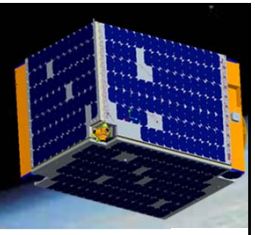
An Iterative Subsystem-Generated Approach to Populating a Satellite Constellation Tradespace

Andrew A. Rader
Franz T. Newland
COM DEV Mission Development Group

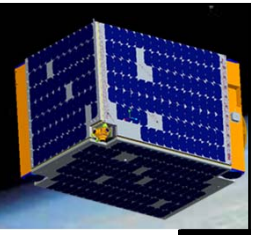
Adam M. Ross
SEAri, MIT

Outline



- Introduction to Tradespace Exploration
- Defining the Tradespace
- Methods:
  - Populating the Tradespace by Iteration
  - Utility Attributes
  - Design Variables and Cost Drivers
- Exploring the Tradespace by Attribute
- Multi Attribute Tradespace Exploration (MATE)
- Conclusions

Tradespace Exploration



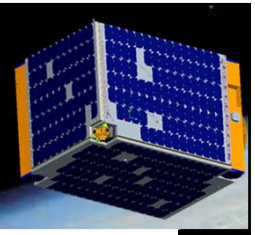
Objectives:

1. Compare potential designs on a common basis (and gain insight in the process)

2. Identify high value (utility/cost) designs to be examined in further detail

3. Examine the performance/cost/schedule implications of system requirements (which are easily achieved and which are significant cost drivers?)

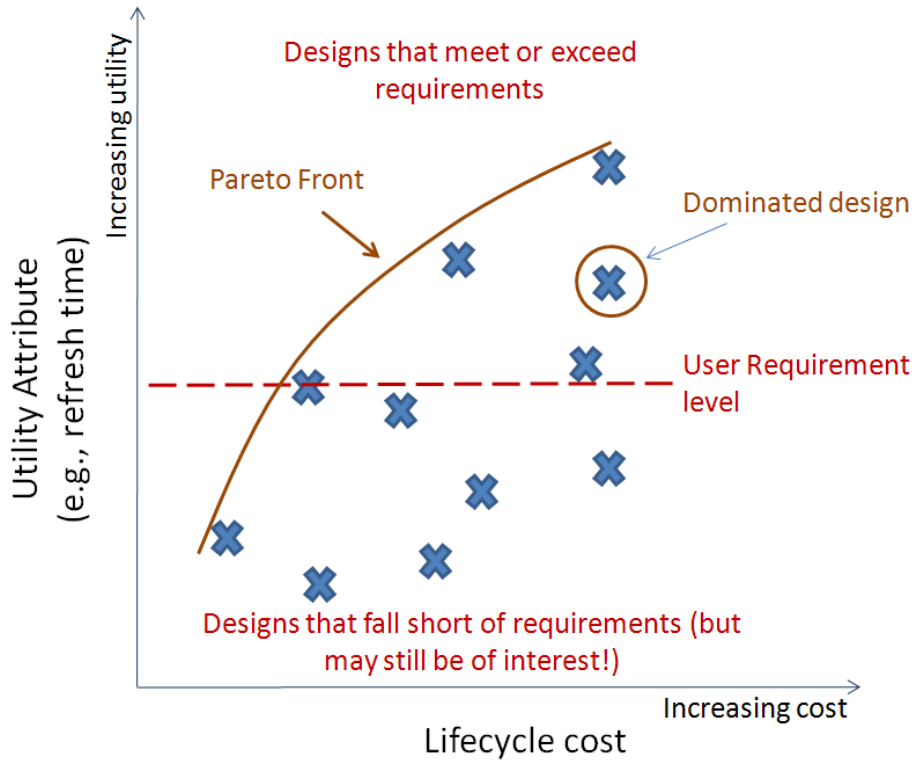
Tradespace Exploration



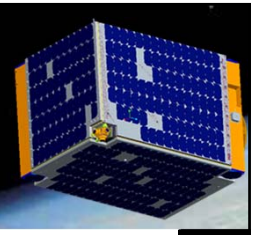
- Cost vs. performance tradeoff:
  - Which is better – a system that meets <u>most</u> objectives at a reasonable cost, or a system that meets <u>all</u> requirements at double the cost?
- Process rather than point analysis: insight gained through analysis is more important than result.
- Complex space missions often have many competing design alternatives, each comprising trade-offs across multiple subsystems.
  - ➢ It is not straightforward to span the tradespace by simply selecting design independent variables at either the system or subsystem level.
- How do we deal with these interactions while efficiently mobilizing domain expert knowledge?

Example Tradespace

- Increasing *value* is composed of:
  - increased utility (increasing along the y-axis, and
  - lower cost (decreasing along the x-axis).
- Pareto Front represents designs that have:
  - the highest utility for a given cost, or
  - the lowest cost for a given utility.
- Designs falling closer to the Pareto Front are higher in value.
- Designs falling father below the Pareto Front are dominated by higher value designs.

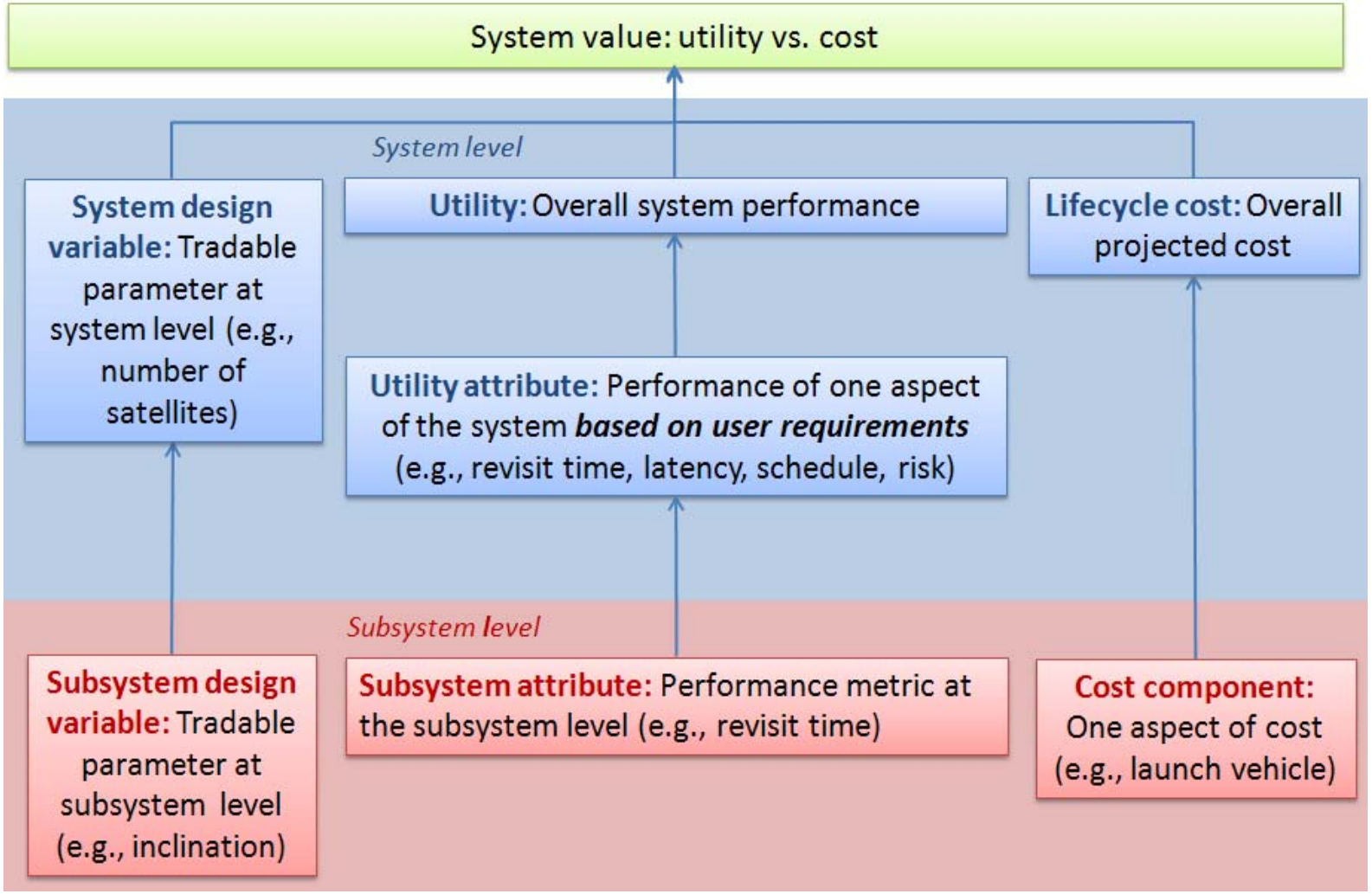


The Mission



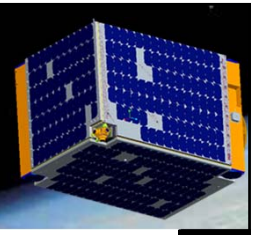
- Earth observation constellation with global coverage and requirements for:
  - high observation times
  - frequent revisits
  - fast downlinks
  - with high reliability for multiple points on Earth
- How these attributes should be traded off versus each other or versus costs?
- How should peak performance be valued versus mean, median, or worst case performance?
- How should global coverage be valued versus coverage over a particular area of interest?
- How much value is derived from additional ground stations as compared with additional satellites?

