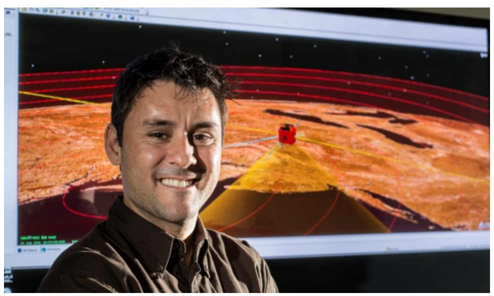
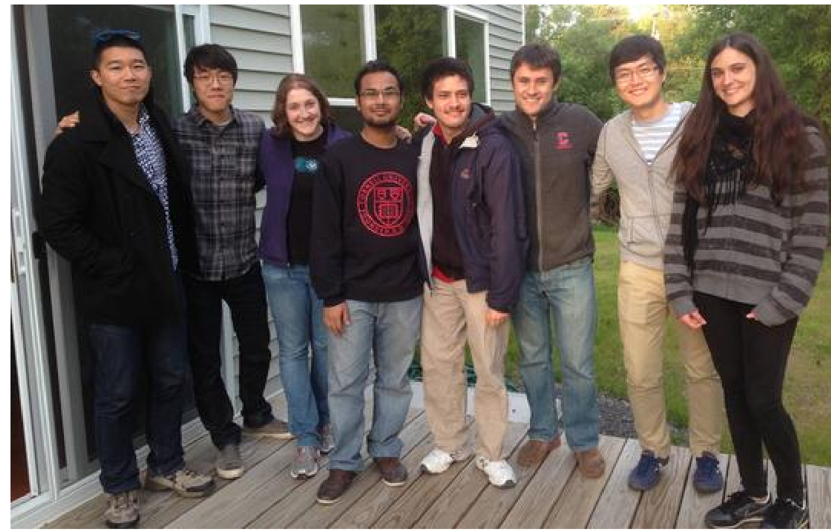
Towards cognitive design assistants and mixed-initiative design of complex systems

Prof. Daniel Selva <daniel.selva@cornell.edu>
University of Connecticut
February 17th 2017

The SEAK Lab at Cornell

5 PhD students + several Meng/UG in Mechanical & Aerospace Engineering and Systems Engineering





SEAK Lab Mission

To support development of the next generation of aerospace concepts and architectures while studying and improving the architectural design process

Understanding space mission design process

Developing mission design tools & methods

Conducting mission architecture/design studies

Global Optimization

Machine Learning

Visual and Data Analytics

Multi-Agent Systems

Knowledge-Based Systems

Human-Agent Interaction

A motivating example… Earth Observing Systems

- Current architecture is single large monolithic satellite
- Future systems will be distributed, collaborative networks of intelligent heterogeneous assets sharing information in real-time and making decisions autonomously

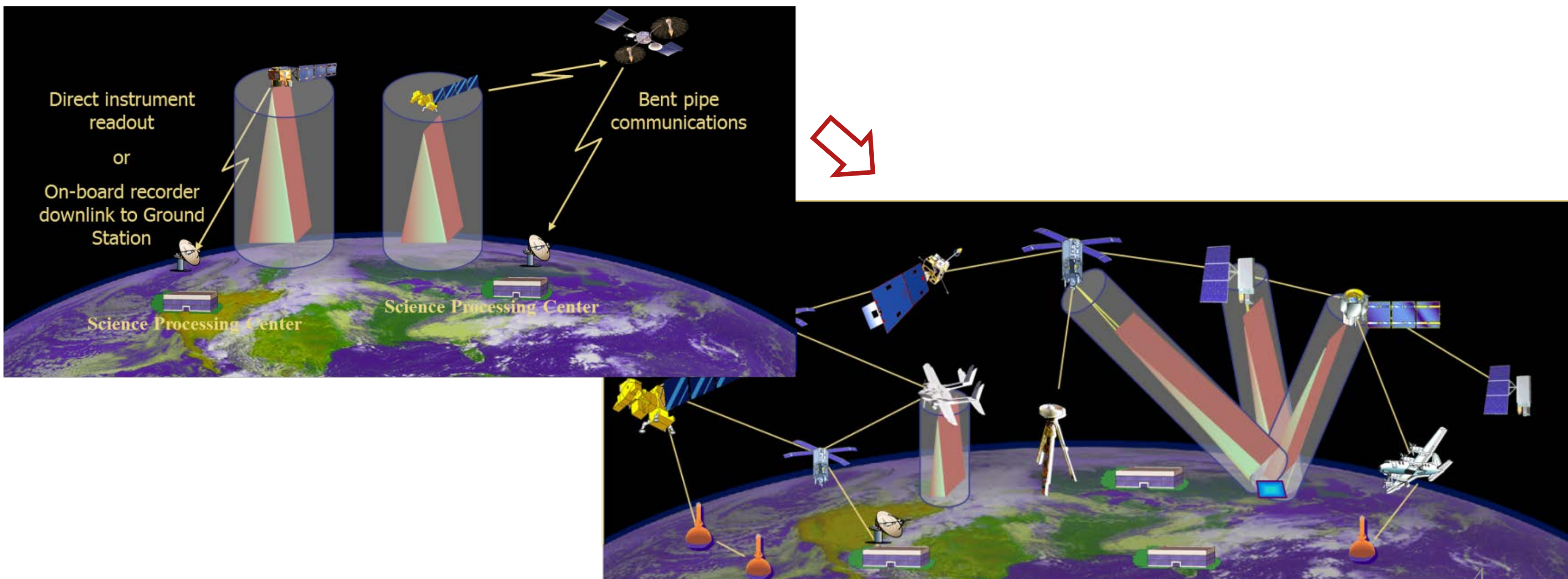


Image credit: NASA

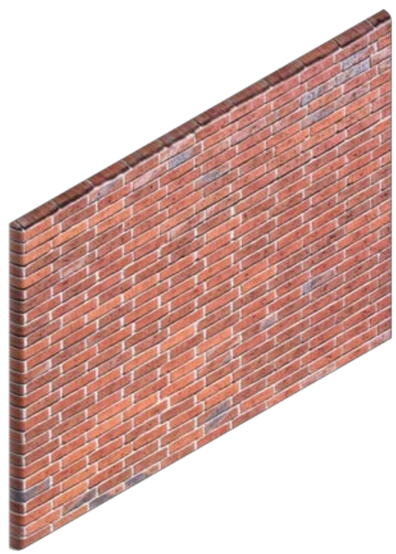
How are complex systems designed today?

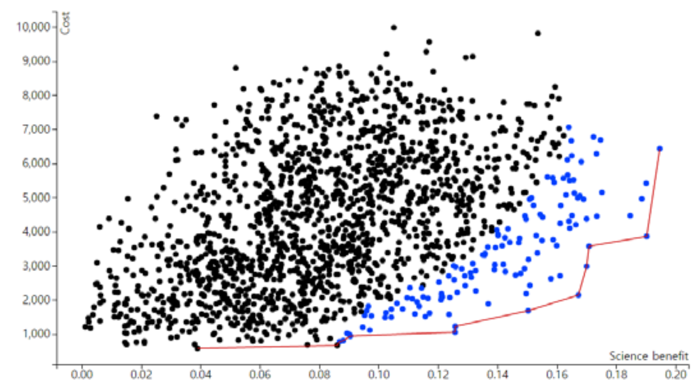
Qualitative approaches

- Focus on creativity (e.g. brainstorming), empathy (e.g., design thinking) and consensus (e.g., Delphi)

Quantitative approaches

- Focus on rigor, consistency, exhaustiveness, and choosing “most preferred” design (e.g., decision-based design)



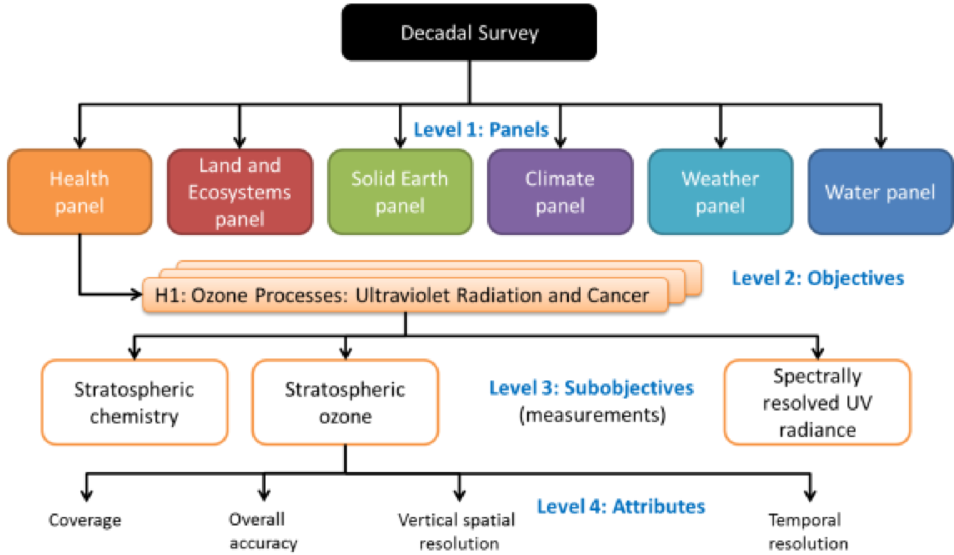


Human-centered design

Automated design

Example Problem & Tool: Decadal Survey

Earth Science Decadal Survey is an NRC study that recommends a set of Earth observing missions to cover needs of Earth sciences



Given the 6 following missions and their 13 instruments:

SMAP (2), ICESAT-II (1), DESDYNI (2), CLARREO (3), ASCENDS (3), HYSPIRI (2)

Is there a better partitioning architecture?

27 million architectures

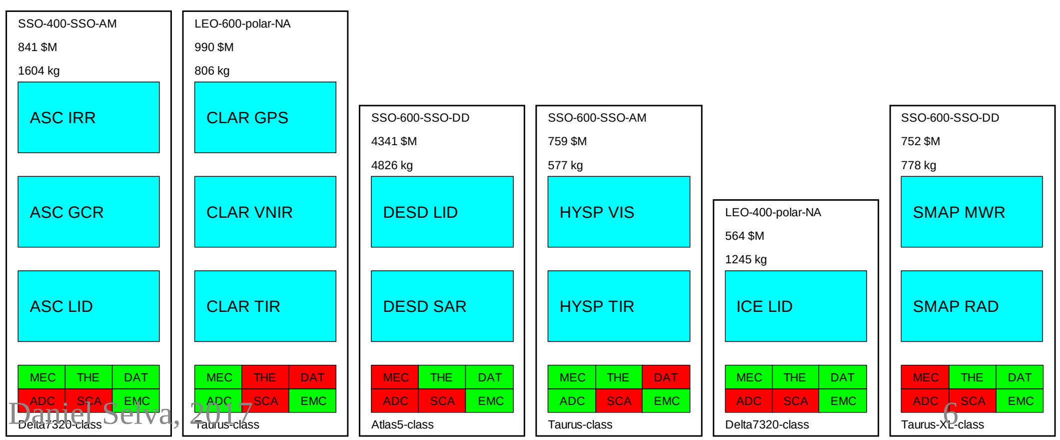
Metrics:

