A novel approach to Large Systems Performance Prediction via Markov analysis

Roberto Petrucci, Domenico Vigilante
Finmeccanica - Selex Sistemi Integrati
Via Tiburtina km 12400, 00131 Rome, Italy

Roadmap

- Introduction
- Mathematical Framework
  - *Definitions and metrics*
  - *Large Systems State Evolution*
  - *Engagement Timeline*
  - *Markov Model*
- Study Case
  - *Oil Camp Protection Overview*
  - *Frontal Assault Scenario Description*
  - *Frontal Assault Scenario Results*
- Conclusions

Introduction

- This paper provides a preliminary answer to the rising needs for performance prediction in the Large System (LS) area.
- Up to now, performance studies have been mostly confined to sub-system level modeling and simulation techniques.
- Even if each company has a mature modeling capability with respect to its own produced sub-system, the overall performance prediction theory is still embryonic and mostly based on huge Monte Carlo simulations , useful only to test and refine a defined system architecture.

Introduction

- However, simulations provide poor support to system design, as it is well known to SE community: they are time and resource consuming, requires a lot of coding effort and are not oriented to sensitivity and trade-off studies.
- What is missing is a closed formula model to be used in the preliminary design stage, to support the system architecture definition and, from a marketing perspective, to support the production of high-quality technical proposals.

Introduction

- Our proposed approach offers a solution to these drawbacks, providing a guideline to those engineers facing the challenge of LS devising.
- Aim of this work has been twofold:
  - Define a suitable mathematical framework, including performance metrics, modeling and calculus procedures.
  - Perform a trade study, using the model mentioned above, to assess the performance improvements obtained adding a radar sub-system to a notional study case: the protection of an Oil Camp sited in an area affected by critical terroristic activity.

Mathematical Framework

Definitions and metrics 1/3

- We propose the following practical definition, mostly because it is functional to the aims of this report.
  - ***An integrated system is a set of heterogeneous items (named subsystems) playing different roles and directed towards a common objective, that is the mission execution.***
- In most cases, different items have no comparable performances: the metrics used to define the "quality" of item A may be completely different from those used for item B, provided that item A could be a ground sensor whereas item B could be a truck, a fence or a command and control system.

Mathematical Framework

Definitions and metrics 2/3

- A straight forward approach to the metrics problem begins with the definition of the overall LS performance as the probability to succeed against a specified set of threatening scenarios.
- The overall success probability depends on the success probability of the items composing the system.
- We propose to "scale" the system approach to the subsystem level:
  - ***The performance of a subsystem is defined as the probability that such subsystem carries out its role within the LS mission in the required timeframe.***

Mathematical Framework

Definitions and metrics 3/3

- This approach leads to a **Markov** model to represent the scenario within following assumptions:
  - *Assumption 1:* The number of states significant and representative of the system evolution is finite
  - *Assumption 2:* Each system state is defined by a combination of each scenario actor state. For scenario actors is intended each item playing a role inside the scenario: threats, subsystems, environment etc.
  - *Assumption 3:* The transition from one state to the next one is ruled by a specific combination of subsystem performances. These probabilities may be non-stationary, i.e. they may change with time.

