Determining Best Weighting Factors for Combining Several Utility Functions

Kerry M. Soileau

November 7, 2026

Abstract

Without careful monitoring of compliance with budget, schedule and design, a big program can get into serious trouble. We describe here a tool to address program risks. This tool, called the "utility function," quantifies the current "health" of a program. The utility function has been defined as a linear combination of the cost, schedule and technical utilities. This article presents a general technique for determining weights for utility function combination. This technique is computationally simple, determines the possible choices of weights which are harmonious with management judgment, and provides insight into the sensitivity of this harmony to small changes in weights. Although the technique was developed for application to cost, schedule and technical utility functions, the technique is a general one and could be used to determine weights for combining any finite set of subordinate functions into a single value.

Keywords. Program management, utility, weights.

1 Introduction

In a perfect world, a large project could be designed, scheduled, and budgeted, and then work would proceed without budget overruns or late milestone completion, and the resulting product would meet or exceed all

requirements and specifications. Risk management is an important function because the world is not perfect.

The term *risk* is used here to denote any obstacle to the on-budget, on-schedule completion of a product. To manage risks well we require early detection so we can take action to reduce their risk impact. The impact of a risk consists of the consequences of the risk together with the likelihood the risk will occur.

We describe here a tool to address program risks. This tool, called the "utility function," quantifies the current "health" of a program. The function combines into a single number the subordinate utility functions for cost, schedule and technical aspects of the program. Each subordinate function is defined so that it returns a value between 0 and 1. The 0 value defines the most undesirable status imaginable, while a value of 1 indicates the most desirable status. The value 0.5 indicates a nominal status. Specifically, the calculation of program utility is performed as follows:

$$P = \omega_c C + \omega_s S + \omega_t T, \tag{1.1}$$

where P is program utility, C is cost utility, S is schedule utility, T is technical utility, ω_c is cost weight, ω_s is schedule weight, and ω_t is technical weight. Weights are assumed to fall within the interval $[0, 1]$.

This leads to the question, What is the best (or if not the best, a good) way to linearly combine the program cost, schedule and technical utility

functions into a single measure of overall program utility? To put it another way, how should we choose the weights indicated by the equation?

2 Selection Criteria

We stipulate that the combination function must have the following properties:

If cost utility increases, and schedule and technical utilities do not decrease, then the combination function should increase.
If schedule utility increases, and cost and technical utilities do not decrease, then the combination function should increase.
If technical utility increases, and cost and schedule utilities do not decrease, then the combination function should increase.
If cost utility decreases, and schedule and technical utilities do not increase, then the combination function should decrease.
If schedule utility decreases, and cost and technical utilities do not increase, then the combination function should decrease.
If technical utility decreases, and cost and schedule utilities do not increase, then the combination function should decrease.

These requirements led immediately to the following constraints:

$$\omega_c \geqslant 0 \tag{2.1}$$

$$\omega_s \geqslant 0 \tag{2.2}$$

$$\omega_t \geqslant 0 \tag{2.3}$$

Without loss of generality we imposed the following constraint:

$$\omega_c + \omega_s + \omega_t = 1. \tag{2.4}$$

Weights.png
Triangle: $\Lambda \approx 1.51632$

3 "Good" and "Bad" Weight Sets

We next considered by which criteria could one distinguish a "good" choice of weights from a "bad" one. To begin with, a utility function with a good set of weights would rank several scenarios into the same order as would a manager. Let $\{S_1, S_2, S_3, \cdots, S_N\}$ be a set of ISS Program scenarios, ranked by Program management according to desirability so that S_1 is least desirable and S_N is most desirable.

Now let (C_i, S_i, T_i) be the point determined by the cost, schedule and technical utilities of the ith scenario. Each choice of weights determines a ranking of the scenarios independent of the Program management ranking. In the following we will refer to a choice $(\omega_c, \omega_s, \omega_t)$ as a *weight-point*. A scenario is considered more desirable than another if its utility, computed

using the weights indicated by the weight-point, is greater than the other. The ranking is then done by arranging the scenarios in order of increasing desirability. We say weights are *good* if the consequent ranking of the scenarios is identical to that of Program management; otherwise we say the weights are *bad*.

