Better risk trade-off discussions via automatic argument reduction tools

Tim Menzies
West Virginia University
tim@menzies.us

James Kiper,
Miami University,
kiperjd@muohio.edu

Martin S. Feather,
Jet Propulsion Laboratory,
California Institute of Technology,
martin.s.feather@jpl.nasa.gov

Abstract

When debating about complex systems with a large number of options, humans can often be slower than an AI system at identifying the clusters of key decisions *that are of most benefit. By focusing a group on these key decision clusters, more time can be devoted to key decisions and less time is wasted on irrelevancies. Our proposed new tool will be based on:*

- *JPL's DDP group decision support tool [3]*
- *WVU's contrast set learners [7])*
- *Miami University's cluster visualization tools*

and tested on case studies at JPL or other NASA applications.

1 Introduction

2 Imagine the Scene

A team of NASA's top experts are debating options on some complex deep space mission. The mission is in its early planning stages so few of the details are fixed. The science team wants to add a new instrument package to the mission. But the propulsion experts are already worried about the payload mass and any addition worries them. Also, the electronics team members are worried about the added stress on the on-board systems. A spirited discussion follows in which each team tries to explain the costs and benefits of their various proposals.

In the midst of this heated debated, a screen flickers. The AI system monitoring the debate has just realized that of the dozens issues currently being debated, a resolution on (e.g.) four matters makes debates about most of the other issues redundant. The team has consensus on two of those matters decisions, so the team quickly adopts them. These two decisions greatly reduce the space of remaining discussions and the group finishes their debates in time for lunch.

3 The DDP Tool

The above scene is not science fiction- some of the technology has already been developed and applied to JPL missions. At JPL, the DDP tool [2] is in use to organize interactive knowledge acquisition and decision making sessions with spacecraft experts . The DDP tool and process works as follows:

- 6 to 20 experts are gathered together for short, intensive knowledge acquisition sessions (typically, 3 to 4 half-day sessions). These sessions *must* be short since

[0]SEDECS2003: the 2nd International Workshop on Software Engineering Decision Support (part of SEKE2003); June 20 2003 http://www.ksi.edu/seke/seke03.html.Date April 15, 2003. WP ref: 03/sekew/star1. URL:http://tim.menzies.us/pdf/03star1.pdf

[0]This work was sponsored by the NASA Office of Safety and Mission Assurance under the Software Assurance Research Program led by the NASA IV&V Facility and conducted at the University of Miami, West Virginia University (partially supported by NASA contract NCC2-0979/NCC5-685) and at the Jet Propulsion Laboratory, California Institute of Technology (under a contract with the National Aeronautics and Space Administration). The JPL work was funded by NASA's Office of Safety and Mission Assurance, Center Initiative UPN 323-08. That activity is managed locally at JPL through the Assurance and Technology Program Office. The second author's research is also supported in part by National Science Foundation Grants CCR-0204139 and CCR-0205588. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

DDP assertions are either:

- *Requirements* (free text) describing the objectives and constraints of the mission and its development process;
- *Weights* (numbers) associated with requirements, reflecting their relative importance;
- *Risks* (free text) describing events that can damage requirements;
- *Mitigations*: (free text) describing actions that can reduce risks;
- *Costs*: (numbers) effort associated with mitigations, and repair costs for correcting Risks detected by Mitigations;
- *Mappings*: directed edges between requirements, mitigations, and risks.
- *Part-of relations* structure the collections of requirements, risks and mitigations;

Figure 1. DDP's ontology

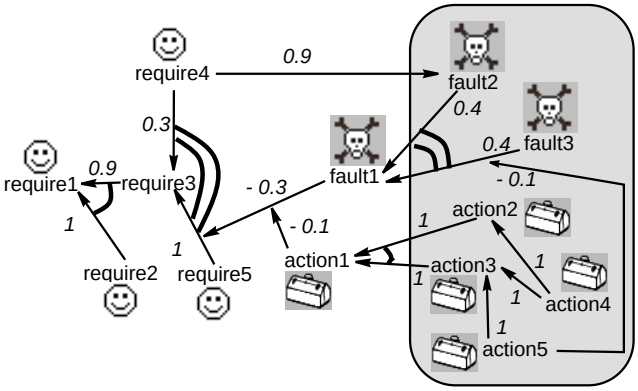
it is hard to gather together these experts for more than a very short period of time.

- The DDP tool supports a graphical interface for the rapid entry of the assertions. Such rapid entry is essential, lest using the tool slows up the debate.
- Assertions from the experts are expressed in using an ultra-lightweight decision ontology (e.g. see Figure 1). The ontology *must* be ultra-lightweight since:
  - Only brief assertions can be collected in short knowledge acquisition sessions.
  - If the assertions get more elaborate, then experts may be unable to understand technical arguments from outside their own field of expertise.