Science

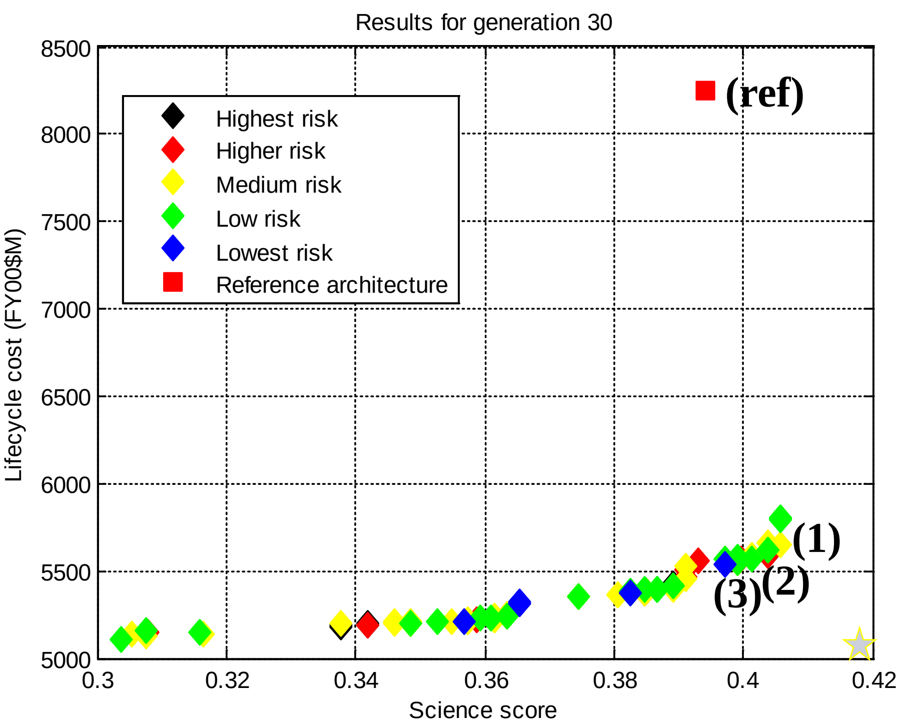
Lifecycle cost

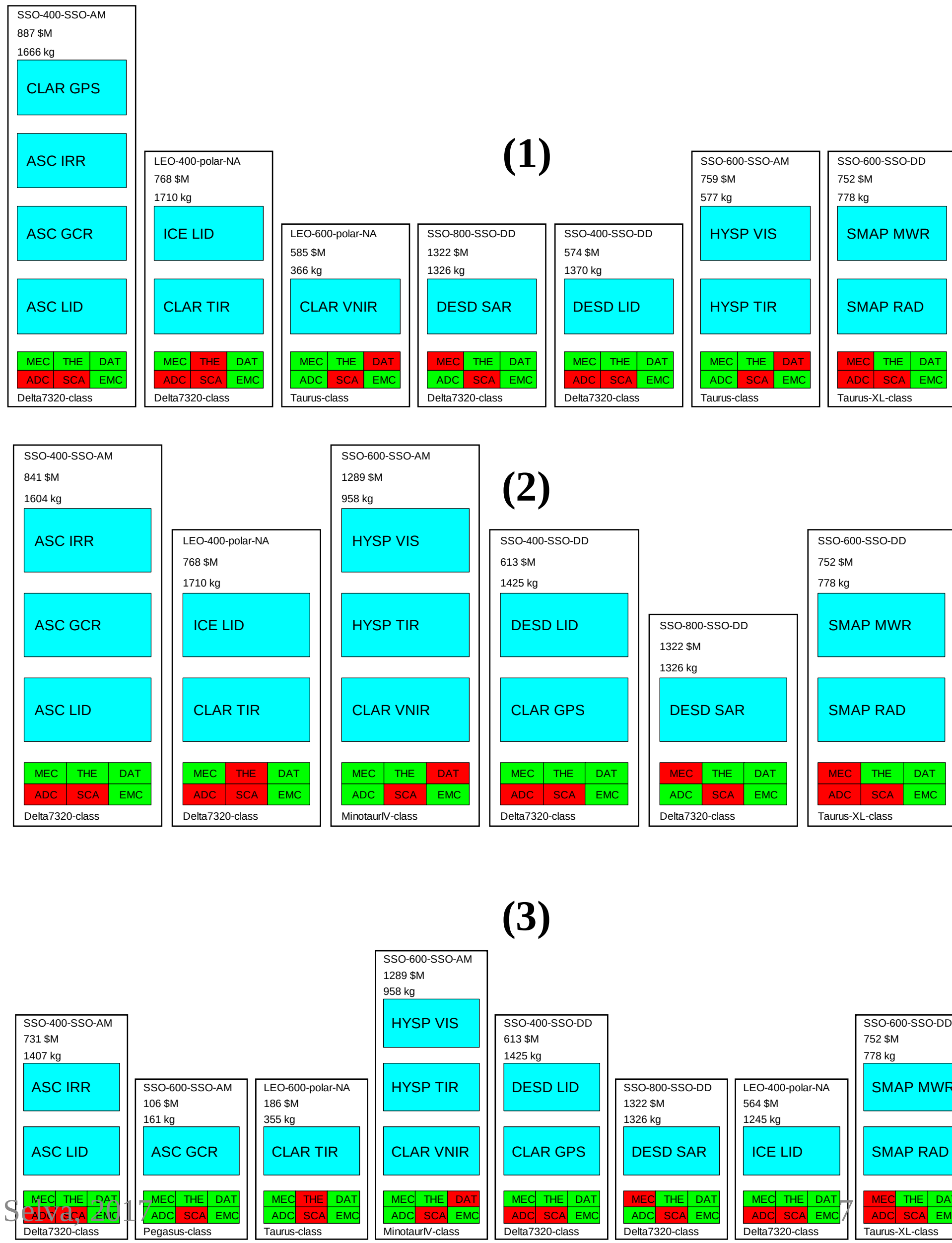
Launch risk

Programmatic risk



Example of results





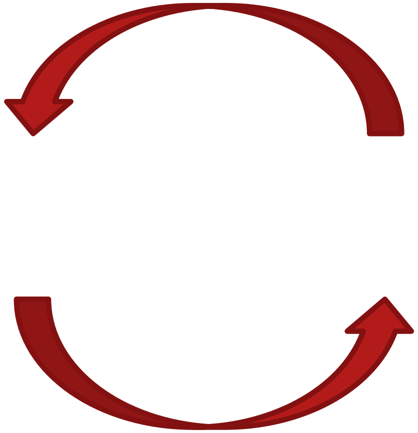
Selva, D., Cameron, B. G., & Crawley, E. F. (2014). Rule-Based System Architecting of Earth Observing Systems: Earth Science Decadal Survey. *Journal of Spacecraft and Rockets*, *51*(5), 1505–1521. http://doi.org/10.2514/1.A32656

Some challenges in design of complex systems

- How can we **represent and leverage the knowledge** human experts have to improve efficiency and effectiveness of design tools?
- How can tools help humans **discover knowledge** such as the high-level design features and trade-offs driving the cost and performance of complex systems?

Outline for today: Duplex “knowledge transfer”

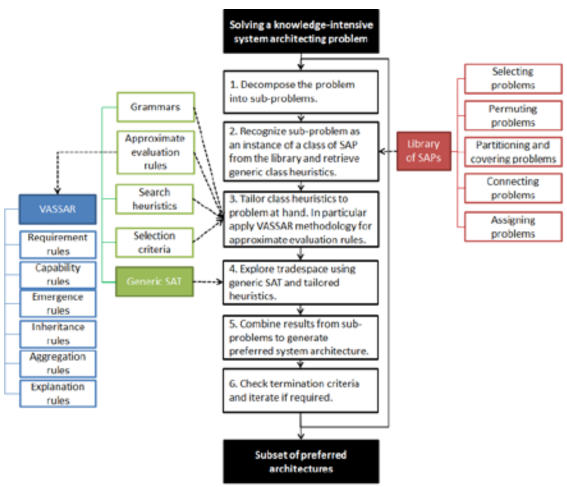
- How can we represent and leverage the knowledge human experts have to improve design of complex systems?
  - **VASSAR:** scalable and traceable knowledge-intensive architecture evaluation
  - **KDO/AOS**: self-organizing robust optimization of complex design spaces

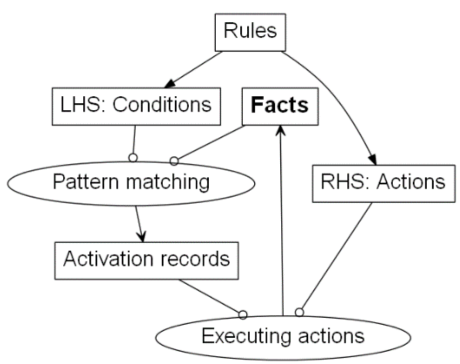


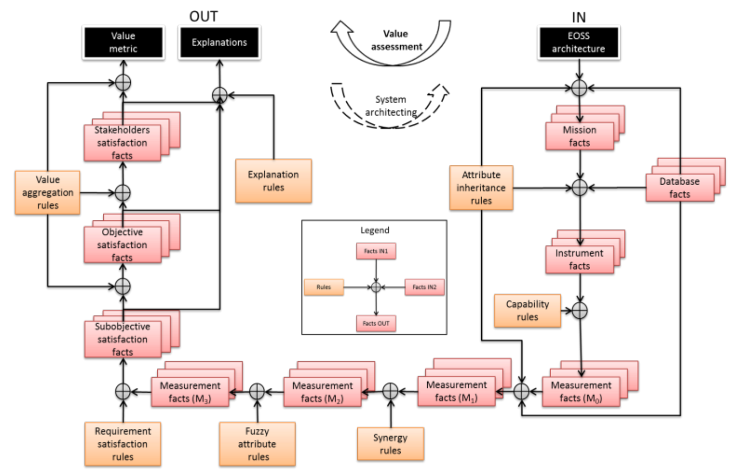
- How can we help humans discover the high-level design features and trade-offs driving the cost and performance of complex systems?
  - **i-feed**: interactive extraction of high-level association rules and compact mental models in design spaces
  - **cityplot:** visualization of multi-objective mixed-integer design spaces

VASSAR: leveraging knowledge in evaluation

- The need for knowledge-intensive evaluation
  - Traceability and Scalability
  - Knowledge-based systems
- An architecture for knowledge-intensive value functions







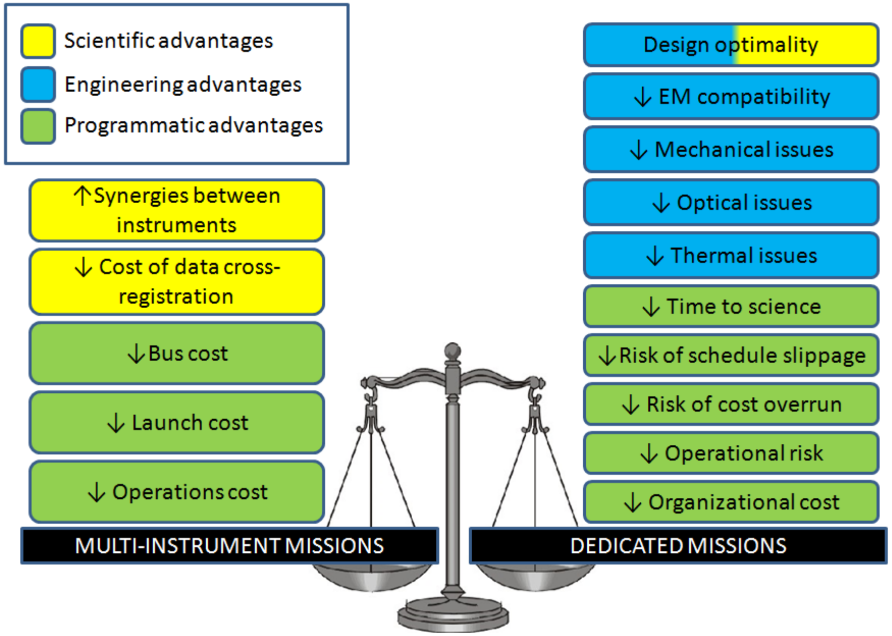
A complicated "knowledge-intensive" problem

- Consider Decadal Survey example

- Lots of knowledge must be put into the model in order for it to be useful

- This also makes it hard to debug the model: when you get a result that doesn't make sense, it's hard to trace it back to individual assumptions.

- Regular models lack scalability and traceability of knowledge. Can we do something better?

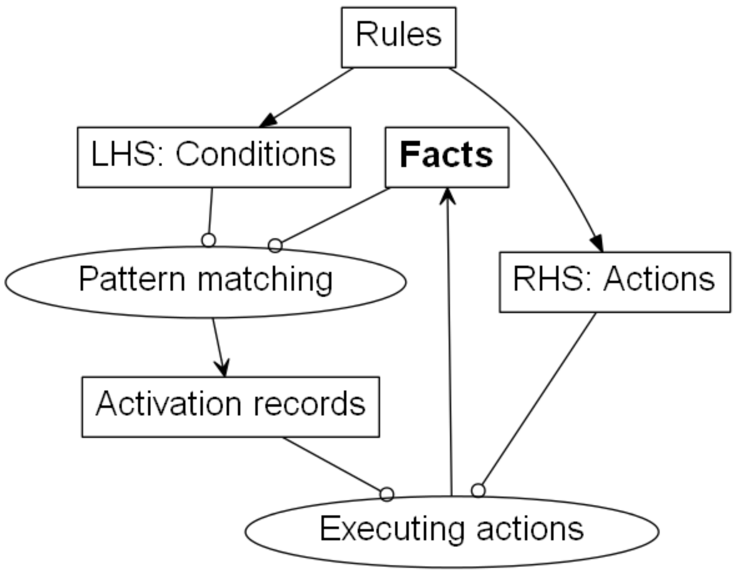


Selva, D. & Crawley, E., 2010. Integrated assessment of packaging architectures in Earth observing programs. In *Aerospace Conference, 2010 IEEE*. Big Sky: IEEE, pp. 3–12.

