Multi-Criteria Optimization of Temporal Preferences

Paper ID: #627 **Keywords: Temporal Reasoning, Constraint Satisfaction**

Abstract

We propose a new framework for multi-criteria optimization in constraint-based temporal reasoning. Motivated by a real-world domain, we augment one of the most expressive current formalisms, the Disjunctive Temporal Problem with Preferences (DTPP), in two crucial ways. First, we model optimality criteria as being attributed to subsets of soft constraints, in contrast to the direct mapping to individual constraints common in previous formulations. Second, using Multi-Attribute Utility Theory (MAUT) we construct an objective function that considers not only the individual values of these separate criteria, but also their mutual interactions. The increased expressive power of the *Multi-Criteria DTPP* (MC-DTPP) allows us to model a broad range of complex preferential optimization problems that existing Temporal Constraint Satisfaction Problems cannot (for instance, capturing the whole Pareto frontier). We propose two algorithms for finding optimal solutions to an MC-DTPP, and demonstrate the computational efficiency of reasoning with MC-DTPPs on a suite of randomized benchmarks and a new collection of real-world scheduling instances.

1 Introduction

Recent studies in constraint-based quantitative temporal reasoning have addressed *preferential optimization* [11; 18; 17]. In this line of research, traditional simple temporal (STP) constraints [3] are augmented with local preference functions that express how well an assignment satisfies the corresponding constraint. For instance, these functions might convey that a certain activity should be as long as possible, or that it is desirable for two activities to be scheduled in close proximity. Early research focused on *Weakest Link Optimality* [12; 19], where the minimum preference value is maximized, while later work has begun to address the more challenging *utilitarian* optimization [18; 17], where the sum of the individual preference values is maximized.

The work presented here is motivated by temporal reasoning problems arising in calendar management, where constraint-based preferential optimization has gained recent attention [16; 1]. Management of one's time is intensely personal by nature; the decision over which scheduling option is "best" is strongly dependent on each individual's personal preferences [1]. Moreover, many criteria factor into this decision and interact in complex ways. Combined, these modelling requirements surpass the expressive power of existing temporal constraint satisfaction formalisms.

In this paper we explore multi-criteria optimization in constraint-based temporal reasoning. Our goal is to obtain a formalism expressive enough to capture problems in domains such as the above, and to develop algorithms capable of solving such problems. We begin with one of the most expressive current formalisms, the *Disjunctive Temporal Problem with Preferences* (DTPP), which features disjunctive constraints with preferences over them [19]. We augment the DTPP to overcome two limitations that make it currently unable to capture our motivating problems. First, we model criteria as being attributed to subsets of soft constraints, in contrast to the direct mapping to individual constraints. Second, using Multi-Attribute Utility Theory (MAUT) we construct an objective function that considers not only the individual values of these separate criteria, but also their mutual interactions.

We propose two algorithms for achieving optimization over this new *Multi-Criteria DTPP* structure. One is a variation on a state-of-the-art branch-and-bound DTPP solver, while the other adopts a recently-proposed Multi-Criteria Search (MCS) framework. We demonstrate the efficiency of our approach on a suite of randomized benchmarks, as well as a new collection of real-world scheduling problems.

2 Related Work

There is a sizeable body of literature on the theory and practice of multi-criteria decision making (MCDM) [10; 4]. The three aspects of finding optimal solutions to a multi-criteria constraint optimization problem are: (1) modelling the problem (including elicitation of the user's preferences), (2) deciding on the combination or trade-off of the decision criteria (and thus the nature of the solutions to user wants to seek), and (3) providing an algorithm to derive the optimal solutions of the chosen type. The two broad approaches are multi-attribute utility theory (MAUT), which focuses on constructing a global utility function (the aggregation function), and multi-objective combinatorial optimization (MOCO), which is concerned with optimizing multiple objectives. In the MAUT approach, which we will adopt for the MC-DTPP, the three above parts are known as the *disaggregation*, *aggregation*, and *optimization* phases [10].

Given the size of the literature, we focus on quantitative approaches in constraint programming (CP), except to note the graphical model of CP-nets [2]. Researchers in CP have, by and large, focused either on the disaggregation and aggregation phases (i.e., the modelling aspect), or on the optimization phase (i.e., the algorithmic aspect) supposing that an aggregation function is available or the type of solution is decided. The modelling aspect is a recognizably important and difficult part of tackling real-world problems, especially where the problem definition or objective features uncertainty; [7] and [14] present case studies.

From MAUT, a common aggregation function is to combine the criteria into a single objective function as a weighted sum. The utilitarian objective for a DTPP can be seen in these lines. The solving aspect can then be addressed by variants of branch-and-bound search with dominance testing [5]. Two drawbacks to a weighted sum are its limited expressiveness, for instance regarding dependency between criteria, and the pragmatic difficulty of setting the weights (and handling the sensitivity of the problem to their values). One main alternative to a weighted sum aggregation of the criteria is to seek the frontier of non-dominated solutions. An efficient CP method to compute this Pareto frontier is given by [6].