Mathematical Framework

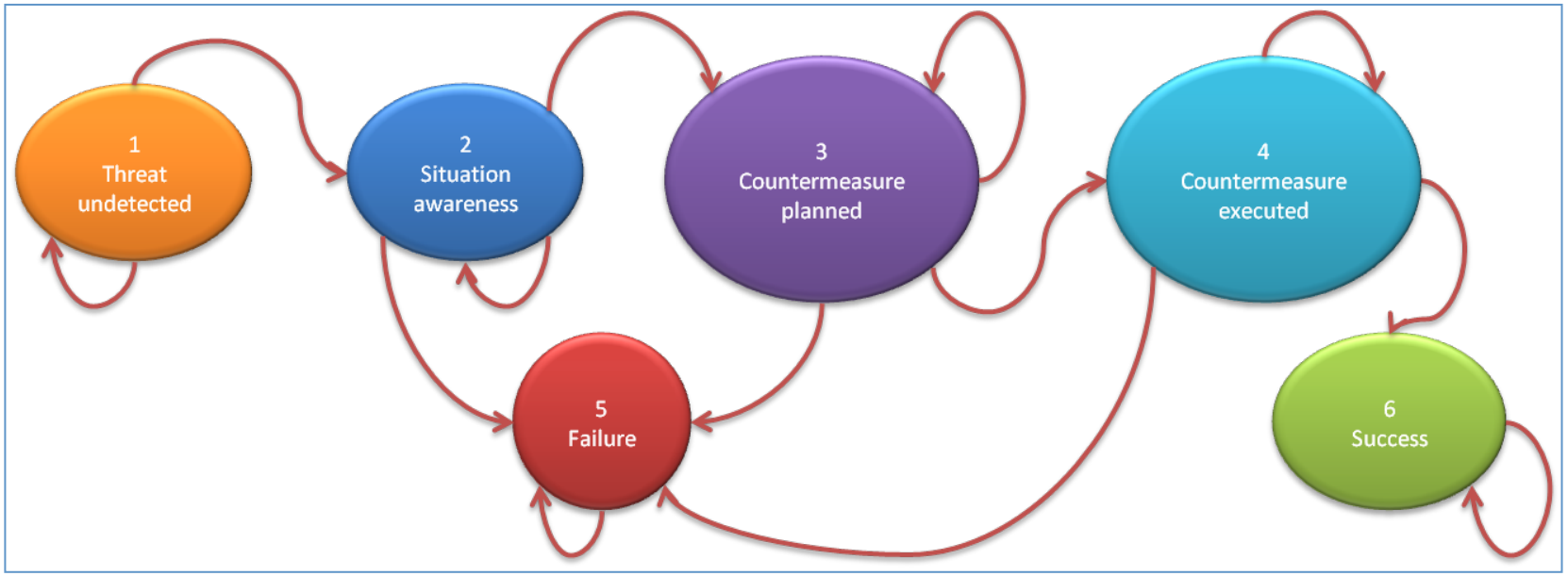
LS state evolution 1/2

- The mission execution of a LS can be summarized in a subset of macro-states:
  1. Threat Undetected: this state represents the "*at rest*" system state, when no active countermeasure is operated.
  2. Situation Awareness: the sensors suite, as well as intelligence and data mining techniques, identify a specific threat.
  3. Countermeasures Planned: the decision makers, supported by information management systems and decision support systems, plan specific countermeasures against the identified threat.
  4. Countermeasures Executed: the subsystems involved in the countermeasures application are activated and forced in the needed states.
  5. End Game: we may have multiple mission endings, from a complete success to a complete failure, with all the intermediate cases.

