Decision Making Over Combinatorially-Structured Domains

Andrea Martin and K. Brent Venable
Department of Computer Science, Tulane University, LA, USA
amartin@tulane.edu and kvenabl@tulane.edu

Abstract

We consider a scenario where a user has to make a set of correlated decisions and we propose a computational modeling of the deliberation process. We assume the user compactly expresses her preferences as soft constraints. We design a sequential procedure that uses Decision Field Theory to model the decision making on each variable. We compare our sequential approach to one where a single decision is made over the set of all alternatives on randomly generated tree-shaped Fuzzy Constraint Satisfaction Problems. Our preliminary result are promising: they show that the sequential approach outperforms the non-sequential one in terms of execution time and returns choices of similar quality.

Introduction and Motivation

Preferences are a very important notion in decision making. As such they have been studied in multiple disciplines such as psychology, philosophy, business and marketing.

Recently, preference reasoning has grown into an important topic in computer science and specifically in Artificial Intelligence. Preference models are currently used in many AI applications, such as scheduling and recommendation engines.

On the other hand, in psychology, Decision Field Theory (DFT) (J.R. Busemeyer 1993) formalizes the process of deliberation in decision making. In fact, when a person is confronted with a decision, she will anticipate each course of action and will try to evaluate all the possible consequences according to different criteria. While DFT has mainly tackled the problem of making a single decision, both in real life and in artificial intelligence applications, decision are more complex and it can be helpful to organized them in a combinatorial structure over which decisions can be applied sequentially. There are several approaches to modeling preferences compactly (Rossi, K.B. Venable, and Walsh 2011), such as, for example, soft constraints (Meseguer, Rossi, and Schiex 2005). In this paper, we focus on the later, a formalism in which variables are assigned values from their domains, and there are constraints, involving subsets of variables and associating to simultaneous assignments of the constrained variables a preference value. We use soft constraints to support a deliberation process performed through DFT and we consider a sequential approach, where deliberation is applied to each variable. In our framework, preferences over the set of alternatives according to a criteria (called attribute, following the DFT literature) are decomposed into a fuzzy constraint satisfaction problem (F CSP). Each attribute is associated with a FCSP. We test our approach on randomly generated tree-shaped FCSPs assuming two attributes and setting DFT parameters in a standard way. We compare the results obtained applying the deliberation process sequentially to those obtained by a single deliberation step over the entire combinatorial domain. As expected the sequential approach outperforms the one-shot procedure in terms of time. Moreover, it focuses on a relatively small set of alternatives compared to the size of the choice space and the deliberated assignments are, on average, of high quality with respect to the preferences in the soft constraint problems representing the attributes. Our work has two objectives: the first one is to provide a computational model which can help understand human decision making over complex domains; the second one is to investigate DFT as means of incorporating a form of uncertainty into the soft constraint formalism. To the first end, we note that our sequential approach appears to be cognitively more plausible as it is far more likely for humans to break complex decisions into interconnected sub-problems, rather than try to deliberate directly on a large set of complex objects. The next step on our agenda in this respect is to study the effect of decomposing complex deliberations in terms of phenomena which have been observed in behavioral studies of human decision making, such as the similarity, attraction and compromise effect, and that DFT has been shown to model effectively(R. Roe 2001). In terms, of using DFT as a paradigm to extend soft constraints, we note that the DFT simulation of deliberation, which is modeled as an oscillation between evaluation criteria for different scenarios, can be used to model uncertainty about which are the true preferences. Furthermore, DFT allows to specify a similarity measure between options and to express how a high preference on an option can influence the preference of similar or dissimilar options. Both aspects extend the expressive power of soft constraints where preferences are assumed to be known (or static), and those of different options influence each other only through shared variable assignments.

This paper is organized as follows. The first section reviews the basic notions regarding soft constraints and multialternative decision field theory (MDFT). The second section presents the description of sequential decision making over soft-constraint networks exemplified on a detailed running example. The third section describes non-sequential decision making on the same running example. The last two sections of the paper, before concluding, present experimental results and a discussion of related work.

Background

In this section, we introduce some basic notions that will be useful in the rest of the paper. First, we will define soft constraint problems. Then, we will briefly describe multialternative decision field theory.