Preference-Based Search (PBS) [9] is a dedicated algorithm for a formalism in the MOCO line that can express preferences between as well as over criteria. PBS can find balanced and extreme solutions, in addition to Pareto-optimal solutions. Like PBS but in the MAUT line, Multi-Criteria Search (MCS) [8] is based on a succession of mono-criteria searches. MCS is dedicated to an expressive formalism of MAUT that encompasses sophisticated aggregation functions beyond a weighted sum, such as the Choquet integral [13].

3 Background

A *Disjunctive Temporal Problem* (DTP) [20] is defined by a pair $\langle X, C\rangle$, where each element $X_i \in X$ designates a time point, and each element $C_i \in C$ is a constraint of the form $c_{i1} \vee c_{i2} \vee ... \vee c_{in_i}$. In turn, each c_{ij} is of the form $a_{ij} \leq x_{ij} - y_{ij} \leq b_{ij}$ with $x_{ij}, y_{ij} \in X$ and $a_{ij}, b_{ij} \in \Re$. DTPs thus generalize Simple Temporal Problems (STPs) [3], in which each constraint is limited to a single disjunct.

One type of solution to a DTP is the *object-level solution*, defined as a numeric assignment to each of the time points in X, such that all the constraints in C are satisfied. A second type of solution is a *meta-CSP solution*. Here, instead of directly considering assignments to the time points in X, a meta-variable is created for each constraint in the DTP, whose domain is simply the set $\{c_{i1}, c_{i2}, \ldots, c_{in_i}\}$ of disjuncts one can choose to satisfy C_i. A complete assignment in the meta-CSP thus involves a selection of a single disjunct for each constraint, commonly referred to as a *component STP*, which represents a set of object-level solutions.

A *DTP with Preferences (DTPP)* [19] extends a DTP by replacing the hard interval of each disjunct c_{ij} with a preference function $\langle f_{ij} : \Re \rightarrow \mathcal{A}\rangle$ that maps every temporal difference to a *preference value* from a set $\mathcal{A}$, expressing its relative utility [11]. Following DTPP convention, we take $\mathcal{A} = \Re$.[1]

Given an assignment S to the DTPP D, the preference value of a disjunctive constraint $C_i \in C$ is defined to be the maximum value achieved by any of its disjuncts:

$$val_D(S, C_i) = \max_{c_{ij} \in D(C_i)} f_{ij}(S(x_{ij}) - S(y_{ij})) \qquad (1)$$

With the inclusion of preferences, we are no longer concerned with simply finding a feasible assignment; we also want a solution of high quality. This requires us to specify an *objective function* with respect to each of the individual preference functions. Previous work has focused on objectives corresponding to three notions of optimality.

One of the earliest objectives to be considered is *maximin* or *Weakest Link Optimality* (WLO) [11; 12], in which the global value of an assignment S is equal to the minimal preference value satisfied by the assignment:

$$val_D(S) = \min_i (val_D(S, C_i)) \qquad (2)$$

WLO corresponds to optimality with the fuzzy semiring [11]. When using WLO, solutions are compared myopically using only the "weakest link" in each solution; no credit is given for satisfying other constraints at very high levels. Despite this drawback, WLO is appropriate in many situations, and optimally solving DTPPs using WLO is only slightly more expensive than solving DTPs [19].

The second type of objective that has been explored is *utilitarian* optimality, in which the global value of an assignment S is equal to the sum of the preference values of the individual constraints [18]:

$$val_D(S) = \sum_i val_D(S, C_i) \qquad (3)$$

Only recently have approaches been developed for performing utilitarian optimization of a DTPP; all, including the leading solver [17], can handle problems containing complex piecewise-constant preference functions.

The third type of objective is *stratified egalitarian* (SE) optimality. SE-optimal solutions are the set of WLO-optimal solutions that are not Pareto-dominated — a subset of the Pareto frontier. In the case of STPPs, this set can be obtained be repeated use of a WLO solver. SE optimality, however, has not been explored for DTPPs.

4 Multi-Criteria Disjunctive Temporal Problems with Preferences

In a sense, the extensions of preferences to temporal CSPs described previously can indeed model situations that contain multiple criteria, in which each constraint is a single criterion on which to optimize, and where the objective function defines a simple, fixed way to combine these criteria. Regardless of which objective function one thus chooses for the DTPP, there is an underlying assumption that criteria and the constraints go hand-in-hand. The following example motivates us to seek a more expressive formalism.

4.1 A Motivating Example

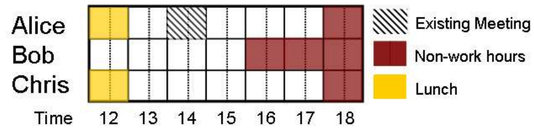

Figure 1: Tuesday afternoon schedule

On Tuesday morning, Alice realizes she needs to schedule a 90 minute project meeting with her colleagues Bob and Chris. The meeting needs to happen soon, but Alice needs time to prepare, so she tasks her automated scheduling tool to arrange a meeting sometime on Tuesday afternoon.

