Characterization of trade-off preferences between non-functional properties

Ulrik Franke

RISE SICS – Swedish Institute of Computer Science
SE-164 29 Kista, Sweden
ulrik.franke@ri.se

Federico Ciccozzi

School of Innovation, Design and Engineering, Mälardalen University
SE-721 23 Västerås, Sweden
federico.ciccozzi@mdh.se

Abstract

Efficient design and evolution of complex software intensive systems rely on the ability to make informed decisions as early as possible in the life cycle. Such informed decisions should take both the intended functional and non-functional properties into account. Especially regarding the latter, it is both necessary to be able to predict properties and to prioritize them according to well-defined criteria. In this paper we focus on the latter problem, that is to say how to make trade-offs between non-functional properties of software intensive systems. We provide an approach based on the elicitation of utility functions from stake-holders and subsequent checks for consistency among these functions. The approach is exploitable through an easy-to-use GUI, which is also presented. Moreover, we describe the setup and the outcome of our two-fold validation based on exploratory elicitations with students and practitioners.

Keywords: Non-functional properties, Decision-making, Trade-offs, Utility functions

1. Introduction

Software is ubiquitous in our society and most companies in any applicative domain rely on IT for their operations. Digitization and automation are no

longer competitive advantages by themselves. Instead, as IT is becoming an irreplaceable asset, proper IT used as a cornerstone of operational excellence is simply essential and expected to be there. A consequence is that decision-makers in any domain face crucial decisions regarding the evolution of their IT portfolios: What should be bought off the shelf? What should be subscribed to as a service? What can be found in open-source communities? What, if anything, should be developed in-house? And, perhaps most importantly from an architectural perspective, how should all these diverse IT components fit together?

This challenge is faced by companies in essentially any domain, from the automotive company deciding on which software to put in the next generation car, to the SCADA system designer outlining the new control system for a power grid or the financial service provider rolling out a new payment system architecture. They all share two wishes: (1) to be able to select the best components throughout their architectures, and (2) to do it in the early phases, before all the details of their intended systems are actually known, in order to limit costs. Indeed, the cost of extracting defects grows as a project progresses and the products are developed – it is in fact much less expensive to correct errors in the concept or design phases, whereas that cost can grow exponentially if corrections are delayed to production and testing phases [1]. Thus, the ability to make informed decisions based on sound reasoning early on in the life cycle is pivotal.

Needless to say, though, it is very difficult to select IT components from several different alternatives, when these alternatives are still on the drawing board. One part of the problem is the estimation of the non-functional properties (hereafter simply "properties") of the future component. How secure will it be? How reliable? How maintainable? This is a classic set of topics that are very interesting in their own right. In this paper, however, we focus on another problem, which remains even when perfect property estimates are achieved: how to make enlightened *trade-offs* between non-functional properties.

To make this problem more concrete, consider the software product quality

model defined in the ISO/IEC 25010 standard [2]. According to this standard, system/software product quality consists of eight properties: (i) functional suitability, (ii) performance efficiency, (iii) compatibility, (iv) usability, (v) reliability, (vi) security, (vii) maintainability and (viii) portability. Assuming for the sake of the argument that the estimates problem is solved (which it most certainly is not), this means that each alternative software product – each option on the decision maker's table – can be characterized by an 8-dimensional vector. Also assuming that the properties can all be measured and mapped onto a scale of, say, 0–10, the problem becomes one of selecting between alternatives of the form $A = (10, 10, 2, 10, 10, 5, 8, 1)$, $B = (4, 9, 8, 10, 7, 0, 8, 9)$, $C = (7, 8, 7, 4, 7, 2, 7, 0)$.

This is a complex problem. Let us consider some of the possible trade-off choice scenarios. By simply considering an unweighted mean of all properties, A is the best. On the other hand, if portability (last) is the property to maximize, B is the best; if the sum of functional suitability (first) and compatibility (third) shall be maximized, then C is the best.

Only when there is dominance, i.e. one alternative being at least as good as the others in each dimension, and strictly better in at least one, the choice becomes trivial; unfortunately this is not the common case. Although difficult, these choices are pivotal for efficient development and good quality of the resulting product. This paper focuses on trade-offs by providing an approach based on the elicitation of utility functions from stake-holders and subsequent checks for consistency among these functions.

This paper is based on a previous conference publication [3]. While most of the theoretical contents are kept from our conference publication, for this paper we ran a set of empirical elicitations with students and practitioners, which are reported in Section 7. This represents the main original contribution of this paper. Additionally, a few conceptual clarifications and modifications have been made in the theoretical chapters, and the concluding discussion has been updated to reflect the empirical results.

The remainder of this paper is organized as follows. Section 2 briefly reviews

some related work in order to put the contribution in context. It is followed by Section 3 which introduces some key concepts, needed to understand the rest of the paper. In Section 4, we introduce the elicitation of preferences with regard to non-functional properties, and in Section 5 we discuss how to ensure the consistency of the preferences thus elicited. Section 6 illustrates the framework devised with an example. Section 7 reports the setup and outcomes of a two-fold empirical elicitation. Section 8 discusses the contribution and Section 9 concludes the paper with a substantial discussion of future work.

2. Related work

There is an abundant literature on decision-making when developing or selecting IT components and services. An early example is King and Schrems' discussion of cost-benefit analysis in developing and operating information systems [4]. From our perspective it is interesting to note that they list five important non-functional properties which have to be taken into account: accuracy, response time, security, reliability, and flexibility. Interestingly, important parts of ISO/IEC 25010 were thus known already in the late 70-ies.

One particular problem that has attracted a lot of attention is the dilemma of in-house development vs. buying commercial off-the-shelf (COTS) products. The problem of identifying appropriate software engineering metrics for evaluating COTS has been studied for a long time [5], as has the problem of setting requirements on such metrics [6]. The actual decision-making is often done using optimization approaches [7], [8], in particular when the trade-off is between two properties such as cost and reliability [9].

However, in general these problems are multi-dimensional, as explained in Section 1, and many studies indeed treat them as such. For example, one approach to solve such multi-criteria problems is to prioritize between the objectives in order to resolve inconsistencies, and then solve the resulting problem algorithmically [10]. Another widely used approach is to apply the analytic hierarchy process (AHP) to decompose the problem into sub-problems and re-

solve differences between stakeholders [11], [12]. The kind of analysis most closely related to ours is Pareto analysis, i.e. identifying alternatives that are not dominated by any other alternatives, and then selecting solutions from this so called Pareto front. For example, Neubauer and Stummer first determine Pareto-efficient alternatives and then let the user interactively explore the solution space to find the desired solution [13]. Michanan et al. apply similar analysis to the trade-off problem between power consumption and performance, using actual live performance data [14].

This paper is similar to much of the existing literature in that it takes the multi-dimensionality of the problem seriously, and in that it aims to involve the stakeholders to elicit important information to solve the problem. In particular, it can be seen as an off-shot from the Pareto analysis strand. It differs from the existing literature in that it attempts to discuss the problem of trade-offs between several non-functional properties systematically based on canonical utility functions from the microeconomic literature, allowing for complications like diminishing marginal utility in a way not captured by e.g. AHP or cumulative voting. Österlind et al. have worked in this direction previously [15], but whereas they require the user to manually enter the parameters of utility functions, a core idea in our paper is to elicit these in a user-friendly manner, so that relatively powerful utility models can be built from relatively straight-forward user input.

3. Preference and utility modeling

The preliminaries introduced in this section are standard. A good textbook dealing with these concepts is Varian [16].

Preferences over bundles of goods (or, in our case, non-functional properties of one good – a software system) are comparisons between vectors. $\mathbf{x} \succeq \mathbf{y}$ means that the decision-maker thinks that the bundle $\mathbf{x}$ is at least as good as the bundle $\mathbf{y}$. For the preference relation $\succeq$ to *order* the bundles, it needs to be complete (apply to all $\mathbf{x}$ and $\mathbf{y}$ in the alternatives set X), reflexive ($\mathbf{x} \succeq \mathbf{x}$),

and transitive ($\mathbf{x} \succeq \mathbf{y}$ & $\mathbf{y} \succeq \mathbf{z} \Rightarrow \mathbf{x} \succeq \mathbf{z}$). The strict preference $\mathbf{x} \succ \mathbf{y}$ can then be defined to mean not $\mathbf{y} \succeq \mathbf{x}$.

Any preference order that is complete, reflexive, transitive, and continuous (i.e. the preference order is preserved in the limit of a sequence of goods) can be represented by a continuous utility function, i.e. a function $u : X \to \mathbb{R}$ such that $\mathbf{x} \succ \mathbf{y}$ if and only if $u(\mathbf{x}) > u(\mathbf{y})$. Such functions are convenient to use in modeling and analysis of preferences. However, the assumptions do not always hold. For example, intransitive preferences are readily found experimentally [17].

There are several utility functions proposed in the literature. In the following, we introduce three of the most common, which are all special cases of the more general constant elasticity of substitution (CES) utility function.

One very simple utility function is the following:

$$u(\mathbf{x}) = \mathbf{a}^T \mathbf{x} \tag{1}$$

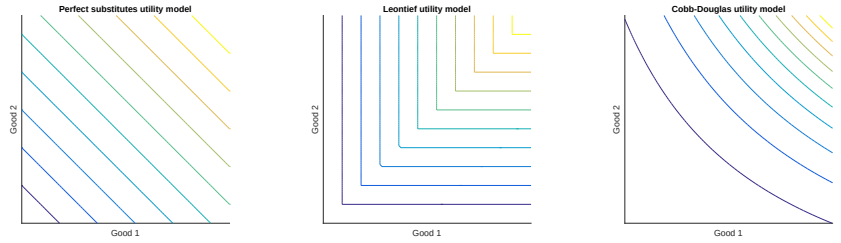
Here, the utility of the bundle $\mathbf{x}$ of the n goods $x_1, x_2, \ldots, x_n$ is just the sum of these goods, weighted by the coefficients $\mathbf{a} = a_1, a_2, \ldots, a_n$. Under such preferences, the goods are *perfect substitutes*, i.e. the decision-maker is willing to switch between goods at a *fixed ratio*, viz. is indifferent between one unit of x_1 and $\frac{a_1}{a_2}$ units of x_2. Backup tape cartridges of 6 TB from brand A and 3 TB from brand B are a good example of (nearly) perfect substitutes, with decision-makers willing to switch one A for 2 B (if they are similar with respect to e.g. failure rates).

Another very simple utility model is the following:

$$u(\mathbf{x}) = \min\{a_1 x_1, a_2 x_2, \ldots, a_n x_n\} \tag{2}$$

Here, the utility of the bundle $\mathbf{x}$ is the smallest x_i as weighted by a_i. This is called Leontief preferences and the goods are *perfect complements*. Such goods have to be consumed together, so additional units of one good without simultaneous increases in all the others are no better. For a personal computer, a decision-maker could have Leontief preferences for processor speed, cache size,

and RAM size, because overall performance will be hampered by the worst of them. If the RAM size is already the bottleneck, it makes no sense to decrease RAM size in order to gain more processor speed – no matter how much processor speed is offered. Thus with Leontief preferences, there are no trade-offs to be made.



(a) Perfect substitutes utility. (b) Leontief (perfect complements) utility (c) Cobb-Douglas utility

Figure 1: Indifference curves for different utility models. Brighter level curves represent higher utilities, i.e. points on these curves are preferred to points on the darker curves. All the combinations along a single level curve are equivalent, i.e. a decision-maker is indifferent between them.