Soft Constraints A soft constraint (Meseguer, Rossi, and Schiex 2005) requires a set of variables and associates each instantiation of its variables to a value from a partially ordered set. More precisely, the underlying structure is a c-semiring which consist of the following, $\langle A, +, \times, 0, 1\rangle$, where A is the set of preference values, $+$ induces an ordering over A (where $a \leq b$ iff $a + b = b$), $\times$ is used to combine preference values, and 0 and 1 are respectively the worst and best element. A Soft Constraint Satisfaction Problem (SCSP) is a tuple $\langle V, D, C, A\rangle$ where V is a set of variables, D is the domain of the variables and C is a set of soft constraints (each one involving a subset of V) associating values from A. An instance of the SCSP framework is obtained by choosing a specific preference structure. Choosing $S_{FCSP} = \langle [0, 1],\ max, min,\ 0,\ 1\rangle$ means that preference values are in $[0, 1]$, we want to maximize the minimum preference value, the worst preference value is 0 and the best preference value is 1. This is the setting of fuzzy CSPs (FCSPs) (Schiex 1992; Meseguer, Rossi, and Schiex 2005), that we will use this paper. Figure 1 shows the constraint graph of an FCSP where $V = \{X, Y, Z\}$, $D = \{a, b\}$ and $C = \{c_X, c_Y, c_Z, c_{XY}, c_{YZ}\}$. Each node models a variable and each arc models a binary constraint, while unary constraints define variables' domains. For example, c_Y is defined by the preference function f_Y that associates preference value 0.4 to $Y = a$ and 0.7 to $Y = b$. Default constraints such as c_X and c_Z, where all variable assignments get value 1, will often be omitted in the following examples.

Solving an SCSP consists of finding the ordering induced by the constraints over the set of all complete variable assignments. In the case of FCSPs, such an ordering is a total order with ties. In the example above, the induced ordering has $(X = a, Y = b, Z = b)$ and $(X = b, Y = b, Z = b)$ at the top with a preference value 0.5, $(X = a, Y = a, Z = a)$ and $(X = b, Y = a, Z = a)$ just below with 0.4, and all others tied at the bottom with preference value 0.2. Often, however, solving an SCSP is interpreted as finding an optimal solution, that is, a complete assignment with an undominated preference value (thus $(X = a, Y = b, Z = b)$ or $(X = b, Y = b, Z = b)$ in this example). Given a variable X, we write $s_{\downarrow X}$ to denote the value of X in s. Unless certain restrictions are imposed, such as a tree-shaped constraint graph, finding an optimal solution is an NP-hard problem.

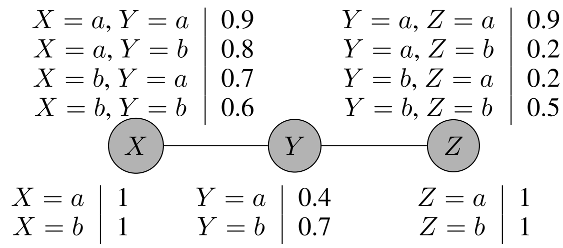


Figure 1: Example of an FCSP.

Constraint propagation may improve the search for an optimal solution. In particular, given a variable ordering o, a FCSP is directional arc-consistent (DAC) if, for any two variables x and y linked by a fuzzy constraint, such that x precedes y in the ordering o, we have that, for each a in the domain of x, $f_x(a) = max_{b \in D(y)}(min(f_x(a), f_{xy}(a, b), f_y(b)))$, where f_x, f_y, and f_{xy} are the preference functions of c_x, c_y and c_{xy}. When the problem has a tree-shape and the variable ordering is compatible with the father-child relation of the tree, DAC is enough to find the preference level of an optimal solution (Meseguer, Rossi, and Schiex 2005). Such an optimal preference level is the best preference level in the domain of the root variable. To find an optimal solution, it is then enough to perform a backtrack-free search which instantiates variables in the same order used for DAC.

In our running example, if we choose the variable ordering $\langle X, Y, Z\rangle$, achieving DAC means first enforcing the property over the constraint on Y and Z and then over the constraint on X and Y. The first phase modifies the preference value of $Y = b$ to $max(min(f_Y(b), f_{Y,Z}(b, a), f_Z(a)), min(f_Y(b), f_{Y,Z}(b, b), f_Z(b))) = max(min(0.7, 0.2, 1), min(0.7, 0.5, 1)) = max(0.2, 0.5) = 0.5$. Similarly, the second phase sets the preference values of both $X = a$ and $X = b$ to 0.5.