Terminology & Hierarchy



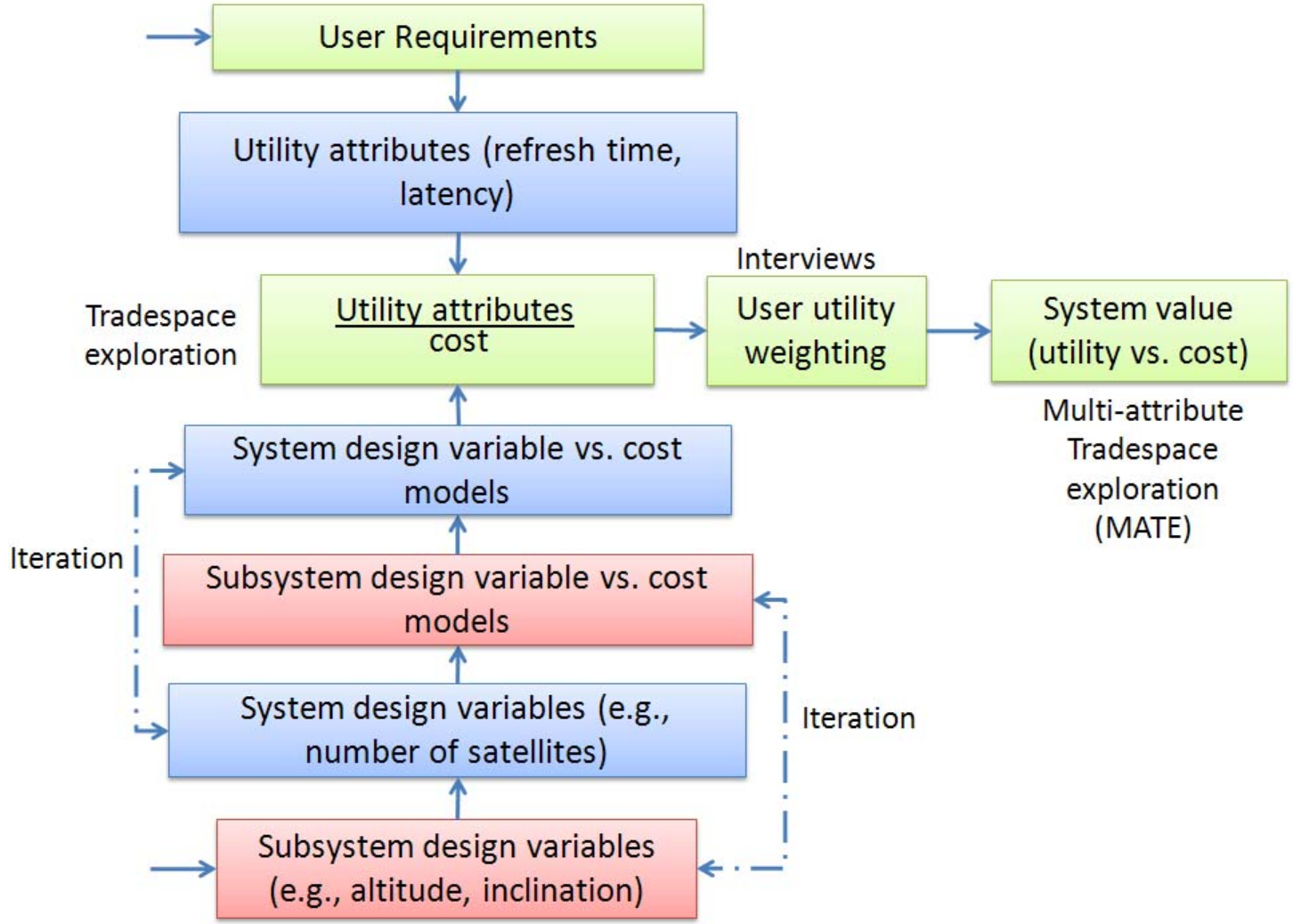
Methods

Method



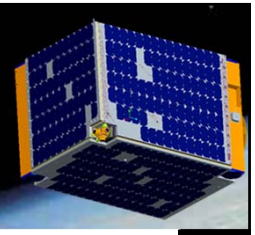
- Subsystem level optimizations performed in parallel by domain experts using their own tools.
- Interactions between subsystems assessed and propagated at system level using key nodal checkpoints.
- Allows independent partitioning of the design space by expertise (e.g., orbit, ground, launch, payload, etc.).
- Assembled system level tradespace automatically captures individual subsystem level trades.
- Tradespace considered multiple architectures, including nanosatellites with low duty cycles and strict power limitations, microsatellites, and hosted payloads on larger satellites.

Tradespace Exploration Flow



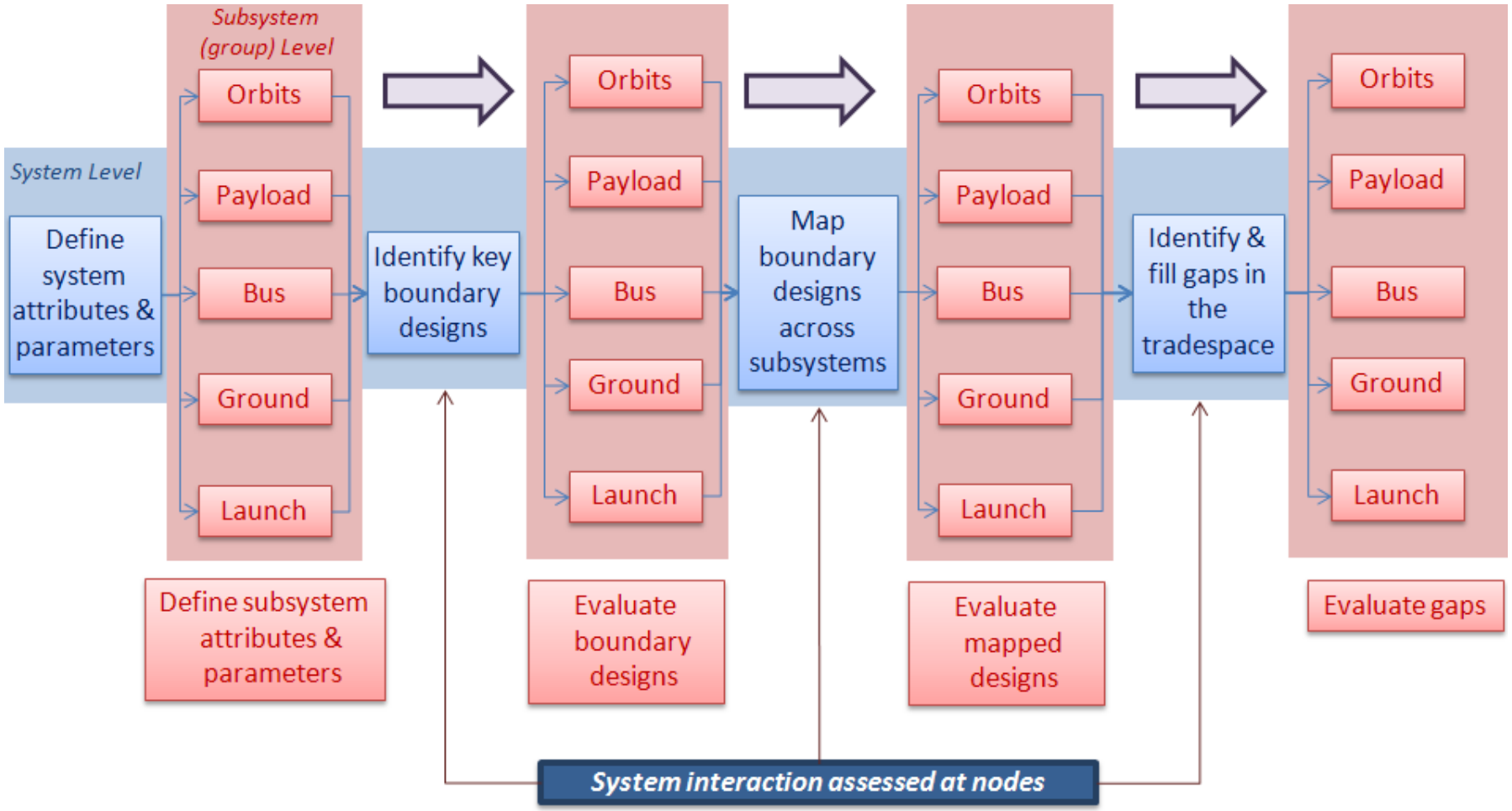
Methods

Subsystem Level Optimizations



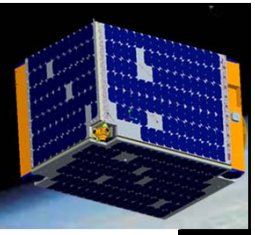
- Subsystem design variables from system design variables e.g., orbit = planes, spacing, inclinations, eccentricities, # per plane, etc.
- Subsystem domain experts are free to use own tools and conduct own optimizations.
- Minimal communication between expert groups is required to ensure that designs are feasible (though not necessarily efficient) across subsystems .
- No need to account for interactions at the subsystem level.
- Note that two subsystems may not produce the same result even with the same goal.