Mathematical Framework

LS state evolution 2/2

States Diagram of a general Large System

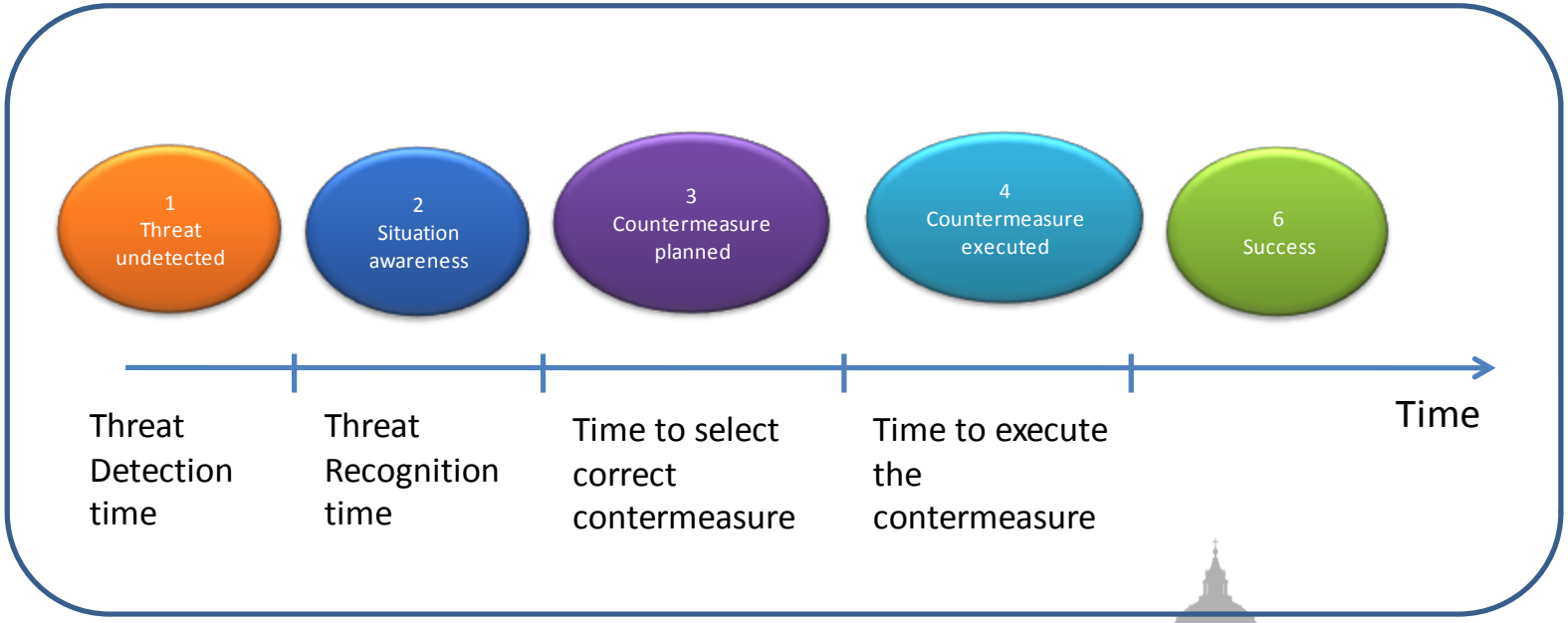


The state transition is provided by the performance of the actors

Mathematical Framework

Engagement Timeline 1/3

- Each sub-system performs its functions according to a time-based script, which we will refer to as *Engagement Timeline*. It is the common language that will help us to integrate metrics coming from different sub-systems



Mathematical Framework

Engagement Timeline 2/3

- From a System Designer point of view, the timeline should be defined from the end point above. That is, from state 6 up to state 1:
  - In order to properly counteract the specific threat of the scenario, the optimal countermeasure needs a certain time to be executed;
  - The command&control systems need time to select that particular countermeasure, after the threat has been correctly identified and recognized;
  - The classification and discrimination sub-systems need to observe the target for a certain time in order to operate;
  - The sensor suite (radar, acoustic, electro-optic, cameras, etc.) needs time to acquire and initiate the observation.

Mathematical Framework

Engagement Timeline 3/3

- The end-state "Failure" has not been reported in this example; however, if one or more subsystems fail to correctly play their role, we can reach the bad end-state following many different routes.
- To further aggravate the complexity of this kind of scenario evolutions, we must acknowledge that many large systems (mostly in homeland protection and defense) are over-specified; that is, they offer redundant solutions to recover from some sub-systems failure.
- In other words, we must recognize that the number of "what if" cases exponentially increases as we increase the complexity of the large system and of its behavior.

Mathematical Framework

Markov Model 1/2

- Markov condition: the system state at time k depends **only** on the system state at time k-1.
- State vector *s* in which the element i^{th} is the probability of the system to lies in the state "i" at a given time:

$$s(k) = \begin{bmatrix} p_1(k) & p_2(k) & \dots & p_n(k) \end{bmatrix}$$

- The system evolution is represented by the transition matrix:

$$T(k) = \begin{bmatrix} p_{11}(k) & p_{12}(k) & \dots & p_{1n}(k) \\ p_{21}(k) & p_{22}(k) & \dots & p_{2n}(k) \\ \dots & \dots & \dots & \dots \\ p_{n1}(k) & p_{n2}(k) & \dots & p_{nn}(k) \end{bmatrix}$$

Mathematical Framework

Markov Model 2/2

- One of the biggest limitations within the Markov theory boundaries is the Markov Condition, which implies that we cannot model (at least in a straight forward manner) memory-based systems with this mathematical tool.
- Nevertheless, common experience says that complex systems are usually far from being memory-less.
- Following example try to explain how to bypass this limitation and make the Markov chain useful to LS evaluation performance

Mathematical Framework

Example 1/6

- An enemy vehicle is approaching our defended site, with the purpose of breaking inside and destroying some facilities.
- The best countermeasure we should select is a plain intercept mission by means of armed security contractors *before* the enemy vehicle reaches the site boundaries.
- If one of the sub-systems fails (i.e. does not execute its task or executes it overtime) the end game will be FAIL instead of SUCCESS

Mathematical Framework

Example 2/6

- The LS system mission can be summarized as follows:
  - SEARCH. Until the threat is detected the system is in the state 1.
  - RECOGNITION: When the threat is detected it is taken in charge by the early warning system that provides the target classification and identification: the time assigned to this task is the threat recognition time.
  - COUTERMEASURE PLANNING: After the recognition time the system has to plan the correct countermeasure to mitigate or neutralize the threat (typically, this phase involves computer-aided decision making); at the end of this time the large system reach the state 3.
  - COUNTERMEASURE EXECUTION: The correct countermeasure execution brings the large system to the end state “success”.

