Graph Methods for Systems-of-Systems Analysis

David J. Marchette

Quantitative Methods for Defense and National Security

Outline

Outline

Basic Definitions

- A graph $G = (V = [n], E)$ is a set of vertices and edges. We write $u \sim v$ or uv for the edge $\{u, v\}$.
- We allow self loops.
- An attributed graph $G = (V, A_V, E, A_E)$ is a graph with
  - A set A_v for each vertex v of vertex attributes.
  - A set A_e for each edge e of edge attributes.
- A time series of graphs is a collection of graphs indexed by time.

Random Graphs

- A random graph is a graph-valued random variable:

$$G : \Omega \rightarrow \mathcal{G},$$

  a probability distribution on the collection of all possible graphs (usually of a given order).
- In an independent edge model, each edge ij is present independently with probability p_{ij}.

Generalizations

- In a directed graph each edge has a direction, so uv is not the same as vu. In this case the adjacency matrix is not symmetric.
- In a graph, each edge occurs exactly once. A multi-graph is one where each edge may occur multiple times. In this case, the adjacency matrix may contain the number of edges between each pair of vertices.
- In a weighted graph, each edge has a weight (e.g. throughput) and the adjacency matrix may contain the weights.

Graph Formulation

- An SoS is a directed graph (G, E).
- Each node corresponds to a subsystem. Each directed edge corresponds to the interaction between the incident subsystems.
- A realization of an SoS is an SoS with vertex and edge attributes.
- The attributes encode information about the subsystems and their communications links/interactions:
  - Readiness level.
  - Number/types of communications/information transferred.
  - Measures of performance.
  - Any information about the system germane to a given situation.
- We write $\widetilde{G} = (G, A_V, A_E)$.

Readiness Levels

- The attributes on the vertices and edges correspond to **readiness levels**.
- The vertex readiness levels are called “technology readiness levels” (TRL) and the edge readiness levels are called “integration readiness levels” (IRL).
- These indicate the stage of development of the subsystem.
- They are a measure of the maturity of the technology and the implementation.
- Readiness levels take on values in $\{0, 1, \ldots, 9\}$.
- In practice there will be other measures as well, such as test results, adherence to specifications, information about sub-components, and so on.
- For this talk we will consider only readiness levels (xRL).

Mission Threads

- A scenario is $s \in \mathcal{S}$. One can think of this as the parameters of the simulation (or real world situation).
- A mission is an SoS realization $\widetilde{G}$ and a scenario s. A mission thread is a mission and time series of subgraphs $\{G_1, \ldots, G_m\}$. These are called events.
- Given a mission pair $M = \{\widetilde{G}, s\}$

$$\widetilde{G}, s \mapsto \{G_1, \ldots, G_m, Y\}.$$

  are the events (G_i) and outcomes (Y) produced by the mission thread.
- We will assume that m is constant and known for any given mission, M.
- $G_1, \ldots, G_m$ and Y are random variables.

Outline

Random Graph Formulation

- Given $\widetilde{G}$ there is an underlying random process that results in the events in the mission thread.
- Thus we can equate with $\widetilde{G}$ a random graph time series model.
- Our goal is to analyze this model, in the context of different missions, to determine which parameters of the model (which subsystem) is critical to the success of the mission, or needs to be improved.
- We will illustrate the ideas with an extremely toy example.

Distributions

- The events $G_1, \ldots, G_m \in \mathcal{G}(G)$, $Y \in \mathcal{Y}(s)$ ($\mathcal{Y}(s) = \{0, 1\}, [0, 1]$ etc.).
- We will suppress the dependence on (G, s).
- We are interested in analyzing (estimating) $F_\theta = F_{\widetilde{G},s}$, the distribution on $\mathcal{G} \times \mathcal{Y}$ for a given scenario.
- In particular, we want to be able to tell what parts of G (vertices, edges) contribute most to the outcome, and how their attributes should be changed (improved) in order to improve the outcome.