For instance, suppose we have the following scenarios:

		Utilities		Overall	Utility
Scenario	Cost	Schedule	Technical	(.35,.26,.39)	(.41,.56,.03)
1	.17	.32	.98	.52	.28
2	.12	.45	.28	.27	.31
3	.98	.13	.45	.55	.49
				2	1
	Bad Weight		Rankings	1	2
	Good Weight			3	3

Table 1

The weights indicated as bad are bad because they resulted in a utility ranking of the scenarios different than 1, 2, 3. The bad weights computed scenario 2 as worst, scenario 1 as middling, and scenario 3 as best. The weights indicated as good are good because their ranking of the scenarios agreed with Program management's, namely 1, 2, 3.

It should be noted that for a given situation there are usually infinitely many good choices. As we will see, if plotted on a three-dimensional coordinate system, the set of good weight-points would form a convex polyhedral region.

4 Good Weight Set Polyhedra

The point $(\omega_c, \omega_s, \omega_t)$ describes a good weight set if and only if it satisfies the following five conditions:

Table 2

$\omega_c C_i + \omega_s S_i + \omega_t T_i \leqslant \omega_c C_{i+1} + \omega_s S_{i+1} + \omega_t T_{i+1}$ for $i = 1, 2, \cdots, N-1$	(1)
$0 \leqslant \omega_c$	(2)
$0 \leqslant \omega_s$	(3)
$0 \leqslant \omega_t$	(4)
$\omega_c + \omega_s + \omega_t = 1$	(5)

Table 3

Equation (1) implies

$$0 \leqslant \omega_c \left(C_{i+1} - C_i\right) + \omega_s \left(S_{i+1} - S_i\right) + \omega_t \left(T_{i+1} - T_i\right)$$

Equation (5) yields

$$\omega_t = 1 - \omega_c - \omega_s$$

hence

$$0 \leqslant \omega_c \left(C_{i+1} - C_i\right) + \omega_s \left(S_{i+1} - S_i\right) + \left(1 - \omega_c - \omega_s\right) \left(T_{i+1} - T_i\right)$$

which on simplifying produces

$$\boxed{0 \leqslant \omega_c \left(C_{i+1} - C_i - T_{i+1} + T_i\right) + \omega_s \left(S_{i+1} - S_i - T_{i+1} + T_i\right) + T_{i+1} - T_i}$$

whence

$$\left\{ \begin{array}{ll} \frac{\omega_c(-C_{i+1}+C_i+T_{i+1}-T_i)-T_{i+1}+T_i}{(S_{i+1}-S_i-T_{i+1}+T_i)} \leqslant \omega_s & \text{if } S_{i+1} - S_i - T_{i+1} + T_i > 0 \\ \frac{\omega_c(-C_{i+1}+C_i+T_{i+1}-T_i)-T_{i+1}+T_i}{(S_{i+1}-S_i-T_{i+1}+T_i)} \geqslant \omega_s & \text{if } S_{i+1} - S_i - T_{i+1} + T_i < 0 \end{array} \right\}$$

Table 3

The definitions

$\alpha_i = S_i - T_i, \quad \beta = C - T$ and $\gamma_i = T_i - T_{i+1}$ yield

$$\left\{ \begin{array}{ll} \frac{\beta_i - \beta_{i+1}}{\alpha_{i+1} - \alpha_i} c + \frac{\gamma_i}{\alpha_{i+1} - \alpha_i} \leqslant s & \text{if } \alpha_{i+1} > \alpha_i \\ \frac{\beta_i - \beta_{i+1}}{\alpha_{i+1} - \alpha_i} c + \frac{\gamma_i}{\alpha_{i+1} - \alpha_i} \geqslant s & \text{if } \alpha_{i+1} < \alpha_i \end{array} \right\}$$

Candidates for vertex points are the points determined by intersections of these lines with the c axis, the s axis, and the line $c + s = 1$. We take these in turn.