Mathematical Framework

Example 3/6

- Let us define following Markov states:
  - State 0: Security Patrol not at Intercept Point
  - State 1: Security Patrol at Intercept Point
  - State 2: Intruder reaches the Intercept Point
  - State 3: Intruder enters the site
- The transition probability from state 2 to state 3 clearly depends on system state *before* state 2; in other words, security patrol presence at IP affects the probability that intruders break in.

Mathematical Framework

Example 4/6

- Mathematically speaking:
  - $P_{23}(k)$ = Pr{state 3 at time k |state 2 at time k-1} depends on time k-2:
  - $P_{023}(k)$ = Pr{state 3 at time k |state 2 at time k-1 AND state 0 at time k-2 }
  - $P_{123}(k)$ = Pr{state 3 at time k |state 2 at time k-1 AND state 1 at time k-2}
- Those transition probabilities clearly violate the Markov Condition.

Mathematical Framework

Example 5/6

- Let us define the system states in a slightly different way:
  - State A: “Security Patrol not at intercept point”
  - State B: “Security Patrol at intercept point”
  - State C: “Intruder reaches the intercept point AND Security not at intercept point”
  - State D: “Intruder reaches the intercept point AND Security at intercept point”
  - State E: “Intruder enter the site”

Mathematical Framework

Example 6/6

- We can now write again the transition probabilities without violating the Markov Condition:
  - $P_{CE}(k) = Pr\{\text{state E at time k} \mid \text{state C at time k-1}\}$
  - $P_{DE}(k) = Pr\{\text{state E at time k} \mid \text{state D at time k-1}\}$
- The memory information is already embedded inside the system status. This "trick" makes easier the mathematical formulation; however, the computational time needed to run the model, depending on the number of states, increases exponentially.
- The total number of states increases; it is easy to verify that this increase is equal to 2^n, where n is the memory depth we wish to emulate.