Knowledge-based systems to the rescue

- Experts store their knowledge in "chunks" that fit well the structure of **logical rules** *(Newell & Simon, 1972)*

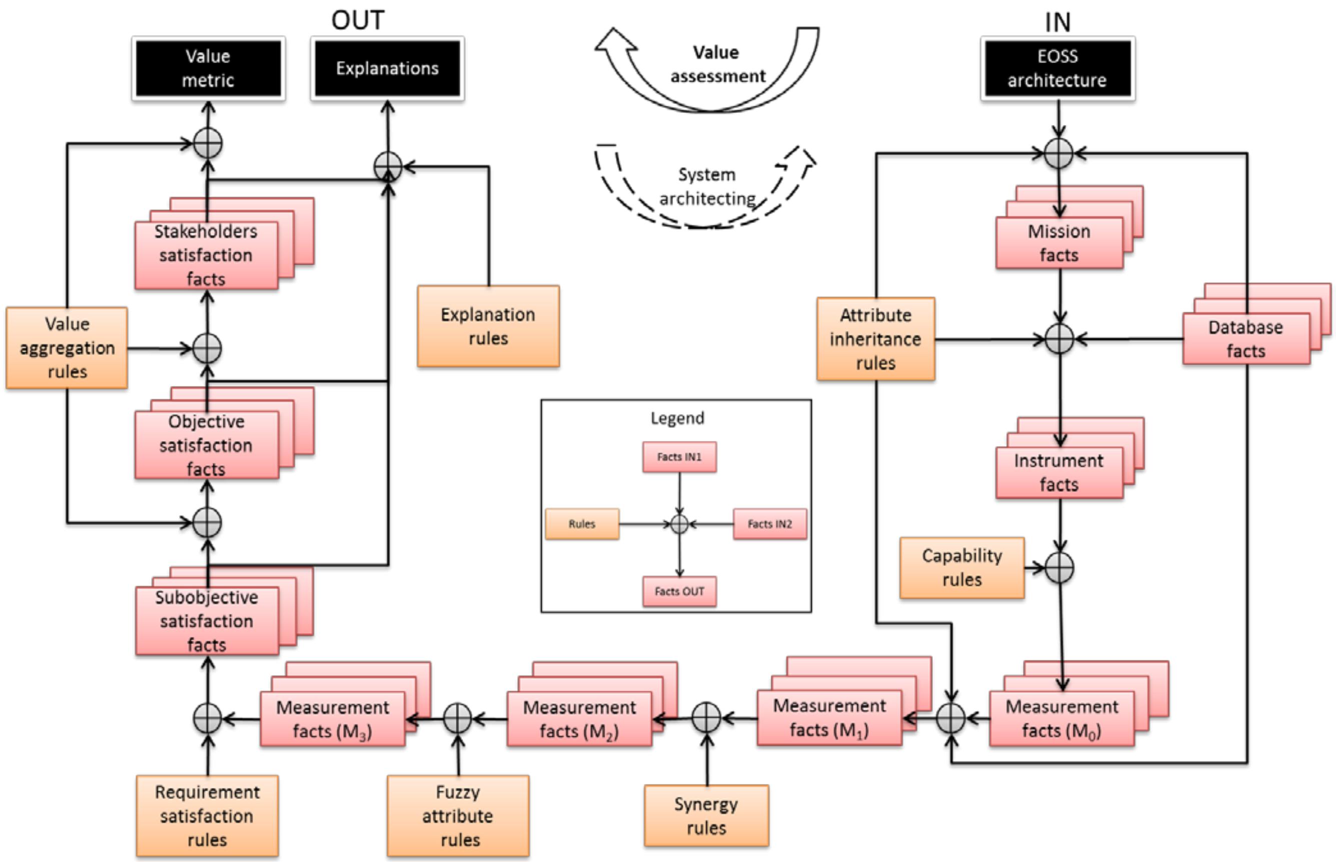
- Rule-based systems solve complex problems by using thousands of logical rules in a computer program, imitating human reasoning
  - First Rule-based System: **MYCIN experiment** *(Buchanan & Shortliffe, 1984)*




```
(defrule soil-moisture-disaggregation
(MEASUREMENT (spatial-resolution High) (accuracy Low))
(MEASUREMENT (spatial-resolution Low) (accuracy High))
=>
(assert (MEASUREMENT (spatial-resolution High) (accuracy High))))
```

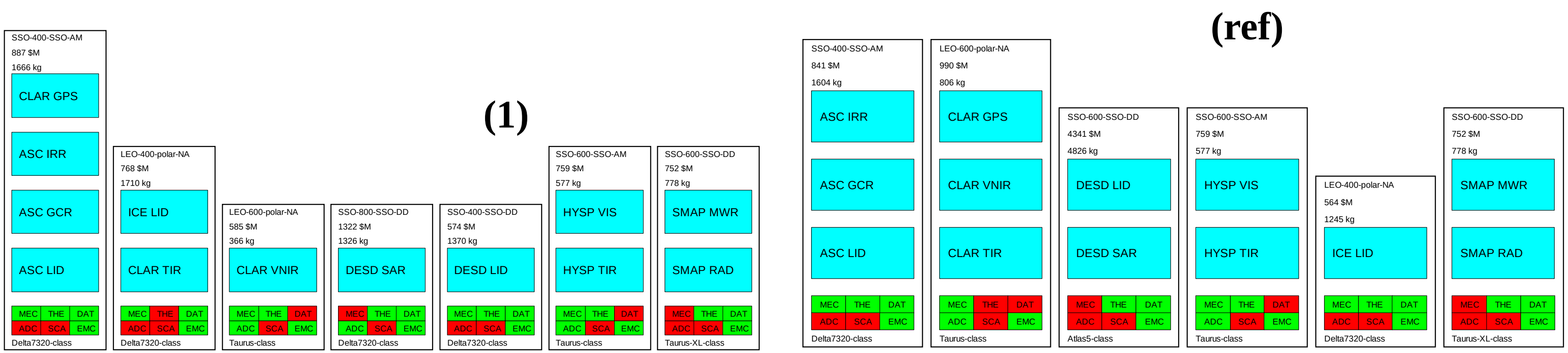
```
(defrule RF-interference
(INSTRUMENT (Illumination Active) (frequency ?f) (spacecraft ?s))
?i <- (INSTRUMENT (Illumination Passive) (frequency ?f) (spacecraft ?s))
=>
(modify ?i (RF-interference High)))
```

VASSAR: An architecture for knowledge-intensive evaluation of space systems



Selva, D. & Crawley, E., 2013. VASSAR: Value Assessment of System Architectures using Rules. In *Aerospace Conference, 2013 IEEE*. Big Sky: IEEE. Conference Best Paper Award.

VASSAR – Automatic Generation of Explanations



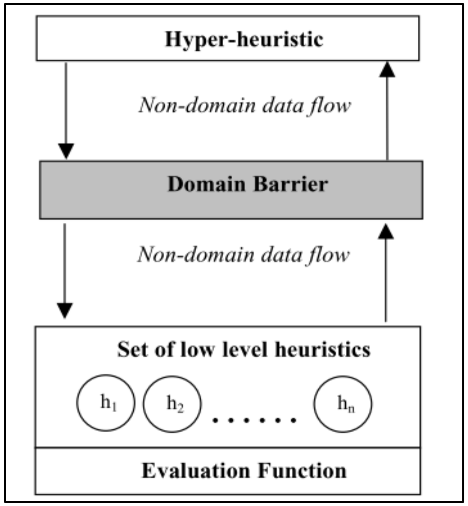
- Science: (1) has higher science than (ref) because:
  - (ref) partially misses subobjectives related to surface deformation and hydrorcabon reservoir monitoring due to SAR flying at 600km compromise orbit

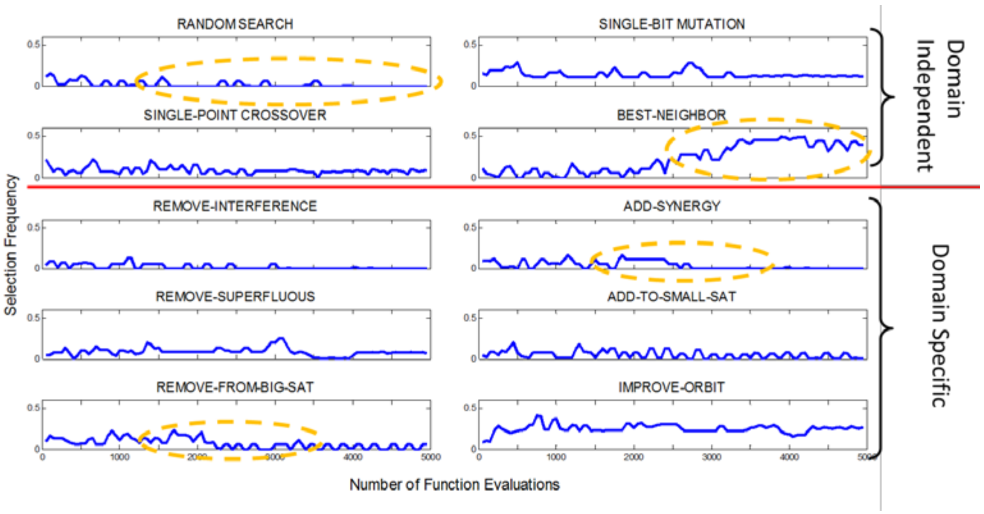
- Cost: (1) also has lower cost than (ref) because:
  - (ref) has to put a lidar at 600km (higher instrument and bus cost)
  - Lower launch costs (1 Atlas 5 = $110M > 2xD7320 = $90M)

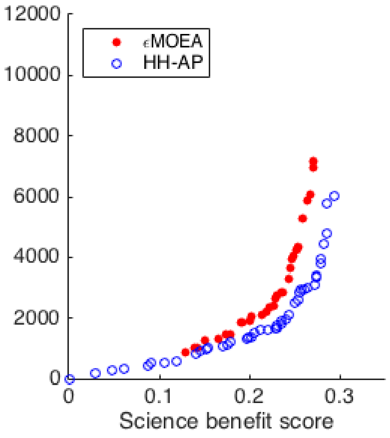
Selva, D., Cameron, B.G. & Crawley, E.F., 2014. Rule-based System Architecting of Earth Observing Systems: The Earth Science Decadal Survey. *Journal of Spacecraft and Rockets*.

KDO/AOS: leveraging knowledge in optimization

- Multi-objective evolutionary optimization
- Domain-specific and domain-independent operators
- Knowledge-driven optimization
- Adaptive operator selection

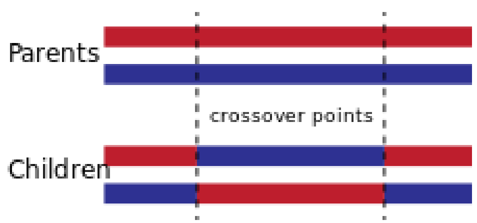




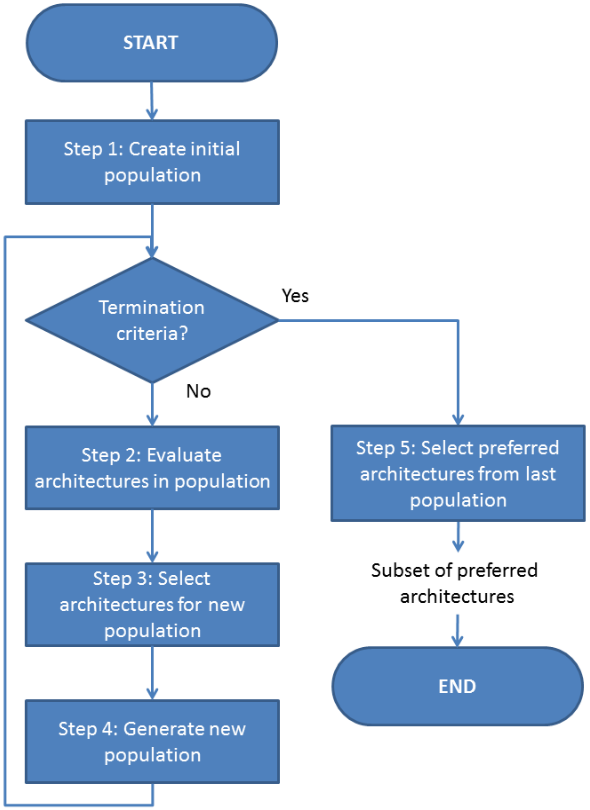


Evolutionary algorithms are useful and popular

- Evolutionary algorithms such as NSGA-II, ϵ-MOEA and others are very popular in design
- Evolve a population of solutions (designs) by iteratively applying a set of operators.
- Domain-independent operators
  - Crossover
  - Mutation



NASA ST5 spacecraft antenna (Wiki)