A third simple utility model is the following:

$$u(\mathbf{x}) = x_1^{a_1} x_2^{a_2}, \ldots, x_n^{a_n} \tag{3}$$

Here, the utility of the bundle $\mathbf{x}$ is a multiplicative function of the goods, weighted by the exponents $a_1, a_2, \ldots, a_n$. This is called Cobb-Douglas utility, and represents a model where goods are neither perfect substitutes nor perfect complements, but somewhere in between.

Level curves (or indifference curves, as they are often called) for the case of two goods are shown in Fig. 1 for each of the three utility models. Higher dimensional cases are analogous.

4. Preference elicitation

4.1. Elicitation from a single indifference curve

A straightforward method for eliciting utility functions from decision-makers is to identify alternatives where they are indifferent, i.e. to find points on an indifference curve.

For convenience, assume that the trade-off is about two properties x and y. If we have a number of pairs $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ on the same indifference curve, that means by definition that they all entail the same utility $\hat{u}$.

In the linear case of perfect substitution, finding the weights a_1 and a_2 in (1) just amounts to solving the (over-determined) linear system of equations:

$$\begin{pmatrix} x_1 & y_1 \\ \vdots & \vdots \\ x_n & y_n \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \approx \begin{pmatrix} \hat{u} \\ \vdots \\ \hat{u} \end{pmatrix} \tag{4}$$

A least squares solution yields the best approximation of a_1 and a_2, and thus the best approximation of a perfect substitution utility function, based on the preferences stated.

The Cobb-Douglas case is almost as straightforward. Finding the weights a_1 and a_2 in (3) amounts to solving the (over-determined) linear system of equations found by taking logarithms of the original problem:

$$\begin{pmatrix} \ln x_1 & \ln y_1 \\ \vdots & \vdots \\ \ln x_n & \ln y_n \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \approx \begin{pmatrix} \ln \hat{u} \\ \vdots \\ \ln \hat{u} \end{pmatrix} \tag{5}$$

A least squares solution yields the best approximation of a_1 and a_2, and thus the best approximation of a Cobb-Douglas utility function, based on the preferences stated.

The Leontief case is somewhat more difficult, due to its non-linear nature. However, it can be solved by Newton's algorithm, using suitable initial guesses for a_1 and a_2 and iteratively correcting them. The objective function $\mathbf{f}$ to

minimize is simply the vector or residuals:

$$\mathbf{f} = \begin{pmatrix} r_1 \\ \vdots \\ r_n \end{pmatrix} = \min_{i=1}^{n} \left\{ a_1 \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}, a_2 \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \right\} - \begin{pmatrix} \hat{u} \\ \vdots \\ \hat{u} \end{pmatrix} \tag{6}$$

(The min operator is, of course, applied n times to yield a vector of the appropriate length – once to each pair $(a_1 x_i, a_2 y_i)$.)

Though $\mathbf{f}$ has kinks, derivatives can be found where it is piece-wise smooth. The Jacobian is the following:

$$\begin{aligned} \mathbf{J} &= \begin{pmatrix} \frac{\partial \mathbf{f}}{\partial a_1} & \frac{\partial \mathbf{f}}{\partial a_2} \end{pmatrix} = \\ &= \begin{pmatrix} \begin{cases} x_1 & \text{if } a_1 x_1 < a_2 y_1 \\ 0 & \text{otherwise} \end{cases} & \begin{cases} y_1 & \text{if } a_2 y_1 < a_1 x_1 \\ 0 & \text{otherwise} \end{cases} \\ \vdots & \vdots \\ \begin{cases} x_n & \text{if } a_1 x_n < a_2 y_n \\ 0 & \text{otherwise} \end{cases} & \begin{cases} y_n & \text{if } a_2 y_n < a_1 x_n \\ 0 & \text{otherwise} \end{cases} \end{pmatrix} \end{aligned} \tag{7}$$

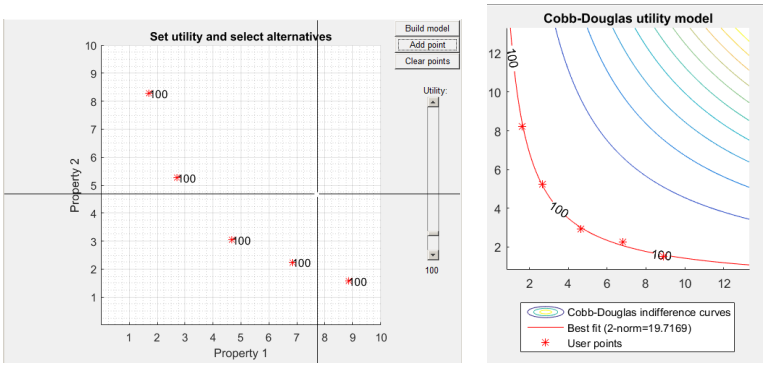
The Jacobian can now be used to find appropriate correction terms δ_1 and δ_2 to be added to a_1 and a_2 in each iteration:

$$-\mathbf{J} \begin{pmatrix} \delta_1 \\ \delta_2 \end{pmatrix} \approx \mathbf{f} \tag{8}$$

Upon reaching convergence, this solution yields a_1 and a_2, defining the best approximation of a Leontief utility function based on the preferences stated.

In practice, of course, it is difficult to know the functional form of the utility function a priori. Instead, solutions for all three alternatives can be found, and the one with the smallest residual, as measured by an appropriate norm such as the Euclidean, can be selected.[1]

[1] However, care must be taken when comparing residual vectors, since their magnitudes depend on the magnitude of $\hat{u}$, which is arbitrary. More precisely, as $\hat{u}$ increases, the residual vector of (4), and the a_i parameters of the solution, increase linearly with $\hat{u}$. This is of course also the case for (5), but *not* for the original Cobb-Douglas problem, which (5) is the logarithm



(a) Elicitation of alternatives.

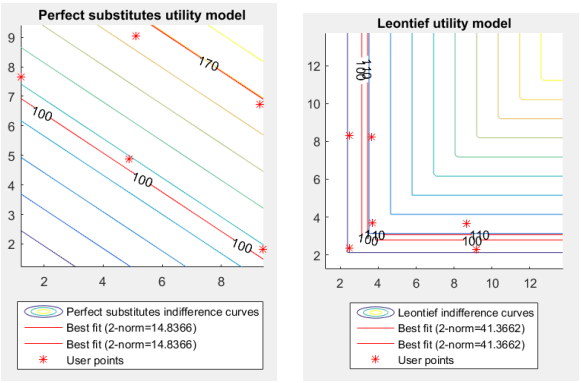
(b) Cobb-Douglas model built from data.

Figure 2: Screenshots showing elicitation of alternatives with the same utility. The crosshairs are used to place a point, using the mouse, in the graph. The functional form of the utility model and its parameters are estimated based on the points placed.

An elicitation system of this kind was implemented for the purpose of being able to estimate utility functions with respect to non-functional properties of IT components. The interface is shown in Fig. 2(a).

To identify a point (i.e. combination of properties 1 and 2) as yielding a certain utility, the decision-maker simply presses the `Add point` button and uses the crosshairs to select the point in the graph. Once enough points have been identified, the button `Build model` is used to build models, according to Eqs. 4, 5, and 6–8, respectively. The solution with the smallest Euclidean residual norm is selected and presented graphically to the decision-maker as

of. The residual vector of the original Cobb-Douglas problem instead increases exponentially, whereas the a_i parameters of the solution increase logarithmically with $\hat{u}$. Therefore, unless residual vectors from different solution alternatives are somehow normalized before being compared according to e.g. the Euclidean norm, the value of $\hat{u}$ will have an undue influence on the selection of utility model. One such normalization is to use the residual of the logarithmic problem in (5), divide it with the arithmetic mean of the a_i parameters, and compare it to the residual of (4), similarly divided with the arithmetic mean of the a_i parameters from that solution. These normalized residuals are invariant under transformations of $\hat{u}$. This complication was not fully appreciated in the original conference article.



(a) Perfect substitutes model built from data.

(b) Leontief model built from data.

Figure 3: Screenshots showing elicitation of alternatives from several indifference curves.

shown in Fig. 2(b). The model built is presented including the points selected by the user as well as the model level curve that most closely approximates them. It should be noted, however, that the two-dimensional nature of the presentation does not fully convey the goodness of an estimate, as the actual residual is calculated as the difference in the third dimension – the utility along the z axis. Thus, points seemingly close to the best fit in the x, y plane can still have a big residual, if the utility function is steep in that vicinity.

4.2. Elicitation from several indifference curves

While we derived Eqs. 4, 5, and 6–8 to solve the case of building models from a single indifference curve, it is straightforward to extend them to the case of elicitation from several indifference curves. The only change needed is to modify the $\hat{\mathbf{u}}$ vector so that it can contain a different utility $\hat{u}_i$ for each pair (x_i, y_i). Thus, points can be supplied on several indifference curves by the user. In the interface shown in Fig. 2(a), this is done by modifying the `Utility` slider to the right (set to 100 in the figure) before adding a point. The utility is displayed together with the point in the graph.

Utility models built from several indifference curves are shown in Fig. 3.

5. Consistency of elicited preferences

In the previous section, we limited ourselves to elicitation of utility functions in pairwise trade-offs between properties. This is partly due to the simple fact that the two-dimensional case is easier to illustrate, but it also has to do with the fact that it is easier and less taxing for the user to do the elicitation as a sequence of pairwise elicitations. However, this also raises the important question of how to ensure consistency in elicited preferences.

More precisely, the property we want to ensure is transitivity – we want to avoid the circular case where $A \prec B$, $B \prec C$, but also $C \prec A$. For each kind of pairwise utility functions elicited in the previous section, this holds locally by virtue of the mathematical nature of the functional forms used. But if several such pairwise elicited utility functions are combined, there is no a priori guarantee of such consistency.

5.1. The linear case

The discussion of consistency is easiest in the perfect substitutes case (1). This is similar to the analysis of monetary exchange rates [18], and to the analytical hierarchy process [19], especially the AHP literature on consistency [20]. Here, as noted above, the decision-maker is indifferent between one unit of x_i and $\frac{a_i}{a_j}$ units of x_j, and is thus willing to switch one unit of x_i for $\frac{a_i}{a_j}$ or more units of x_j.

This is a special case of a general observation that can be made by considering a small movement along x_i and x_j on an indifference curve, so that there is no total change in utility:

$$\frac{\partial u(\mathbf{x})}{\partial x_i} dx_i + \frac{\partial u(\mathbf{x})}{\partial x_j} dx_j = 0 \Rightarrow \frac{dx_j}{dx_i} = -\frac{\frac{\partial u(\mathbf{x})}{\partial x_i}}{\frac{\partial u(\mathbf{x})}{\partial x_j}} \tag{9}$$

It is common to refer to the absolute value (omitting the minus sign) of this slope as the *rate of substitution* between x_i and x_j.

In the perfect substitutes case, the derivatives in (9) are constant, so it can be applied also for non-infinitesimal changes to x_i and x_j. Applying it to the

exchanging of one unit of x_i for $\frac{a_i}{a_j}$ (so that $\Delta x_i = -1$, $\Delta x_j = +\frac{a_i}{a_j}$) gives the following zero change in utility, just a expected:

$$\frac{\partial u(\mathbf{x})}{\partial x_i}\Delta x_i + \frac{\partial u(\mathbf{x})}{\partial x_j}\Delta x_j = a_i(-1) + a_j\frac{a_i}{a_j} = 0 \tag{10}$$

For n properties, we can now form a matrix R of rates of substitution $\left|\frac{dx_j}{dx_i}\right|$ from x_i to x_j, which in the linear utility case are just the following constant quotients:

$$R = \begin{pmatrix} \frac{a_1}{a_1} & \frac{a_1}{a_2} & \dots & \frac{a_1}{a_n} \\ \frac{a_2}{a_1} & \frac{a_2}{a_2} & \dots & \frac{a_2}{a_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{a_n}{a_1} & \frac{a_n}{a_2} & \dots & \frac{a_n}{a_n} \end{pmatrix} \tag{11}$$

The fact that the diagonal elements are all equal to unity follows by definition: one unit of x_i is worth precisely one unit of x_i, for all i.

