Geographic Distribution of Design Teams: A Probabilistic Analysis of Reliability Effects Illustrated with a Satellite Payload

Russ Garber and

Elisabeth Pate-Cornell

Engineering Risk Research Group

Stanford University

Outline

- Distributed Teams
- System—Action—Management (SAM) model
- Bayesian Analysis and use of Expert Opinion
- Results

Distributed Teams

- Definition – collaborative work done by physically remote teams.
- Can lead to many complications, specifically:
  - Communication problems
    - Management of engineering changes.
  - Interface interaction issues.

Real-life Example – Mars Climate Orbiter (MCO)

- Communications difficulties very much responsible for the MCO disaster.
- Miscommunications between Lockheed Martin and JPL.
- Problem of "inadequate communications between project elements" (phase I report).
- Failure report made many suggestions concerning group interactions.

Basic Tradeoffs

- Geographic Distribution of system elements
  - Can save money.
  - Can make up for an in-house lack of technological skill.
  - Can provide increased reliability.

VS:

- Increased interface errors and communications problems when distributed teams are employed.

Our Model:

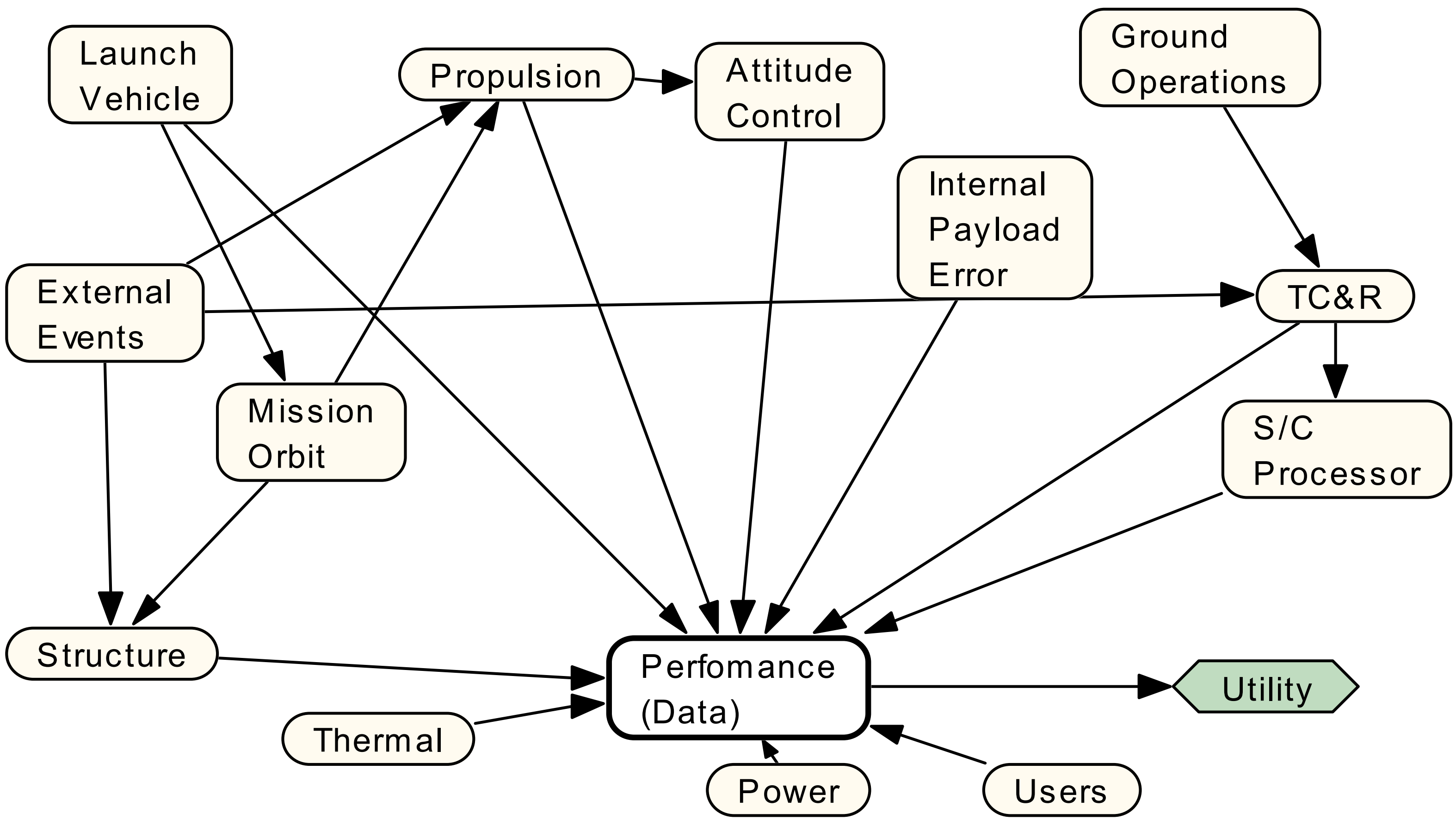
- We analyzed the specific problem of whether or not to geographically distribute the design of the payload from the other subsystems.
- Decision depends upon:
  - Ability of prime contractor to produce the payload.
  - Expected performance of the offsite team.
  - Quality and quantity of the interaction.
  - Amount of resources allocated to project.