Hyper-heuristics – Self-organizing optimization

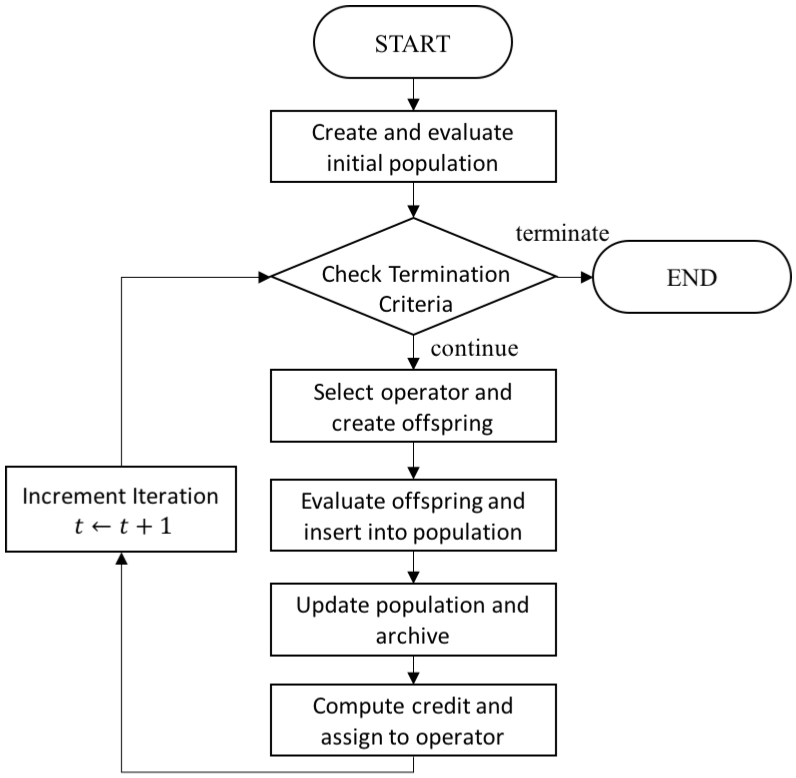
- Use a set of operators/heuristics O
  - Domain-independent: Different kinds of Crossover, mutation, etc.
  - Or Domain-specific!
- Credit assignment: Measure performance of each operator over time
  - $c_{i,t}$ = credit received by o_i at iteration t
  - Example: $c_{i,t} \propto f(\vec{x}^p) - f(\vec{x}^{o_i,t})$
- Operator selection: Assign solutions to operators proportionally to their quality ($q_{i,t}$ = quality of o_i at iteration t)

$$q_{i,t+1} = (1-\alpha)\cdot q_{i,t} + \alpha \cdot c_{i,t}$$

$$p_{i,t+1} = p_{min} + (1 - |O| \cdot p_{min}) \cdot \frac{q_{i,t+1}}{\sum_{j=1}^{|O|} q_{j,t+1}}$$

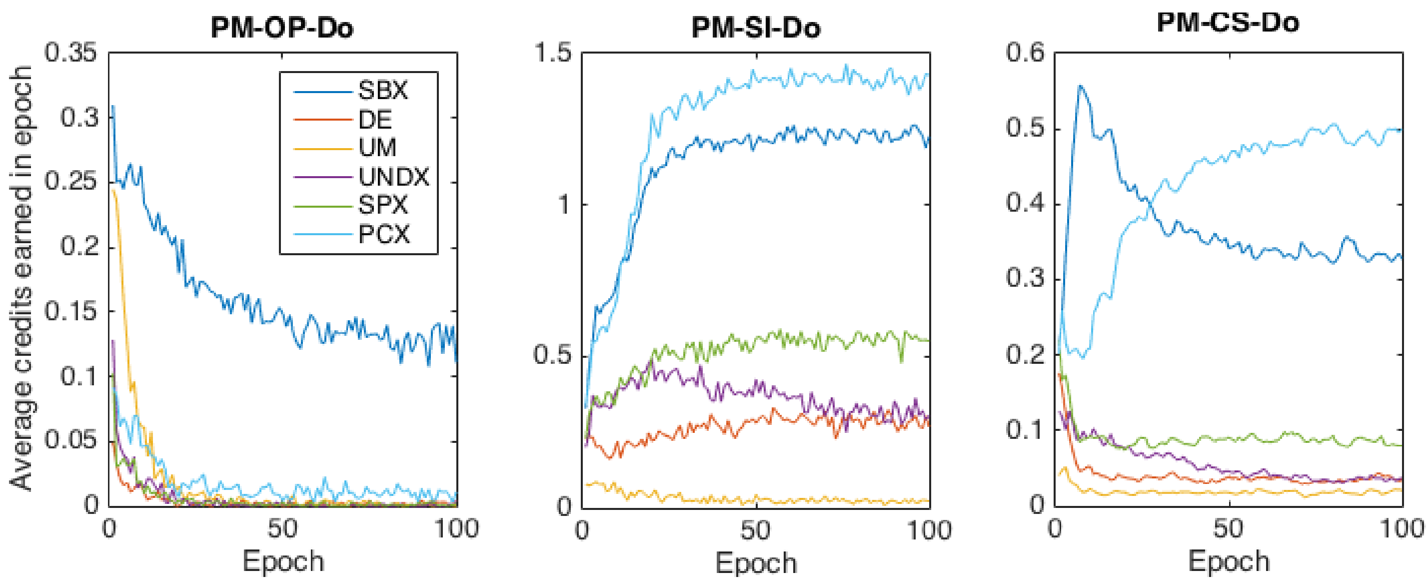
$a \in [0,1]$ = adaptation rate

p_{min} = minimum selection probability



Hyper-heuristics outperform state-of-the-art EA

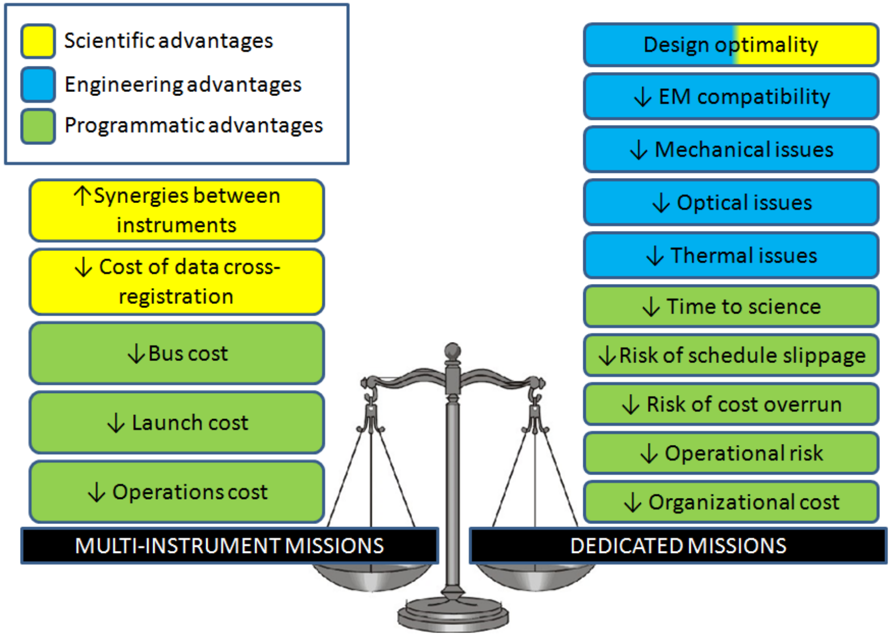
- We measured performance of 9 different HH (new and existing) on 26 different benchmarking problems (WFG, UF, DTLZ)
- Our experiments show that HH consistently outperform state-of-the-art EA over wide range of problems
- HH are able to discover the operator(s) that work better for each problem



Hitomi, N., and Selva, D., A Classification and Comparison of Credit Assignment Strategies in Multiobjective Adaptive Operator Selection. Under review in *IEEE Transactions on Evolutionary Computation*.

Domain knowledge is available – why not use it?

- Examples of chunks of expert knowledge
  - Put synergistic instruments together
  - Putting 2+ high-energy instruments together not good
  - Don't put optical instruments in low-light orbits
  - Avoid mechanical, thermal and electromagnetic interferences
- Why not take advantage of the domain-specific knowledge available?

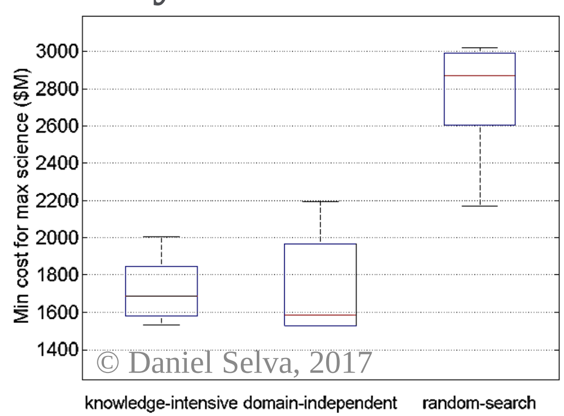


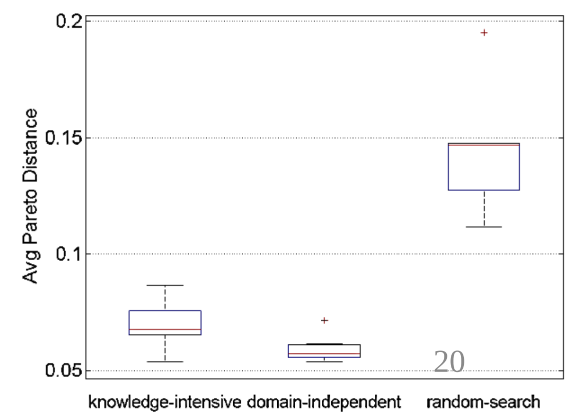
Incorporating domain knowledge is not trivial

- Incorporating domain knowledge can be done by means of
  - Constraints
  - Initial population
  - Operators
  - Human in the loop
- Our early experiments showed that using domain-specific operators lead to faster but **premature convergence** and **lack of diversity** in the population .

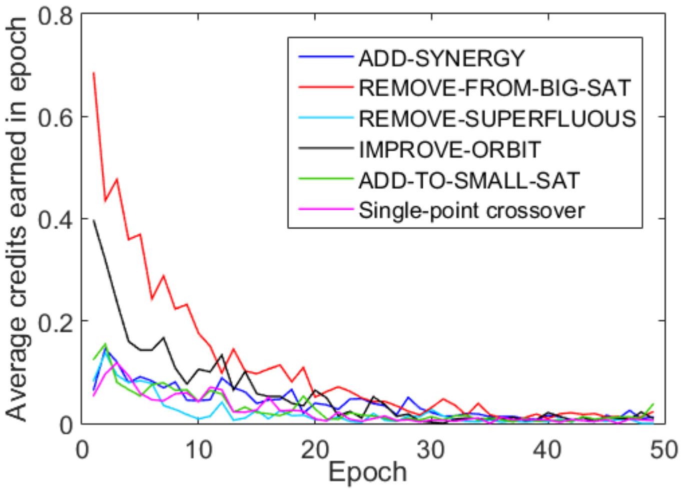
Example of domain-specific operators

Heuristic	Description
ADD-SYNERGY	Adds instrument to a random orbit so as to capture a currently missed synergy
REMOVE-INTERFERENCE	Removes instrument from random orbit so as to eliminate a current interference
IMPROVE-ORBIT	Moves random instrument to a better orbit
REMOVE SUPERFLUOUS	Removes superfluous instrument from a random orbit
ADD-TO-SMALL-SAT	Adds random instrument to a random small satellite
REMOVE-FROM-BIG-SAT	Removes random instrument from a random big satellite

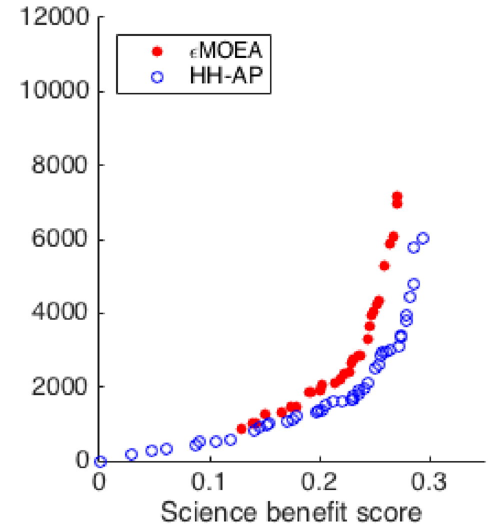


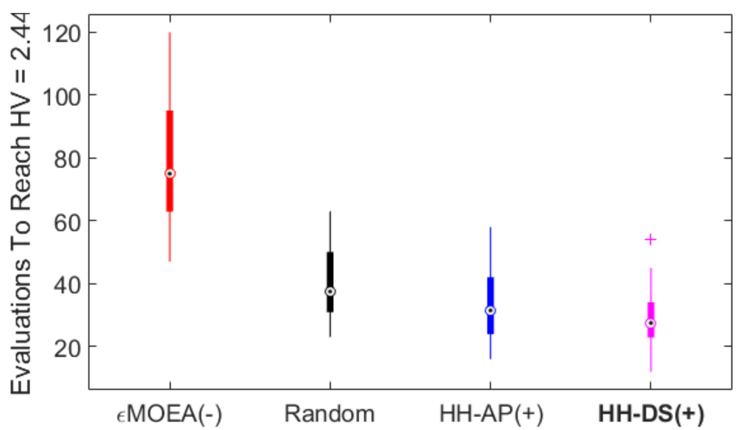


Hyper-heuristics enable using domain knowledge



Hyper-heuristics can use domain-specific heuristics to accelerate optimization at beginning and then shut them down to avoid premature convergence!