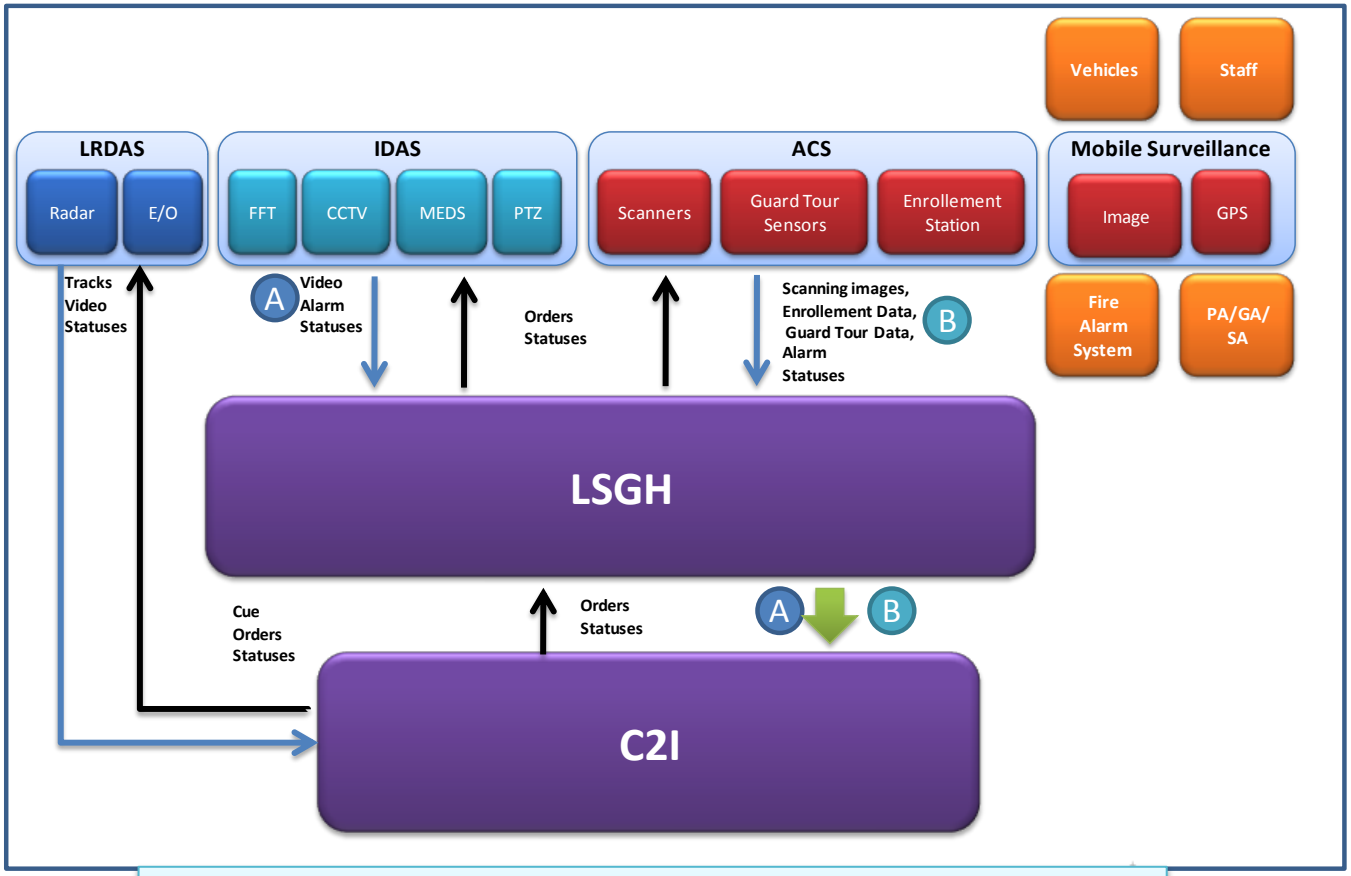
Study case

Oil Camp Protection Overview 1/2

- This is a notional representation of a typical design process for the protection of an oil camp in a hostile territory.
- It has been derived from a technical proposal produced by Finmeccanica for the protection of Oil Fields in Iraq.
- The proposed model has been applied, in that context, in order to perform a trade study and support the inclusion of a radar sensor in the Oil Field Security System (OFSS).

Study case

Oil Camp Protection Overview 2/2



OFSS top level architecture

Study case

Frontal Assault Scenario Description 1/3

- The scenario storyboard can be resumed in the following scenes:
  - The hostile vehicle is moving toward the camp
  - The Early Warning system detect, classify and identify the threat providing these information to the Command and Control System
  - The operator recognizes the threat and selects the correct procedure.
- In that specific scenario the right procedure is composed by two specific actions:
  - Alert the personnel inside the camp and lead them in a safe bunker
  - Lead a security patrol toward the predicted attack point.