The result of these sessions is a network of influences connecting project requirements to risks to possible mitigations. A (highly) stylized version of that network is shown in Figure 2.

The ontology of Figure 1 may appear too weak for useful reasoning. However, in repeated sessions with DDP, it has been seen that the ontology is rich enough to structure and simplify debates between NASA experts. For example, DDP has been applied to over a dozen applications to study advanced technologies such as

- a computer memory device;



Faces denote requirements;

Toolboxes denote mitigations;

Skulls denote risks;

Conjunctions are marked with one arc; e.g. *require1* if *require2* and *require3*.

Disjunctions are marked with two arcs; e.g. *fault1* if *fault2* or *fault3*.

Numbers denote weights; e.g. *action5* reduces the contribution of *fault3* to *fault1*, *fault1* reduces the impact of *require5*, and *action1* reduces the negative impact of *fault1*.

Oval denotes structures that are expressible in the latest version of the JPL semantic net editor (under construction).

Figure 2. A semantic net of the type used at JPL [4].

- gyroscope design;
- software code generation;
- a low temperature experiment's apparatus;
- an imaging device;
- circuit board like fabrication;
- micro electromechanical devices;
- a sun sensor;
- a motor controller;
- photonics; and
- interferometry.

In those studies, DDP sessions has found cost savings exceeding $1 million in at least two of these studies, and lesser amounts (exceeding $100,000) in the other studies. The DDP meetings have also generated numerous design improvements such as savings power or mass and shifting of risks from uncertain architectural to better understood design. Further, at these meetings, some non-obvious sig-

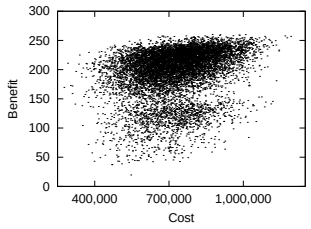


Figure 3.A: Before. Here, one dot is one project plan; i.e. one possible setting to the 99 risk mitigation options.

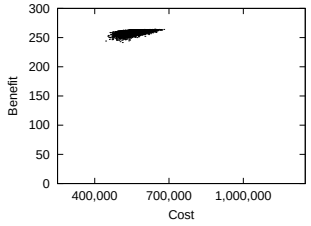


Figure 3.B: After. Results from applying the constraints learnt by the TAR2 contrast set learner.

Figure 3. An application of TAR2. X-axis= "cost" = sum of the cost of selected risk mitigations (lower is better). Y-axis= "benefit"= requirements coverage, less the effects of risk (more is better)

nificant risks have been identified and mitigated. Lastly, DDP can be used to document resolutions to those debates. Hence, DDP is in use at JPL:

- not only as a group decision support tool (as it was designed to do);
- but also a design rationale tool to document decisions.

That is not to say DDP cannot be improved.

4 Improving DDP with the TAR2 Tool

The current version is a *manual* tool. Users sketch out mappings between requirements, risks, and mitigations then search for the cheapest mitigations that most reduce risks. This search can be overwhelming large. For example, one deep space mission analyzed using DDP has 99 possible mitigations; i.e. $2^{99} \approx 10^{30}$ possibilities. This space is too large to explore thoroughly. Figure 3.A shows the results of 50,000 runs with DDP. In each run, a random set of mitigations were selected each time. Note the huge range of possible costs and benefits.

The range of possibilities shown in Figure 3.A seems dauntingly large. However, our TAR2 *contrast set learner* has shown that a heated discussion on most of the risk mitigations would be a *complete waste of time*. A contrast set learner finds the differences in variable settings seen in different situations. For example, an analyst could ask a contrast set leaner "what are the differences between people with Ph.D. and bachelor degrees?". TAR2 differs from other contrast set learners such as TARZAN [8] and STUCCO [1] in that it searches for the *smallest* contrast set that *most* separates preferred and undesired behavior.

TAR2 divided Figure 3.A into "preferred" and "undesired" regions (here, "preferred" means lower costs and higher benefits and "undesired" means not preferred). With knowledge of that division, TAR2 learnt a set of constraints that select for the preferred outcomes while avoiding the undesired regions. The goal of TAR2 is to *improve the mean* and *reduce the variance* in the behavior of a system.

Figure 3.A shows 50,000 runs with DDP using mitigations compatible with the constraints learnt by TAR2. Comparing Figure 3.A withFigure 3.B, we see that the variance in behavior has indeed been greatly *reduced* while *decreasing* mean costs and *increasing* mean benefits.

TAR2 generated Figure 3.B using only a small subset of the available risks mitigations. TAR2 made recommendations on only $\frac{1}{3}rd$ of the 99 mitigations available in this DDP models.