We note that, by achieving DAC w.r.t. ordering o, we obtain a total order with ties over the values of the first variable in o, where each value is associated to the preference of the best solution having such a variable instantiated to such a value. In our running example, achieving DAC brings both values of X in a tie and in $top(X, P)$, with preference value 0.5, that is the preference value of an optimal solution.

Multialternative Decision Field Theory Decision Field Theory attempts to formalize the deliberation process by assuming that a decision maker's preference for each option evolves during deliberation and by integrating a stream of comparisons of evaluations among options on attributes over time (J.R. Busemeyer 1993). DFT has been extended to multialternative preferential choice, where settings with more than two options are considered. In DFT a valence value $v_i(t)$ is associated with a choice to be made at any moment in time t, which represents the advantage or disadvantage of option i when compared with other options with respect to some attribute(R. Roe 2001). For example, options could be different car models with attributes being fuel efficiency and comfort. The valence vector, $V(t) = CMW(t)$ is a product of three matrices (M,$W(t)$ and C), has a dimension equal to the number of options and represents the valence of each option. The first matrix, M, contains the personal evaluation of each option with respect to the attributes. The second component is a vector of attention weights, $W(t)$, allocated to each attribute at a particular moment in time. The third and final matrix C contains parameters describing how to aggregate the evaluation of an option with the evaluation of other options in order to obtain its advantage (or disadvantage). Furthermore, at any moment in time, each alternative is associated with a preference strength $P(t)$. Strength for alternative i at time t, denoted $P_i(t)$ represents the integration of all the valences considered for alternative i from the start of the deliberation process to time t. A new state of preference $P(t+1)$ is formed at each time step from the previous preference $P(t)$ and the new input valence vector, $V(t)$, as follows: $P(t+1) = SP(t) + V(t+1)$. In this formula, matrix S models how the preference of one option influences the preference of another option. For example, one can assume a higher (negative) interaction among options which are very similar. We provide a detailed example in the next section.

Sequential decision making over soft constraint networks

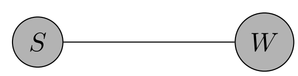
We assume a set of correlated decisions to be made, $X = \{X_1, \ldots, X_n\}$, where each variable X_i can take different values from its domain $D(X_i) = \{\sigma_1, ..., \sigma_m\}$. To represent the agent's personal evaluation we use a FCSP defined over the variables in X. We use one FCSP for each attribute. In the car example mentioned above, variables would describe different car models, and could be, for example, brand, engine type and number of seats. Then, we would have a FCSP for each attribute, that is one describing the preferences in terms of fuel efficiency and the other in terms of comfort. The preference values are used to populate the M matrix at each step of the decision process. We recall that a FCSP is associated with a graph where nodes correspond to variables and edges to constraints. As an initial step we consider FCSPs where the constraint-graph is tree-shaped. This allows us to topologically sort the variables in an ordering which we denote by $O = X_1 > X_2 > ..... > X_n$. The idea is to sequentially find a value for each variable X_i via a DFT deliberation process following order O. The sequential procedure is a sequence of n steps, where at each step i:

1. We extract the subjective preference of the user on the values in the domain of X_i. To do this, we will enforce DAC on the FCSP, in reverse order w.r.t. O.
2. Then, DFT is applied to X_i, returning a deliberated assignment for variable X_i, say σ_i. We write this as: $DFT(X_i) = \sigma_i$.
3. Finally, DAC is applied to propagate the effect of the assignment in both soft constraint networks following order O.

After n steps have been executed, we obtain a final combinatorial decision, that is, we will have selected one value for each variable. One may wonder why step 1 is necessary. As we have mentioned in the background section on soft constraints, the effect of applying DAC on a tree-shaped Fuzzy CSP is to have the preferences on the domain values of each variable coincide with the highest preference of a solution containing that assignment. Intuitively, this means that the new preferences take into account the information induced on that assignment by the rest of the network. For example, one may in general prefer a certain car brand, but, if they only have two-seaters available and she is looking for a family car, then this should impact her preferences when deliberating on the brand. If we were to skip this step then the deliberation process would be using preferences which are not realistic, that is, are completely disconnected from what is actually achievable given the rest of the network.