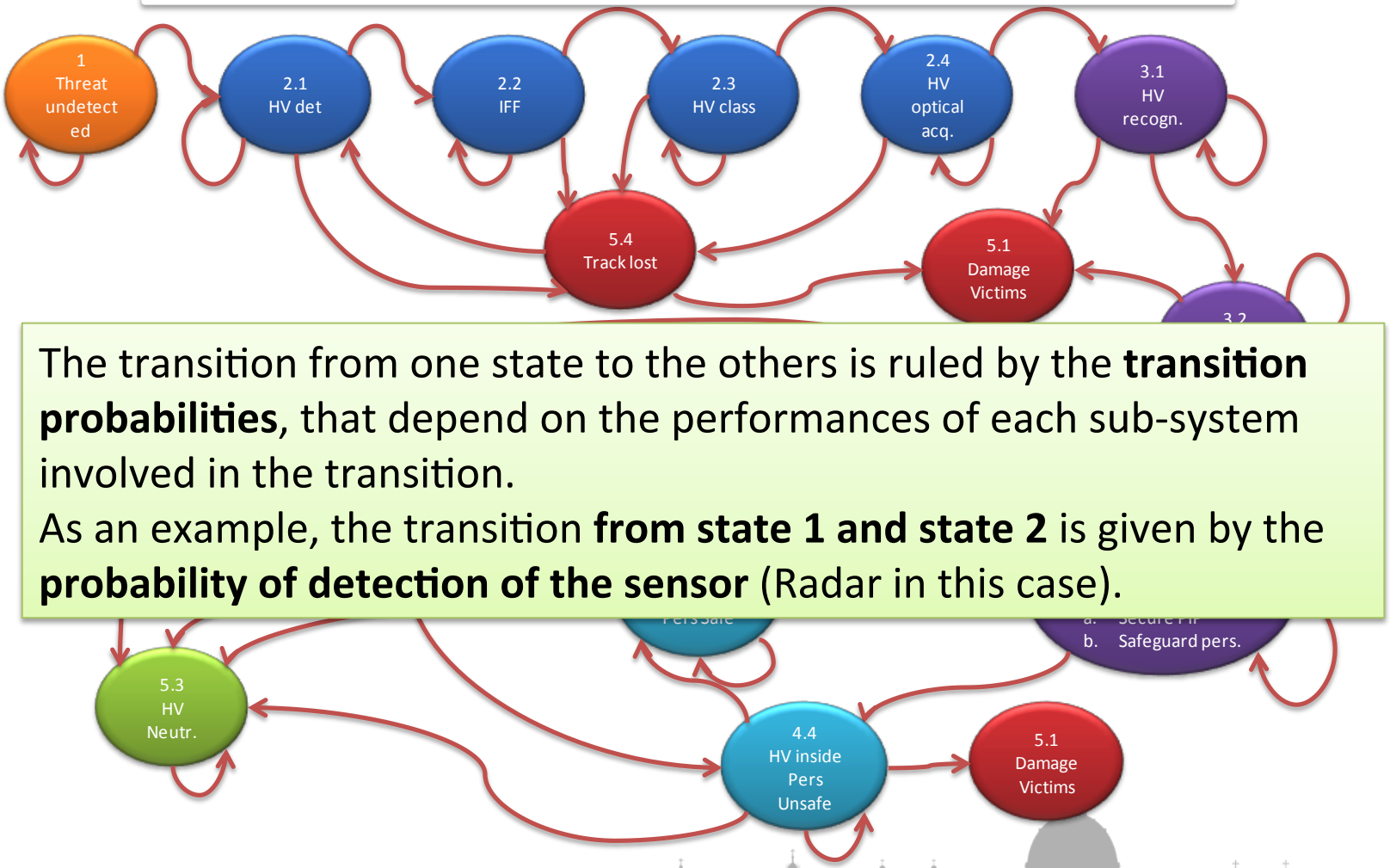
Study case

Frontal Assault Scenario Description 2/3

Threat undetected	Situation awareness	Counter-measure planned	Counter-measure executed	End Game
	• Hostile Vehicle (HV) detected by radar • Friend or Foe Identification (IFF) completed • HV classified • HV acquired by optical system	• HV recognized by the operator • Situation assessed • Correct actions directed	• Predicted Intercept Point (PIP) secured, Personnel safe • PIP secured, Personnel unsafe • HV inside camp, Personnel safe • HV inside camp, Personnel unsafe	• Damages, victims • Damages, no victims • HV neutralized by Security Personnel.

Study case

Frontal Assault Scenario Description 3/3



The transition from one state to the others is ruled by the **transition probabilities**, that depend on the performances of each sub-system involved in the transition.

As an example, the transition **from state 1 and state 2** is given by the **probability of detection of the sensor** (Radar in this case).

Study case

Frontal Assault Scenario Input Data 1/2

- **CASE A:**
  - Hostile vehicle speed: 90 km/hr
  - Sensor suite coverage: 20 km radar / 5 km EO system.
  - Probability of correct HV recognition within 30 sec: 95%
  - Probability of correct situation assessment within 30 sec: 95%
  - Probability of correct procedure selection within 30 sec: 95%
  - Probability to secure PIP within 5 minutes from alarm: 95%
  - Probability to safeguard personnel within 5 minutes: 95%
  - Probability to win fight in case of PIP secured: 100%
  - Probability to win fight inside camp within 10 minutes: 95%