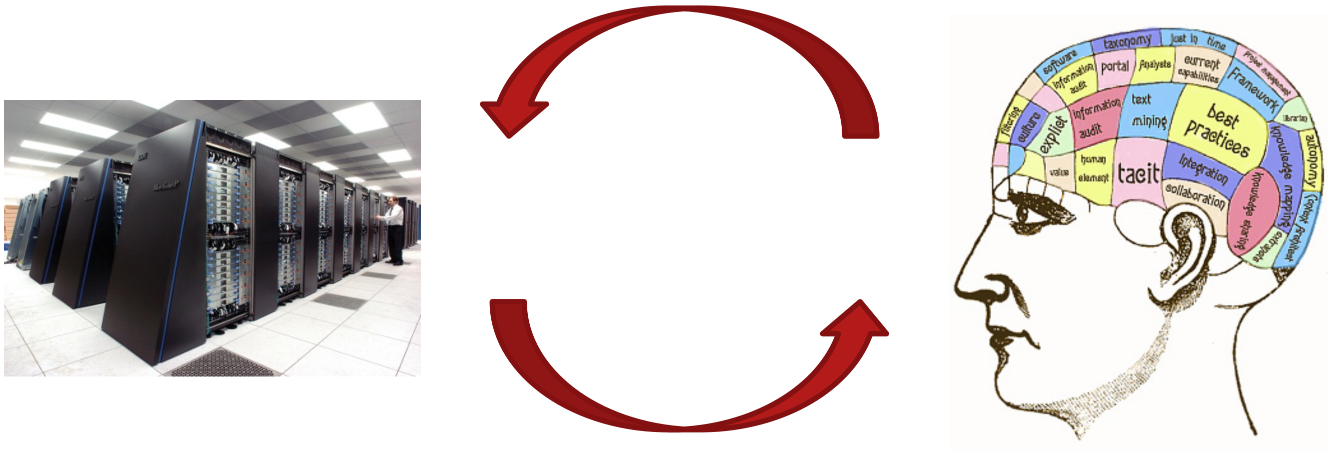




Hitomi, N., & Selva, D. (2016). A hyperheuristic approach to leveraging domain knowledge in multi-objective evolutionary algorithms. Accepted to *ASME 2016 International Design Engineering Technical Conferences*. Charlotte, NC.

Outline for today: Duplex “knowledge transfer”

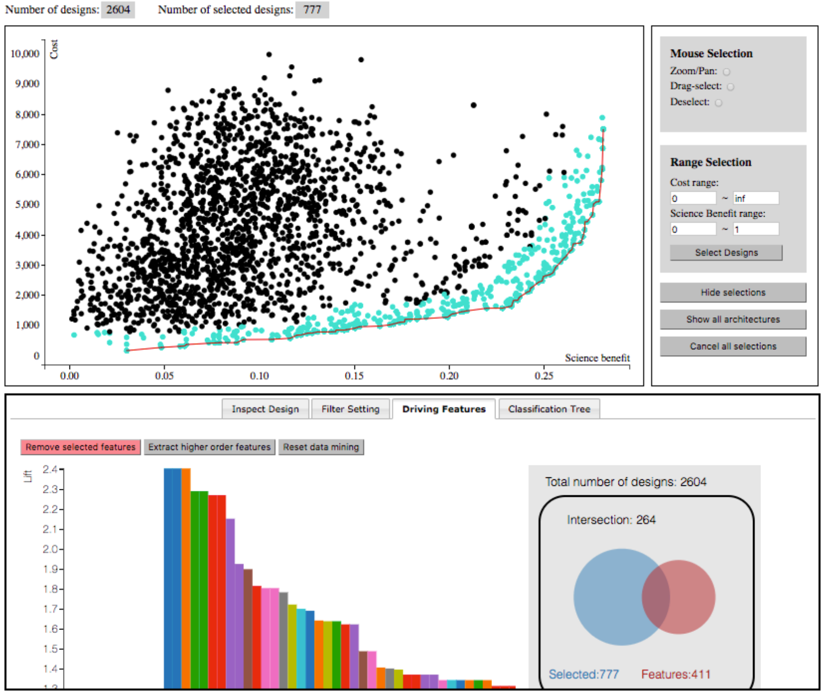
- How can we represent and leverage the knowledge human experts have to improve design of complex systems?
  - **VASSAR:** scalable and traceable knowledge-intensive architecture evaluation
  - **KDO/AOS**: self-organizing robust optimization of complex design spaces



- How can we help humans discover the high-level design features and trade-offs driving the cost and performance of complex systems?
  - **i-feed**: interactive extraction of high-level association rules and compact mental models in design spaces
  - **cityplot:** visualization of multi-objective mixed-integer design spaces

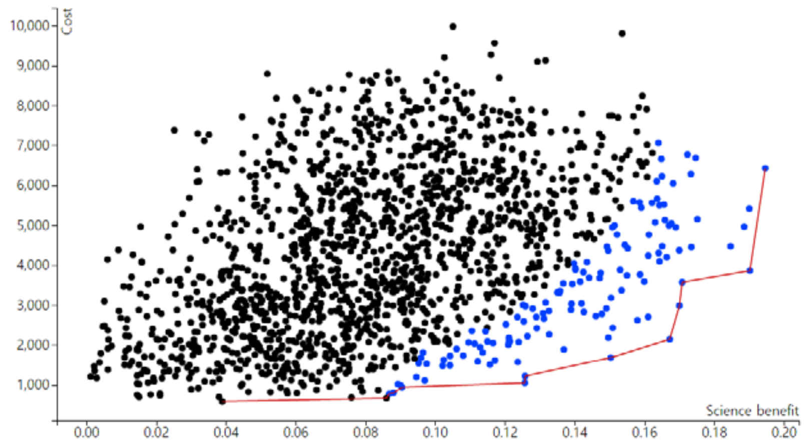
iFEED: Interactive Feature Extraction for Engineering Design

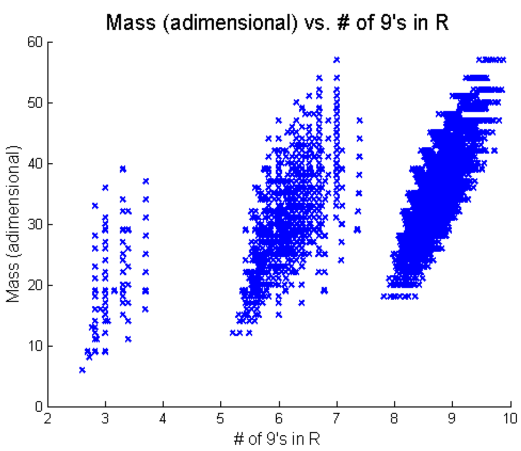
- Beyond optimization – “learning”
- Feature selection with association rule mining
- Feature structure and prediction with classification tree



It’s not only about optimizing – also about “learning”

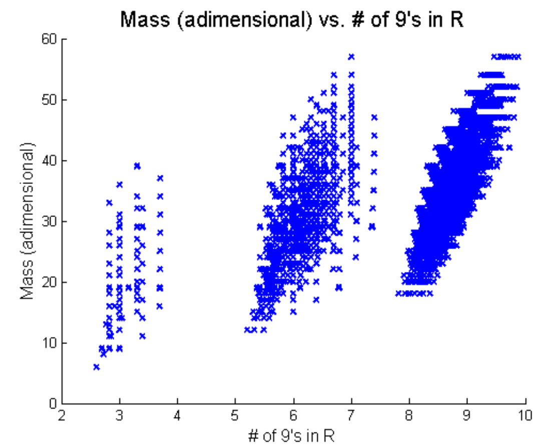
- Users of design optimization tools: *“Actual output of the tool is not the most important part; it’s all about what is learnt while using the tool.”*
- What is it that they learn?
  - What metrics are more sensitive to what design variables?
  - What, if anything, do good (or bad) architectures have in common?
  - What combinations of variables drive the formation of clusters in the design space?
  - How much of this is generalizable to other concepts?



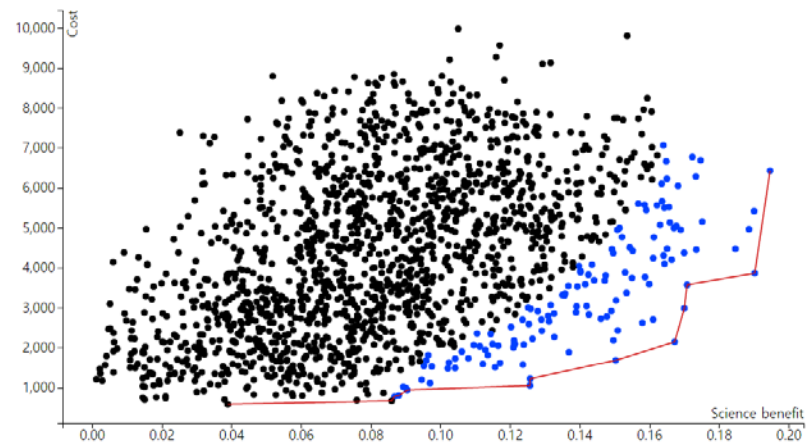


Driving features

- <u>Feature</u>: Any design variable or function of several design variables
  - **Driving feature**: A feature that is consistently found in a class of designs more often than in the others
- <u>Goal</u>: Automatically identify a <u>small</u> set of driving features with <u>high predictive power</u> for goodness of a design (**feature extraction problem**)
  - Spoiler alert: Driving features are rarely just design variables;
  - instead, they tend to be high-level combinations of multiple design variables
  - Intermediate variables can help generalize



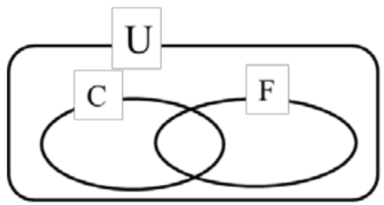
Driving feature: min(#sensors, #computers)



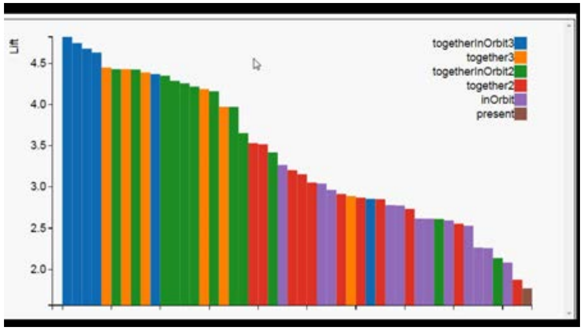
Driving features:
At least 3 orbits populated
No 2+ large instruments together
IRS + MWS + CHEM on same orbit

Driving Feature Detection – Association Rules

- Given a binary feature space, **association rule mining** is a simple unsupervised ML technique to find rules of the form: F $\Longrightarrow$ C where F and C are two features.
- Based on the following metrics:
  - $supp(F) \equiv \frac{|F|}{|U|}$
  - $conf(F \Rightarrow C) = \frac{supp(F \cap C)}{supp(F)}$
  - $lift(F \Rightarrow C) \equiv \frac{conf(F \Longrightarrow C)}{supp(C)}$



U: All possible designs
C: Designs within target region
F: Designs with the feature

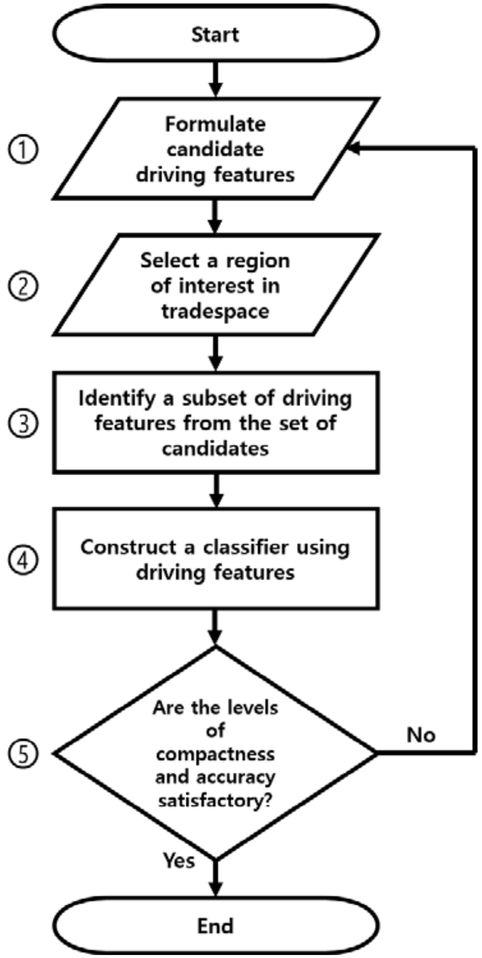


- Simple algorithm: Try all possible rules. A rule on a high-support feature that has high confidence and high lift is a driving feature.
  - What is high? Thresholds must be defined.
- <u>Problem</u>: the feature space is larger than the design space, possibly with infinite representations (**curse of dimensionality**).
  - Example: the number of features in a binary space of length N is $3^N > 2^N$

iFEED: Bringing the human into the loop

1. Formulation of candidate driving features
2. Selection of a region of interest in the tradespace (e.g. high performance, low cost, and low Pareto ranking designs)
3. Identifying a subset of driving features from the candidate features, using association rule mining
4. Building a compact form of classifier using driving features as predictors and the region of interest as the label
5. Evaluation of the classifier.