SAM Model

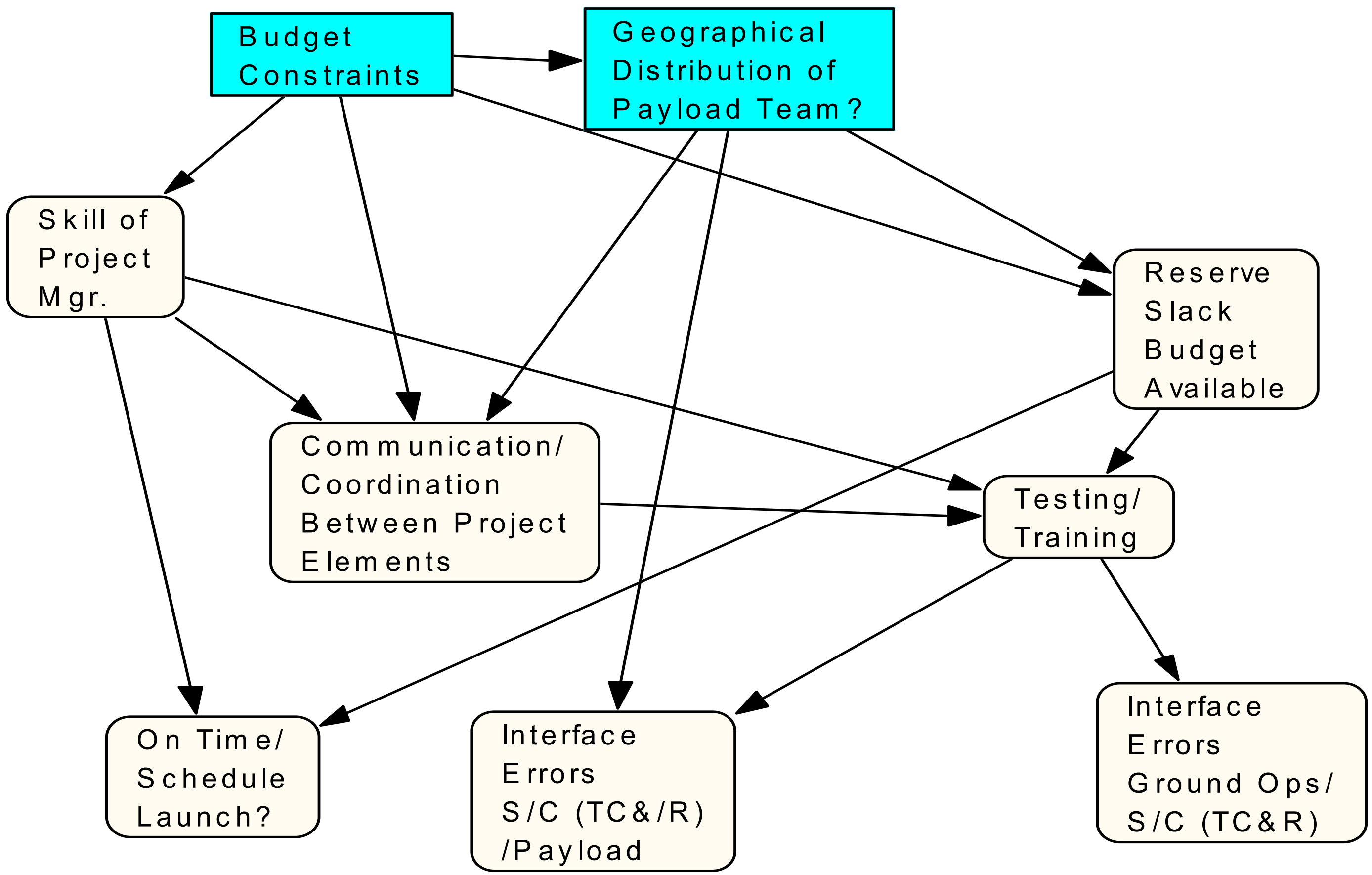
- SAM – System—Action—Management (Murphy and Pate-Cornell, 1996)
  - used to analyze the effects of geographically distributing the payload.
  - Starts with physical system.
  - Links middle level decisions and actions to physical system.
  - These "decisions and actions" predicated by upper management decisions.
  - Bayesian analysis used.
  - Expert opinion often required in lieu of comprehensive database.