The fact that $r_{i,j} = 1/r_{j,i}$ for $r_{i,j} \in R$ is also important, as it fixes the value of one unit of x_i. Suppose that we trade one unit of x_i for $r_{i,j}$ units of x_j, and then trade the resulting $r_{i,j}$ units of x_j back in to x_i again: this circular trade must be on an indifference curve, so $u(x_i) = u(x_i r_{i,j} r_{j,i})$, as is of the course the case for all rates in (11).

Expanding this requirement to hold also for longer chains of exchanges, we see that the following n equations hold for (11):

$$\begin{pmatrix} r_{1,j} \\ r_{2,j} \\ \vdots \\ r_{n,j} \end{pmatrix} r_{j,j+1} = \begin{pmatrix} r_{1,j+1} \\ r_{2,j+1} \\ \vdots \\ r_{n,j+1} \end{pmatrix} \tag{12}$$

In words, each column j of exchange rates (expressing the rates of going from each of the properties $1, 2, \dots, n$ to property j) can be turned into the adjacent column $j+1$ by multiplying with the scalar exchange rate from j to $j+1$, $\frac{a_j}{a_{j+1}}$. This applies to the entire matrix if applied circularly, so that the n:th column

is turned into the 1st, using $r_{n,1} = \frac{a_n}{a_1}$. This property also entails a convenient necessary property of R (noted in AHP context in [19] and [20]):

Theorem 1 (Rank of R)**.** *Any R expressing non-circular preferences has rank 1.*

Proof. R has non-zero columns, so the rank is positive, but by (12), every pair of columns is linearly dependent, so the rank is 1. □

To make matters more concrete, say that we have elicited linear rates of substitution between functional suitability (x_1), performance efficiency (x_2), and compatibility (x_3) for some future software product in some circumstances. The elicitation results are the following three 3×3 matrices of rates of substitution for x_1 and x_2, x_1 and x_3, and x_2 and x_3, respectively, with the unknown elements in each matrix left blank.

$$\hat{R} = \begin{pmatrix} 1 & \frac{2}{3} & \\ \frac{3}{2} & 1 & \\ & & \end{pmatrix} \quad \tilde{R} = \begin{pmatrix} 1 & & \frac{3}{4} \\ & & \\ \frac{4}{3} & & 1 \end{pmatrix} \quad \check{R} = \begin{pmatrix} & & \\ & 1 & \frac{4}{5} \\ & \frac{5}{4} & 1 \end{pmatrix} \tag{13}$$

Taken as 2×2 matrices they all imply non-circular preferences, as $\hat{r}_{1,2} = 1/r_{2,1}$, $\tilde{r}_{1,3} = 1/r_{3,1}$, and $\check{r}_{2,3} = 1/r_{3,2}$. They also complement each other in the sense that values for the missing elements of each matrix can be found in the others. However, just naïvely combining these values into a full 3×3 matrix R' does not yield a consistent utility function:

$$R' = \begin{pmatrix} 1 & \frac{2}{3} & \frac{3}{4} \\ \frac{3}{2} & 1 & \frac{4}{5} \\ \frac{4}{3} & \frac{5}{4} & 1 \end{pmatrix} \tag{14}$$

To see this, we can note that $\text{rank}(R') = 3$, or just calculate e.g. $r'_{1,2} \cdot r'_{2,3} = \frac{2}{3} \cdot \frac{4}{5} = \frac{8}{15} \neq r'_{1,3} = \frac{3}{4}$.

There are several ways to make R' consistent. If one of the elicited rates is known to be less certain than the others, or problematic in some other way,

it is straightforward to re-calculate this particular rate from the others. If all rates are equally trustworthy, the derivatives of eigenvalues method from [20] can be applied. However, noting that the rates in R' are actually quotients of a_i parameters from (1), we can understand the inconsistency in another way, viz. as inconsistencies in the observations of a_i. Specifically, $\hat{R}$ implies that $\hat{a}_1 = 2$ and $\hat{a}_2 = 3$, $\tilde{R}$ implies that $\tilde{a}_1 = 3$ and $\tilde{a}_3 = 4$, and $\check{R}$ implies that $\check{a}_2 = 4$ and $\check{a}_3 = 5$.

Luckily, this also suggests a family of straight-forward methods for forming a consistent joint 3×3 matrix R, namely combining the different parameter values into a single (thus consistent) parameter. For example, using arithmetic means as combination method, we form a consistent a_1 simply as follows: $a_1 = (\hat{a}_1 + \tilde{a}_1)/2$. In the example, we thus find a consistent matrix R by plugging $a_1 = 5/2$, $a_2 = 7/2$, and $a_3 = 9/2$ into (11):

$$R = \begin{pmatrix} 1 & \frac{5/2}{7/2} & \frac{5/2}{9/2} \\ \frac{7/2}{5/2} & 1 & \frac{7/2}{9/2} \\ \frac{9/2}{5/2} & \frac{9/2}{7/2} & 1 \end{pmatrix} \tag{15}$$

This matrix corresponds to one single utility function of the form (1). We can also easily verify that $\mathrm{rank}(R) = 1$.

5.2. The Cobb-Douglas case

Moving on to the Cobb-Douglas case, the rates of substitution between the properties x_i and x_j are no longer constant, but vary depending on the magnitudes of x_i and x_j. This also differs from the use of constant weights in AHP. In particular, properties exhibit *diminishing marginal utility*, which is very reasonable in many cases: going from 1 to 2 in terms of reliability, say, may well be worth more than going from 2 to 3. (But again, this may not be the case, which can be captured by utility functions such as the Leontief or linear one.)

More precisely, to find the rates of substitution between x_i and x_j, we can apply (9) to (3), and obtain the following well known result about Cobb-Douglas

rates of substitution:

$$r_{i,j} = \frac{\frac{\partial u(\mathbf{x})}{\partial x_i}}{\frac{\partial u(\mathbf{x})}{\partial x_j}} = \frac{a_i x_i^{a_i - 1} \prod_{k \neq i} x_k^{a_k}}{a_j x_j^{a_j - 1} \prod_{k \neq j} x_k^{a_k}} = \frac{a_i x_i^{a_i - 1} x_j^{a_j}}{a_j x_j^{a_j - 1} x_i^{a_i}} = \frac{a_i x_j}{a_j x_i} \tag{16}$$

Inspecting (16), we see that as the amount of x_j relative to x_i grows, the exchange rate required for indifference increases, i.e. diminishing marginal utility of x_j.

For these rates to be consistent, again we can form the following condition that must hold:

$$r_{i,k} = r_{i,j} r_{j,k} = \frac{a_i x_j}{a_j x_i} \frac{a_j x_k}{a_k x_j} = \frac{a_i x_k}{a_k x_i} \tag{17}$$

This means that (12) applies here as well: each column in R can be transformed into the next by multiplication with a conversion factor, the important difference being that this exchange rate is not fixed, but depends on the amounts of the properties involved.

However, (17) is just a necessary condition, not a sufficient one. For even though (16) simplifies beautifully as a quotient, we need the $\frac{\partial u(\mathbf{x})}{\partial x_i}$ from each elicitation where property x_i is involved to be the same – including, in the Cobb-Douglas case, the long product conveniently canceled out in (16). The matrix of rates of substitution is a matrix of functions that must satisfy (9) for any input argument (i.e. magnitudes of x_i). This is one reason why the arithmetic means method of ensuring consistency is attractive in its simplicity. Methods such as taking derivatives of the eigenvalues of the matrix are more difficult to apply in the general case, compared to when the matrix is constant.

5.3. The Leontief case

In the Leontief case, the difficulties identified for linear and Cobb-Douglas utility in principle still hold. However, due to its particular functional form, it also exhibits additional difficulties. Applying (9) to (2), we obtain the following

expression:

$$\begin{aligned}\frac{\partial u(\mathbf{x})}{\partial x_i}dx_i + \frac{\partial u(\mathbf{x})}{\partial x_j}dx_j = \\ =dx_i \cdot \begin{cases} a_i & \text{if } a_i x_i = \min_{k=1}^{n}\{a_k x_k\} \\ 0 & \text{otherwise} \end{cases} + \\ dx_j \cdot \begin{cases} a_j & \text{if } a_j x_j = \min_{k=1}^{n}\{a_k x_k\} \\ 0 & \text{otherwise} \end{cases} = \\ = \begin{cases} dx_i a_i \neq 0 & \text{if } a_i x_i = \min_{k=1}^{n}\{a_k x_k\} \\ dx_j a_j \neq 0 & \text{if } a_j x_j = \min_{k=1}^{n}\{a_k x_k\} \\ 0 & \text{otherwise} \end{cases}\end{aligned} \tag{18}$$

Putting (18) into simple words: Trade-offs between x_i and x_j cannot be made along Leontief indifference curves. If $a_i x_i$ is the smallest of all weighted properties, then the utility gained by increasing x_i cannot be balanced by any utility lost by decreasing x_j because decreasing x_j does not decrease utility. Mutatis mutandis if $a_j x_j$ is the smallest of all weighted properties. A "trade-off" can only be made if neither $a_i x_i$ nor $a_j x_j$ is the smallest of all weighted properties, meaning that changes in x_i and x_j do not matter at all for utility.

This is just the perfect complements property expressed formally – Leontief goods are not substitutable. Forming rates of substitution as quotients of partial derivatives is revealing: if $a_i x_i$ is the smallest of all weighted properties, then the exchange rate $r_{i,j} = \frac{a_i}{0} = \infty$, and conversely the exchange rate $r_{j,i} = \frac{0}{a_i} = 0$, because it is never worth trading even an infinitesimal amount of x_i even for an arbitrarily large amount of x_j.

Again, to ensure consistency among several pairwise elicited Leontief rates of substitution, we need to find consistent parameters that determine a unique instance of (2). As is evident from (18), in the Leontief case consistent parameters do not only influence the magnitudes of the partial derivatives, but also decide which one of them is non-zero. Here, it becomes clear that we cannot just do this comparison pairwise (as elicited), but that the min function needs to be applied globally to a vector of properties, consistently weighted, e.g. by

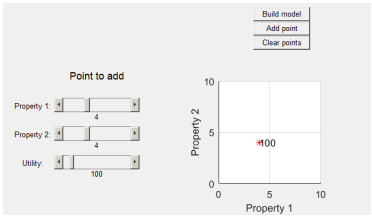


Figure 4: Screenshot showing alternative elicitation interface, more amenable to be applied in the higher dimensional case.

taking arithmetic means of the parameters pairwise elicited.

5.4. Enforcing consistency in elicitation

Another strategy for ensuring consistency would be to change the elicitation model. (4)–(8) can easily be modified to elicit further parameters, in addition to a_1 and a_2. However, such an interface might also become less intuitive to the user, as the two-dimensional graphical depiction of the trade-off (seen in Fig. 2(a)) could no longer be used. Instead, such an interface would require values along the property axes (which would be eight, if adhering to ISO/IEC 25010) to be set in some other way. Fig. 4 shows an example of such an interface, where the property values are entered through sliders rather than mouse clicks in the two dimensional plane. The number of sliders could easily be expanded (e.g. to eight) allowing elicitation of consistent preferences that could be fed into higher dimensional versions of (4)–(8). However, the graphical aid to the right, depicting the choices made, is not as easy to expand. The problem with such elicitation is that though it is mathematically convenient, the problem faced by the user rapidly approaches the kinds of high dimensional vector comparisons illustrated in Section 1. In other words, this elicitation may be too taxing to use in practice as it pushes the original problem of vector comparison back to the decision-maker.