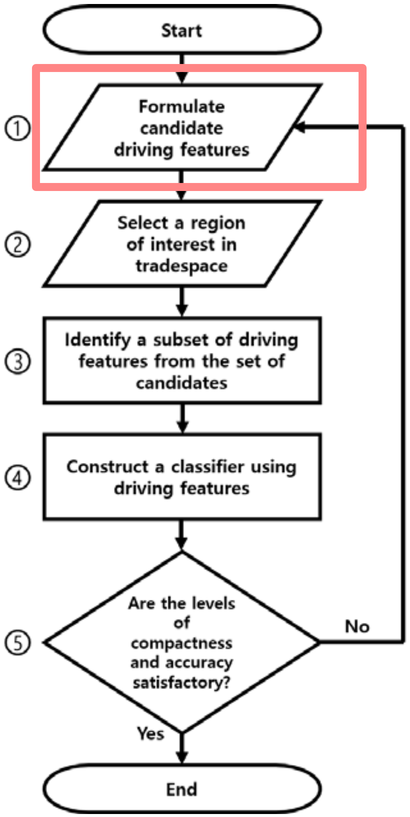
Iterate if needed



1. Formulation of candidate driving features

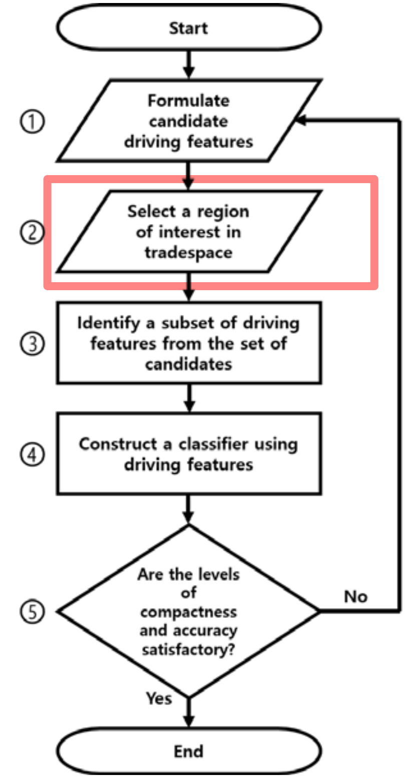
- Formulation of candidate driving features using **domain-specific knowledge** and insights obtained by observing **the structure of the problem**
- Limits the feature space to be searched (number of all possible features is unbounded!)
- Opportunities to search high level features

Name of the feature	Arguments	Description
Present	I_i	Instrument I_i is present in at least one of the orbits
Absent	I_i	Instrument I_i is absent in all the orbits
InOrbit	O_i, I_j	Instrument I_j is present in orbit O_i
NotInOrbit	O_i, I_j	Instrument I_j is not present in orbit O_i
Together	$I_i, I_j, (I_k)$	Instruments I_i, I_j (and I_k) are present together in any orbit
TogetherInOrbit	$O_i, I_j, I_k, (I_l)$	Instruments I_j, I_k (and I_l) are present together in orbit O_i
Separate	$I_i, I_j, (I_k)$	Instruments I_i, I_j (and I_k) are not present together in any single orbit
emptyOrbit	O_i	No instrument is present in orbit O_i
numOrbitUsed	n	The number of orbits that have at least one instrument assigned is n

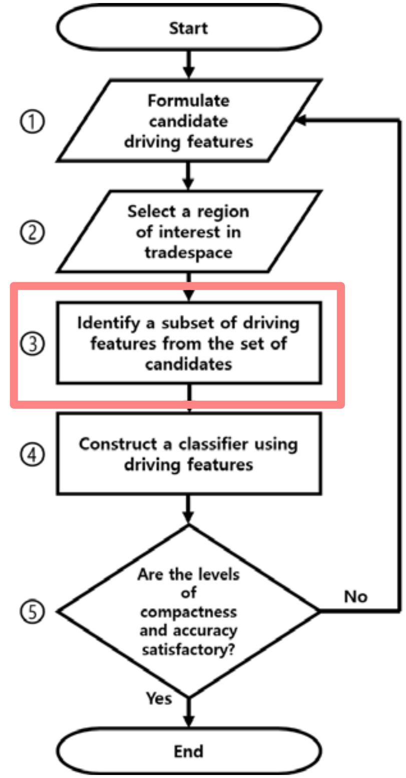


2. Selection of a region of interest

- Selection of the target region in the objective space
  - Defining goodness function (e.g. expected utility, NPV, Pareto ranking)
  - Visual selection of designs



3. Identifying driving features

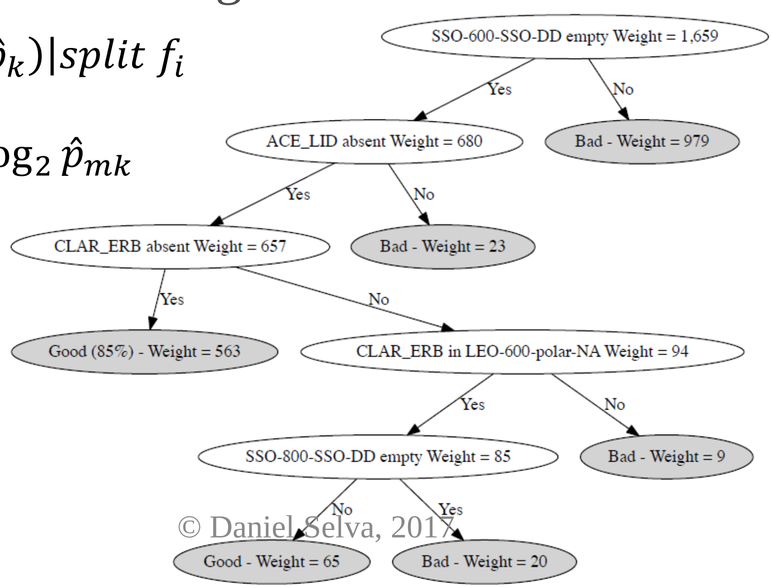


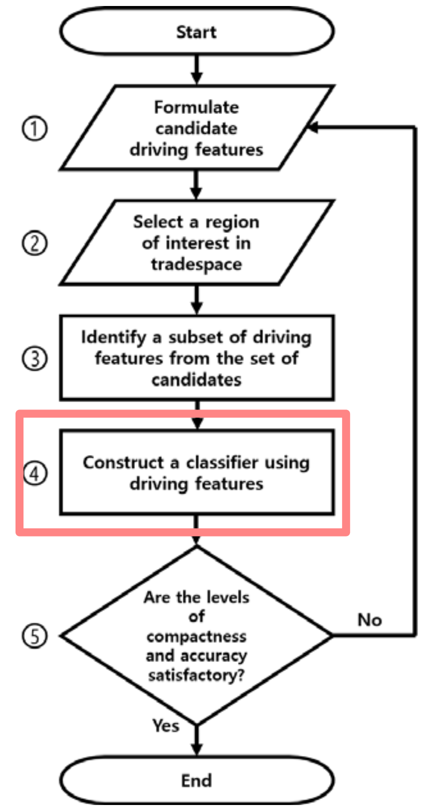
4. Building classification tree using driving features

- **Classification trees** introduce hierarchical structure and priority information about driving features
  - They are among the most human-understandable machine learning models
- Driving features are selected as test nodes of the classification tree. C4.5 algorithm recursively selects features with largest information gain

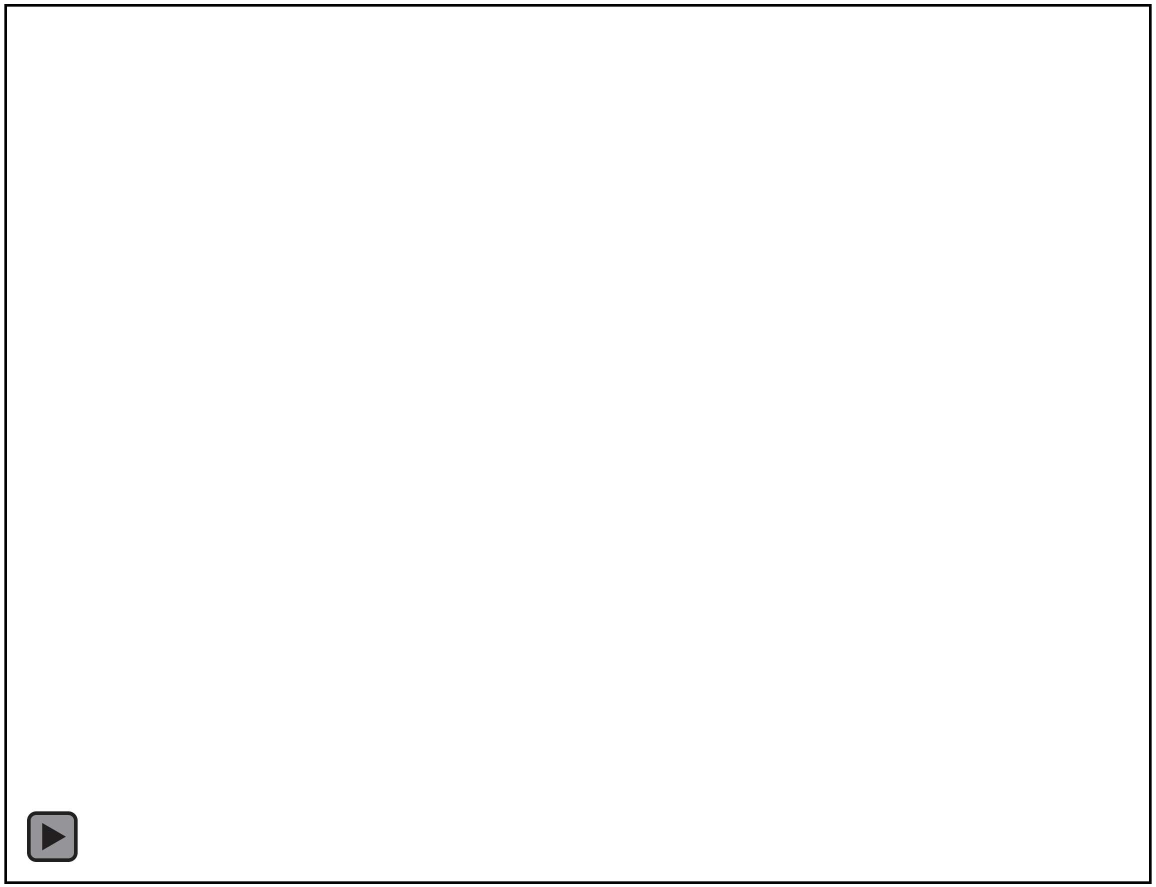
$$IG(f_i) = H(\hat{p}_k)|no\ split\ - H(\hat{p}_k)|split\ f_i$$

$$H(\hat{p}_k) = -\sum_{m=1}^{M} \hat{p}_m \sum_{k=1}^{K} \hat{p}_{mk} \log_2 \hat{p}_{mk}$$





4. Building classification tree using driving features



The Vision

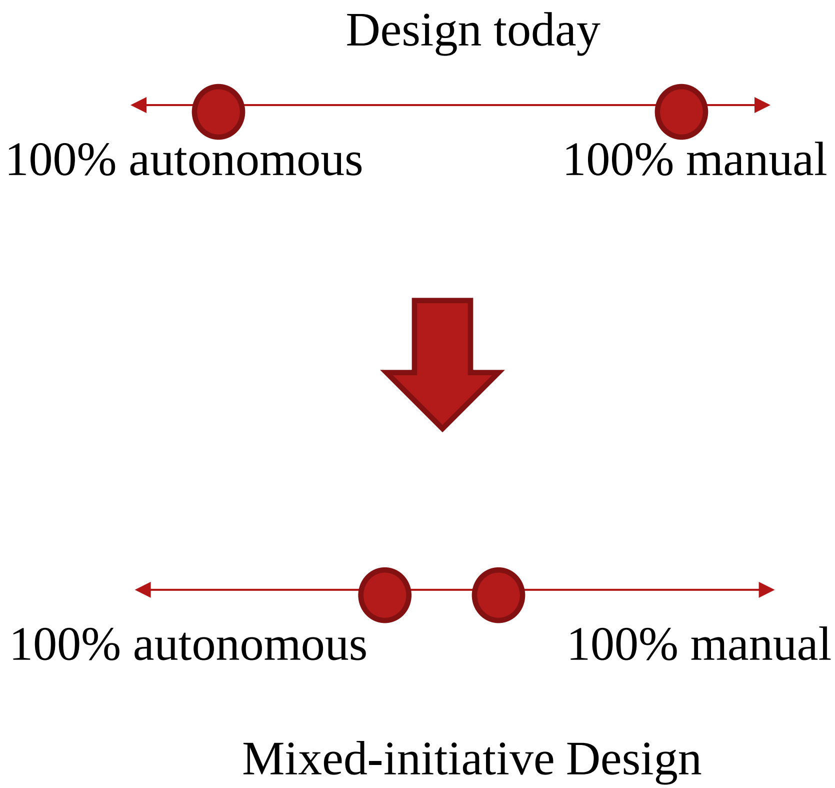
Towards cognitive design assistants and mixed-initiative design

So far, focus on making design tools smarter

- **Emphasis in engineering design has been to make design tools more intelligent = adaptive**
- **Adaptive**…
  - Formulations (variables)
    - Design vector "zooms in" on variables (I. Y. Kim and O. L. de Weck, "Variable chromosome length genetic algorithm for progressive refinement in topology optimization," Struct. Multidiscip. Optim., vol. 29, no. 6, pp. 445–456, Jan. 2005)
  - Models (objective function)
    - Bayesian Optimization (P. Frazier and J. Wang, "Bayesian Optimization for Materials Design," in Information science for materials discovery and design, vol. 225, T. Lookman, F. J. Alexander, and K. Rajan, Eds. 2015, pp. 45–75)
  - Search strategies
    - Parameter tuning (G. S. Tewolde, D. M. Hanna, and R. E. Haskell, "Enhancing performance of PSO with automatic parameter tuning technique," 2009 IEEE Swarm Intell. Symp., no. 1, pp. 67–73, Mar. 2009)
    - Adaptive operator selection (E. Burke, G. Kendall, J. Newall, and E. Hart, "Hyper-heuristics: An emerging direction in modern search technology," in International series in operations research and management science, 2003, pp. 457–474)