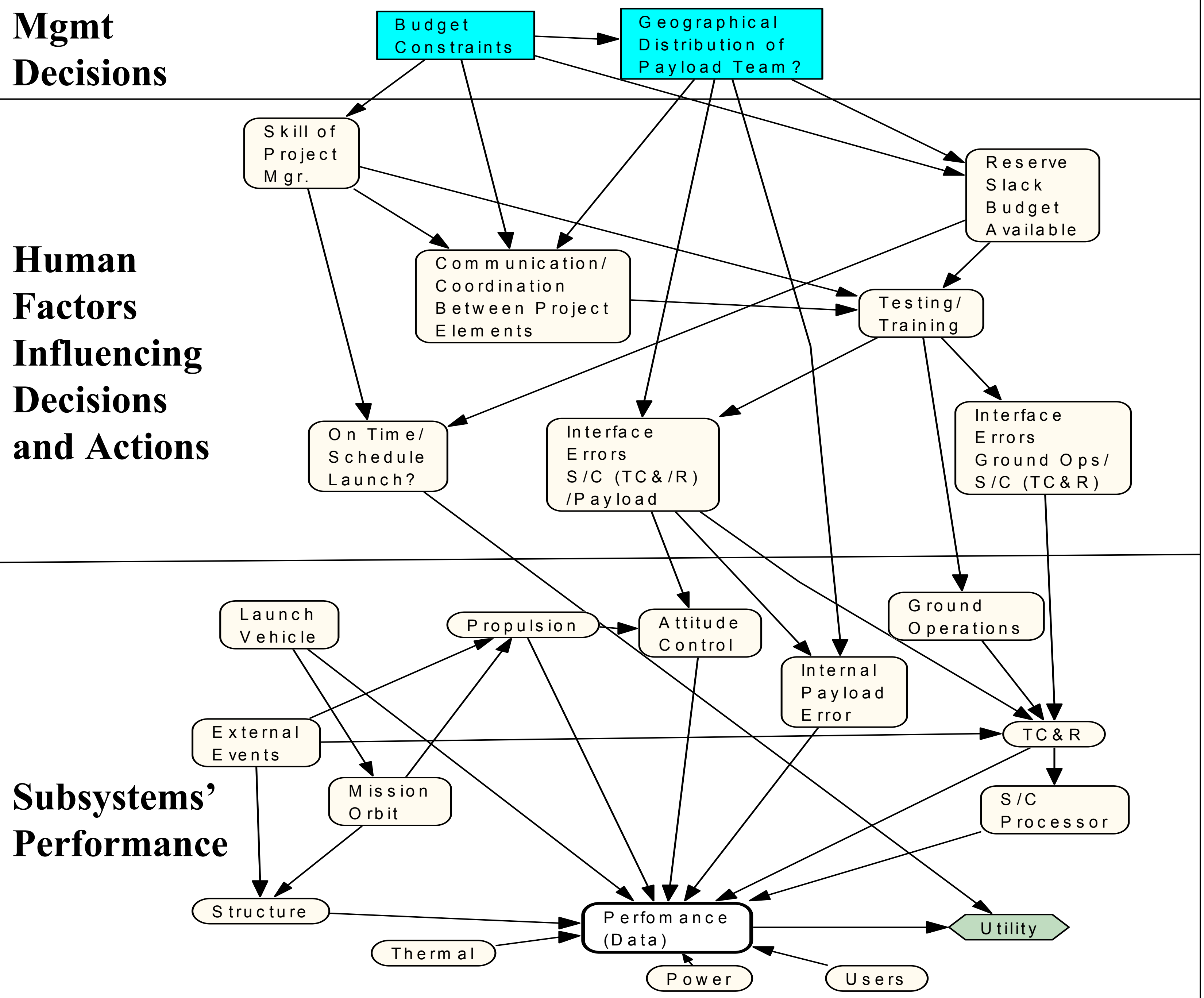
Geo Satellite System

(particular thanks to Joel Sercel for help with this model)