An alternative middle road could be to elicit higher dimensional preferences using sliders, but also to modify the graphical aid so that the user can use it to show projections of the higher dimensional locations of the points set onto the

plane or onto a three dimensional space.

5.5. The case of several models

An implication of the discussion above is that for three properties x_1, x_2, x_3, we cannot have the trade-offs between 1 and 2, and 2 and 3, respectively to be of one functional kind (e.g. Cobb-Douglas), but the trade-off between 1 and 3 to be of another (e.g. linear). This is prohibited by (12), which for this example translates into (17) (with $i = 1$, $j = 2$, $k = 3$) showing that the trade-off between 1 and 3 is also Cobb-Douglas.

However, if one set of properties $x_1, \ldots, x_n$ has a consistent utility function of one kind, and another set of properties $x_{n+1}, \ldots, x_{n+m}$, has a consistent utility function of another kind, it is possible to find rates of substitution from one property of the first kind to another property of the second kind.

Consider first the case of substitution between Cobb-Douglas properties and linear utility properties. Without loss of generality, assume an ordering such that the first n properties jointly contribute to a Cobb-Douglas utility function, and the next m properties jointly contribute to a linear utility function. We then have a total utility function of the following form:

$$u(\mathbf{x}) = \prod_{i=1}^{n} x_i^{a_i} + \sum_{j=n+1}^{n+m} a_j x_j \tag{19}$$

The rates of substitution between any pair among the first n ones will be according to (16), for the derivatives of the sum part of (19) will be zero with respect to all the first n properties, so (9) holds just as for a pure Cobb-Douglas utility function. By a symmetric argument, the rates of substitution between any pair among the next m properties will be the quotients of (11).

To find the rates of substitution between the two sets, we again form quotients of partial derivatives according to (9):

$$r_{i,j} = \frac{\frac{\partial u(\mathbf{x})}{\partial x_i}}{\frac{\partial u(\mathbf{x})}{\partial x_j}} = \frac{a_i x_i^{a_i - 1} \prod_{k \in \{1,\ldots,n\} \setminus i}^{n} x_k^{a_k}}{a_j} \tag{20}$$

This rate of substitution is not constant, as in the linear case, but exhibits diminishing marginal utility – the rate goes up as more x_i is traded for x_j

(Cobb-Douglas parameters a_i are typically less than unity, so $x_i^{a_i-1}$ grows as x_i decreases). However, as opposed to the diminishing marginal utility of the Cobb-Douglas rate, (20) is independent of x_j. This is expected – a unit of x_j is always worth a_j, no matter how many one already has, according to (19). Rather the whole effect comes from each unit of x_i becoming more valuable as the level of x_i decreases.

Moving on to the case of substitution between Cobb-Douglas utility properties and Leontief utility properties, using ordering as above, we have a total utility function of the following form:

$$u(\mathbf{x}) = \prod_{i=1}^{n} x_i^{a_i} + \min_{j=n+1}^{n+m} a_j x_j \tag{21}$$

To find the rates of substitution between these two sets, we again form quotients of partial derivatives according to (9). But the numerator is the same as the numerator in (20), and the denominator is either a_j, if $a_j x_j = \min_{k=n+1}^{n+m}\{a_k x_k\}$, or zero. So the rates of substitution between Cobb-Douglas utility properties and Leontief utility properties is either the same as that between Cobb-Douglas utility properties and linear utility properties, i.e. (20), or infinite. This is reasonable, as the Leontief utility either grows linearly in x_j, or not at all.

Finally considering the case of substitution between linear utility properties and Leontief utility properties, using ordering as above, we have a total utility function of the following form:

$$u(\mathbf{x}) = \sum_{i=1}^{n} a_i x_i + \min_{j=n+1}^{n+m} a_j x_j \tag{22}$$

Again forming quotients of partial derivatives according to (9), we find rates of substitution that are either the same as that between linear utility properties, i.e. $r_{i,j} = \frac{a_i}{a_j}$ if $a_j x_j = \min_{k=n+1}^{n+m}\{a_k x_k\}$, or infinite.

6. Grand unifying example

The scene is now all set for a grand unifying example, exhibiting the full power of the framework devised in the preceding sections. Suppose that we face

trade-offs between the eight non-functional properties from ISO/IEC 25010. Using the elicitation framework described in Section 4, we can find trade-offs between pairs of properties.

Let us say that we have elicited a Cobb-Douglas utility function for properties x_1, x_2, x_3, a linear utility function for properties x_4, x_5, x_6, and a Leontief utility function for properties x_7, x_8. Each of these have been made consistent for instance by forming means of parameters from each pairwise elicitation, as described in Section 5. We thus have three matrices of rates of substitution – R_{123}, R_{456}, and R_{78}. Based on the discussion in Section 5.5, we can now form a grand total 8×8 matrix of rates of substitution, seen in (23).

$$
R = \left(\begin{array}{ccc|ccc|cc}
1 & \frac{a_1 x_2}{a_2 x_1} & \frac{a_1 x_3}{a_3 x_1} & \frac{a_1 x_1^{a_1-1} x_2^{a_2} x_3^{a_3}}{a_4} & \frac{a_1 x_1^{a_1-1} x_2^{a_2} x_3^{a_3}}{a_5} & \frac{a_1 x_1^{a_1-1} x_2^{a_2} x_3^{a_3}}{a_6} & \begin{cases} \frac{a_1 x_1^{a_1-1} x_2^{a_2} x_3^{a_3}}{a_7} \\ \infty \end{cases} & \begin{cases} \frac{a_1 x_1^{a_1-1} x_2^{a_2} x_3^{a_3}}{a_8} \\ \infty \end{cases} \\
\frac{a_2 x_1}{a_1 x_2} & 1 & \frac{a_2 x_3}{a_3 x_2} & \frac{a_2 x_2^{a_2-1} x_1^{a_1} x_3^{a_3}}{a_4} & \frac{a_2 x_2^{a_2-1} x_1^{a_1} x_3^{a_3}}{a_5} & \frac{a_2 x_2^{a_2-1} x_1^{a_1} x_3^{a_3}}{a_6} & \begin{cases} \frac{a_2 x_2^{a_2-1} x_1^{a_1} x_3^{a_3}}{a_7} \\ \infty \end{cases} & \begin{cases} \frac{a_2 x_2^{a_2-1} x_1^{a_1} x_3^{a_3}}{a_8} \\ \infty \end{cases} \\
\frac{a_3 x_1}{a_1 x_3} & \frac{a_3 x_2}{a_2 x_3} & 1 & \frac{a_3 x_3^{a_3-1} x_1^{a_1} x_2^{a_2}}{a_4} & \frac{a_3 x_3^{a_3-1} x_1^{a_1} x_2^{a_2}}{a_5} & \frac{a_3 x_3^{a_3-1} x_1^{a_1} x_2^{a_2}}{a_6} & \begin{cases} \frac{a_3 x_3^{a_3-1} x_2^{a_2} x_1^{a_1}}{a_7} \\ \infty \end{cases} & \begin{cases} \frac{a_3 x_3^{a_3-1} x_2^{a_2} x_1^{a_1}}{a_8} \\ \infty \end{cases} \\
\hline
 & & & 1 & \frac{a_4}{a_5} & \frac{a_4}{a_6} & \begin{cases} \frac{a_4}{a_7} \\ \infty \end{cases} & \begin{cases} \frac{a_4}{a_8} \\ \infty \end{cases} \\
 & & & \frac{a_5}{a_4} & 1 & \frac{a_5}{a_6} & \begin{cases} \frac{a_5}{a_7} \\ \infty \end{cases} & \begin{cases} \frac{a_5}{a_8} \\ \infty \end{cases} \\
 & & & \frac{a_6}{a_4} & \frac{a_6}{a_5} & 1 & \begin{cases} \frac{a_6}{a_7} \\ \infty \end{cases} & \begin{cases} \frac{a_6}{a_8} \\ \infty \end{cases} \\
\hline
 & & & & & & 1 & \begin{cases} \infty \text{ if } a_7 x_7 < a_8 x_8 \\ 0 \text{ otherwise} \end{cases} \\
 & & & & & & \begin{cases} 0 \text{ if } a_7 x_7 < a_8 x_8 \\ \infty \text{ otherwise} \end{cases} & 1
\end{array}\right) \tag{23}
$$

The blocks along the diagonal are just the three matrices R_{123}, R_{456}, and R_{78} from the elicitation. The subdiagonal blocks are omitted for convenience – we know already that $r_{i,j} = 1/r_{j,i}$. The superdiagonal blocks are formed as described in Section 5.5 – showing rates of substitution between Cobb-Douglas and linear (rows $1-3$ $\times$ columns $4-6$), between Cobb-Douglas and Leontief (rows $1-3$ $\times$ columns $7-8$), and between linear and Leontief (rows $4-6$ $\times$ columns $7-8$). Whenever trade-offs are made with the Leontief properties x_7

and x_8, rates will be ∞ when the denominator is 0 (and conversely, rates will be 0 when the numerator is 0 on the subdiagonal).

Two things are worth noticing about the matrix in (23). First, with the exception of the linear trade-offs in the very middle, the values are not fixed, but functions of the values in the vector $\mathbf{x}$. This also means that trade-offs made according to any particular rate are in general only valid for infinitesimal changes dx_i and dx_j. For larger changes Δx_i and Δx_j, the rates found at the initial values $\mathbf{x}_0$ are linear approximations. Second, (12) holds for R in (23).

(23) implies a utility function with three components – the sum of a Cobb-Douglas, a linear, and a Leontief utility function. Given the parameters $a_1, \ldots, a_8$, we are now in a position to answer the question posed in the introduction: Which of the alternatives A, B, or C should be preferred?

Assuming, for the sake of the example, that the elicitation gave $\mathbf{a} = (0.5, 0.3, 0.2, 2, 3, 4, 5, 6)$, we find $u(A) = 83.25$, $u(B) = 86.86$, and $u(C) = 44.29$, so under these preferences B turns out to be the best. The numbers of the example are arbitrary, of course, but the trade-off method illustrated is not.

7. Empirical results

In order to test the conceptual ideas outlined above, we conducted a set of exploratory preference elicitations. While these cannot be considered proper experiments, they nevertheless gave some interesting results, including insights that are relevant for future rigorous experiments. This section outlines the elicitation instrument and setup, the various pilot elicitations, and the two mature elicitations finally conducted.

7.1. Elicitation instrument and setup

Based on the interface idea shown in Fig. 2(a), an actual elicitation instrument was constructed. The method used was an idea originally proposed in the discussion of the conference article, viz. to start from an existing product with known non-functional properties, employing it as a baseline. Stake-holders

would then be asked to identify their indifference curves in regions corresponding to small perturbations in those properties.

For this idea to be feasible, suitable non-functional properties had to be identified. These properties would need to meet three important requirements: (i) being known (to a reasonable precision) for the baseline product, and (ii) being measurable on a numeric scale, preferably a ratio scale, and (iii) there being at least three properties in the chosen set, to allow for analysis of consistency as described in Section 5. Though initially discussed, the idea of letting the subjects themselves select the non-functional properties was discarded, as it seemed to entail a high risk of not meeting criterion (ii). Criterion (iii), in practice, became an exact number rather than a lower bound, to avoid unnecessarily taxing elicitation. In the end, the following three properties were selected and fixed for use in all the elicitations:

Mean Time to Failure (MTTF): This is the time it takes, on average, before the service fails and becomes unavailable. Example: 83 days.

Mean Time to Restore (MTTR): This is the time it takes, on average, to get the service available again after a failure. Example: 3 hours.