Populating the Tradespace by Iteration



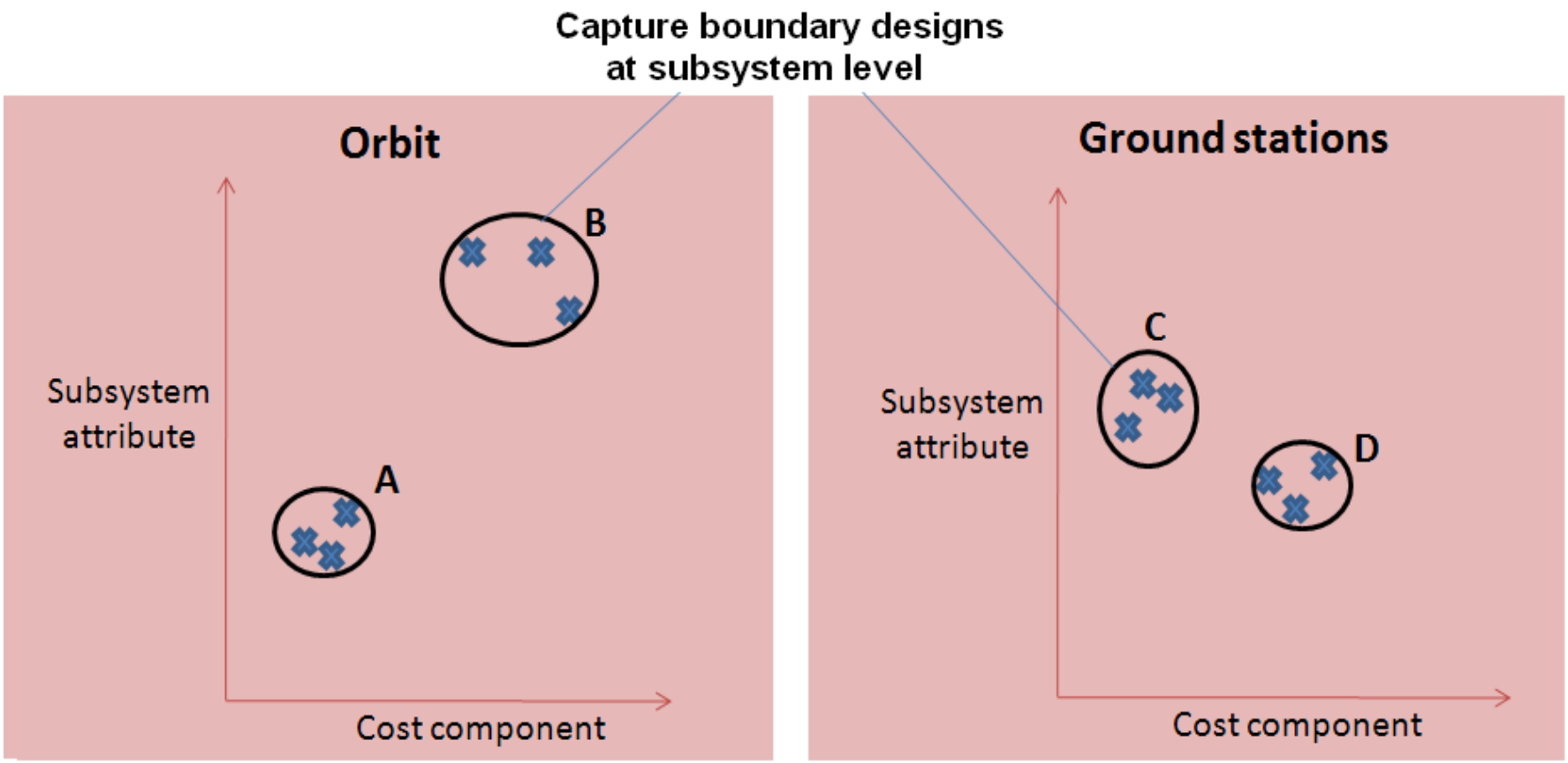
Methods

Propagating the Tradespace at the System Level

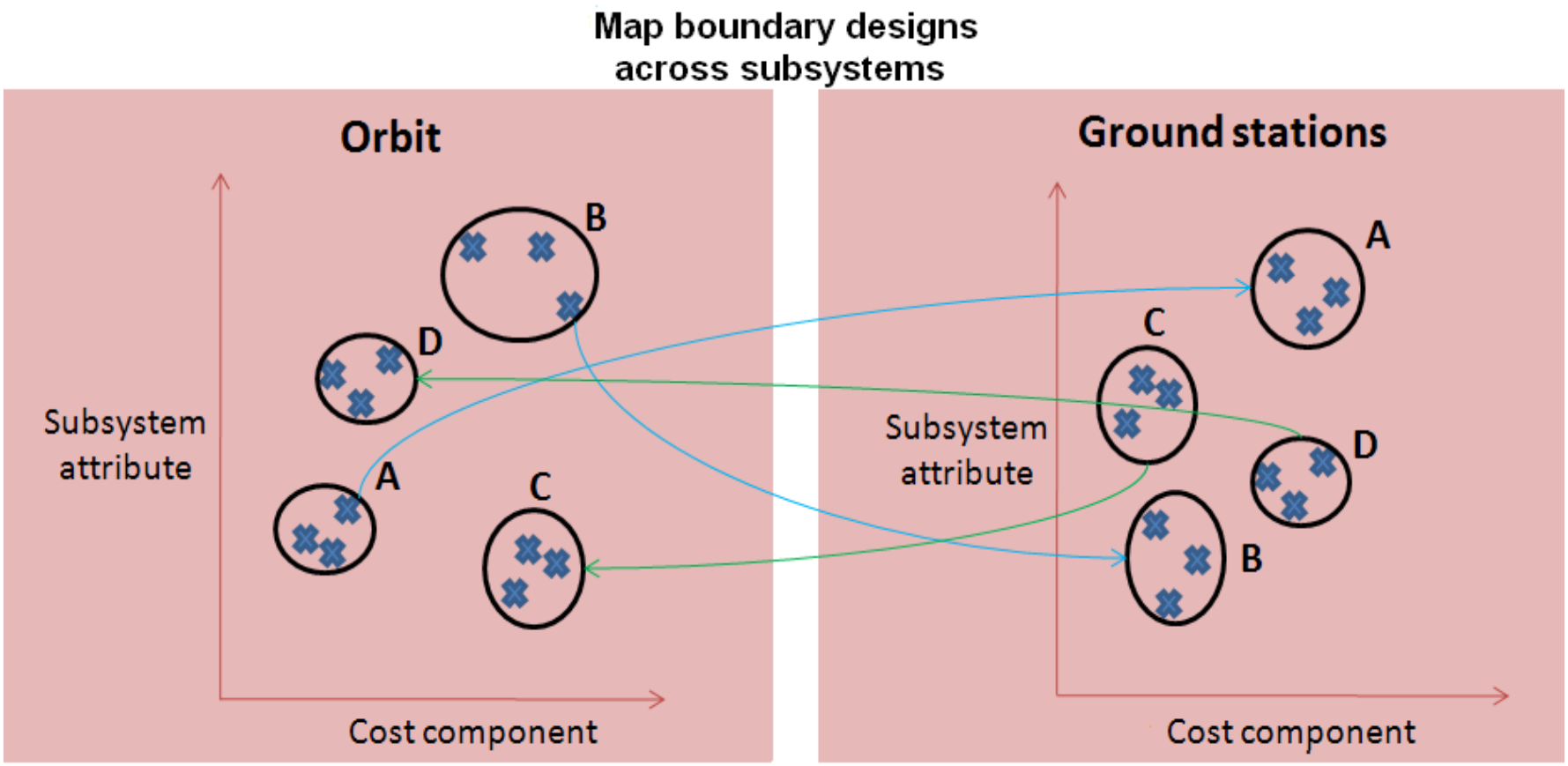


1. Evaluate interactions at system level: i.e., how do subsystem design variables influence system level utilities and costs?
2. Map subsystem-generated design variables across subsystems (possible full factorial expansion, e.g., 5 orbit x 3 ground station x 2 launch = 5 x 3 x 2 = 30, but overlaps are likely).
3. Potential for pruning the initial Tradespace.
4. Add missing “gap” designs based on exploration of initial Tradespace – especially when two or more subsystems share design variable elements (e.g., launch analysis suggests cost savings from implementing an orbital configuration).