Mgmt Decisions and Human Actions Parts of SAM Model




Mgmt Decisions
Human Factors Influencing Decisions and Actions
Subsystems' Performance
Budget Constraints
Geographical Distribution of Payload Team?
Skill of Project Mgr.
Reserve Slack Budget Available
Communication/ Coordination Between Project Elements
Testing/ Training
On Time/ Schedule Launch?
Interface Errors S/C (TC&/R) /Payload
Interface Errors Ground Ops/ S/C (TC&R)
Launch Vehicle
Propulsion
Attitude Control
Ground Operations
Internal Payload Error
External Events
TC&R
Mission Orbit
S/C Processor
Structure
Perfomance (Data)
Utility
Thermal
Power
Users

Model Intricacies

- Utility dependent upon data received and losses due to launch delays/cost overruns.
- Utility function is concave (logarithmic):
  - **U (mission) = 10 + 299 x (log(D) + 1) – (10 x LD)**
    - U = utility
    - D = amount of data received by the users
    - LD = launch delay cost.
- Utility curve (arbitrary) set such that
  - U = 0 in the case of launch delays and cost overruns AND completely failed mission.
  - U = 1 when there are no delays and and the mission is a complete success.

Model Intricacies (II)

- Data received dependent upon state of systems.
  - If any part has completely failed, data received = 0.
  - If a connecting subsystem is partially working, the amount of data received is reduced by an amount dependant upon how important the damaged system is to the amount of data received.

Model Intricacies (III)

- Data = LV*A*P*SCP*T*S*PP*TCR*U*PL
  - Above equation holds when partial failures are independent.
  - Above variables are not probabilities but reflect degree of performance degradation of a particular independent partial failure.
  - For example, if there is a partial failure in the attitude control due to a sensor failure inhibiting earth/sun energy transfer, than A => .9, all others remain at 1, and Data = .9.

Partial Failure Dependencies

- Dependencies among partial failures are also taken into account.
- Approximated in this model by squaring amount lost due to degradation of each subsystem.
- Dependencies considered:
  - Attitude Control and Power.
  - Attitude Control and Propulsion.
  - Propulsion and Launch Vehicle.
  - TC&R and Users.

Example

- Partial failures in both Launch Vehicle and Propulsion
  - LV(partial failure) = .6, Propulsion (partial failure) = .8.

  - But, if both partial failures occur, then the losses are exacerbated due to dependency. Extra propellant will be needed to get satellite into proper orbit and a leakage, along with extra initial usage, could significantly shorten mission life.
  - Performance = (.6)^2*(.8)^2 = .23.
  - Without taking dependency into account, Perf = .48.

Data on Subsystem Failures

Total Missions	Problem Area	Number of Anomalies	Percentage (total)
116	All Failures	62 (20 total)	53.4% (17%)
107	Attitude Control	11 (5 total)	10.3% (5%)
116	Launch Failure	10 (9 total)	8.6% (8%)
107	Payload	15 (0 total)	14.0% (0%)
107	Power	10 (1 total)	9.3% (1%)
107	Propulsion	8 (2 total)	7.5% (2%)
107	Processor	2 (1 total)	1.9% (1%)
107	Spacecraft Structure	0	0.0% (0%)
107	TCR	3 (1 total)	2.8% (1%)
107	Thermal	1 (0 total)	0.9% (0%)
107	Other/unknown	2 (0 total)	1.9% (0%)