Mean development time for a new feature: This is the time it takes, on average, to implement, test, and commit a new feature in the service. Example: 14 days.

The definitions and examples given above were the ones given to the subjects when introducing the elicitation.

With these properties fixed, an elicitation instrument was built. Google sheets, rather than the Matlab based instrument described in Section 4, were used in order to allow (i) simple data collection from several users, (ii) at the same time, without (iii) any need to install separate software on the computers of subjects, while (iv) still allowing competent management and display of numeric results. The computational machinery required to solve Eqs. 4, 5, and 6-8 was not implemented in Google sheets, however, as the estimated utility functions

did not have to be displayed in real time to the subjects in the elicitations conducted.

The interface is shown in Fig. 5. It allows elicitation of the three pairwise trade-offs (i) MTTF vs. MTTR, (ii) MTTF vs. Mean development time, and (iii) MTTR vs. Mean development time. In each case, four hypothetical alternatives – characterized by their property values – are elicited, forming an indifference curve together with the baseline original property values. The resulting indifference curve is displayed on the right-hand side, as the numbers are entered.

At the bottom of the elicitation instrument, subjects were also asked to follow a URL to fill out a short feedback form evaluating the task and the elicitation process. It contained the following substantial questions:

1. How interesting did you find the task? (5 point Likert scale)
2. How clear did you find the task? (5 point Likert scale)
3. How relevant and helpful do you think the graphical representation was for solving the task? (5 point Likert scale)
4. What is your main take away (or understanding) from the task? (Free text)
5. Additional feedback or comments (Free text)

Subjects could choose to answer this form anonymously, so that their elicited preferences and their feedback could not be matched, or allow the two to be identified via a pseudonym (still maintaining anonymity with regard to their real names).

The overall setup proceeded as follows. The subjects were given a short introduction to trade-offs by the elicitation leader. The presentation emphasized that trade-offs are everywhere in engineering, and that most of the time there are no obvious 'right' or 'wrong' solutions. Whenever alternatives with vectors of properties do not dominate each other, non-trivial trade-offs have to be made. Next, the three properties MTTF, MTTR, and Mean development time were introduced, as above. Once these properties had been introduced, the first task

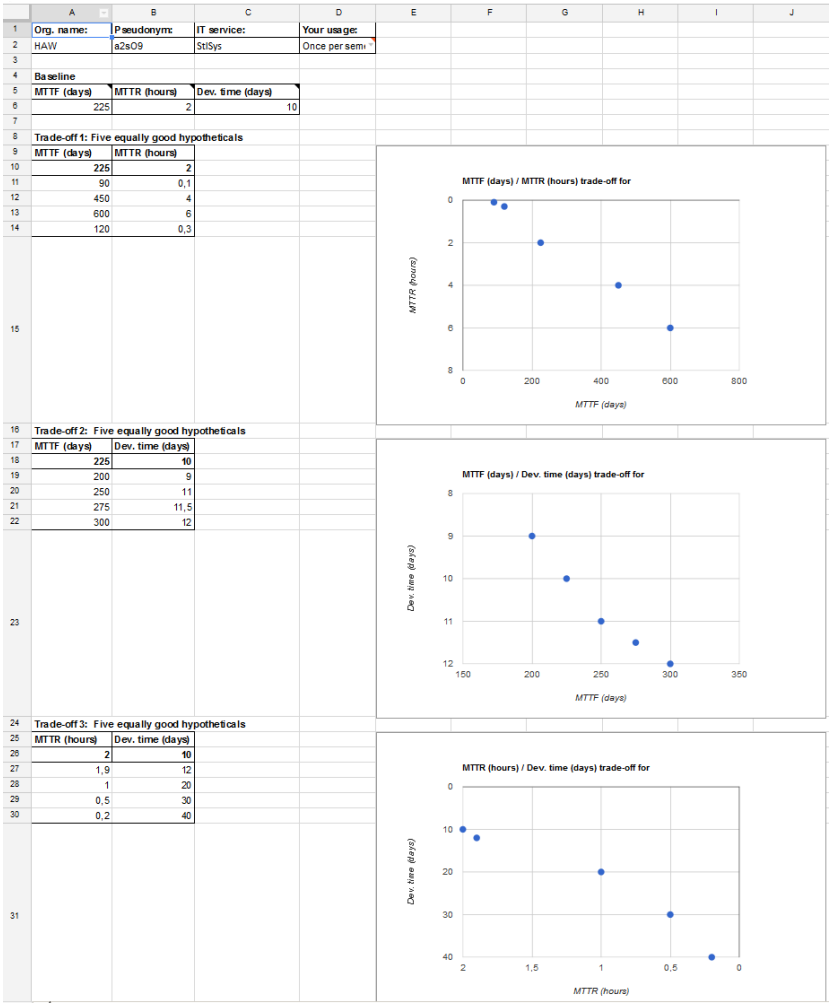


Figure 5: Screenshot showing elicitation of alternatives with the same utility. For each of the three pair-wise trade-offs, property values of four hypothetical alternatives are entered below the baseline, bolded, showing the actual property values. As the hypotheticals are entered, the diagrams to the right are populated, showing the points on the indifference curves as they are entered.

was to find the baseline to be used. Subjects were asked, collectively, to think of an IT service that they use, and then discuss and agree upon numeric values for the three properties. The elicitation leader moderated the discussion as needed, and also entered the resulting figures into the Google sheet based elicitation

instrument.

Once subjects had agreed on the baseline, their task was introduced in greater detail. It was explained that the aim was to investigate trade-offs by eliciting four hypothetical alternative services, assessed to be equally good as the baseline. To do this, while avoiding introducing any bias with regard to the particular properties at hand, a non-IT example was used. Subjects were asked to consider a baseline hotel with properties a two element property vector (cost = 100 €/night, distance from city center = 2 km). Against this baseline, the following examples of equally good hypotheticals were given:

- (cost = 120 €/night, distance = 1 km)
- (cost = 180 €/night, distance = 0.5 km)
- (cost = 70 €/night, distance = 5 km)
- (cost = 50 €/night, distance = 10 km)

It was explained that eliciting these alternatives would allow the construction of a so called indifference curve. It was also stressed that subjects should not think of the technical feasibility of their hypotheticals – the focus of the elicitation is not on feasibility but on preferences. Despite this, of course, it is still possible that some subjects were constrained in the elicitation by their self-perceived knowledge of feasible engineering choices. Finally, it was emphasized that there are no right or wrong answers, and that the trade-offs are context dependent. Subjects were asked to take the task seriously and do their best to make sensible trade-offs in their own context.

At this stage, the URL to the elicitation instrument was distributed to the subjects, who could then start filling out their data.

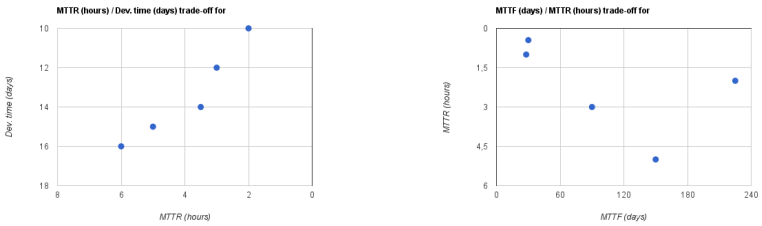
7.2. Pilot elicitations

In order to test the elicitation instrument before using it for real, two pilot elicitations were conducted. The first pilot round was less formal, using the Google sheets based instrument only, omitting the presentation and guiding by

the elicitation leader. The test subjects were colleagues of the authors, i.e. researchers with PhDs in computer science or information systems. Out of three invited colleagues, two completed the form. In this pilot, there was no fixed IT service for the subjects to consider, but they were rather allowed to make something up themselves. Explicit feedback included the need for more instructions to participants (which in part prompted the development of the presentation and guidance by the elicitation leader), the need to provide explicit definitions of the three properties, and the need to highlight the baseline values, so that subjects would not overwrite them. An implicit lesson learned from the results rather than the subject feedback was that the third trade-off was challenging to subjects. Both pilot subjects in fact offered a series of hypotheticals that dominate each other, as seen in Fig. 6(a), rather than hypotheticals that could reasonably form an indifference curve. Such a systematically spurious result could possibly be attributed to poor understanding among subjects of the properties, i.e. what constitutes better or worse MTTF, MTTR, and Mean development time.

The second pilot round included the full setup, including the presentation and guiding by the elicitation leader. The test subjects were students in computer science recruited by the authors, one undergraduate and one graduate student. As opposed to the first pilot, subjects were asked to think of real IT services, but the two subjects did not have to think of the same service. Another change from the first pilot was that the questionnaire evaluating the task and the elicitation process was tested for the first time. These answers revealed that subjects found the task both interesting and clear (average of 4 on 5 point Likert scale), and also that the graphical representation in the diagrams was relevant and helpful for solving the task (average of 4 on 5 point Likert scale). One of the pilot subjects expressed it as follows: "The graphical representation was very useful. Without it, the task might seem a bit too abstract." The second pilot also confirmed that the third trade-off is challenging to subjects, as these two subjects also offered series of hypotheticals that dominate each other.

Having thus tested and improved the elicitation instrument and setup in the pilots, two actual elicitations with two quite different sets of subjects were



(a) A systematically spurious indifference curve.

(b) A more random spurious indifference curve.

Figure 6: Examples of spurious elicitation results (from elicitation 1), where some hypotheticals dominate others in a systematic or more random way.

conducted, described in Sections 7.3, and 7.4, respectively.

In addition to these two, a third elicitation was attempted with subjects being colleagues of the authors, working on an automotive software development research project [21], which has previously been suitable for research activities [22]. However, when elicitation was attempted, it became evident that it was very difficult to get a common idea of a suitable IT service, the properties of which to reason about. In this respect, a research project where the service is constantly being re-developed and re-defined turned out to be ill-suited.

7.3. Elicitation 1: University course

The first set of subjects were a class of university students in their final year towards the bachelor's degree, now taking a specialization course in Enterprise Architecture, and the professor teaching the course. The first author gave a guest lecture in their course, before moving on to present the elicitation task.

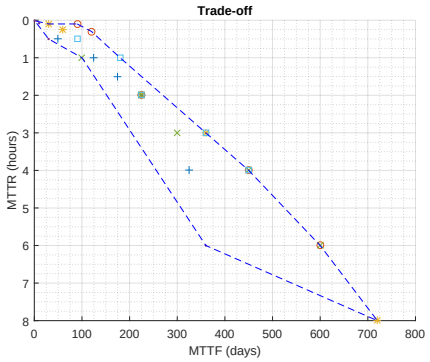
In this elicitation, a suitable IT service known and used by the students had been identified beforehand: the Student Information Systems (StISys), used to sign up for exams and labs, as well as to access examination results. Once the three properties had been introduced, the first elicitation took place, collectively, by discussing and agreeing upon numeric values for the properties. In accordance with the procedure described above, the elicitation leader moderated the

discussion, eventually arriving at a baseline of 225 days of MTTF, 2 hours of MTTR, and 10 days of Mean development time for a new feature. These figures do not necessarily correspond to the ground truth as might be found e.g. from logs, but they do constitute a valid baseline to elicit preferences. For the development time, the presence of the professor was instrumental in getting a good estimate, as the students only have a limited perception of the new features introduced in the service.