Capture Boundary Designs

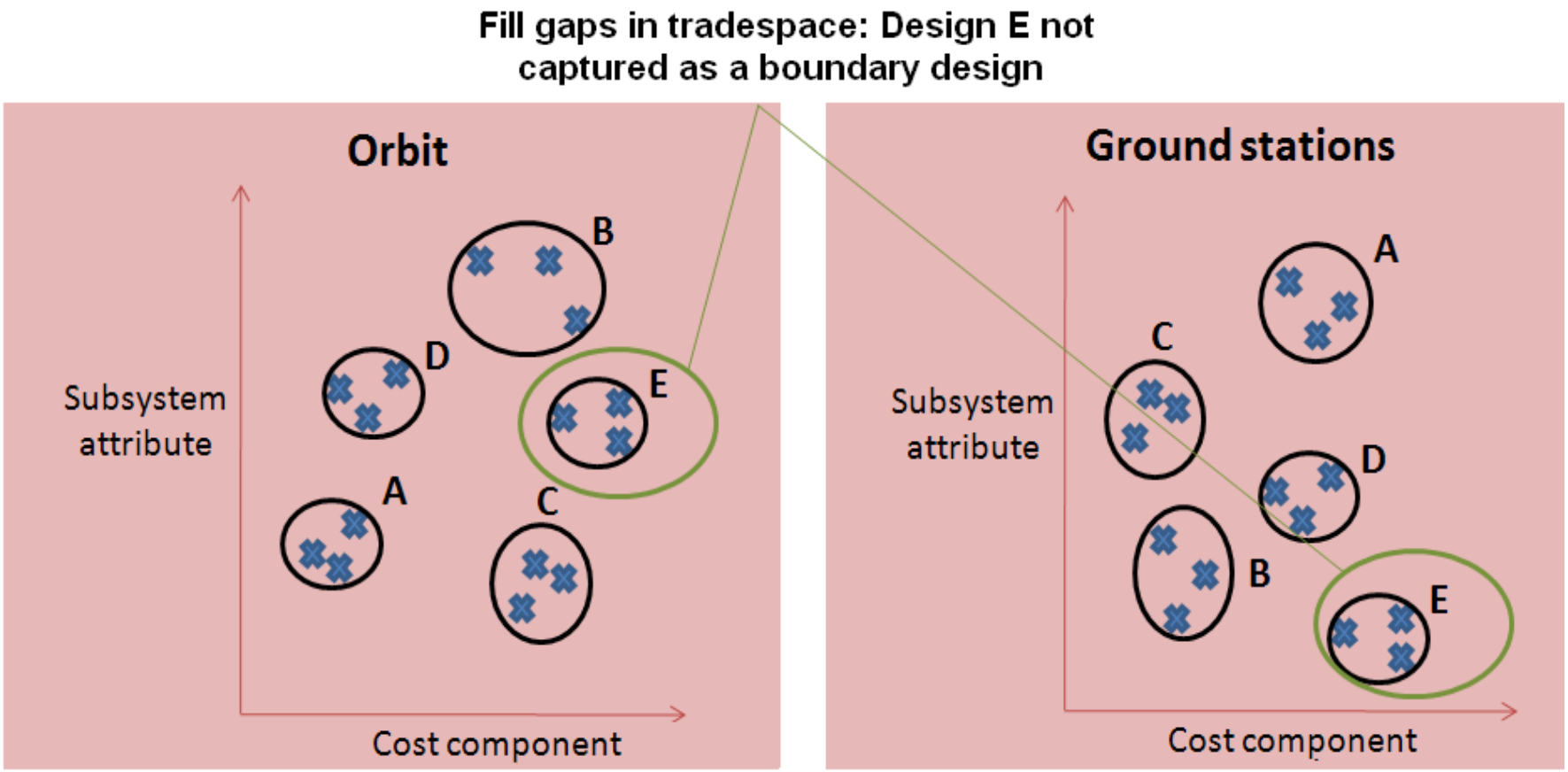


Map Across Subsystems



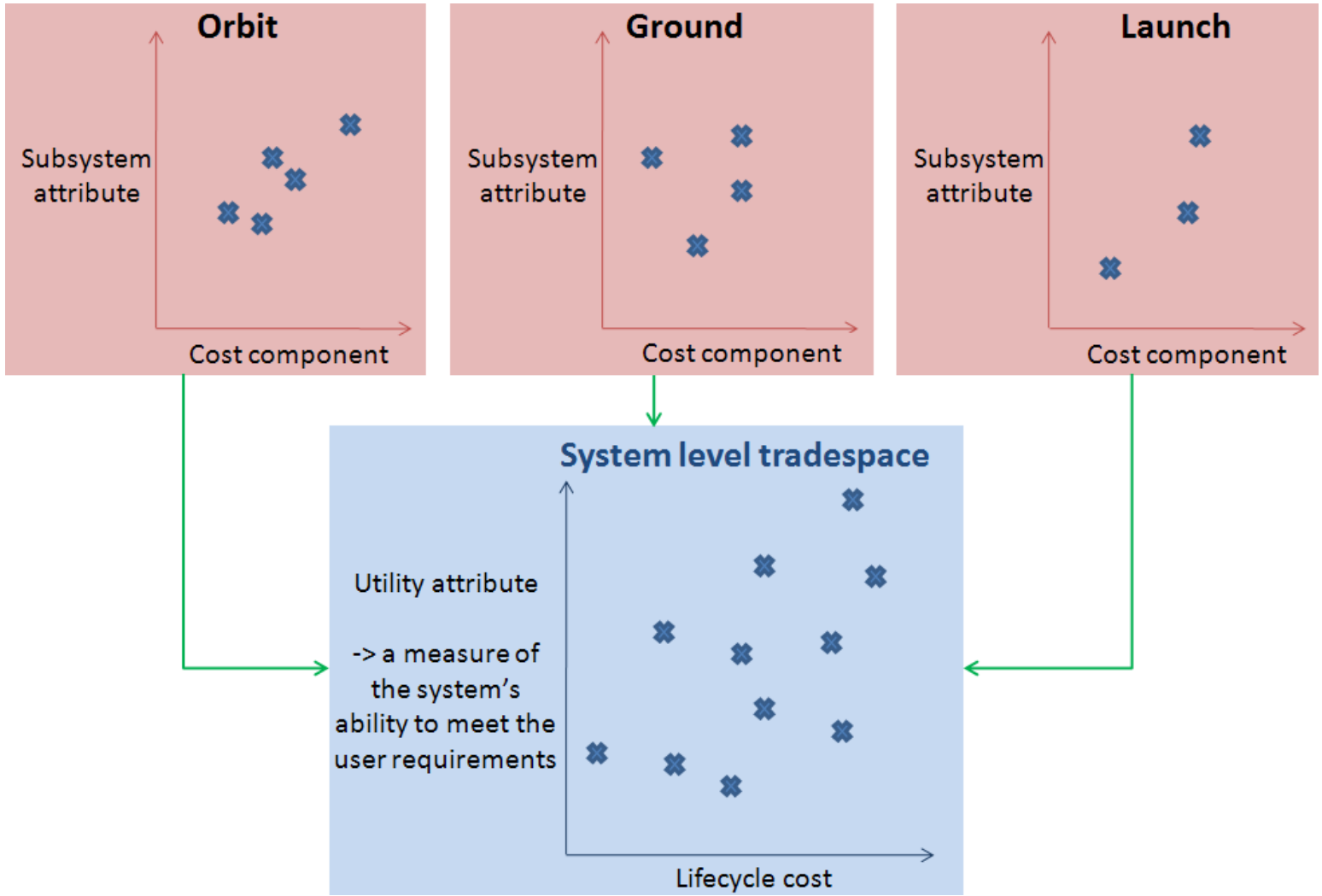
Methods

Fill Gaps in Tradespace



Methods

Build System Level Tradespace



Methods

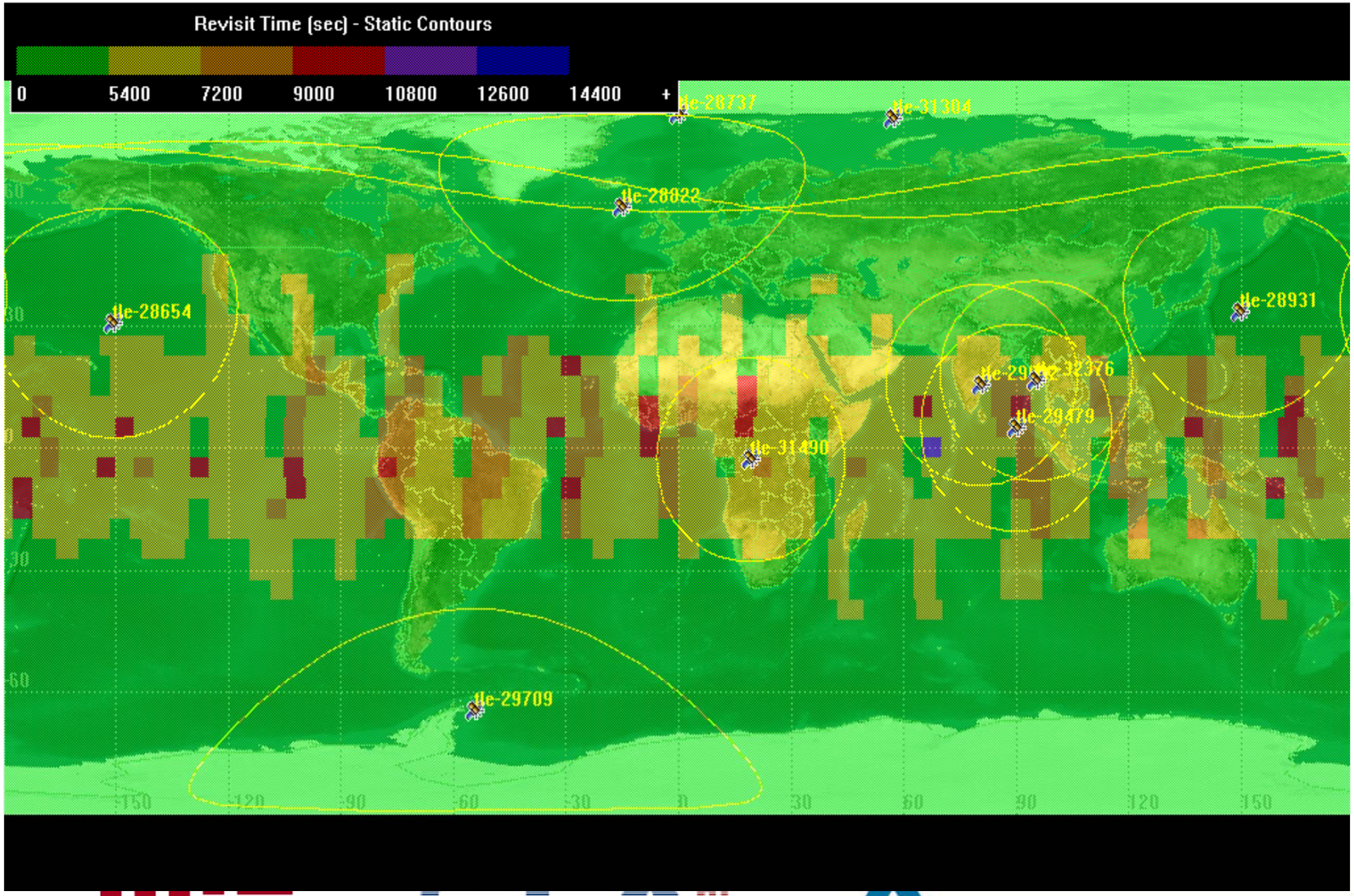
Utility Attributes

- Expert domain teams trace subsystem design variables to subsystem attributes using their own tools.
- Many attributes are multidimensional, e.g., latency is combination of:
  - Time from data collection to downlink
  - Time taken to downlink the data set
  - Time to taken distribute to data from ground station to the user
  - Time to process data
- Not always straightforward to define metrics. E.g., revisit time varies by:
  - Satellite
  - Orbital parameters
  - Orbital epoch
  - Location of interest on Earth
  - Transient factors (weather, duty cycle, etc.)
- Worst case, best case, mean, median, modal, etc., or combination could be considered most relevant performance metric.

Utility Attributes

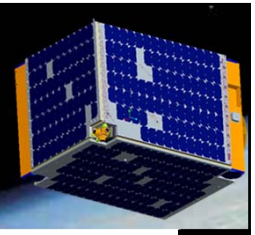
- Expert domain teams trace subsystem design variables to subsystem attributes using their own tools.
- Many attributes are multidimensional, e.g., latency is combination of:
  - Time from data collection to downlink
  - Time taken to downlink the data set
  - Time to taken distribute to data from ground station to the user
  - Time to process data
- Not always straightforward to define metrics. E.g., revisit time varies by:
  - Satellite
  - Orbital parameters
  - Orbital epoch
  - Location of interest on Earth
  - Transient factors (weather, duty cycle, etc.)
- Worst case, best case, mean, median, modal, etc., or combination could be considered most relevant performance metric.

Example: Maximum Revisit Time



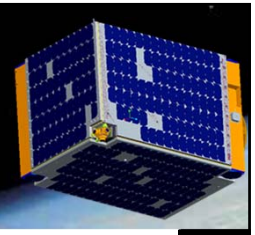
Attributes & design variables