Vision

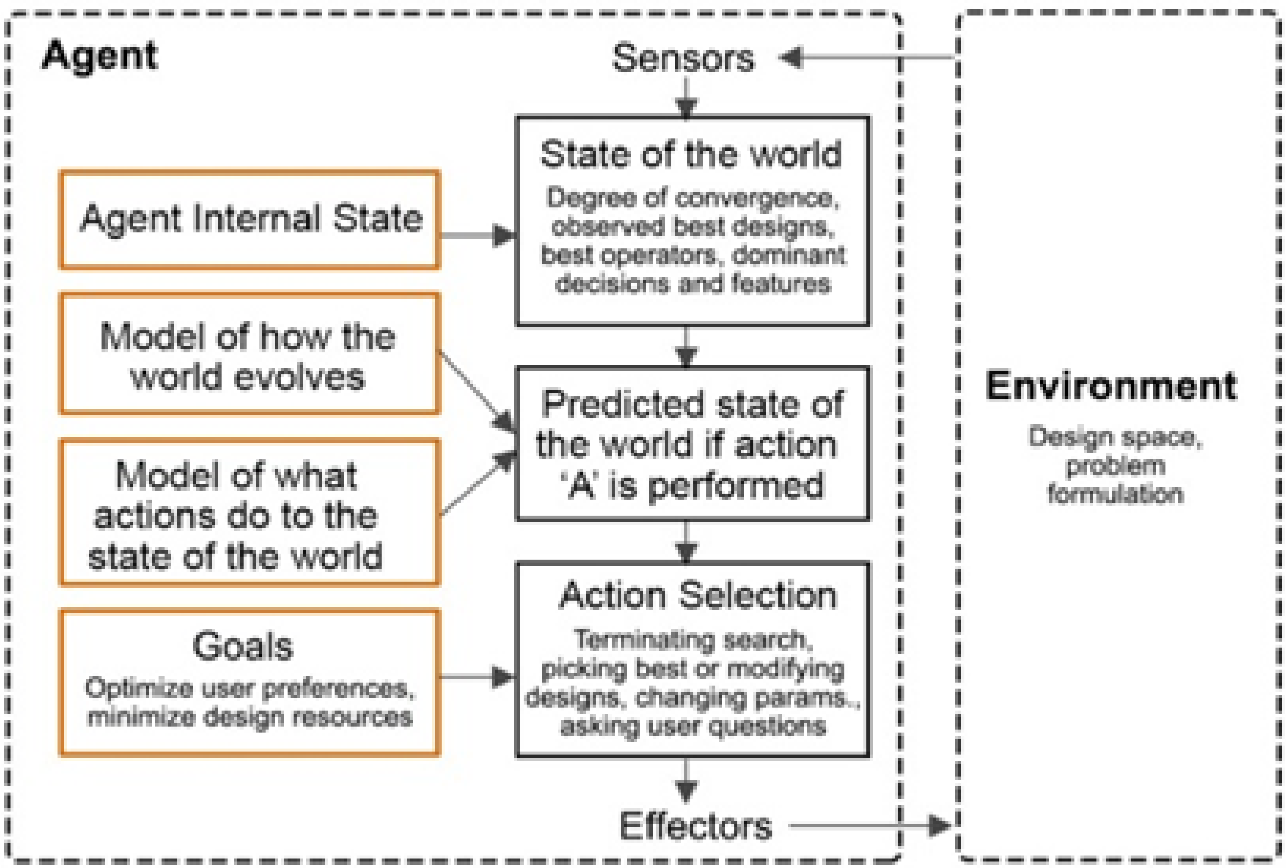
- Cast design tools as intelligent agents
- Try to learn from intelligent systems (e.g., robotics) community
  - For unstructured and uncertain tasks, robotics community has moved away from autonomy to human-robot collaboration and mixed initiative
- Currently, either 100% human or fully automated design (autonomy)
- Wait: Design IS a highly unstructured task+!
- Propose mixed-initiative design#!!



Mixed-initiative Design

- Informally: no clearly defined master-slave roles, more balanced role allocation
- Shared plans and intentions
- Dynamic role/function allocation
- Shared attention and “common ground”
- Trust-building, collaborative interaction
- Embodied interaction

Design tools as intelligent agents



A plan to implement that vision

State of the art

- Some intelligence in design agents
- No dialogues or interactions
- Static & unbalanced role allocation, emphasizing either 100% manual or 100% automated design
- Mostly static visual human-agent communication

Improved human-computer interaction

- Self-explaining abilities
- New roles, balanced role allocation & mixed-initiative
- Meaningful dialogues & interactions
- Verbal & non-verbal communication, embodied interaction

New paradigm: Mixed-initiative human-agent design

- Shared plans and intentions
- Dynamic role allocation & gradation of autonomy
- Shared attention & common ground
- Trust-building, collaborative embodied interaction

present → **3 years** → **6 years**

Cognitive design assistants

IBM-SoftBank

IBM-Hilton

- Self-explaining abilities
- Meaningful dialogues and interactions
- Verbal and non-verbal interactions
- New roles
  - Design Analyst
  - Design Critic
- Highlighted by NSF ESD PD as a new promising area of research
  - Still declined my CAREER!

Summary and Concluding remarks

- Design of complex systems is a hard problem
- Emphasis in the past was design automation
  - I still do a lot of that!
- More collaborative approaches are promising
- (Part of) the future of design is cognitive design assistants!

Current and future work

- **KDO/AOS**: Incorporating chunks of knowledge through different implementations of soft constraints and repair operators
- **iFEED**: Driving feature generalization – incorporating intermediate design attributes. Which visualizations work better for experts/novices?
- **Daphne** (design assistant for Earth observing sats)
  - Natural Language Processing Layer
  - Design Analyst Role
  - Design Critic Role
  - Embodied interaction

References

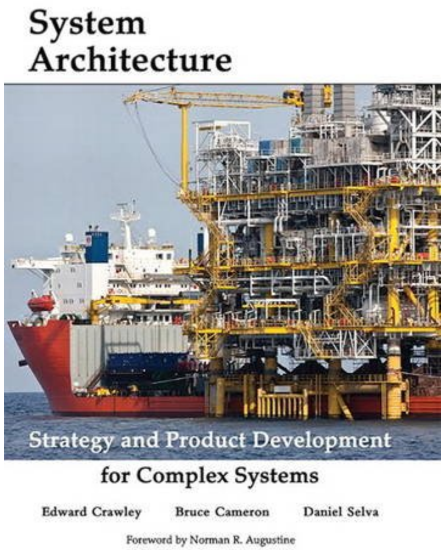
- Crawley, E., Cameron, B. & Selva, D., 2015. *Systems Architecture: Strategy and Product Development for Complex Systems*, Prentice Hall.
- Das, S.J., Selva, D. & Golkar, A., 2015. An Intelligent Spacecraft Configuration Tool for Mission Architecture Space Exploration. In *AIAA SPACE 2015 Conference and Exposition*.
- Dwyer, M. & Selva, D., 2013. The impact of technical complexity on the decision to collaborate and combine. In *2013 IEEE Aerospace Conference*.
- Hitomi, N. & Selva, D., 2016. A hyper-heuristic approach to leveraging domain knowledge in multi-objective evolutionary algorithms. In *Proceedings of the ASME 2016 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference IDETC/CIE 2016*.
- Hitomi, N. & Selva, D., 2015a. Experiments with Human Integration in Asynchronous and Sequential Multi-Agent Frameworks for Architecture Optimization. *Procedia Computer Science*.
- Hitomi, N. & Selva, D., 2015b. The effect of credit definition and aggregation strategies on multi-objective hyper-heuristics. In *Proceedings of the ASME 2015 International Design Engineering Technical Conferences*. Boston, MA.
- Jacobs, M. & Selva, D., 2015. A CubeSat Catalog Design Tool for a Multi-Agent Architecture Development Framework. In *Aerospace Conference, 2015 IEEE*.
- Knerr, N. & Selva, D., Cityplot: visualization of high-dimensional design spaces with multiple criteria. *Journal of Mechanical Design*.
- Krueger, J. et al., 2009. Spacecraft and Constellation Design for a Continuous Responsive Imaging System in Space. In *AIAA SPACE 2009 Conference & Exposition*.
- Net, M.S., Selva, D. & Golkar, A., 2014. Exploring classification algorithms for early mission formulation cost estimation. *2014 IEEE Aerospace Conference*, pp.1–14.

References

- Sanchez, M. et al., 2014. Results of the MIT Space Communication and Navigation Architecture Study. *2014 IEEE Aerospace Conference, Big Sky, Montana*.
- Sanchez-Net, M. et al., 2015. Integrated Tradespace Analysis of Space Network Architectures. *Journal of Aerospace Information Systems*, 12(8), pp.564–578.
- Selva, D., 2014b. Experiments in Knowledge-intensive System Architecture: Interactive Architecture Optimization. In *2014 IEEE Aerospace Conference*. Big Sky, Montana: IEEE.
- Selva, D., 2014. Knowledge-intensive global optimization of Earth observing system architectures: a climate-centric case study. In *Proceedings of SPIE 9241, Sensors, Systems, and Next-Generation Satellites XVIII, 92411S*.
- Selva, D., Abello, C. & Hitomi, N., 2015. Preliminary Experiments with Learning Agents in an Interactive Multi-agent Systems Architecture Tradespace Exploration Tool. In *IEEE Systems Conference SysCon 2015*.
- Selva, D., Cameron, B. & Crawley, E.F., 2014. A rule-based method for scalable and traceable evaluation of system architectures. *Research in Engineering Design*, 25(4), pp.325–349.
- Selva, D., Cameron, B.G. & Crawley, E.F., 2014a. Rule-based System Architecting of Earth Observing Systems: The Earth Science Decadal Survey. *Journal of Spacecraft and Rockets*.
- Selva, D. & Crawley, E., 2010. Integrated assessment of packaging architectures in Earth observing programs. In *Aerospace Conference, 2010 IEEE*. Big Sky: IEEE, pp. 3–12.
- Selva, D. & Crawley, E., 2013. VASSAR: Value Assessment of System Architectures using Rules. In *Aerospace Conference, 2013 IEEE*. Big Sky: IEEE.
- Selva, D. & Krejci, D., 2012. A Survey and Assessment of the Capabilities of Cubesats for Earth Observation. *Acta Astronautica*, 74, pp.50–68.
- Selva, D. & Krejci, D., 2013a. Preliminary Assessment of Performance and Cost of a Cubesat Component of the Earth Science Decadal Survey. In *27th Annual AIAA/USU Conference on Small Satellites*. Logan, Utah.

Thank you!

Questions?



Cityplot: visualizing multi-objective design spaces

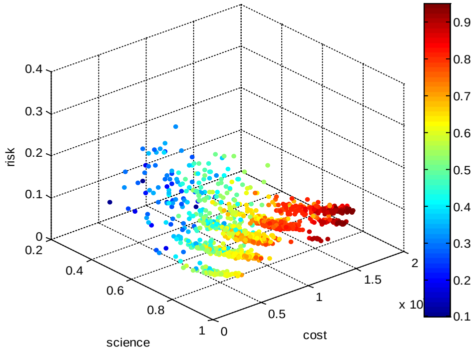
Goal: Visualize design "landscape"

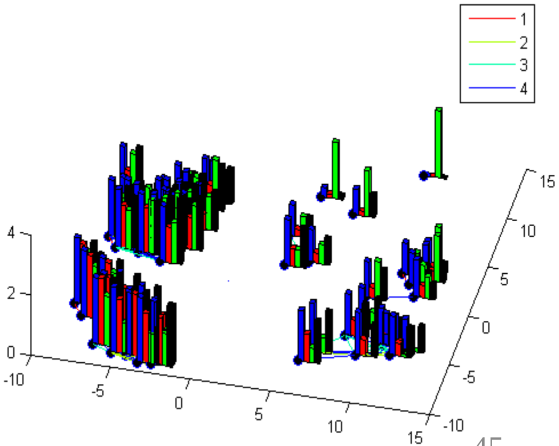
Handle large spaces

Discrete and continuous variables.