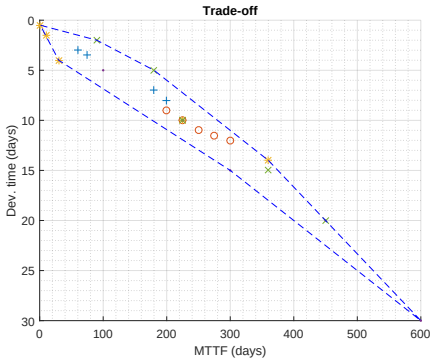
In total, 8 participants filled out the elicitation instrument. Out of these, one participant only stated one of the three requested trade-offs, resulting in 22 distinct elicited indifference curves. Out of these, 14 were reasonable in the weak sense of not containing any dominating/dominated hypotheticals on the indifference curve, whereas the remaining 8 did contain such hypotheticals. Two typical examples (from elicitation 1) are shown in Fig. 6. As noted above, systematically spurious results (but not seemingly random spuriousness) could possibly be attributed to poor understanding among subjects of the properties, i.e. what constitutes better or worse MTTF, MTTR, and Mean development time. As seen in Fig. 2(a), the elicitation instrument was built to help subjects in this respect, by inverting the axes for properties where a smaller value is better, so that better alternatives are always graphically depicted to the right and/or above worse alternatives. However, it is clear that this mechanism is not sufficient to prevent systematically spurious preferences.

Fig. 7 gives an overview of the elicitation results with the individual responses exhibiting dominating/dominated hypotheticals removed. Note that here again axis directions correspond to the convention that moving to the right or up is better (as illustrated in Fig. 1), i.e. the axis directions for MTTR and Dev. time have been reversed, as these properties are better the smaller they are.

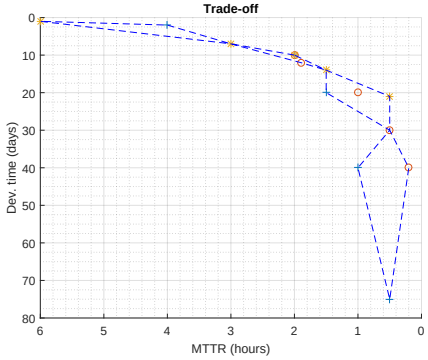
In each of the plots in Fig. 7, the elicited hypotheticals of a particular subject are shown using a unique marker, e.g. hypotheticals from one subject are shown as $\square$, hypotheticals from another as $\times$. When hypotheticals from several subjects coincide, the markers are superimposed (and thus, unfortunately, not



(a) The MTTF/MTTR trade-off, showing 6 responses (2 removed).



(b) The MTTF/Dev. time trade-off, showing 6 responses (2 removed).



(c) The MTTR/Dev. time trade-off, showing 4 responses (4 removed).

Figure 7: Elicitation results from the university course. The baseline is (MTTF= 225 days, MTTR= 2 hours, Dev. time= 10 days). In each case, a boundary plot (blue, dashed) encompasses all elicited hypotheticals to give a rough visual indication of how the set of indifference curves looks.

very legible). The baseline, of course, can be seen as the superimposition of all the markers.

Two interesting observations can immediately be made. First, as is to be expected, subjects do not agree. For example, as seen in Fig. 7(a), three different subjects $(\times, +, \square)$ put hypotheticals $\times$ =(MTTF = 100, MTTR = 1),

+ =(MTTF = 125, MTTR = 1), and □ =(MTTF = 180, MTTR = 1) on the same indifference curve as the baseline (MTTF = 225, MTTR = 2). But these three hypotheticals cannot all be on a non-Leontief indifference curve, since they all share the same MTTR but differ in MTTF. Such spans in preferences are illustrated, roughly, by the boundary plots encompassing all the elicited hypotheticals for each trade-off. A tight boundary plot indicates more agreement – a wider boundary plot indicates less agreement. Of course, as argued in Section 4, disagreement between subjects can be resolved using least squares solutions of utility functions, but it should also be noted that it can be dangerous to apply this approach mechanically. This will be further discussed in Section 8.

The second observation relates precisely to this. Looking at Fig. 7(c), it is relatively clear that the collective indifference curve is not linear, but exhibits a curvature. However, it is *concave* rather than *convex*, like the Cobb-Douglas utility depicted in Fig. 1(c). Naïvely solving (5) gives a solution, $a_1 = 1.71 > 0$, $b_1 = 1.54 > 0$, where *increasing* MTTR and *increasing* development time yields higher utility. In other words, the directions – what is good and what is bad – for both properties have been confused. This is a drawback of the elicitation method outlined in Section 4.1, and the elicitation instrument employed in the empirical study. Using the method outlined in Section 4.2 instead would solve this particular problem, but only at the cost of requiring the subject to specify cardinal utilities, as opposed to mere ordinal ones. Though this might be feasible in some cases, it would often impose unreasonable requirements on the subjects.

7.3.1. Subject feedback

Turning to the feedback form answered by subjects after the elicitation was complete, 7 out of the 8 subjects completed it. The quantitative results are given in Table 1. It is evident that most subjects found the task both interesting and clear. Opinions are more diverse on the usefulness of the graphical representation, i.e. the diagrams seen to the right in Fig. 5. As only 3 subjects had allowed their elicited preferences and their feedback to be matched, it is difficult to say anything specific about the relation between the perceived use-

Table 1: Quantitative subject feedback, on 5 point Likert scale, from elicitation 1.

Question:	How interesting did you find the task?	How clear did you find the task?	How relevant and helpful do you think the graphical representation was for solving the task?
	4	5	2
	5	4	4
	5	3	2
	5	4	5
	3	5	5
	5	5	5
	5	4	4
Mean:	4.57	4.29	3.86
Median:	5	4	4

fulness of the graphical representation and the quality of the results in terms of existence of dominating/dominated hypotheticals on the indifference curve. The data that is available is mixed indeed – two subjects ranking the usefulness of the graphical representation as 5 and 2, respectively, both gave only dominance free indifference curves, whereas 1 subject who also ranked the usefulness of the graphical representation as 2, gave two dominance free indifference curves and one with systematically dominating/dominated hypotheticals.

As for the free text questions in the feedback form, the main take aways of the subjects in general were reasonable and thoughtful, focusing on the necessity of trade-offs, that every person might exhibit individual trade-offs, and the fact that some properties are more important than others, but that translating this into a numeric value is more difficult. One subject also commented on linear (Fig. 1(a)) vs. curved indifference curves, saying that the latter might be more realistic.

7.4. Elicitation 2: Integration service provider

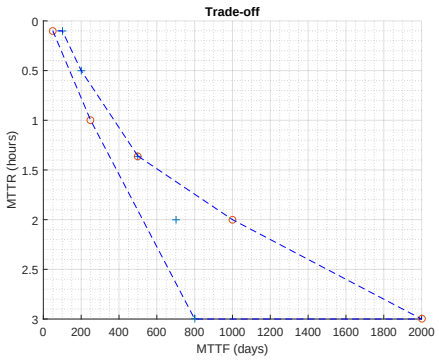
The second set of subjects were two employees of a commercial integration service provider. One subject had a role in sales, the other in technology.

In this elicitation, the IT service had not been decided upon beforehand. However, the subjects very quickly agreed that the most suitable service was the integration platform that constitutes the core business of the company, i.e. processing and transferring information, on behalf of customers, between various applications. Once the three properties had been introduced, the first elicitation took place, collectively, by discussing and agreeing upon numeric values for the properties. In accordance with the procedure described above, the elicitation leader moderated the discussion, eventually arriving at a baseline of 500 days of MTTF, 1.36 hours of MTTR, and 20 days of Mean development time for a new feature. In this case, these figures should be regarded as having reasonably high precision, though it should be remembered that as mean values, they can serve only as a rough characterization of the full underlying distributions. The subject with the technology role was instrumental in achieving high precision in the estimates.

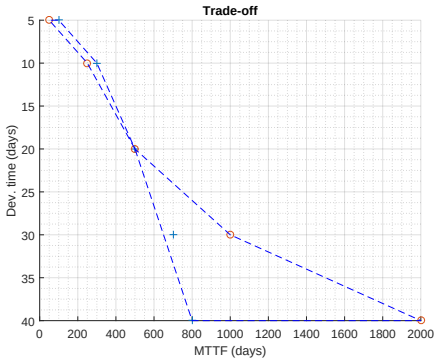
Both subjects fully completed the elicitation instrument. There were no dominating/dominated hypotheticals on the indifference curves.

Fig. 8 gives an overview of the elicitation results. As no individual responses exhibited dominating/dominated hypotheticals, nothing has been removed. Axis directions and unique markers per subject are as before.

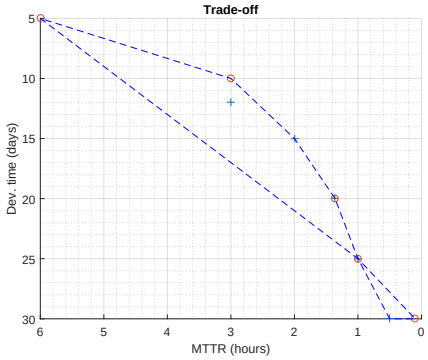
Similar observations can be made as in the case of the university course. Again, as is to be expected, subjects do not agree. This is particularly evident in the cases of greater MTTF values. For example, as seen in Fig. 8(a), the two different subjects $(\circ, +)$ put hypotheticals $+ =$(MTTF = 800, MTTR = 3), $\circ =$(MTTF = 2000, MTTR = 3) on the same indifference curve as the baseline (MTTF = 500, MTTR = 1.36). But these two hypotheticals cannot both be on a non-Leontief indifference curve, since they both share the same MTTR but differ in MTTF. The same tendency is seen in Fig. 8(b), where hypotheticals $+ =$(MTTF = 800, Dev. time = 40), $\circ =$(MTTF = 2 000, Dev. time = 40) were



(a) The MTTF/MTTR trade-off, showing 2 responses.



(b) The MTTF/Dev. time trade-off, showing 2 responses.



(c) The MTTR/Dev. time trade-off, showing 2 responses.

Figure 8: Elicitation results from the integration service provider. The baseline is (MTTF= 500 days, MTTR= 1.36 hours, Dev. time= 20 days). In each case, a boundary plot (blue, dashed) encompasses all elicited hypotheticals to give a rough visual indication of how the set of indifference curves looks.

put on the same indifference curve as the baseline (MTTF = 500, Dev. time = 20). The MTTF issue will be further discussed in Section 8.

The second observation can also be made again: looking at Fig. 8(c), it is relatively clear that the collective indifference curve is not linear, but exhibits a concave curvature. (The boundary plot is visually misleading here, as there

are no actual hypotethicals between the boundaries of the long line segment connecting MTTR = 6 with MTTR = 1. A better boundary would have passed the + =(MTTR = 3, Dev. time = 13) point as well.) Again, solving (5) gives a solution, $a_1 = 0.66 > 0$, $b_1 = 1.56 > 0$, where *increasing* MTTR and *increasing* development time yields higher utility. The concavity issue will be further discussed in Section 8.

However, a third, important, observation can also be made. The subjects in this elicitation did not exhibit any dominating/dominated hypotheticals. With just two subjects, it cannot be ruled out that this is a mere coincidence, but it does not seem like an unreasonable hypothesis to believe that this is due to a better understanding of the IT service which is the core business of the subjects to develop and sell, compared to the students in elicitation 1, who are mere users and customers, and can have preferences that are not necessarily well founded. One of the subjects explicitly stated this hypothesis after the elicitation, noting that "To us, this is not just an abstraction", when being told about the prevalence of dominating/dominated hypotheticals among the student subjects. (Of course, the credibility of the hypothesis is not substantially strengthened by such a remark that might obviously be subject to self-serving bias.)

7.4.1. Subject feedback

Table 2: Quantitative subject feedback, on 5 point Likert scale, from elicitation 2.