Satellite Constellation Attributes



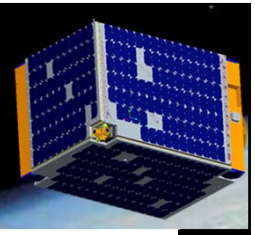
Utility attribute	Contributing subsystems
Refresh time globally	• Orbit • Launch (only as an orbit driver) • Ground
Refresh time over particular area of interest	• Orbit • Launch (only as an orbit driver) • Ground
Time on target	• Orbit • Launch (only as an orbit driver) • Payload (footprint, duty cycle)
Latency	• Orbit • Launch (only as an orbit driver) • Payload (data volume and format, downlink rate) • Ground (time to downlink, distribute, & process data)
Redundancy	• Bus (reliability/availability) • Orbit (refresh time with loss of spacecraft) • Ground (refresh time with loss of ground station)

Satellite Constellation Design Variables



System design variables	Subsystem design variables	Associated cost drivers
Spacecraft/bus	• Number of spacecraft • Nanosat/microsat/smallsat/hosted payload • Controlled/uncontrolled	• Design & build • Launch • Operations
Orbital characteristics	• Orbit planes • Eccentricities • Separation • Special cases (e.g., sun-synchronous)	• Launch • Propulsion & delta-v
Launch vehicle	• Number of satellites per launch • Primary vs. secondary payload • Replacement availability	• Launch vehicles • Launch operations
Payload	• Selection of payload for mission	• Equipment design & build cost • Operations
Downlink	• Frequency • Bandwidth • Geographic availability	• Equipment design & build cost • Regulatory issues
Redundancy, reliability, & replacement strategy	• On-orbit (hot or cold) spares vs. replacement • Spacecraft reliability (expected lifespan)	• Initial cost vs. replacement cost
Ground station	• Number and placement of stations • Build vs. buy • Stationary vs. mobile	• Equipment design & build cost • Start-up vs. operational costs
Data processing & handling	• Storage • Security	• Start-up costs • Operational costs