Goals

- Given data, we might seek to estimate $F_{\widetilde{G},s,Y}$, or we might seek to estimate $F_{\widetilde{G},Y}$ (integrating across scenarios).
- This would allow us to determine which parameters (A_V, A_E) contribute to the desired outcome ($Y = 1$). This tells us **what**.
- Additionally, given $G_1, \ldots, G_m$ we can answer **why**.

Classification Task

- Suppose $Y \in \{0, 1\}$ (failure or success).
- We observe (through experiment, simulation, or what have you)

$$\begin{pmatrix} G_{11} & \dots & G_{1m} & Y_1 \\ G_{21} & \dots & G_{2m} & Y_2 \\ & \dots & & \\ G_{k1} & \dots & G_{km} & Y_k \end{pmatrix}$$

- Classification involves building a classifier

$$g : G_1, \dots, G_m \mapsto Y$$

Classification Task

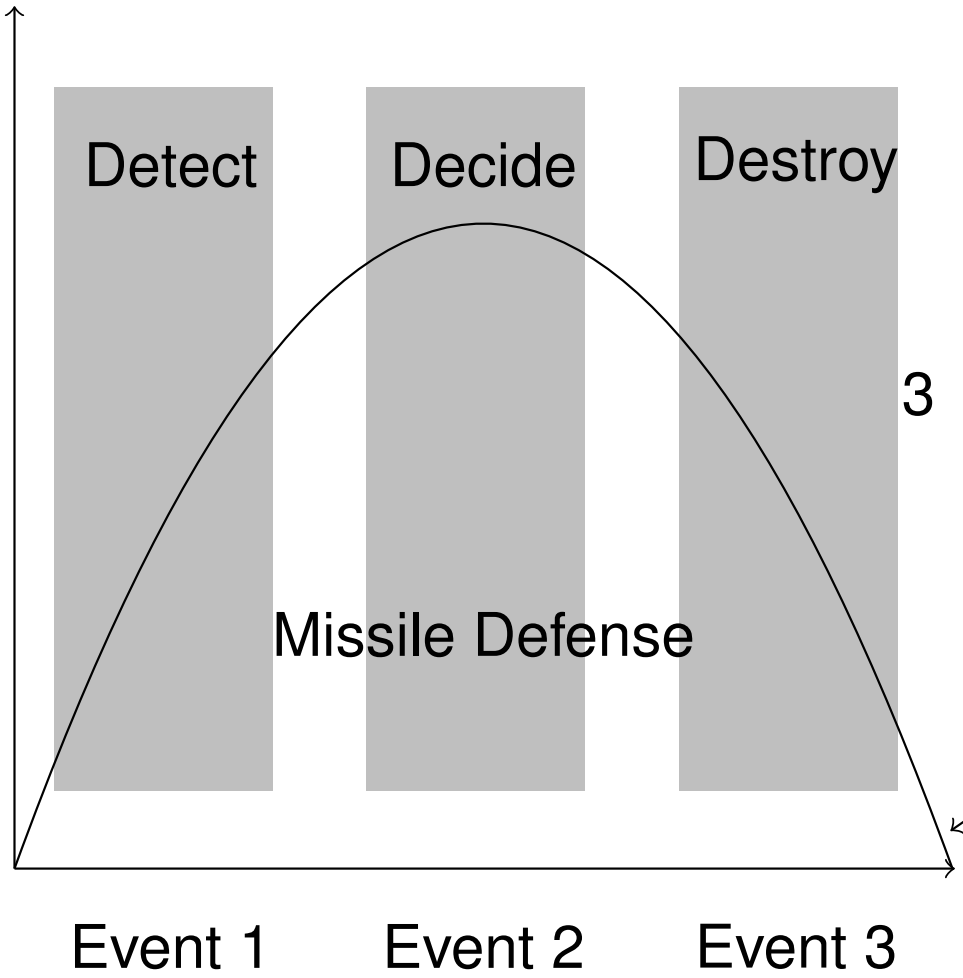
- One might think that this classifier not really what we want. We want a classifier

$$g : \widetilde{G} \mapsto Y$$

- Note however that we need to know not only which components (vertices, edges) contribute most to success/failure, but how they do so in the context of the mission.
- In particular, we need to know where in the mission thread the critical component fails, and why.
- With this information the systems engineer can decide the cost/benefits of making a particular change.