Question:	How interesting did you find the task?	How clear did you find the task?	How relevant and helpful do you think the graphical representation was for solving the task?
	5	4	5
	3	5	3
Mean = Median:	4	4.5	4

Turning to the feedback form answered by subjects after the elicitation was complete, both subjects completed it. The quantitative results are given in Table 2. It is clear that subjects were reasonably happy with the task and the graphical representation. Just a single free text comment was submitted, noting that some trade-offs are more easy or evident than others.

8. Discussion

The preceding sections have discussed how to elicit preferences with regard to non-functional properties (Section 4), how to ensure the consistency of these preferences (Section 5), and how the results of such elicitations might look like in practice (Section 7). The GUI and mathematical apparatus allow relatively powerful utility models to be built from relatively straight-forward user input. These utility models, in turn, offer solutions to the trade-off problem introduced in Section 1. However, there are also limitations, to be discussed in the following.

8.1. The limits of automatic identification of utility functions

In the original conference article, it was argued that the use of least squares solutions is a strength of the method proposed, as it allows for letting many stake-holders enter data independently, and then construct the utility function based on all inputs. This might be a way to get rid of some individual biases in order to construct collectively valid utility models. Based on the empirical results in Section 7, this vision must be complemented with a few more caveats.

First, the prevalence of dominating/dominated hypotheticals, illustrated in Fig. 6, shows a need to either (i) disregard some subject input completely (as was the case in the rest of the analysis in Section 7), or, preferably, (ii) to capture spurious results during elicitation and thereby force the subject to revise those preferences.

Second, the choice of proper utility functions is probably more important than foreseen in the original conference article. In particular, the elicitations described in Section 7 highlighted the risk of misidentifying the directions of

what is good and what is bad for the properties involved. Thus, there seems to be a need for more tailoring and application of specifically suitable utility functions for various properties, rather than just generic ones like those introduced in Section 3. Sections 8.2 and 8.3 therefore discuss the question of specifically suitable utility functions further along two different lines.

8.2. Concave utility curves

The observation that Figs. 7(c) (in particular) and 8(c) look concave merits us revisiting the discussion in the original conference article about *utility functions* as opposed to *production functions*.

Each of the three functions introduced in Section 3 can also be interpreted as a production function. Whereas a utility function is designed to model preferences over bundles of goods, with level curves corresponding to a decision-maker being indifferent between bundles, a production function is designed to model output from combinations of inputs, with level curves corresponding to the same amount of output from combinations of input.

In the case of production functions, there are good arguments why they are often convex, or more precisely, why their input requirement sets (the set of input goods $\mathbf{x}$ from which it is feasible to produce output good y) are often convex [16]. One is the *replication* argument. If there are two technologies available which can produce y from different input vectors $\mathbf{x}_1$ and $\mathbf{x}_2$, then to produce $100y$, we could do it either with $100\mathbf{x}_1$ or with $100\mathbf{x}_2$. But, crucially, we could also do it by applying technology 1 to $50\mathbf{x}_1$ and technology 2 to $50\mathbf{x}_2$. Taking this argument to its limit, we could produce $100y$ by *any* combination of $100t\mathbf{x}_1$ and $100(1-t)\mathbf{x}_2$ for $t \in [0,1]$. This is precisely convexity of the input requirement set. There is also a *temporal version* of the replication argument, where output y is considered per time (e.g. per month), and fractional use of technologies 1 and 2 means using the one during the first part of the month and the other during the second part of the month. If this is feasible, and switching costs are negligible, then again this translates into convexity of the input requirement set. Clearly, both versions of the argument depends on scale –

for a small production operation, the discreteness of inputs makes the replication argument less convincing. It is not always meaningful to speak of fractions of production machines, trucks or people.

These arguments do not necessarily hold for utility functions. Nevertheless, concave preferences are typically considered relatively unrealistic for ordinary goods [23]. In the context of non-functional properties, the replication argument is much less credible. First, the number of feasible technological solutions is likely to be relatively small, so that their discreteness becomes important. Second, in the temporal version, it seems unreasonable to assume that switching costs are negligible.

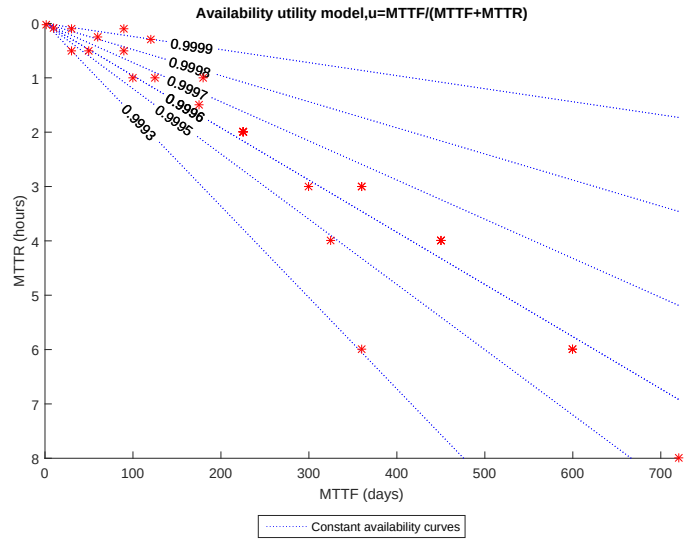
However, and most importantly, non-functional properties are *not* ordinary goods. The MTTF, MTTR, and Mean development time of a piece of software cannot be consumed independently. They come together – in this sense, they are not a bundle of goods, but rather aspects of a single good. Whereas it is in principle fully possible to buy any amount of a factor input such as machinery and try to combine it with any amount of another factor input like manpower for production, combinations of non-functional properties in software are much more constrained and intertwined. As a corollary, whereas with traditional production functions the prices on the factor inputs can be used to determine economically optimal combinations, this is not feasible with non-functional properties. Security cannot be purchased at a particular price per unit, to be combined with performance efficiency purchased at another price per unit and maintainability purchased at a third price. The observation that non-functional properties are *not* ordinary goods sets the stage for the next section, where we discuss some theoretical and practical aspects of trade-offs between such properties.

8.3. Trade-offs between non-functional properties

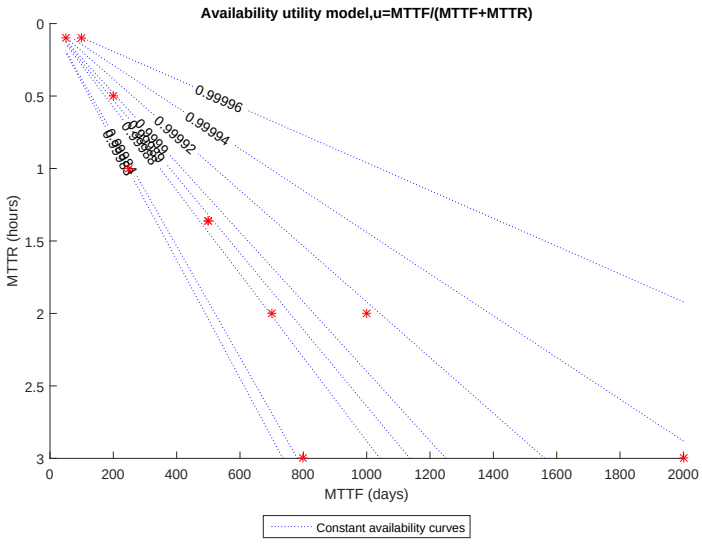
As opposed to factor inputs, which can be freely combined, non-functional properties are aspects of a single good. Feasible designs, giving rise to feasible combinations of non-functional properties, are generated in a design process.

These product alternatives are based on what is deemed feasible from the engineering perspective, and these are the points to be evaluated using the utility functions. As remarked by one of the referees, it is important to make the distinction between *preferences* on the one hand, and *feasible designs* on the other. The preferences need not be constrained by what is deemed technologically feasible at the time, whereas this is precisely the role of designs. This is illustrated by the hotel example in Section 7.1 – a utility curve can be found from hypothetical alternatives even if there are no real hotels corresponding to those alternatives. New technology (or new zoning laws, regulations of real estate investment, etc., in the example) could make such alternatives feasible in the future. Therefore, it is prudent to elicit such preferences beforehand, and allow such preferences to be based also on non-technically feasible alternatives. Thus, the goal is to first elicit these utility functions, and subsequently use them to evaluate different design alternatives. Indeed, if users supply their preferences at an early stage in the life cycle of the would be product, then not only are the alternatives for which they do so hypothetical, but it might also be unknown where, precisely, the border between feasible and infeasible technical solutions is located. This will become (approximately) known only as part of the more elaborate design and engineering work that takes place over the course of the product development. Nevertheless, the preferences expressed in a utility curve can be very useful to guide this product development towards interesting (abstract) regions in the solution space for exploration. Conversely, it can also guide the product development away from solutions that do not correspond user preferences.

To further improve our understanding of trade-offs between non-functional properties, it is useful to consider the elicited MTTF-MTTR trade-offs in greater detail, as graphically illustrated in Figs. 7(a) and 8(a). The reason is that there is a natural candidate for a utility function here, viz. the *availability* resulting from particular combination of MTTF and MTTR. More specifically, the *steady state availability* can be computed as MTTF divided by the total time of operation,



(a) Elicitation 1: University course.



(b) Elicitation 2: Integration service provider.

Figure 9: Elicitation results for the MTTF-MTTR trade-offs plotted together with constant availability curves.

which consists of MTTF plus MTTR:

$$u(\text{MTTF}, \text{MTTFR}) = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}} \tag{23}$$

The result is a fraction, which approaches 100% availability in the limit where MTTF $\to \infty$ (failures never occur) or MTTR $\to 0$ (failures are immediately restored). In practice, of course, availability never reaches 100%, but is often measured as the number of 'nines', such as four (99.99 %), five (99.999 %), or six (99.9999 %).

It is illuminating to plot the elicited MTTF-MTTR trade-offs together with this utility function as seen in Fig. 9(a) for elicitation 1, and Fig. 9(b) for elicitation 2.

We noted above that inter-subjects disagreement is particularly evident in the cases of greater MTTF values. The availability plots put this observation into perspective. *If* availability is the relevant utility, it becomes clear that whether a difference in MTTF or MTTR values should be seen as large or small is highly dependent on the actual values of those properties. Graphically, this is reflected in the closeness of the availability levels in the upper left corner: when both MTTF and MTTR are small, small changes in their values have a large impact on availability. Conversely, towards the lower right corner, availability levels are further apart, so that larger changes in the MTTF and MTTR values are required to have the same impact on availability, i.e. to 'jump' from one availability level curve to the next.

Mathematically, we can describe this by applying (9), as before, to (23):

$$\frac{\partial u}{\partial \text{MTTF}} = \frac{\text{MTTR}}{(\text{MTTF} + \text{MTTR})^2} \tag{24}$$

$$\frac{\partial u}{\partial \text{MTTR}} = -\frac{\text{MTTF}}{(\text{MTTF} + \text{MTTR})^2} \tag{25}$$

$$\frac{d\text{MTTR}}{d\text{MTTF}} = -\frac{\frac{\partial u}{\partial \text{MTTF}}}{\frac{\partial u}{\partial \text{MTTR}}} = \frac{\text{MTTR}}{\text{MTTF}} \tag{26}$$

(26) gives the rate of substitution between MTTR and MTTF, under the assumption that availability is the relevant utility. Applying it, for example, in

the lower right corner of Fig. 8(a), where MTTF= 2 000 days and MTTR= 3 hours we find that the infinitesimal rate of substitution from MTTF to MTTR is just $3/(24 \cdot 2\,000) = 1/16\,000$. Conversely, the infinitesimal rate of substitution from MTTR to MTTF is 16 000.