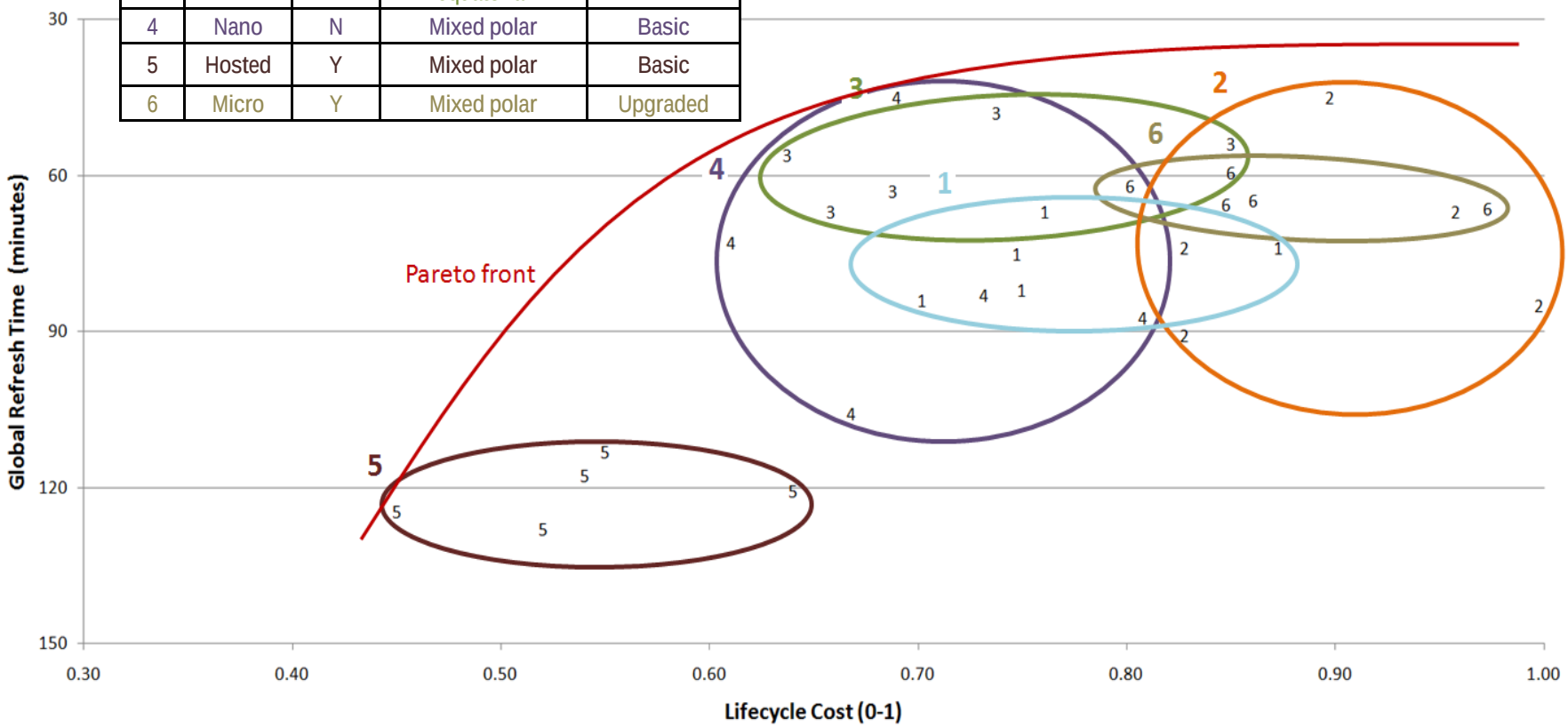
Example Designs



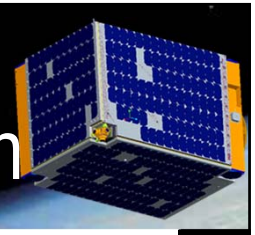
Design number	Bus	Propulsion?	Orbit type	Ground station configuration
1	Microsatellites	Y	Mixed polar	Basic
2	Microsatellites	N	Mixed polar	Basic
3	Microsatellites	N	Mixed including equatorial	Upgraded
4	Nanosatellites	N	Mixed polar	Basic
5	Hosted payload	Y	Mixed polar in historical locations	Basic
6	Microsatellites	Y	Mixed polar	Upgraded

Tradespace Exploration by Attribute: Global Refresh

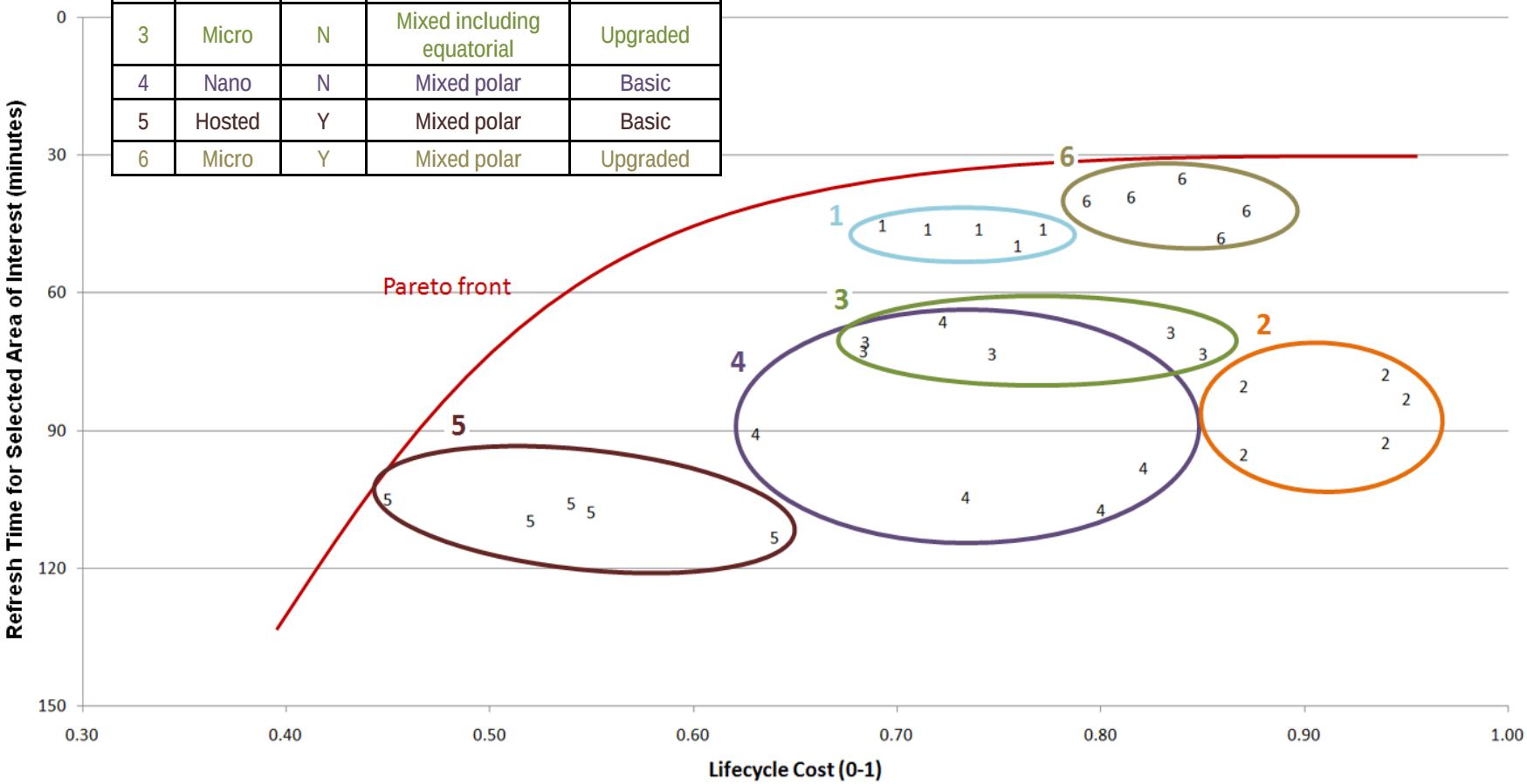
	Bus	Prop?	Orbit	Ground station
1	Micro	Y	Mixed polar	Basic
2	Micro	N	Mixed polar	Basic
3	Micro	N	Mixed including equatorial	Upgraded
4	Nano	N	Mixed polar	Basic
5	Hosted	Y	Mixed polar	Basic
6	Micro	Y	Mixed polar	Upgraded



Tradespace Exploration by Attribute: Local Refresh

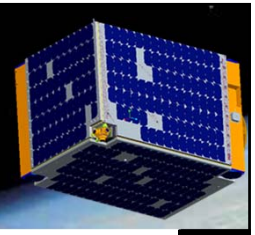


	Bus	Prop?	Orbit	Ground station
1	Micro	Y	Mixed polar	Basic
2	Micro	N	Mixed polar	Basic
3	Micro	N	Mixed including equatorial	Upgraded
4	Nano	N	Mixed polar	Basic
5	Hosted	Y	Mixed polar	Basic
6	Micro	Y	Mixed polar	Upgraded