Further details on this use of TAR2 on a DDP dataset are found in reference [5].

5 Drawbacks with TAR2

Figure 3.B shows that it is possible to use DDP models to optimize risk mitigation actions for complex systems, using only a *small subset* of the available options. However, in two aspects, the TAR2 experiment was a failure:

- *The runtime problems:* TAR2 is too slow. The DDP model had to be executed 50,000 times to learn the constraints that generated Figure 2b. This runtime is too long to support interactive argument support. Worse still, bigger DDP models would take even longer to execute. Clearly, a faster method is required.
- *The hiding problem*. TAR2's output can hide important details. Recall from Figure 3.B that there exists a cluster of results that are the best TAR2 can find. While any point in those clusters are the best TAR2 can offer, adjacent points in the cluster may represent very different mitigations, some of which are more acceptable to the users than others.

TAR2 ran slow since it sampled a large run where mitigations were selected randomly. Perhaps some other search

might be more appropriate? In the sequel, we will discuss the merits of TAR2's search vs *simulated annealing*.

As to the *hiding problem*, we believe it is best addressed as a *quantitative value* problem. A limitation of the DDP ontology is that it asks a set of experts to agree upon some numeric quantity (using a number between 0.0 and 1.0) to rate various relationships: e.g., the impact of risks on objects, the effect of mitigations on risk, etc. In our experience, these experts have been able to do this. However, it is clear that the resulting model is less robust than the numeric values may suggest.

It is our hypothesis that such experts may be more comfortable agreeing upon a probability distribution that represents the impact of a risk on an objective or the effect of a mitigation on a risk. That is, they would be given a small number of possible distributions *having previously been informed about the characteristics of each* and asked to pick among them. They would also have to pick the appropriate parameters for each – e.g. mean and standard deviation for a normal distribution.

The thought is that, although this is more information for the experts to agree upon, it might get agreement faster since they would be recognizing that there is some embedded uncertainty in these values. agreeing upon a probability distribution that represents the impact of a risk on an objective or the effect of a mitigation on a risk. That is, they would be given a small number of possible distributions (having previously been informed about the characteristics of each) and asked to pick among them. They would also have to pick the appropriate parameters for each – e.g. mean and standard deviation for a normal distribution. Our thinking is that, although this is more information for the experts to agree upon, it might get agreement faster since they would be recognizing that there is some embedded uncertainty in these values.

6 Is Simulated Annealing Better Than TAR2?

Optimizing risk mitigations means *minimizing* costs while *maximizing* benefits. That is, it is a classic *optimization problem*. A commonly-used search technique for such optimization is *simulated annealing* [6], illustrated in Figure 4. Simulated annealing is a kind of hill-climbing search for finding a good solution. A simple hill-climber simply jumps to the next best solution and can hence miss globally optimal solutions since it can't move to a near-by higher peak if, to do so, means travelling down-hill across a valley. Simulated annealing avoids this problem using a "jump" factor that is a function of a "temperature" variable. At high "temperatures", simulated annealing can sample more of the local terrain since it can jump up-hill *or* down-hill. As the search proceeds and the "temperature" cools, simulated annealing jumps less and less. Eventually, the jumping mechanism "freezes" and simulated annealing completes its search like a simple hill climber. A simulated annealing capability is now part of the DDP tool [9].

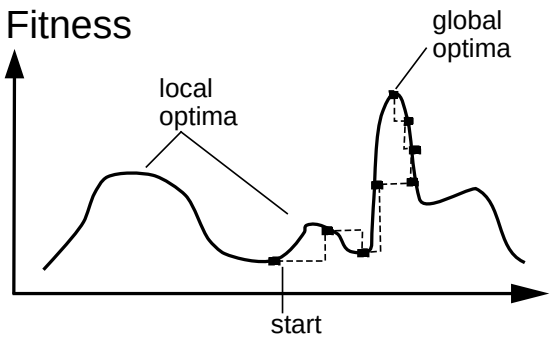


Figure 4. Simulated annealing, an example.

Figure 5 compares TAR2 and simulated annealing. At each round X (shown on the x-axis), simulated annealing or TAR2 was used to extract key decisions from a log of runs of a DDP model. A new log is generated, with the inputs constrained to the key decisions found between round zero and round X. Further rounds of learning continue until the observed changes on costs and benefits stabilizes.

It is insightful to compare the results from TAR2 and simulated annealing:

- As seen in Figure 5, simulated annealing and TAR2 terminate in (nearly) the same cost-benefit zone.
- Simulated annealing did so using only 40% of the data needed by TAR2; i.e. while TAR2 needed 50,000 runs of DDP, the simulated annealing method needed only 20,000.
- The bad news is that, while TAR2 proposed constraints on 33% of the mitigations, simulated annealing proposed actions on 100% of the mitigations. Such a result is consistent with the nature of simulated annealing- this search is a global search through all options. Hence, it tends to propose solutions to a large part of the model.