Multidimensional scaling to reduce design space to 2D

Bar plot to show normalized objectives





Visualization for knowledge discovery: Challenges

- Visual analytics has become a popular approach to knowledge discovery in design
- Histograms, parallel coordinates, glyphs and scatter plots are often used
- Challenges:
  - Linking design space information with objective space information
  - Scaling to **high-dimensional spaces**
  - Handling **discrete variables**

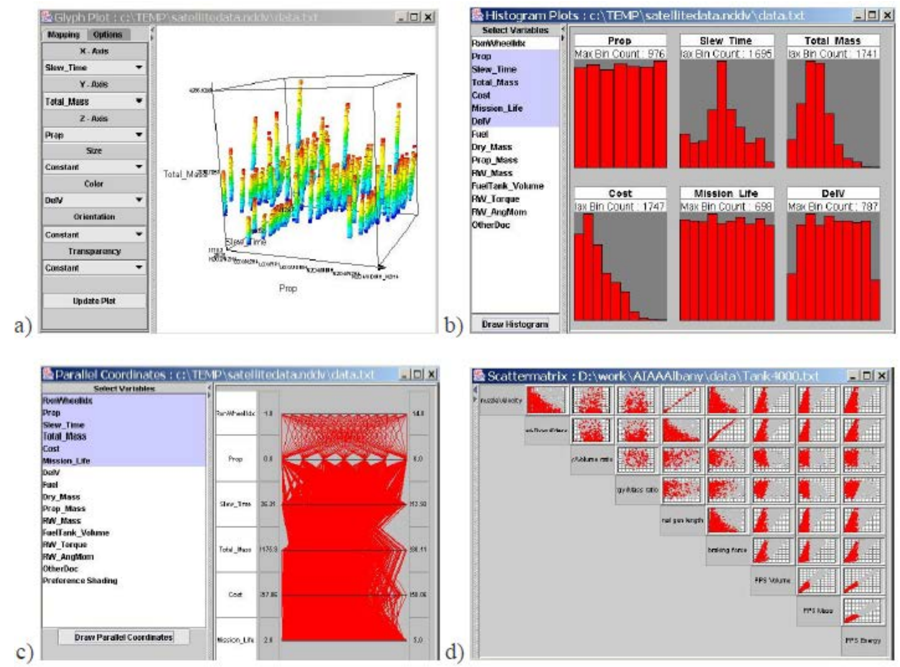
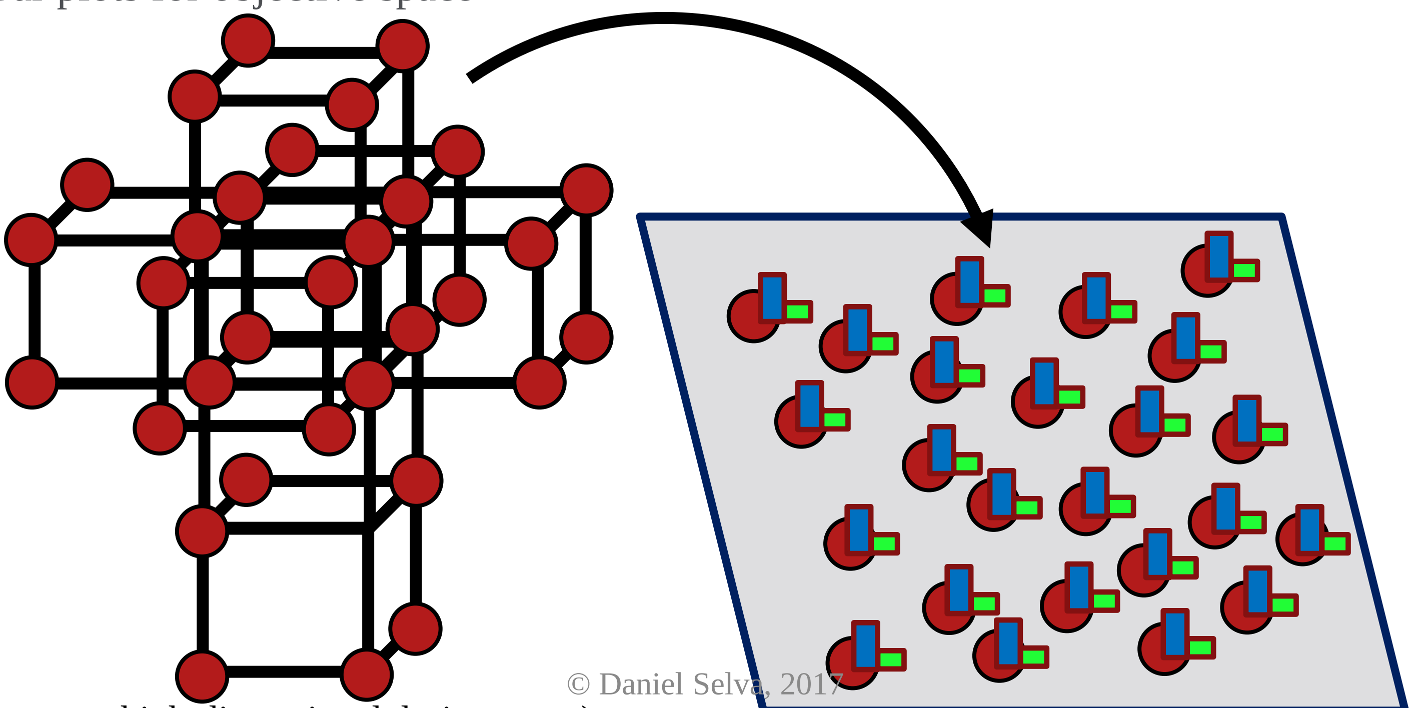


Figure 1. a) Glyph Plot, b)Histogram Plots, c) Parallel Coordinates, d) Scatter

ARL Tradespace Visualizer by (Stump et al. 2004)

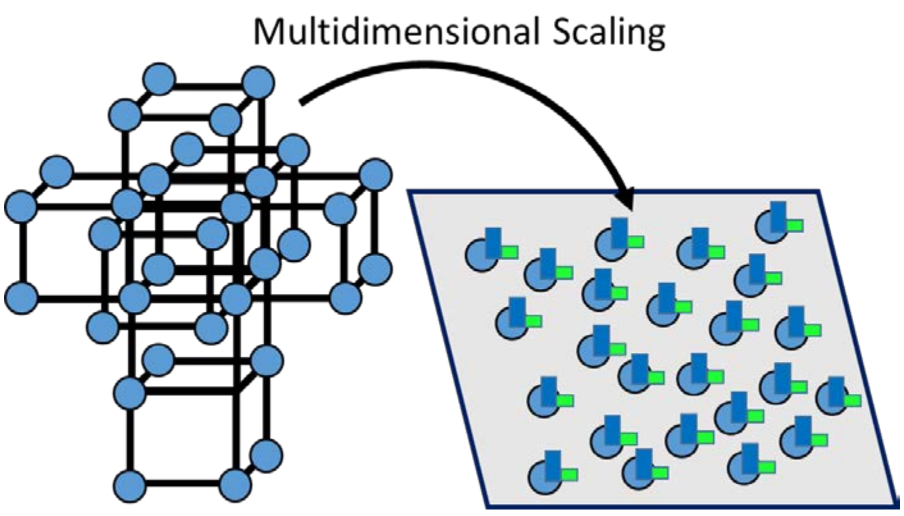
Visualizing the design landscape

- If there were only 1 metric z and 2 decisions x,y we could easily view what the design landscape looks like
- How can we do the same for higher dimension spaces and discrete variables?
  - Multidimensional scaling to project design space to 2D
  - Bar plots for objective space



(represents high-dimensional design space)

Multidimensional scaling



Look at all pairs of designs as drawn in the plane and in the hypercube

$$\underset{z_i}{\operatorname{argmin}} \sum_{i<j} \left| \left|\vec{z_i} - \vec{z_j}\right|_2 - d\left(\vec{x_i} - \vec{x_j}\right) \right|^2$$

Locations in the Plane

Distance is L_2 in the plane

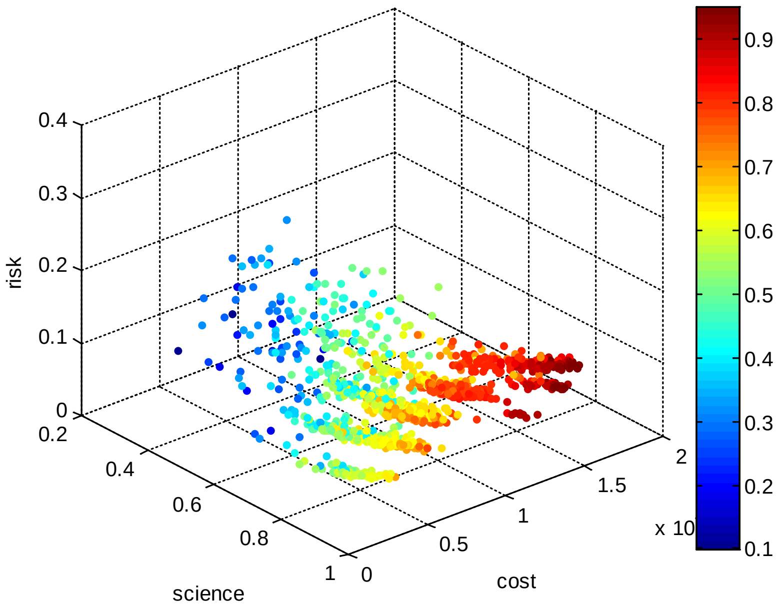
Locations on the Hypercube

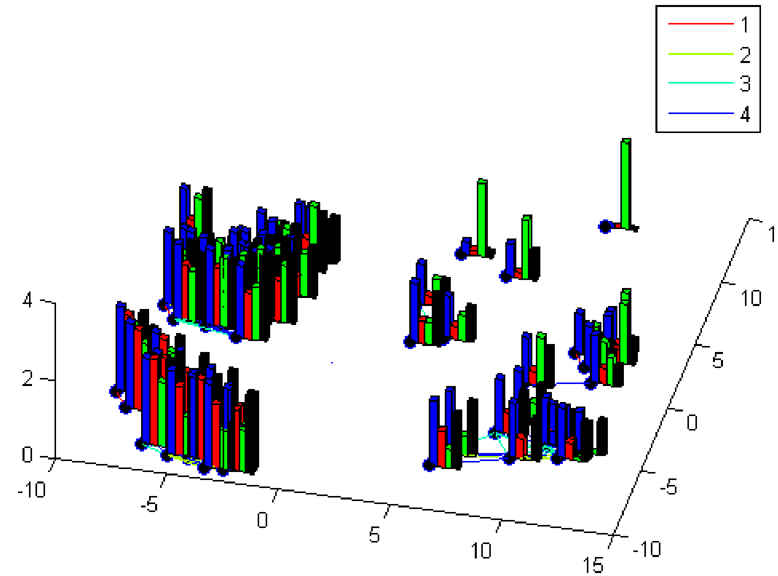
Use the distance for specific designs

Edit distance in earlier example

Make the difference in the distances as close as possible on the average

Example Cityplot: Decadal Survey problem





“Cities” are designs

“Roads” indicate distances between designs in decision space

“skyscrapers” are normalized objective function values

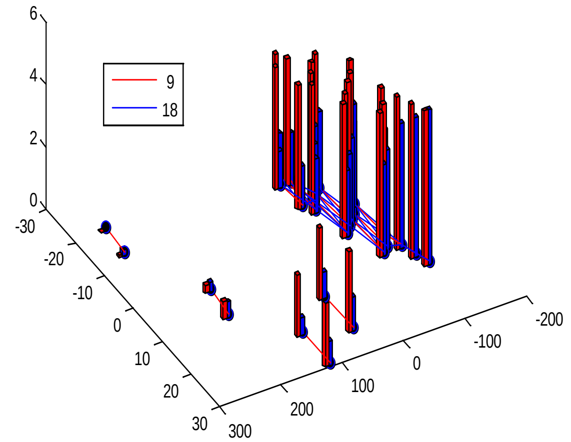
Cities are placed in the image to reflect the distances in the design space

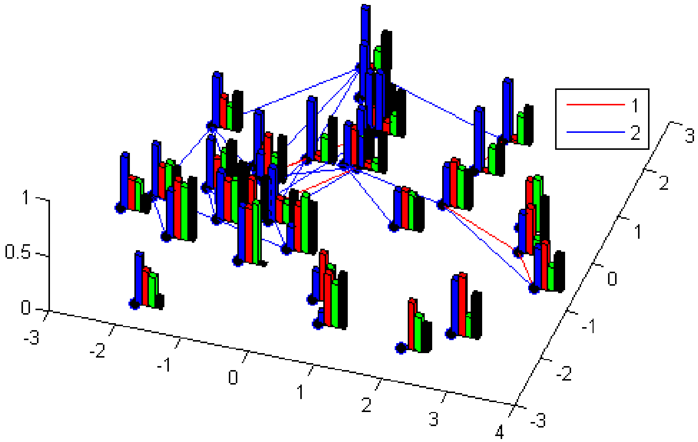
Clustering indicates “families”

Can see “smoothness” of design space

Challenges of cityplot and future work

- Challenges:
  - How do we define a distance function in the design space?
  - How do we choose a set to visualize?
    - Entire space may not be computationally feasible
    - Even when feasible, too large set may lead to messy plot
    - If rely on sampling, results may strongly depend on sampling
- Future work
  - Automatic family characterization
  - Add auto zoom-in feature
  - Study distance functions in design
  - Study effect of sampling





Knerr, N. and Selva, D., “Cityplot: Visualization of multi-Objective Design Spaces”. To be submitted to Journal of Mechanical Design.