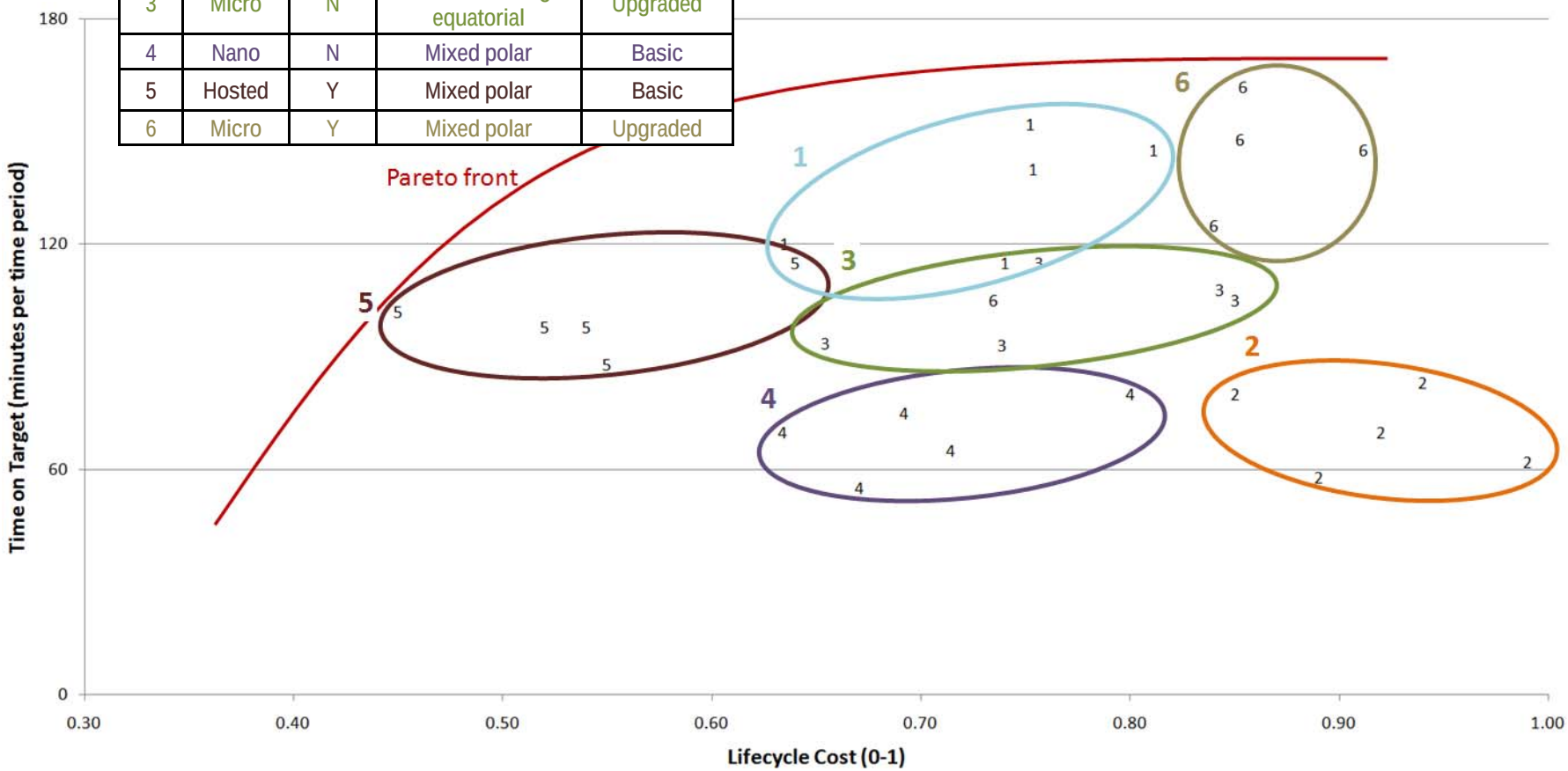


Tradespace Exploration

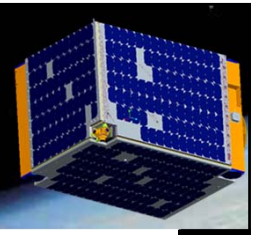
Tradespace Exploration by Attribute: Observation Time



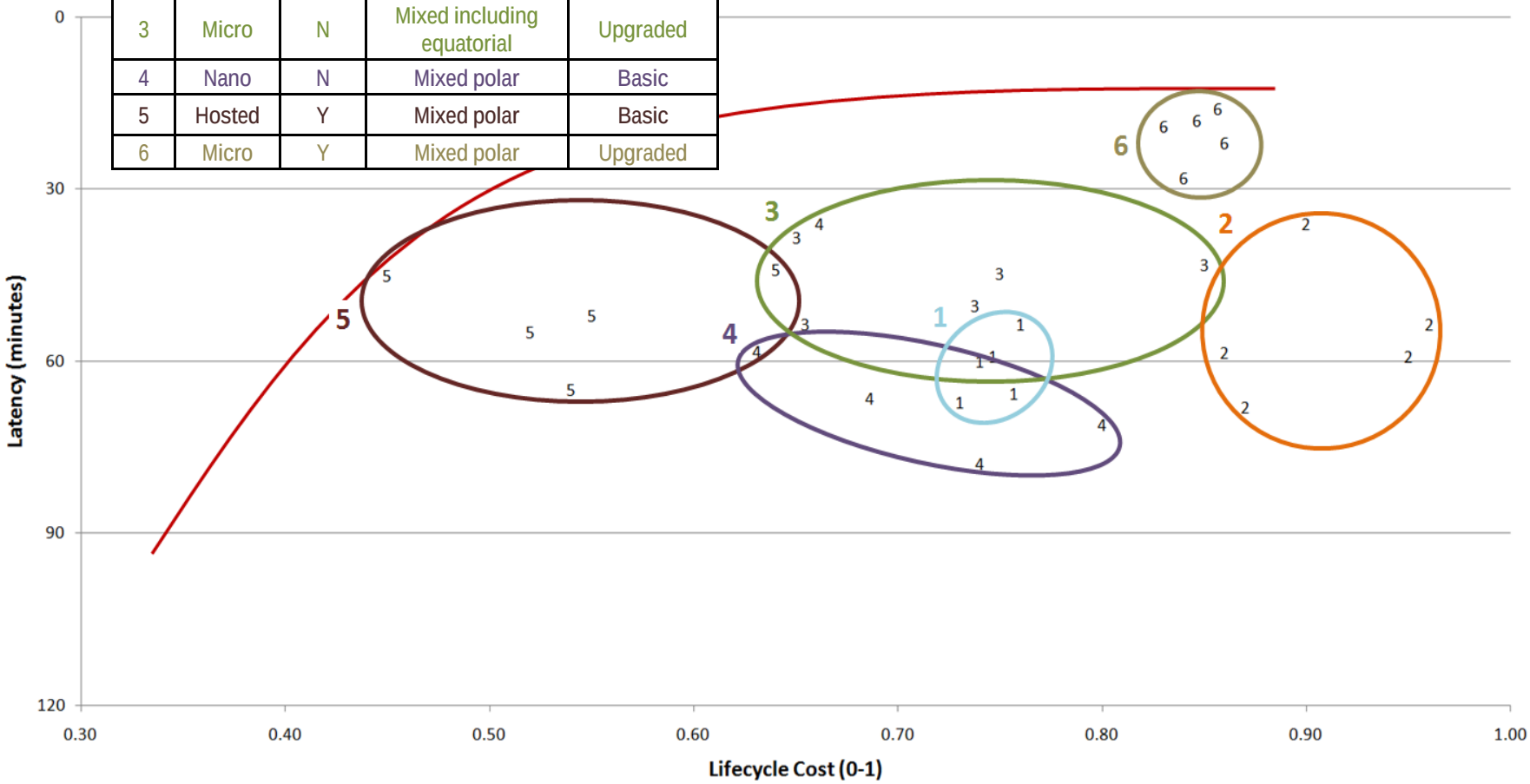
	Bus	Prop?	Orbit	Ground station
1	Micro	Y	Mixed polar	Basic
2	Micro	N	Mixed polar	Basic
3	Micro	N	Mixed including equatorial	Upgraded
4	Nano	N	Mixed polar	Basic
5	Hosted	Y	Mixed polar	Basic
6	Micro	Y	Mixed polar	Upgraded



Tradespace Exploration by Attribute: Observation Time

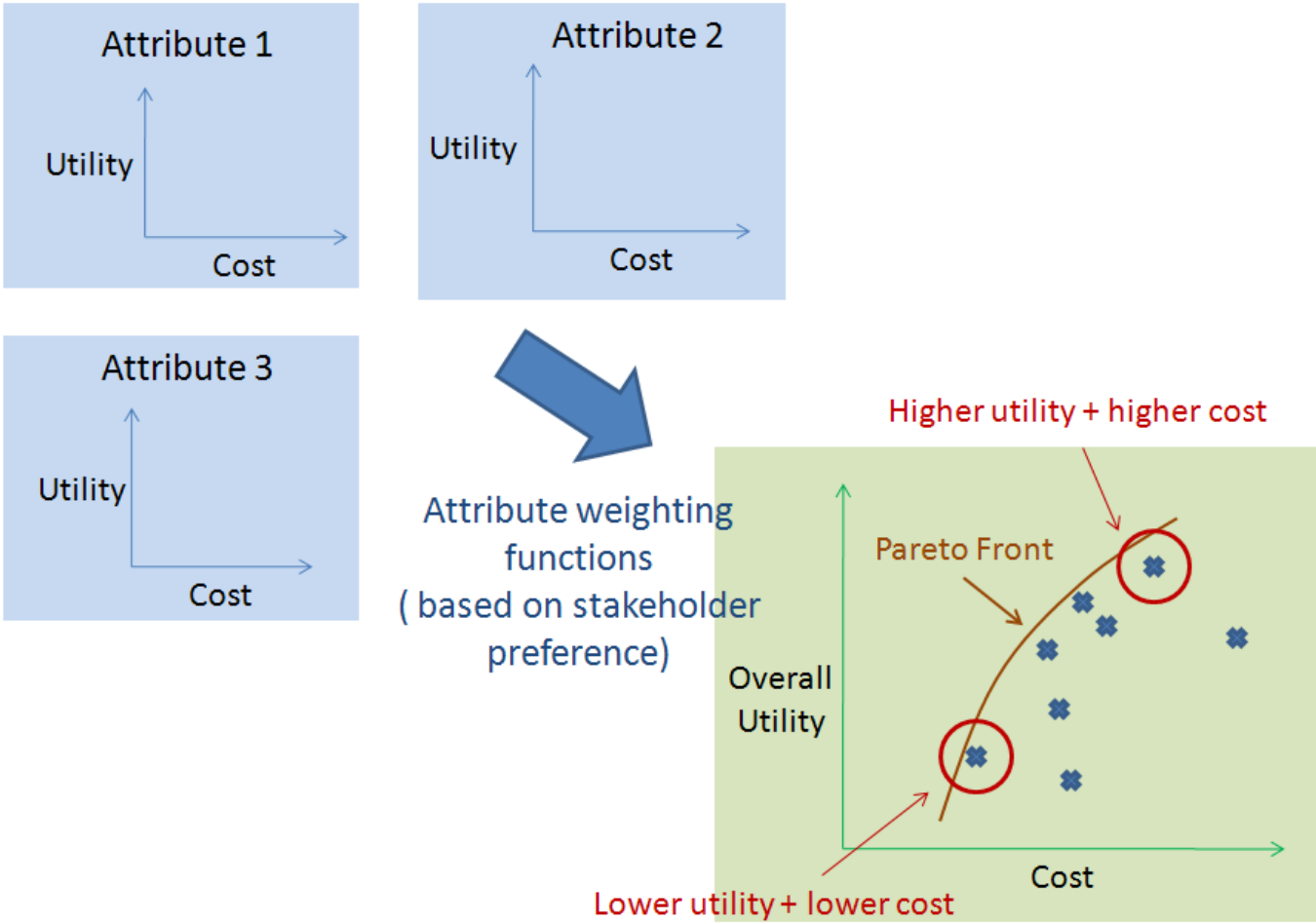


	Bus	Prop?	Orbit	Ground station
1	Micro	Y	Mixed polar	Basic
2	Micro	N	Mixed polar	Basic
3	Micro	N	Mixed including equatorial	Upgraded
4	Nano	N	Mixed polar	Basic
5	Hosted	Y	Mixed polar	Basic
6	Micro	Y	Mixed polar	Upgraded