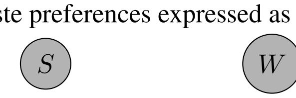
We will now describe the procedure in more detail using an example. Assume we have a user who has to decide on what to have for dinner. His preferences in terms of attributes taste and health on the available options are expressed by two FCSPs depicted respectively in Figure 2 and Figure 3.

$S=v, W=r$	1
$S=v, W=w$	0.3
$S=m, W=r$	0.7
$S=m, W=w$	0.4



$S=v$	0.7	$W=r$	1
$S=m$	1	$W=w$	0.8

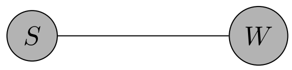
Figure 2: Taste preferences expressed as a Fuzzy CSP.



$S=v$	0.8	$W=r$	0.7
$S=m$	0.4	$W=w$	0.3

Figure 3: Health preferences expressed as a Fuzzy CSP.

$S=v, W=r$	1
$S=v, W=w$	0.3
$S=m, W=r$	0.7
$S=m, W=w$	0.4

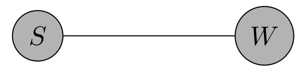


$S=v$	0.7
$S=m$	0.7

$W=r$	1
$W=w$	0.8

Figure 4: Taste preferences expressed by a FCSP after achieving DAC.

$S=v, W=r$	1
$S=v, W=w$	0.3
$S=m, W=r$	0
$S=m, W=w$	0



$S=v$	0.7
$S=m$	0

$W=r$	1
$W=w$	0.7

Figure 5: Taste preferences expressed by a FCSP after applying DFT to S, and assuming $S=v$ is deliberated.

Both FCSPs are defined on same set of variables, that is, Soup, denoted by S, with domain $\{m, v\}$ for meat and vegetable respectively and Wine, denoted with W, with values $\{w, r\}$ for white and red. The user, for example, prefers meat soup to vegetable soup and prefers red wine with it (see Figure 2 for all the values of preferences). On the other hand, the user is also aware tha vegetable soup is healthier than the meat soup and red wine is healthier than white wine.

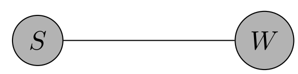
In step 1 of our procedure, the user will achieve DAC on both graphs. This, for example, changes the preferences of the user over Soup. First, it modifies the taste preference value of $S = v$ to maximum preference of any complete assignment involving it, that is, to $max(min(0.7, 1, 1), min(0.7, 0.3, 0.8)) = 0.7$. Similarly, it modifies the preference value of $S = m$ to 0.7.

The taste FCSP is already DAC since there is no constraint between the two variables. In step 2 DFT is applied to S using the four preference values from the two FCSPs as input to matrix M_S. Let us assume, now, that DFT(S) returns $S = v$. Then, all of the preferences tuples with $S = m$ are set to are set to 0 in both FCSPs and the effect is propagated to all the network through DAC following O (see Figure 4 and Figure 5).

This step modifies the preference value of $W = r$ to $max(min(1, 1, 0.7), min(1, 0, 0)) = 0.7$ and then modifies the preference value of $W = w$ to $max(min(0.7, 0.3, 0.7), min(0.7, 0, 0)) = 0.3$ (see Figure 6). Finally, DFT is applied to W with inputs 0.7 and 0.3 and the final result is reached. For example, if $W = r$ is deliberated then, overall, we would have chosen to have vegetable soup and red wine.

Let us now focus on the DFT process performed in Step 2 for variable S. Intuitively, this means that we will simulate the deliberation process carried out by the user when choosing between vegetable and meat soup. This is done by assuming she alternates between considering the options in terms of taste and health and converges to a final preference

$S=v, W=r$	1
$S=v, W=w$	0.3
$S=m, W=r$	0
$S=m, W=w$	0



$S=v$	0.7
$S=m$	0

$W=r$	0.7
$W=w$	0.3

Figure 6: Taste preferences after the second round of DAC, right before DFT is applied to W.

after a number of iterations. More specifically, in our example we are considering two alternatives, i.e. m and v, and two attributes , i.e. taste and health. At any moment in time each alternative is associated with a valence value. For example, the choice among soups produces a two dimensional valence vector $V(t) = [v_{S_m}(t), v_{S_v}(t)]$. As mentioned before, this valence vector is determined by three different components. $V(t) = CMW(t)$. We recall that matrix, M, contains the personal evaluation of each option with respect to its attribute, $W(t)$, is the vector containing the weights associated to each attribute at a given point in time t and matrix C contains parameters describing how to aggregate the evaluation of an option with the evaluation the other options.