Intersection with c axis:

$$\frac{\beta_i - \beta_{i+1}}{\alpha_{i+1} - \alpha_i} c + \frac{\gamma_i}{\alpha_{i+1} - \alpha_i} = 0$$

$$c = \frac{\gamma_i}{\beta_{i+1} - \beta_i}$$

Candidate is the point $\left(\frac{\gamma_i}{\beta_{i+1} - \beta_i}, 0 \right)$

Intersection with s axis:

$$s = \frac{\gamma_i}{\alpha_{i+1} - \alpha_i}$$

Candidate is the point $\left(0, \frac{\gamma_i}{\alpha_{i+1} - \alpha_i} \right)$

Intersection with line $c + s = 1$:

$$\frac{\beta_i - \beta_{i+1}}{\alpha_{i+1} - \alpha_i} c + \frac{\gamma_i}{\alpha_{i+1} - \alpha_i} = 1 - c$$

$$c = \frac{\alpha_{i+1} - \alpha_i - \gamma_i}{\alpha_{i+1} - \alpha_i + \beta_i - \beta_{i+1}}$$

Candidate is the point $\left(\frac{\alpha_{i+1}-\alpha_i-\gamma_i}{\alpha_{i+1}-\alpha_i+\beta_i-\beta_{i+1}}, \frac{\beta_i-\beta_{i+1}+\gamma_i}{\alpha_{i+1}-\alpha_i+\beta_i-\beta_{i+1}}\right)$

For the data given in Section 3, we have

The variable **T** denotes the value of the technical weight for the indicated weight-point

5 General Definition

Let X be a nonempty, totally ordered set with order $\preccurlyeq$. Let m utility functions $u_i : X \to \mathbb{R}$ be given for $1 \leqslant i \leqslant m$. Suppose

$S = \{x_1, x_2, \cdots, x_n\} \subseteq X$ is such that $i \leqslant j$ implies $x_i \preccurlyeq x_j$ for $1 \leqslant i, j \leqslant n$.

We seek m weights $w_1, w_2, \cdots, w_m \in \mathbb{R}_{\geqslant 0}$ such that the total utility for each x_i, namely $\sum_{k=1}^{m} w_k u_k(x_i)$, satisfies

$$\sum_{k=1}^{m} w_k u_k(x_i) \leqslant \sum_{k=1}^{m} w_k u_k(x_j) \tag{5.1}$$

whenever $x_i \preceq x_j$.

5.1 Convexity

If $(w_1, w_2, \cdots, w_m)$ and $(w'_1, w'_2, \cdots, w'_m)$ both satisfy this inequality, then any convex combination

$(\theta w_1 + (1-\theta)w'_1, \theta w_2 + (1-\theta)w'_2, \cdots, \theta w_m + (1-\theta)w'_m)$ also satisfies it.

Accordingly, we define

$$\begin{aligned}
&\Phi\left(\{u_i\}_{i=1}^m, \{x_j\}_{j=1}^n\right) \\
&\quad \equiv \left\{\{w_k\}_{k=1}^m \subseteq \mathbb{R}; \sum_{k=1}^m w_k u_k(x_i) \leqslant \sum_{k=1}^m w_k u_k(x_j) \text{ for } 1 \leqslant i < j \leqslant n\right\} \\
&\qquad = \bigcap_{i=1}^n \bigcap_{j=i+1}^n \left\{\{w_k\}_{k=1}^m \subseteq \mathbb{R}; 0 \leqslant \sum_{k=1}^m w_k (u_k(x_j) - u_k(x_i))\right\} \\
&\qquad = \bigcap_{i=1}^n \bigcap_{j=i+1}^n \left\{\{w_k\}_{k=1}^m \subseteq \mathbb{R}; 0 \leqslant \sum_{k=1}^m w_k (u_k(x_j) - u_k(x_i))\right\} \\
&\qquad\qquad = \bigcap_{i=1}^n \bigcap_{j=i+1}^n \left\{\{w_k\}_{k=1}^m \subseteq \mathbb{R}; 0 \leqslant \vec{w} \cdot (\vec{u}(x_j) - \vec{u}(x_i))\right\} \quad (5.2)
\end{aligned}$$

This is clearly an intersection of hyperplanes, and is therefore a convex set.

6 Conclusion

We have demonstrated a general technique for determining weights for utility function combination. This technique is computationally simple, determines the possible choices of weights which are harmonious with management judgment, and provides insight into the sensitivity of this harmony to small changes in weights. Although the technique was developed for application to the ISS cost, schedule and technical utility functions, the technique is a general one and could be used to determine weights for combining any finite set of subordinate functions into a single value.