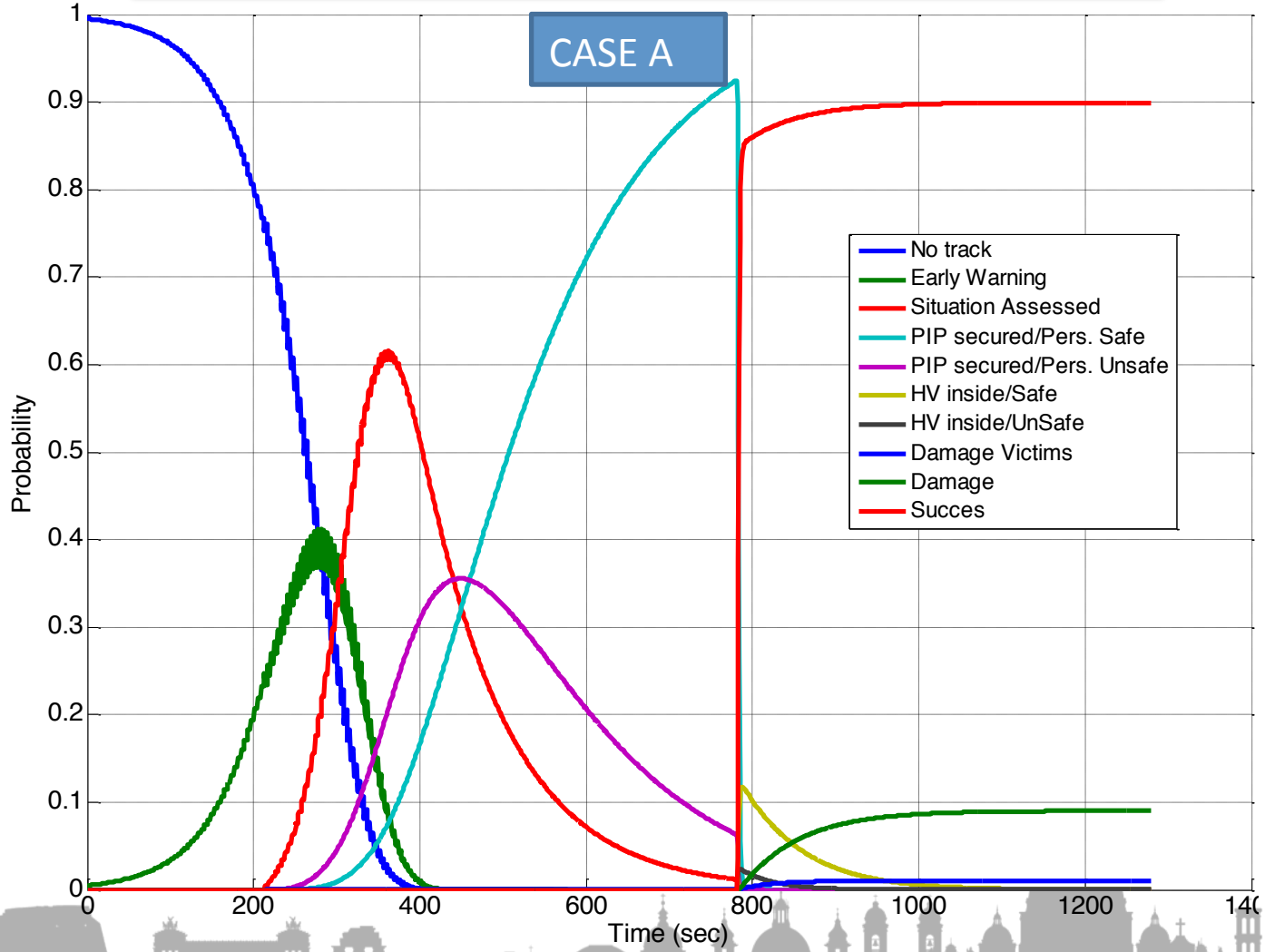
Study case

Frontal Assault Scenario Input Data 2/2

- **CASE B (degraded early warning system):**
  - Hostile vehicle speed: 90 km/hr
  - Sensor suite coverage: 2 km EO system.
  - Probability of correct HV recognition within 30 sec: 95%
  - Probability of correct situation assessment within 30 sec: 95%
  - Probability of correct procedure selection within 30 sec: 95%
  - Probability to secure PIP within 5 minutes from alarm: 95%
  - Probability to safeguard personnel within 5 minutes: 95%
  - Probability to win fight in case of PIP secured: 100%
  - Probability to win fight inside camp within 10 minutes: 95%