Outline

A Toy Problem

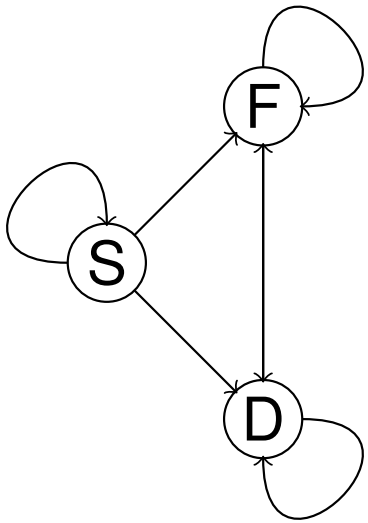


3 Nodes:
- Sensor
- Decision Authority
- Fire Control

A Target (T)

The Toy SoS

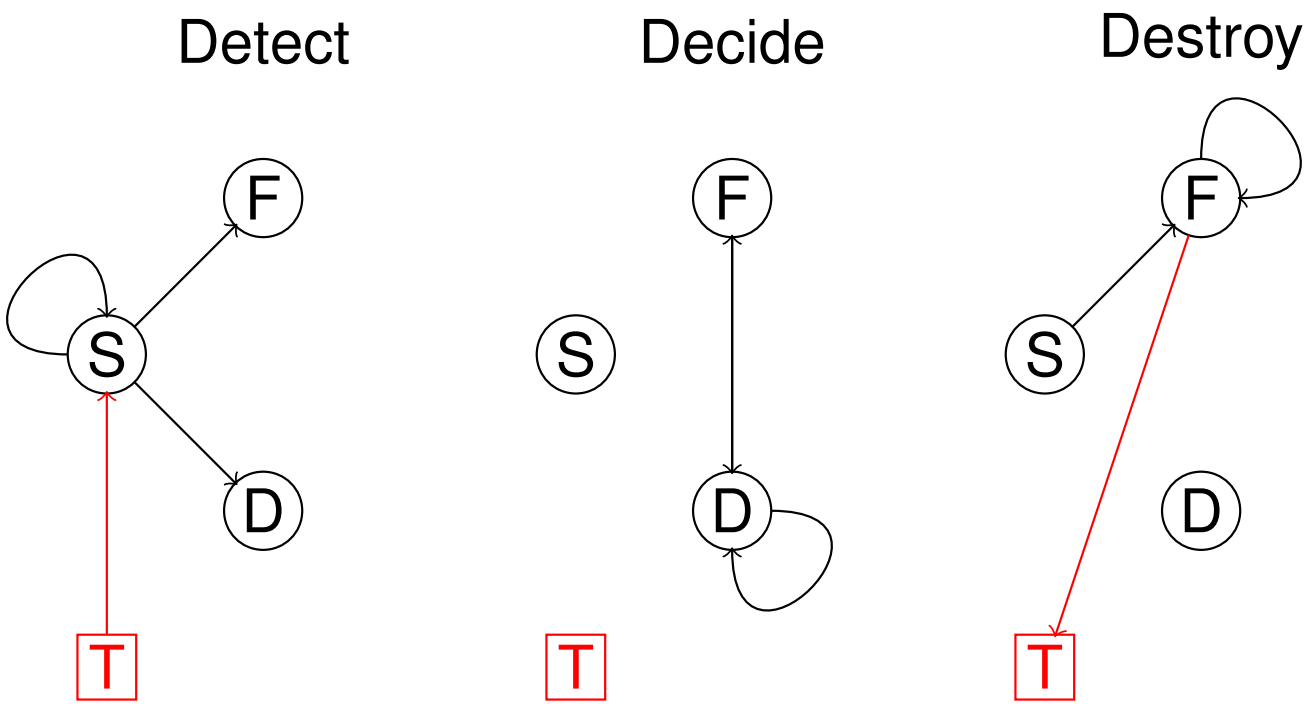
G



$$\begin{pmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{pmatrix} = \begin{pmatrix} p_{SS} & p_{SD} & p_{SF} \\ 0 & p_{DD} & p_{DF} \\ 0 & p_{FD} & p_{FF} \end{pmatrix}$$

For simplicity we assume an independent edge model for this graph.

A Simple Mission Thread



The Model

$$P[Y = 1|G_1, G_2, G_3] = P_1 P_2 P_3,$$

where each P_i corresponds to G_i and

$$\begin{aligned} P_1 &= 0.3I[S \sim S] + 0.3I[S \sim F] + 0.3I[S \sim D] \\ P_2 &= 0.5I[D \sim D] + 0.4I[D \sim F] + 0.1I[F \sim D] \\ P_3 &= 0.7I[F \sim F] + 0.3I[S \sim F] \end{aligned}$$

- Thus the probability of success given the mission thread is 0.9.
- Since the graphs are random (random weather patterns, environment, target(s), etc.) we don't necessarily see this exact mission thread in practice.
- We don't know a priori either the edge probabilities or the weights above.

The Model

- If we assume independent graphs and the random graph model above

$$\begin{aligned} P_1 &= 0.3p_{SS} + 0.3p_{SF} + 0.3p_{SD} \\ P_2 &= 0.5p_{DD} + 0.4p_{DF} + 0.1p_{FD} \\ P_3 &= 0.7p_{FF} + 0.3p_{SF} \end{aligned}$$

- We will assume in the simulation that the probability of a loop vv is the readiness level divided by 9, and similarly for an edge.
- In practice, we only know (assume) that the edge probabilities are a function of the xRL (and perhaps other attributes).

Analysis

$$P[Y = 1] = (0.3p_{SS} + 0.3p_{SF} + 0.3p_{SD})(0.5p_{DD} + 0.4p_{DF} + 0.1p_{FD})(0.7p_{FF} + 0.3p_{SF})$$

- If all probabilities are 1, P[Y=1]=0.9.
- More generally, if all probabilities are equal to p, $P[Y = 1] = 0.9p$.
- If $p_{SF} = 0$ the probability drops to 0.48.