Tradespace Exploration

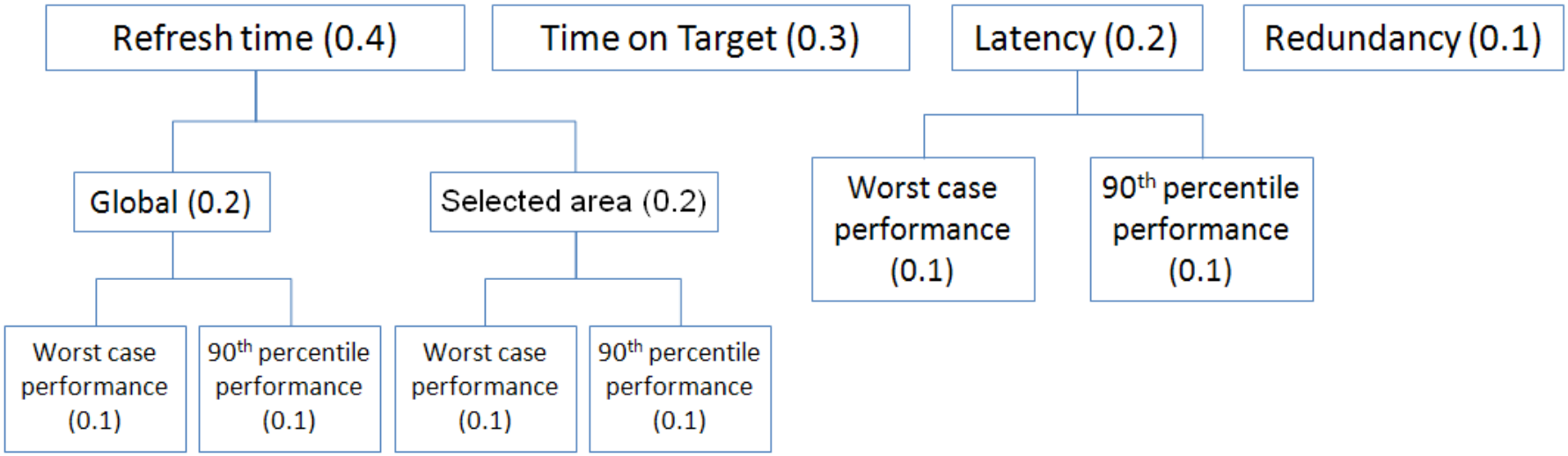
MATE: Assembling the Multi-Attribute Tradespace



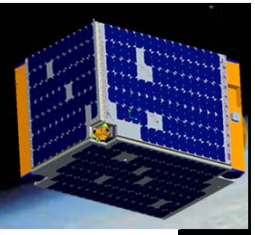
Tradespace Exploration

MATE: Utility Weightings

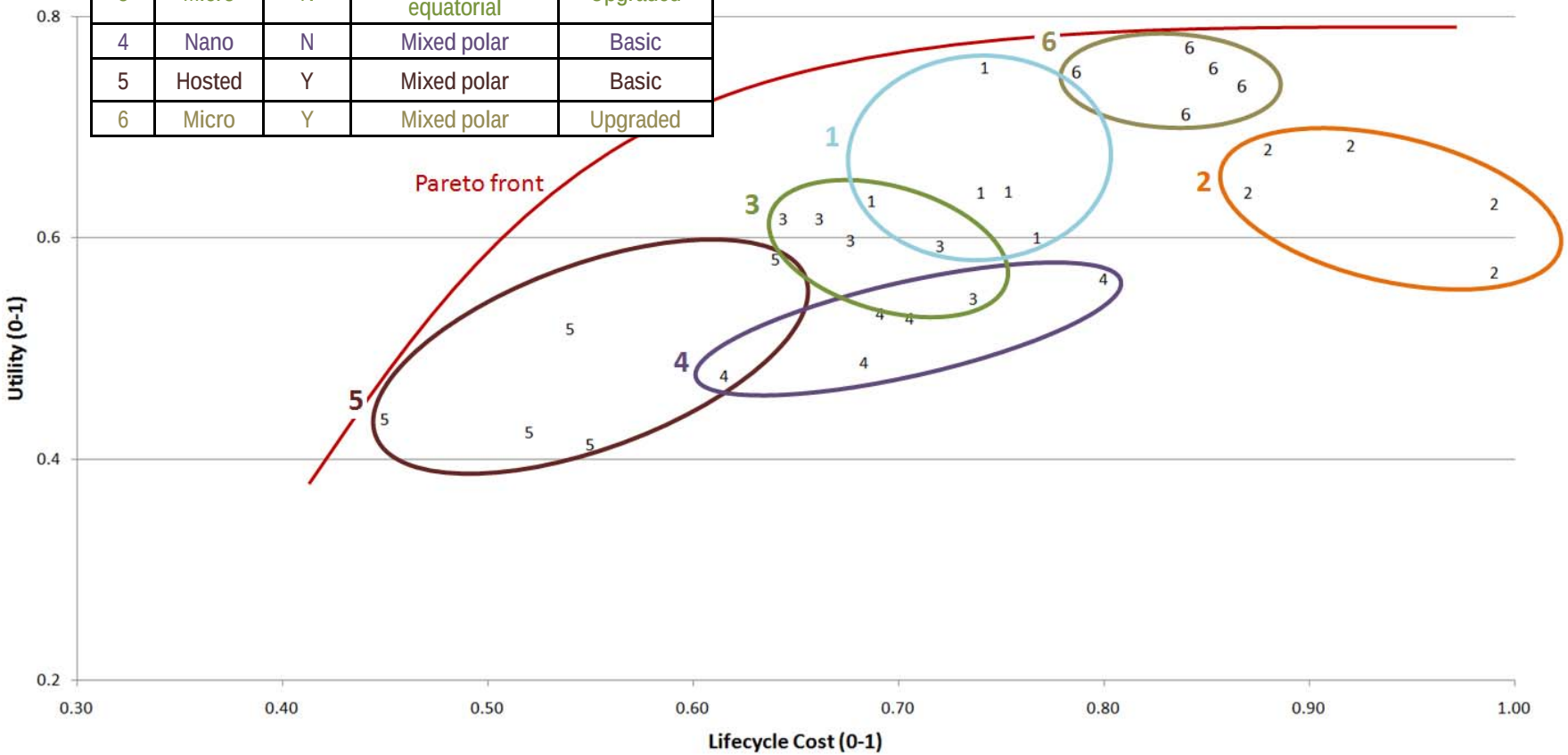
- Conjoint Analysis used to elicit utility weightings:
  - User selects preference from 2 “equal” alternatives of varying attributes multiple times.
  - Difference between nominal attribute rank and true attribute rank allows inference of attribute weightings.



Multi-Attribute Tradespace

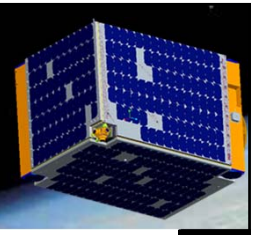


	Bus	Prop?	Orbit	Ground station
1	Micro	Y	Mixed polar	Basic
2	Micro	N	Mixed polar	Basic
3	Micro	N	Mixed including equatorial	Upgraded
4	Nano	N	Mixed polar	Basic
5	Hosted	Y	Mixed polar	Basic
6	Micro	Y	Mixed polar	Upgraded



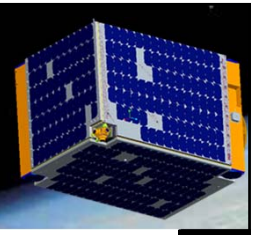
Tradespace Exploration

Conclusions



- Tradespace is problem specific: process of assembling it can reveal predefined biases and lead to a re-examination of user requirements.
  - E.g., some designs that otherwise perform well, are excluded because they fail to meet a single requirement's minimum threshold.
- Engaging subsystem domain experts early:
  - Capitalizes on expert knowledge
  - Saves time
  - Reduces risk

Conclusions



- Instead of generating a full system tradespace that might contain hundreds or thousands of potential designs, this method uses expert knowledge to generate a filtered subset containing only high value solutions
  - Performance optimizations at the subsystem level have already been performed in the assembled system level tradespace.
- Potential drawback: possible missed solutions in filtered tradespace.
- This bottom-up iterative approach is particularly appropriate for industry problems where it might be difficult to assemble a full computational system model.