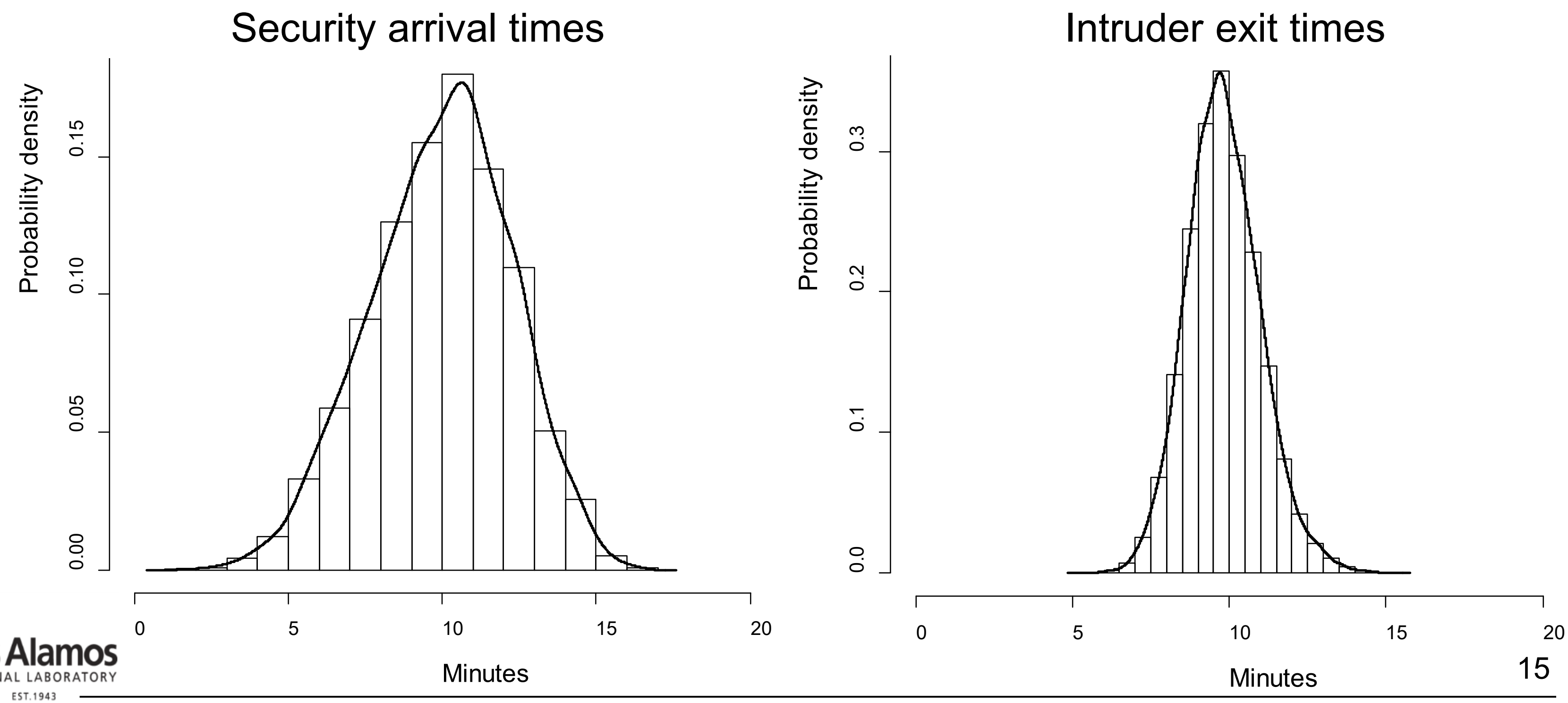


Implementation options for Petri net simulation

- We are using the statistical programming language R
- Scenario-specific procedural code
  - Fast implementation for simple nets
  - Error-prone for complex nets, difficult to debug
  - Can't be extended to provide user-friendly net definition, graphical user interface
- Generalized object-oriented framework
  - More transparent: classes for Net, Place, Transition, etc.
  - Extensible to provide easy net definition, graphical user interface(s)
- Simulation allows sensitivity analysis, optimization over defensive countermeasures

Example simulation output – storage locker break-in

```
Monte Carlo iterations: 10000
Elapsed time: 0.281 seconds
Probability of escape = 0.7479
Probability of security alarm = 0.5416
Mean travel time for the security company = 9.96 minutes
Mean exit time for the intruders = 9.82 minutes
```



Summary

- Approaches to modeling adversarial scenarios with concurrent autonomous/interactive actors
  - Markov and semi-Markov processes (flowgraphs)
  - Stochastic game theory
  - Event graphs
  - Bayesian networks
- Petri nets overcome some deficiencies of other methods
  - Ability to model parallel, concurrent flows of events (e.g., attacker and defender actions)
  - Stochastic extensions allow statistical analysis (including Bayesian)
  - Can be used as a scenario elicitation tool, as well as for simulation
- Ongoing work
  - Analysis/representation of more complex scenarios
  - Object-oriented Petri net simulation framework
  - Optimization over defender actions and costs

References

- Bailey, M. D., Shechter, S. M., Schaefer, A. J. (2006), "SPAR: Stochastic programming with adversarial recourse," *Operations Research Letters* 34, 307-315.
- Collins, D., Warr, R., Huzurbazar, A (2013), "An Introduction to Flowgraph Models for Engineering Systems," *Journal of Risk and Reliability* 227(5), 461-470.
- Goeree, J. K., Holt, C. A. (1999), "Stochastic game theory: For playing games, not just for doing theory," *Proceedings of the National Academy of Sciences* 96, 10564-10567.
- Huzurbazar, A. (2005), *Flowgraph Models for Multistate Time-to-Event Data*, New York: Wiley.
- Peterson, J. L. (1977), "Petri nets,", *ACM Computing Surveys* 9 (3), 223-252.
- Petri, C. A. (1962), *Kommunikation mit Automaten*, Ph.D. Thesis, University of Bonn.
- Pietre-Cambacedes, L., and Bouissou, M. (2010), "Modeling Safety and Security Interdependencies with BDMP (Boolean Logic Driven Markov Processes)," *2010 IEEE International Conference on Man and Cybernetics*, 2852-2861.
- Reisig, W. (1982), *Petri Nets: An Introduction*, Heidelberg: Springer-Verlag.
- Reisig, W. (1992), *A Primer in Petri Net Design*, Heidelberg: Springer-Verlag.
- Schruben, L. (1983), "Simulation modeling with event graphs," *Communications of the ACM* 28 (11), 957-963.
- Villacorta, P. J., Pelta, D. A. (2010), "Evolutionary design and statistical assessment of strategies in an adversarial domain," *2010 IEEE Congress on Evolutionary Computation*, 1-7.
- Ye, N., Zhang, Y., Borror, C. (2004), "Robustness of the Markov chain model for cyber-attack detection," *IEEE Transactions on Reliability*, 53(1), 116-123.