In this example the matrix C is the same for all variables and is represented by the following values:

$$\mathbf{C} = \begin{bmatrix} 1 & \text{-1} \\ \text{-1} & 1 \end{bmatrix}$$

In other words, the advantage/disadvantage of an option w.r.t. the other is simply the difference between their preference values.

The attention weights, $[W_T(t), W_H(t)]$ for the attention weight matrix $W(t)$ are assumed to fluctuate between $[1, 0]$ and $[0, 1]$ over time steps according to a simple Bernoulli process where we assume that the probability of attending taste (i.e. of picking $[1, 0]$) is 0.55, and health (i.e. $[0, 1]$) is 0.45.

The entries for matrix M are obtained from the preferences over the domain values of S after DAC is performed in step 1. They are:

$$\mathbf{M_S} = \begin{array}{c} \text{T} \quad\; \text{H} \\ \begin{bmatrix} 0.7 & 0.8 \\ 0.7 & 0.4 \end{bmatrix} \end{array} \begin{array}{c} S_v \\ S_m \end{array}$$

Summarizing, if we assume that at time t we have $W(t) = [0, 1]$, then $V(t) = [v_{S_m}(t), v_{S_v}(t)]$, where: $v_{S_m}(t) = W_T(t) \cdot m_{S_mT} + W_H(t) \cdot m_{S_mH} - W_T(t) \cdot m_{S_vT} - W_H(t) \cdot m_{S_vH} = 0 \cdot 0.7 + 1 \cdot 0.4 - 0 \cdot 0.7 - 1 \cdot 0.8 = -0.4$ and $v_{S_v}(t) = W_T(t) \cdot m_{S_vT} + W_H(t) \cdot m_{S_vH} - W_T(t) \cdot m_{S_mT} - W_H(t) \cdot m_{S_mH} = 0 \cdot 0.7 + 1 \cdot 0.8 - 0 \cdot 0.7 - 1 \cdot 0.4 = 0.4$.

In the formula above we have denoted with m_{XY} the entry of matrix M corresponding to option X and attribue Y. A choice among both soups produces a two-dimensional preference state $P(t) = [P_{S_m}(t), P_{S_v}(t)]$, where a new state of preference $P(t + 1)$ is formed at each instant from the previous preference state $P(t)$ and the new input valence vector,$V(t)$, according to the following stochastic equation: $P(t+1) = SP(t)+V(t+1)$, where S is the feedback matrix. We set the parameters for the feedback matrix as follows: the self-connections are set to a high value ($S_{ii} = 0.94$) and the inhibitory connections between the attributes are set to very low values ($S_{s_m,s_v} = S_{s_v,s_m} = -0.001$) because we treat these two alternatives as being very different (for example a spicy, creamy meat soup and a clear, mild vegetable soup).

$$\mathbf{S} = \begin{array}{c} S_v \qquad S_m \\ \begin{bmatrix} 0.95 & \text{-0.001} \\ \text{-0.001} & 0.95 \end{bmatrix} \end{array} \begin{array}{c} S_v \\ S_m \end{array}$$

It is important to mention that we define the parameters in the DFT model in a similar way as the DFT literature (J.R. Busemeyer 1993). In the DFT literature (J.R. Busemeyer 1993) two criteria are considered to decide when to stop the deliberation process. In the first one, a fixed time is set and the option with the highest preference after that time has elapsed is returned. Another possibility is to define a threshold on preferences and to return the first option which passes that threshold. In this paper we consider the first option and we define the number of iterations the deliberation takes place for each variable. In the experiments which we describe later, we set this number to be the same on all variables, since they all have the same domain size.

Non-Sequential approach

Given our domain, another possibility for making an overall decision is to run the deliberation process only once over the set of candidate options consisting of all complete assignments. We believe this is unrealistic as, indeed, DFT is intended to model human decision making under the assumption that the number of possible options is manageable. Nevertheless, we consider this alternative method primarily as a means of comparison with the sequential approach.