In summary, the directed search of simulated annealing needs less data than TAR2, but in doing so, we lose the main advantage of TAR2; i.e. no drastic reduction in the space of options.

7 STAR1= simulated annealing + TAR2

In summary, the directed search of SA needs less data than TAR2, but in doing so, we lose the main advantage

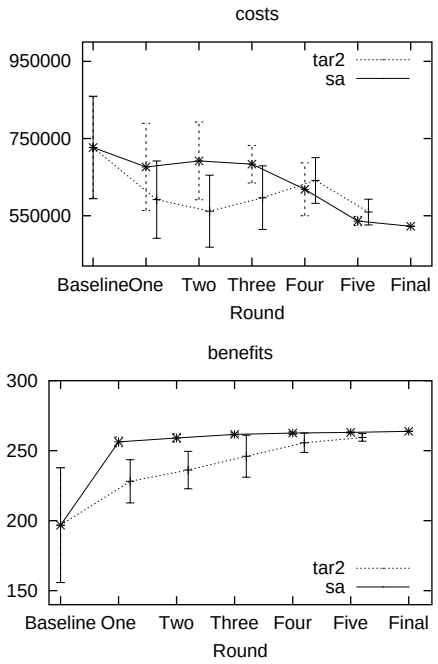


Figure 5. Comparison of TAR2 and simulated annealing.

of TAR2; i.e. no drastic reduction in the space of options. Perhaps we can get the best of both approaches.

Our research is exploring combining the advantages of TAR2 (the selection of a small number of critical decisions) with SA (faster, directed search and an exploration of a larger space of possibilities). The "jumps" in simulated annealing are generated by mutating the best solution seen so far. In traditional SA, these mutations are selected at random. In our proposed approach, we would run a contrast set learner in parallel with the SA to build up a probability profile on settings that were most associated with worse solutions. The mutation sub-routine of the SA would then be modified to avoid mutations that include settings from the worst solutions.

Our analogy for this process is that of a rocket flying down towards some preferred solution. SA is the *gravity* that pulls the rocket down faster while the contrast set learning is the *booster* than thrusts the rocket away from undesired situations.

Specifically, our goals are:

- Implement STAR1, a combination of SA (or other AI search algorithms) and TAR2, and integrate the result with DDP
- Tune the STAR1 such that it such that it terminates in < 10 seconds (i.e. in time to interact with some active debate on some part of a DDP model).
- Augment this integrated tool (STAR1) with a decision clustering tool
- Improve the modeling of risk in DDP through probability distributions
- Test this supplemented version of DDP during live debates on system options by JPL analysts.

References

[1] S. Bay and M. Pazzani. Detecting change in categorical data: Mining contrast sets. In *Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining*, 1999. Available from `http://www.ics.uci.edu/~pazzani/Publications/stucco.pdf`.

[2] S. Cornford, M. Feather, and K. Hicks. Ddp a tool for life-cycle risk management. In *IEEE Aerospace Conference, Big Sky, Montana*, pages 441–451, March 2001.

[3] M. Feather and S. Cornford. Quantitative risk-based requirements reasoning. *Requirements Engineering Journal*, 2003. Available from `http://eis.jpl.nasa.gov/~mfeather/PublicationsAvailable/`.

[4] M. Feather, H. In, J. Kiper, J. Kurtz, and T. Menzies. First contract: Better, earlier decisions for software projects. In *ECE UBC tech report*, 2001. Available from `http://menzies.us/pdf/01first.pdf`.

[5] M. Feather and T. Menzies. Converging on the optimal attainment of requirements. In *IEEE Joint Conference On Requirements Engineering ICRE'02 and RE'02, 9-13th September, University of Essen, Germany*, 2002. Available from `http://menzies.us/pdf/02re02.pdf`.

[6] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science, Number 4598, 13 May 1983*, 220, 4598:671–680, 1983.

[7] T. Menzies, E. Chiang, M. Feather, Y. Hu, and J. Kiper. Condensing uncertainty via incremental treatment learning. In *Annals of Software Engineering*, 2002. Available from `http://menzies.us/pdf/02itar2.pdf`.

[8] T. Menzies and E. Sinsel. Practical large scale what-if queries: Case studies with software risk assessment. In *Proceedings ASE 2000*, 2000. Available from `http://menzies.us/pdf/00ase.pdf`.

[9] J. D. . M. F. S. Cornford. Optimizing the design of end-to-end spacecraft systems using risk as a currency. In *IEEE Aerospace Conference, Big Sky Montana*, pages 9–16, March 2002.