Example of Bayesian Analysis:
Propulsion Node

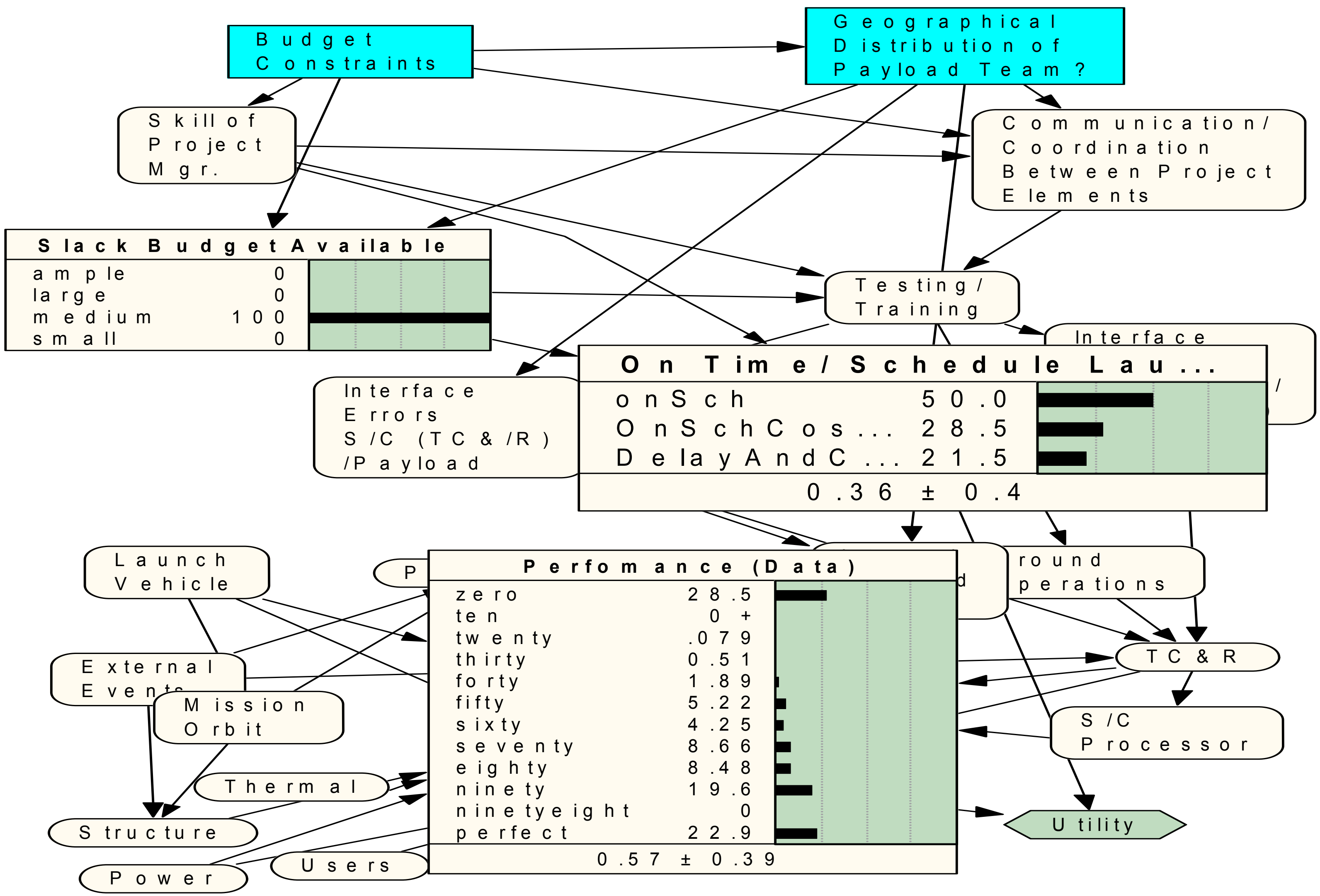
Mission Orbit	External Events	Perfect	Partial Failure	Total Failure
On Course	yes	0.85	0.11	0.04
On Course	no	0.93	0.05	0.02
Off Course	yes	0.70	0.24	0.06
Off Course	no	0.75	0.22	0.03

Example (II):
Project element comm./coordination (for budget constraints = “Ample”)

Budget Constraints	**Payload Production**	**Skill of Proj. Mgr.**	**Comm/Coordination High**	**Med**	**Low**
Ample	Combined	High	0.95	0.045	0.005
Ample	Combined	Mediocre	0.91	0.08	0.01
Ample	Separate	High	0.8	0.13	0.07
Ample	Separate	Mediocre	0.7	0.2	0.1
Tight	Combined	High	0.85	0.09	0.06
Tight	Combined	Mediocre	0.75	0.15	0.10
Tight	Separate	High	**0.35**	**0.35**	**0.30**
Tight	Separate	Mediocre	**0.25**	**0.40**	**0.35**

Resource Constraints (Super, Ample or Tight)	Payload Distribution (Payload/Subsystems Co-located vs. Distributed)	Utility 0 - 100
Super	Co-located	71.22
Super	Distributed	65.51
		Delta = 5.71
Ample	Co-located	66.69
Ample	Distributed	63.10
		Delta = 3.59
Tight	Co-located	63.61
Tight	Distributed	59.44
		Delta = 4.17

Model Snapshot (with budget = tight and payload separated)



Results of Illustrative Simulation

Perfomance (Data)

zero	28.5
ten	0 +
twenty	.079
thirty	0.51
forty	1.89
fifty	5.22
sixty	4.25
seventy	8.66
eighty	8.48
ninety	19.6
ninetyeight	0
perfect	22.9

0.57 ± 0.39

Summary

- Separating Payload/Spacecraft Subsystems can decrease cost but increase chance of interface error.
- We have provided a framework for analyzing the effects of separating the payload production from the other subsystems.
- Our results are determined by the particular data that we used. The user needs to enter his own beliefs and probabilities.
- Can easily add complexity to model.