To take a concrete, non-infinitesimal example, going from where MTTF= 800 days to MTTF= 2 000 days while holding MTTR fixed at 3 hours (i.e. the large discrepancy at the bottom of the graph) increases availability from 99.984% to 99.994%. But moving between these availability levels at the baseline MTTR of 1.36 hours corresponds to a move from MTTF= 363 days to MTTF= 907 days. At the very small MTTR of 0.1 hours (6 minutes) which both subjects used as a hypothetical, moving between the same availability levels corresponds to a move from MTTF= 27 days to MTTF= 67 days. This move of 40 days is actually *smaller* than the actual discrepancy between subjects recorded at MTTR = 0.1 hours, where one subject set MTTF= 50 days and the other set MTTF= 100 days (as illustrated in the upper left corner of Fig. 8(a)).

Thus, *if* availability is the relevant utility, inter-subject agreement is not worse for greater MTTF values than for smaller ones. The graphical depiction with the boundary plot seen in Fig. 8(a) is in this sense a bit misleading, since MTTF exhibits so steeply diminishing returns in terms of availability, as seen in (24).

But *is* availability the relevant utility? The preceding reasoning uses that assumption as an illuminating starting point, but that does not make it true. On the contrary, there is reason to believe that true utilities of combinations of MTTF and MTTR exhibit some deviations from the strict availability perspective. This is empirically indicated in Figs. 7(a) and 8(a), where subjects apparently have deemed hypotheticals on different availability levels equally good. However, as argued in [24], it is also theoretically justified if the costs of downtime are not always the same, or if costs depend on outage duration, or if there is a fixed cost for every outage. For example, 99.9% availability 24 hours a day, 7 days a week, means almost 9 hours of annual downtime. If all downtime costs the same, availability might be the relevant utility. But if the costs of

downtime vary, as is the case e.g. for downtime in a retail payment system, 100 separate 5 minute outages might be preferable to a single 9 hour outage, because it evens out the costs and limits the risk of a long outage at a critical time, such as sales just before Christmas. Conversely, if the IT service in question controls a physical industrial process such as a paper mill, then every outage, no matter how short it is, comes with a significant cost, and the single 9 hour outage might be the preferable. Thus, for a given availability level, different combinations of MTTF and MTTR could be preferred by different stakeholders in different contexts, meaning that the equation of utility with availability is overly simplistic. Nevertheless, availability could serve as a starting point for an appropriate utility function, to be modified as diverging preferences are revealed.

However, it is not always the case that deviations from the utility equals availability baseline are systematic and consistent. Franke and Buschle used an experimental economics approach to investigate preferences for different hypothetical availability service level agreements among enterprise IT professionals [25]. They found a surprising number of non-monotonic preferences, i.e. choices that could not straightforwardly be explained by systematic deviations from an expected value maximizing behavior in a risk-seeking or risk-averse direction. Such preferences are challenging to capture and model descriptively, but also offer an opportunity for prescriptive corrections as part of an elicitation system, where unreasonable preferences can be weeded out already as the user enters them.

To summarize the preceding discussion, it is clear that it is challenging to find appropriate utility functions for products with different sets of non-functional properties. Even in a seemingly straightforward case such as the MTTF-MTTR trade-off, with a strong theoretical candidate for a utility function, unexpected difficulties turn out. Nevertheless, considering more trade-offs, and explicitly cataloging utility function candidates along with their strengths and weaknesses is expected to be rewarding in terms of new insights.

9. Conclusion and future work

This paper shows a first attempt to employ elicitation of utility functions with regard to non-functional properties of software components from stakeholders. Feasibility and usability of the approach has been demonstrated by two exploratory elicitations conducted on students and practitioners. Based on these elicitations, the following empirical claims can, tentatively, be made:

Dominating/dominated hypotheticals: Many subjects exhibit preferences that, taken at face value, are not consistent. This calls for more elaborate tool support to capture spurious preferences as they are entered, guiding users towards consistent preferences.

Utility functions: It is not clear that standard utility functions (such as the linear, Cobb-Douglas, and Leontief functions investigated here) are always sufficient to properly describe utility functions of non-functional properties. This is theoretically reasonable as non-functional properties are not ordinary goods, and the empirical observation that some utility functions elicited are concave lends some support to this conclusion. However, it must be stressed that this warrants further empirical research. From a theoretical point of view, property-specific utility functions such as *steady state availability*, might be better suited starting points to find suitable utility functions for non-functional properties.

Inter-subject disagreement: The elicited preferences exhibit considerable inter-subject disagreement. This confirms the research goal to offer tool support that can guide users towards mutually consistent preferences.

Problem understanding: The elicitation results can be interpreted as to indicate that a considerable number of subjects find the trade-off problem difficult to understand. The problem of dominating/dominated hypotheticals is one aspect of this, but the problem seems to extend beyond this (as shown by the occurrence of not only systematically, but also randomly,

spurious preferences). On an abstract level, this is in line with other results [25].

These observations point in the direction of several enhancements that could be interesting future work.

The primacy of the stake-holders must be stressed – the mathematical apparatus is just a tool, not an end in itself. Therefore, if stake-holders are not satisfied with the solutions, it is important that a future mature decision-support system allows for iterative solutions, where preferences once elicited are possible to change and revise. Furthermore, such iterative elicitation also makes sense from the perspective of avoiding unreasonable preferences, such as the ones illustrated in Fig. 6 or reported in [25]. A mature preference elicitation and modeling system for non-functional properties should thus include more decision-support to the user at the elicitation stage. The use of graphical depictions of indifference curves is a good start, as judged by the subjects represented in Tables 1 and 2, but can surely be improved, e.g. by allowing for alternative representations and user-defined filtering options, and complemented with other alternatives, such as heat maps for visually guiding the stake-holder in the elicitation process. It would also be interesting to investigate how qualitative preferences, not based on utility functions but comparisons only, fare – this might be more intuitive to users. Another possible improvement would be to allow certain stakeholders a bigger say for some properties, e.g. a chief security officer might need veto power in some areas. In this respect, it would certainly be useful to provide support for multiple stake-holder roles entailing different levels of editing and visualization rights.

A more conceptual observation is that there is a need for better characterization of trade-off preferences, not just in terms of standard utility functions, but in terms of functions specifically tailored to suit particular properties. These might have non-standard features, such as concavity. The discussion of the utility function defined by the availability metric in Section 8.3 is one example of a more specific utility function, but more work is clearly needed here. Such work

would also shed more light on when it is appropriate to use elicitation from one vs. several indifference curves.

Additionally, several interesting future research directions are suggested by the results. A first such question has to do with how property trade-off preferences look like in practice, i.e. by doing more empirical elicitations of the kind reported in Section 7. Such data would not only be interesting in its own right, but could also be used to investigate whether preferences are consistent across individuals, stake-holder types and roles, companies, or even entire industry sectors such as automotive or telecom.

Another interesting research direction not previously addressed concerns *uncertainty* in the property values, particularly relevant if these are based on estimates. What if, for instance, one property estimate has a large confidence interval, but another one has a small one? How should they then be traded-off against each other? Effective ways to elicit and model uncertainty are definitely needed. Other interesting questions also stem from considering uncertainty, such as the risk appetite of decision-makers and how that should best be elicited.

Acknowledgment

The work is partially supported by a research grant for the ORION project (reference number 20140218) from The Knowledge Foundation in Sweden. The authors would like to thank Efi Papatheocharous, Kai Petersen, and Séverine Sentilles for beneficial discussions. Thanks are also due to four anonymous referees from the EDOC conference for very good comments on the conference version of the paper, and to two anonymous referees for equally insightful and relevant comments on the journal version.

References

[1] C. Haskins, K. Forsberg, M. Krueger, D. Walden, D. Hamelin, Systems engineering handbook, International Council on Systems Engineering, 2006.

[2] International Organization for Standardization, Systems and software engineering – Systems and software Quality Requirements and Evaluation (SquaRE) – System and software quality models, International standard ISO/IEC 25010:2011(E), International Organization for Standardization, 2011.

[3] U. Franke, Towards preference elicitation for trade-offs between non-functional properties, in: IEEE 20th International Enterprise Distributed Object Computing Conference (EDOC 2016), IEEE, 2016, pp. 89–98. doi:10.1109/EDOC.2016.7579389.

[4] J. L. King, E. L. Schrems, Cost-benefit analysis in information systems development and operation, ACM Computing Surveys 10 (1978) 19–34.

[5] S. Sedigh-Ali, A. Ghafoor, R. A. Paul, Software engineering metrics for COTS-based systems, Computer 34 (2001) 44–50.

[6] N. A. Maiden, C. Ncube, Acquiring COTS software selection requirements, Software, IEEE 15 (1998) 46–56.

[7] V. Cortellessa, F. Marinelli, P. Potena, An optimization framework for "build-or-buy" decisions in software architecture, Computers & Operations Research 35 (2008) 3090–3106.

[8] H.-W. Jung, B. Choi, Optimization models for quality and cost of modular software systems, European Journal of Operational Research 112 (1999) 613–619.

[9] O. Berman, M. Cutler, Optimal software implementation considering reliability and cost, Computers & operations research 25 (1998) 857–868.

[10] B. Zachariah, R. Rattihalli, A multicriteria optimization model for quality of modular software systems, Asia-Pacific Journal of Operational Research 24 (2007) 797–811.

[11] V. S. Lai, B. K. Wong, W. Cheung, Group decision making in a multiple criteria environment: A case using the AHP in software selection, European Journal of Operational Research 137 (2002) 134–144.

[12] C.-C. Wei, C.-F. Chien, M.-J. J. Wang, An AHP-based approach to ERP system selection, International journal of production economics 96 (2005) 47–62.

[13] T. Neubauer, C. Stummer, Interactive decision support for multiobjective COTS selection, in: System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on, IEEE, 2007, pp. 283b–283b.

[14] J. Michanan, R. Dewri, M. J. Rutherford, Understanding the power-performance tradeoff through Pareto analysis of live performance data, in: Green Computing Conference (IGCC), 2014 International, IEEE, 2014, pp. 1–8.

[15] M. Österlind, P. Johnson, K. Karnati, R. Lagerström, M. Välja, Enterprise architecture evaluation using utility theory, in: Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2013 17th IEEE International, IEEE, 2013, pp. 347–351.

[16] H. R. Varian, Microeconomic analysis, WW Norton, 1992. 3rd edition.

[17] A. Tversky, Intransitivity of preferences., Psychological review 76 (1969) 31–48.

[18] H. E. Bray, Rates of exchange, The American Mathematical Monthly 29 (1922) 365–371.

[19] T. L. Saaty, A scaling method for priorities in hierarchical structures, Journal of mathematical psychology 15 (1977) 234–281.

[20] K. M. Dadkhah, F. Zahedi, A mathematical treatment of inconsistency in the analytic hierarchy process, Mathematical and computer modelling 17 (1993) 111–122.

[21] J. Axelsson, A. Kobetski, Z. Ni, S. Zhang, E. Johansson, Moped: A mobile open platform for experimental design of cyber-physical systems, in: 2014 40th EUROMICRO Conference on Software Engineering and Advanced Applications, 2014, pp. 423–430. doi:10.1109/SEAA.2014.38.

[22] E. Papatheocharous, U. Franke, Decision-making in automotive software development – An observational study, in: The 15th International Conference on Intelligent Software Methodologies tools and Techniques (SoMeT 2016), IOS, 2016, pp. 59–68. doi:10.3233/978-1-61499-674-3-59.

[23] J. Hey, Intermediate Microeconomics: People are Different, McGraw-Hill, 2003.

[24] U. Franke, Optimal IT Service Availability: Shorter Outages, or Fewer?, IEEE Transactions on Network and Service Management 9 (2012) 22–33.

[25] U. Franke, M. Buschle, Experimental evidence on decision-making in availability service level agreements, IEEE Transactions on Network and Service Management 13 (2016) 58–70.