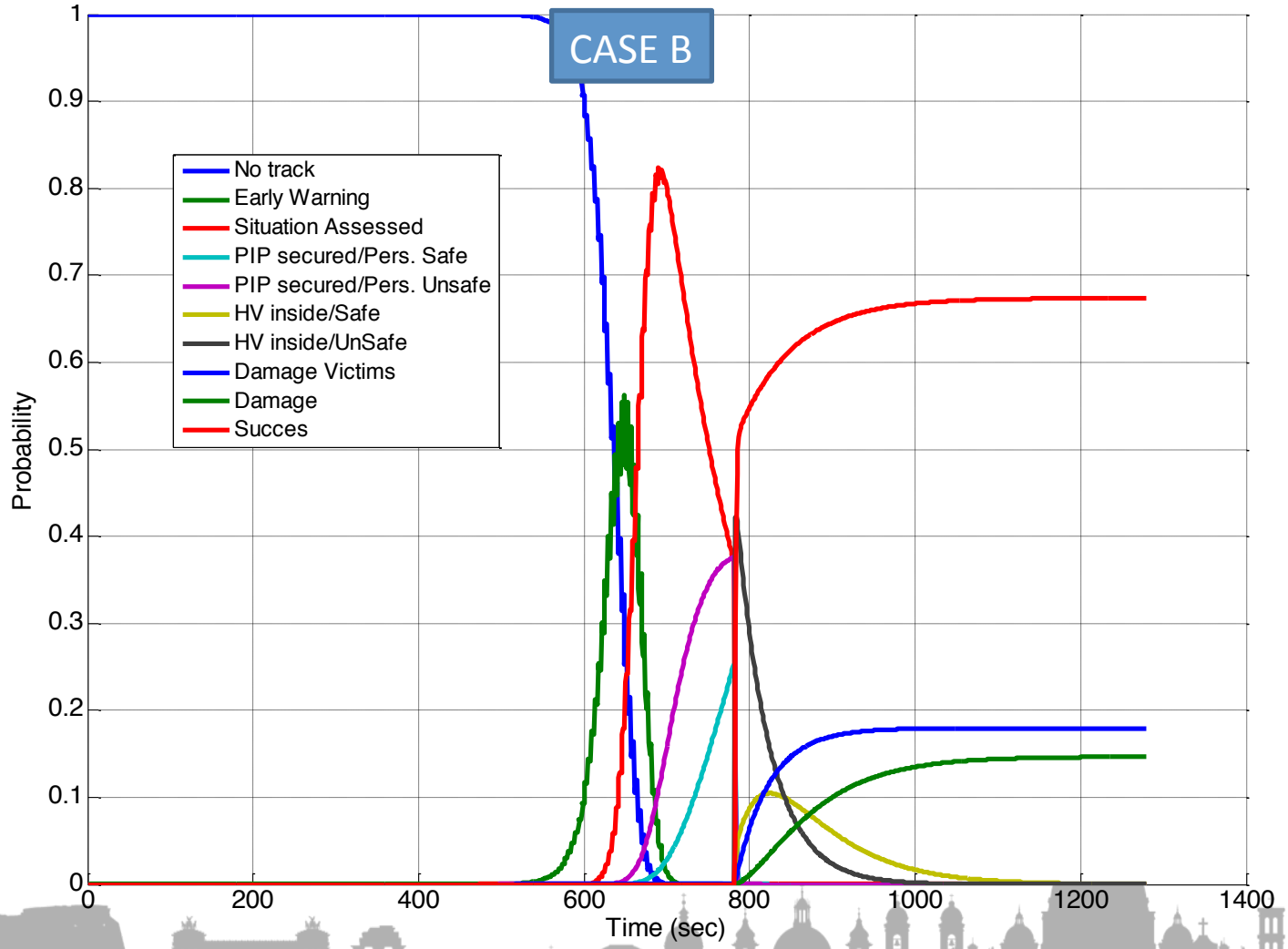
Study case

Frontal Assault Scenario Results 1/3



Study case

Frontal Assault Scenario Results 2/3



Study case

Frontal Assault Scenario Results 3/3

- The probability of success is 90% in Case A and only 67% in case B.
- The main reason is that the Radar system can warn the operators well in advance, so that the security personnel can be alerted and directed towards the intercept point before the intruder breaks through the fence.

End Game	CASE A	CASE B
Success	90%	67%
Damage	9%	15%
Damage/Victims	1%	18%

Conclusions

- Our engineering judgment is that this path may lead to significant results in producing a supporting tool able to provide quantitative predictions even in the preliminary phases of the project (i.e., technical proposal definition).
- The model is based on closed formulas calculations, thus being very fast in terms of computational time. This is a critical feature for system architects, where virtual fast-prototyping approaches are often used, and “real time” results are therefore needed.
- In other words, the proposed approach suites very well the designer needs in terms of sensitivity studies, design, parameter tuning, trade-off analyses, and top level architecture definition.

Conclusions

- In order to fully exploit this technique, some additional effort should be devoted to the following aspects:
  - The library of subsystems should be enriched with additional components (metal detectors, body scanners, enforcements, weapon systems, etc.).
  - A rich set of notional scenarios, for performance comparisons and trade studies, should be built.
  - A user friendly software tool should be coded, starting from the core functions produced for this report, in order to be usable within the company
  - A sensitivity study, as well as some performance measurements in scaled and controlled environment, should be executed in order to validate and corroborate our prediction method.

Questions?