We assume a single decision to be made over a combinatorial structure. In other words, each complete assignment to all variables in X is treated as an option, and its preference according to the CSPs is its evaluation with respect to the associated attribute. We evaluate all of complete assignments via the FCSPs and we use such preferences to populate matrix M. The non-sequential procedure consist of two steps:

1. We compute the subjective preferences for all complete assignments for each attribute. For one attribute, this can be implemented, for example, by running DAC on the FCSPs and then, for each complete assignment, taking the minimum preference associated in the FCSP to its single variable assignments.
2. DFT is applied to the set of all complete assignments returning a final deliberation $(X_1 = \sigma_1, X_1 = \sigma_2, \ldots, X_n = \sigma_n)$.

We will now describe the procedure in more detail using the same example with two binary variables and two attributes introduced in the previous section (see Figure 2 and Figure 3). In step 1, the we run DAC on both graphs and we compute the preferences all complete assignments according to both FCSPs. The results are coded into matrix M, as shown below:

$$\mathbf{M} = \begin{array}{c} \text{T} \quad\; \text{H} \\ \begin{bmatrix} 0.7 & 0.7 \\ 0.3 & 0.7 \\ 0.7 & 1 \\ 0.4 & 0.8 \end{bmatrix} \end{array} \begin{array}{c} S_v, W_r \\ S_v, W_w \\ S_m, W_r \\ S_m, W_w \end{array}$$

Then, similarly to what was done for each variable in the case of the sequential procedure, DFT is applied to the set of all complete assignments (4 in this case). The values for the C and S matrices are defined using the same rationale as in the single variable case:

$$\mathbf{C} = \begin{bmatrix} 1 & \text{-1/3} & -1/3 & -1/3 \\ -1/3 & 1 & -1/3 & -1/3 \\ -1/3 & \text{-1/3} & 1 & -1/3 \\ -1/3 & \text{-1/3} & -1/3 & 1 \end{bmatrix}$$

$$\mathbf{S} = \begin{array}{c} \\ \\ \\ \\ \end{array} \begin{array}{cccc} S_vW_r & S_vW_w & S_mW_r & S_mW_w \\ \left[\begin{array}{c} 0.95 \\ -0.005 \\ -0.005 \\ 0 \end{array}\right. & \begin{array}{c} \text{-0.005} \\ 0.95 \\ 0 \\ \text{-0.005} \end{array} & \begin{array}{c} -0.005 \\ 0 \\ 0.95 \\ -0.005 \end{array} & \left.\begin{array}{c} 0 \\ -0.005 \\ -0.005 \\ 0.95 \end{array}\right] \end{array} \begin{array}{c} S_vW_r \\ S_vW_w \\ S_mW_r \\ S_mW_w \end{array}$$

More in detail, we set the parameters for the feedback matrix S as follows: the self-connections are set to a high value (S_{ii} = 0.95) and the inhibitory connection between two options is a negative number obtained by dividing the number of variables that are assigned the same value in both options by the total number of variables. This preserves the intuition that the more two options are similar the smaller the inhibitory effect between them should be.

As far as the attention weights on attributes, they are defined exactly as in the previous section, that is, as fluctuating between $[1, 0]$ and $[0, 1]$ over time steps with probability of attending taste of 0.55 and health of 0.45. The stopping criterion is a limit on the number of iterations. For example, in our experiments, which we describe below, we set this number to be $20 \cdot n$, where n is the number of variables.

Exeprimental results

We have implemented both the sequential and non-sequential decision making approach and we have tested them on randomly generated problems. We consider a setting with two attributes. Thus, each generated instance comprises of a pair of tree-shaped fuzzy problems defined over the same set of binary variables. We consider a number of variables ranging between 2 and 8 with increments of 2 and a constraint tightness of 20%, meaning that, in each constraint, 20% of the tuples are associated with preference 0. For both of the approaches, the values for the DFT matrices are set as described in our running example. In the sequential case, deliberation is stopped after 20 iterations on each variable and in the non-sequential case after $20 \cdot n$ iterations where n is the number of variables.

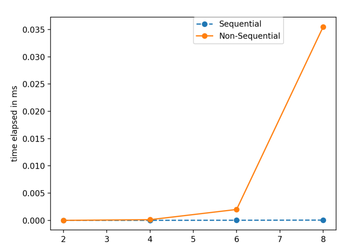


Figure 7: Average execution time for the sequential and non sequential approach when varying the number of variables.