Analysis

- In fact

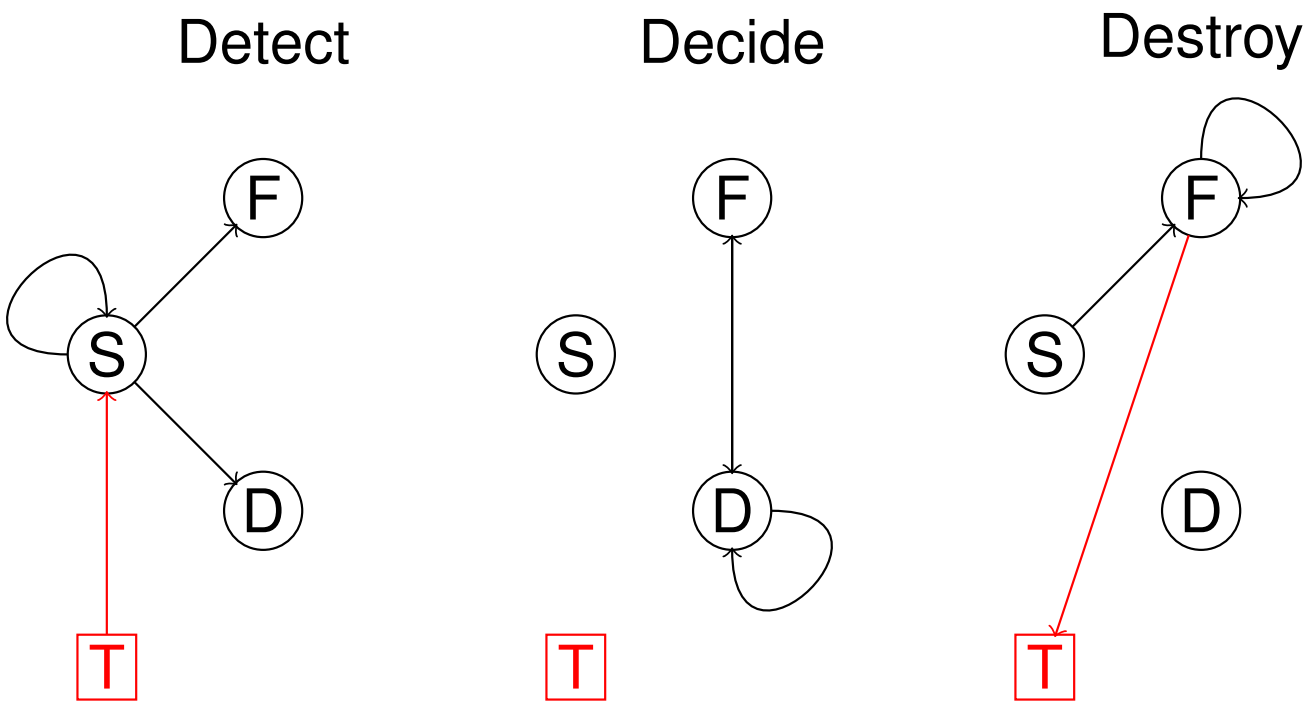
$$\frac{\partial P[Y=1]}{\partial p_{SF}} = 0.3(0.5p_{DD} + 0.4p_{DF} + 0.1p_{FD})$$
$$(0.3p_{SS} + 0.3p_{SD} + 0.7p_{FF} + 0.6p_{SF})$$

- By analyzing the sensitivity of outcome to the parameters that we can change (readiness levels) we can determine where the changes should be made to optimally improve the system.

Simulation Details/Results

- To illustrate the ideas, we ran simulations of the toy problem.
- The details are:
  - Set G, the attributed graph with given xRLs.
  - $P[u \sim v] = G[u, v]/9$
  - $n = 100$ times
    - Generate the mission thread.
    - Use probability of success as above to assign success/failure to the mission.
  - Average the adjacency matrices for success and for failure.
- We compare the difference between success and failure for the three events.

A Simple Mission Thread



Simulation Results

- All xRL's are 6, difference between probability of that link present and success and failure. Then $S \sim F$ is increased to 8 (in parentheses). The links with differences are:
  - Event 1:

$$\begin{aligned} S \sim D &= 0.15 \ (0.17) \\ S \sim F &= 0.36 \ (0.06) \end{aligned}$$

  - Event 2:

$$\begin{aligned} D \sim D &= 0.11 \ (0.15) \\ D \sim F &= 0.14 \ (0.17) \\ F \sim D &= -0.08 \ (0.04) \end{aligned}$$

  - Event 3:

$$\begin{aligned} S \sim F &= 0.0 \ (0.07) \\ F \sim F &= 0.25 \ (0.18) \end{aligned}$$

Outline

Conclusions

- This talk describes the first steps toward a theory of engineering systems of systems.
- The work is very preliminary; many simplifications have been made and much more work is needed.
- The basic framework seems promising. We need:
  - More realistic attribute sets.
  - Good random graph models appropriate to the problem.
  - Realistic simulations.
  - Real data.
- Some of these are available or in the works.