In Figure 7 we show the comparison of the two approaches in terms of running time when varying the number of variables. The plotted results are an average over 10000 runs for each number of variables. For both approaches, we take into account also the time necessary to extract the values for the M matrices from the FCSPs. Not surprisingly, the sequential procedure is faster and the gap between the two running times becomes exponentially larger as the number of variables grows.

Ideally one would want the two approaches to return the same set of choices, modulo the uncertainty inherent in the DFT process. In other words, after running the two approaches the same number of times on a given instance, solutions would ideally be returned as deliberated with similar frequency by both approaches. However, this is not the case, as, by decomposing the decision-making into a set of local deliberation steps, we lose information and we incur in a situation similar to the discursive dilemma in judgment aggregation (Kornhauser and Sager 1986) and sequential voting (Lang and Xia 2009; Pozza et al. 2011).

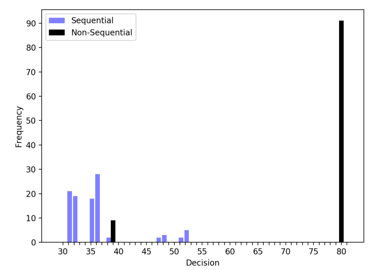


Figure 8: Frequency with which solutions are returned as deliberated in one instance of eight variables.

Figure 8 shows the frequency with which solutions are returned in a problem with 8 nodes after running both approaches 100 times. As we can see, out of the 2^8 possible choices, only 9 are returned by the sequential approach and 2 are returned by the non-sequential one. We performed this same experiment on 100 different instances of 8 nodes and we observed a similar trend. The average size of the set containing solutions returned at least once by the sequential approach is 3.57, over a total of 256, while the size of the non-sequential approach is 1.61. Indeed most of the time the two sets are disjoint, as the average size of their intersection is 0.07. This can also be seen in Figure 9, where we plot the average frequency difference for each of the 256 solutions, where we have only counted cases in which the solution appeared at least once as output for at least one procedure.

It is not surprising that the sequential approach has slightly more variability in its outputs. In fact, the uncertainty modeled by the probability distribution over the weights of the attributes affects the decision of each variables. Instead, in the case of the non-sequential approach it only contributes once and at the global level. Nonetheless, both methods are capable of focusing only a few alternatives and manage to efficiently weed out unattractive candidates.

This is further corroborated by an analysis we have performed on the relationship between the solutions deliberated by the DFT-based decision making procedures and the optimal solutions of the FCSPs (that is, the most preferred complete assignments according to each attribute). The results,

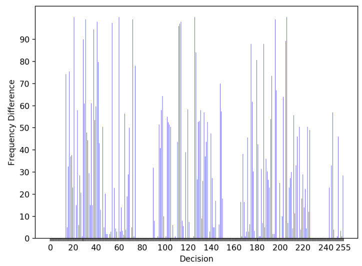


Figure 9: Average over 100 instances of eight variables of the difference in frequency of being returned by the two approaches for each solution.

obtained from the same set of 100 instances are shown in Figure 10.

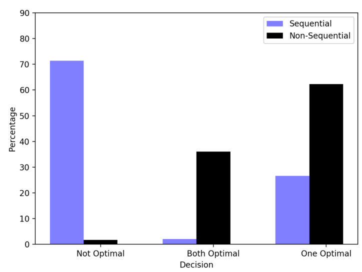


Figure 10: Percentage of times the decision making procedures return solution that are not optimal/ optimal in one/ optimal in both FCSPs.

In most cases the non sequential deliberation process returns a solution which is optimal in at least one of the FCSPs. This suggests that the subjective preferences given in input by matrix M have the effect of focusing the deliberation process only on solutions which are highly ranked by at least one attribute. The non-sequential approach, instead, seams to suffer more from the partial loss of information caused by decomposing the decision process. In fact, running the deliberation process variable by variable implies applying the effect of both FCSPs variable by variable. The selected assignments to previous variables affect future deliberations on those which are connected to them via constraints. This makes it much more likely for the sequential approach to deviate from standard constraint propagation. In order to asses the magnitude of this effect in terms of quality of the deliberated solutions with respect to the subjective preferences represented by the attribute FCSPs, we computed the average difference in preference between the solutions returned by the two procedures and the the optimal preference of the FCSPs. In the table below we show these results.

	Min Distance	Max Distance
Sequential	0.06	0.17
Non-sequential	0.01	0.04

The averages for the non-sequential approach are aligned with the fact that it often returns a solution which is optimal in at least one FCSP. However, the distance from optimal of the solutions deliberated by the sequential approach is also very small. Thus, it appears that the preference propagation obtained via DAC is sufficient to guarantee the selection of a solution which is of high quality for both attributes.

Related Work

While a large literature has been dedicated to studying the human deliberation process in settings with few, unstructured alternatives, understanding how humans make decisions over complex or combinatorial structures is for the most part an unexplored topic. In a recent paper (Samuel J. Gershman 2017) the authors developed a theory of decision making on combinatorial domains. They adopt compositionally structured utility functions as a means to represent preferences and they use probabilistic reasoning to predict preferences over new unseen alternatives. While the combinatorial structure of the alternatives is shared, our approach is very different from theirs in methods and objectives. The preference models are different as they use utilities and we use a fuzzy constraint-based approach. Furthermore, our goal is to model the variability which is observed in human decision making when preferences come from different criteria as opposed to predicting preferences over unseen options from known (or learned) ones.

Sequencing preference reasoning over soft constraint networks is not new. the most related work to ours in this context is (Pozza et al. 2011) where a sequential voting method over fuzzy constraint problems is presented. In that setting each voter expresses his preferences as a SCSP and votes are aggregated variable by variable using a voting rule. Our approach bears some similarities, in that also in that case DAC is used as a preprocessing step before voting on each variable. However, while one could view attributes as voters the way the preference is aggregated by DFT is very different from the application of a voting rule.

Conclusions and Future Work

We have presented an approach for modeling deliberation on combinatorially structured domains. We show that decomposing decision making into a sequence of deliberation steps performed with DFT is a feasible approach to tackling this problem. Our future agenda will first focus on understanding if the properties such as the similarity, attraction and compromise effect, which have been identified in behavioral studies of humans, can be modeled by the sequential approach. We also plan to test our model on behavioral data of human decision making over complex domain and in data sets in PrefLib (Mattei and Walsh 2013). Finally, we also intend to investigate the use of other compact preference models, such as probabilistic CP-nets(Boutilier et al. 2004) which look like a very interesting candidate to represent preferences according to attributes and their weights in the deliberation procedure.

References

Boutilier, C.; Brafman, R. I.; Domshlak, C.; Hoos, H. H.; and Poole, D. 2004. CP-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. *JAIR* 21:135–191.

J.R. Busemeyer, J. T. 1993. Decision field theory: a dynamic cognition approach to decision making. *Psychological Review* 100:432–450.

Kornhauser, L., and Sager, L. G. 1986. Unpacking the court. *Yale Law Journal* 96:82–117.

Lang, J., and Xia, L. 2009. Sequential composition of voting rules in multi-issue domains. *Mathematical social sciences* 57:304–324.

Mattei, N., and Walsh, T. 2013. Preflib: A library of preference data. In *Proceedings of ADT 2013*. Springer.

Meseguer, P.; Rossi, F.; and Schiex, T. 2005. Soft constraints. In Rossi, F.; Beek, P. V.; and Walsh, T., eds., *Handbook of Constraint Programming*. Elsevier.

Pozza, G. D.; Pini, M. S.; Rossi, F.; and Venable, K. B. 2011. Multi-agent soft constraint aggregation via sequential voting. In *Proceedings of IJCAI 2011*, 172–177.

R. Roe, J.R. Busemeyer, J. T. 2001. Multi-alternative decision field theory: A dynamic connectionist model of decision-making. *Psychological Review* 108:370–392.

Rossi, F.; K.B. Venable; and Walsh, T. 2011. *A Short Introduction to Preferences: Between Artificial Intelligence and Social Choice*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers.

Samuel J. Gershman, Jonathan Malmaud, J. B. T. 2017. Structured representations of utility in combinatorial domains. *American Psychological Association* 4:67–86.

Schiex, T. 1992. Possibilistic constraint satisfaction problems or "how to handle soft constraints?". In *Proceedings of UAI 1992*, 268–275.