Bob's schedule is formulated around his young children. He gets to the office early but leaves at 16:00 to collect them from childcare. He makes up the time by not taking lunch. Chris, like Alice, has a more typical workday that includes lunch from 12:00 to 13:00. Alice has an existing commitment from 14:00 to 15:00, but it is of lower priority.

The schedules of the three people for the afternoon are shown in Fig. 1. The heavily shaded times are outside working hours, the lightly shaded times represent lunch preferences, and the ruled times represent existing commitments.

If we represent the start- and end- points of meetings and activities as temporal variables, the scheduling problem arising from this meeting request can be cast as a DTP. However, it can be seen by inspection that Alice's request cannot be fulfilled: the problem is over-constrained. A solution can be found by relaxing the constraints in various ways, such as: (a) shorten desired duration; (b) relax non-work hours (e.g., Bob leaves later); (c) overlap with the existing event; (d) move the existing event; (e) shorten or cancel the existing event; or (f) change requested meeting day (e.g., allow Wednesday).

To encode these various relaxations, one can widen the feasible region of the original temporal constraints and define

[1] Our definition refines the original DTPP specified in [19], which defined preference functions only over feasible regions of disjuncts. By convention, when $\mathcal{A} = \Re$, 0 is the minimum preference value that can be obtained by an assignment that satisfies the constraint.

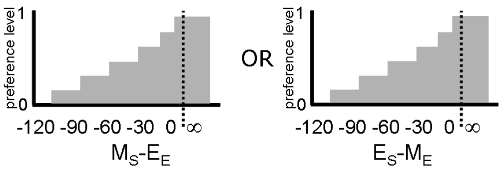

Figure 2: Example of a soft overlap constraint for a new meeting (M) and an existing meeting (E)

preferences over the larger intervals (resulting in DTPP constraints). For example, we can express that the new meeting (M) cannot overlap Alice's existing meeting (E) by the following hard DTP constraint:

$$M_S - E_E \geq 0 \vee E_S - M_E \geq 0, \tag{4}$$

which can be relaxed to the constraint and preference functions shown in Fig. 2; these functions express that no overlap is preferred, and that small overlaps are preferred to large ones.

Although the problem can be cast as a DTPP in this way, the choice of which (possibly relaxed) candidate solution is *best* varies with the individual. In evaluating the candidates, emphasis may vary between the importance and the satisfaction of criteria such as: (1) personal meeting time preferences; (2) duration; (3) participants; (4) meeting location; (5) degree of overlap with existing calendar events; and (6) the meeting time preferences of others. Besides showing that the choice of an optimal solution is a personal one, the example demonstrates that multiple criteria influence this decision.

A common attempt to capture the desired trade-offs between criteria is with a weighted sum objective function (i.e., utilitarianism). However, it is well-known that a weighted sum can express only limited forms of dependencies between criteria [9; 13]. For instance, a weighted sum can express that criteria are equally important, but it cannot express that criteria should be *balanced*. In our example, consider three criteria, $\mathcal{C}_1$, $\mathcal{C}_2$, and $\mathcal{C}_3$ (one for the time preferences of each person), and two candidate time slots 12:30–14:00 and 15:00–16:30. The first time slot violates the lunch-time plans of both Alice and Chris; the second slot is a stronger violation of Bob's need to pick up his children. The choice is therefore between a time very undesirable for one person and a time slightly undesirable for two. The latter will likely be preferred if Alice desires balance between the criteria.

More concretely, consider mapping these qualitative preferences onto a $[0, 1]$ scale for each criterion, where 1 indicates full satisfaction. We may express that meeting during lunchtime reduces satisfaction to 0.6, and that meeting during non-work hours reduces satisfaction to 0.3. The 12:30 slot violates the lunch preference for both Alice and Chris ($\mathcal{C}_1 = 0.6$, $\mathcal{C}_2 = 1.0$, and $\mathcal{C}_3 = 0.6$) and achieves a weighted sum of 2.2 (assuming all weights are 1). The 15:00 slot violates Bob's non-work hours ($\mathcal{C}_1 = 1.0, \mathcal{C}_2 = 0.3$, and $\mathcal{C}_3 = 1.0$) and achieves the greater sum of 2.3. Thus, the time significantly inconvenient for Bob (the unbalanced solution) is chosen. For this trivial example, it is easy to choose satisfaction levels in a way that coerces the weighted sum into choosing the balanced solution. This method fails, however, when complexity increases, e.g, a second inter-criterion preference may require levels inconsistent with the first.

In contrast to utilitarian optimality, maximin optimality can somewhat capture the notion of balanced solutions, but its myopic nature causes it to ignore all constraints except the weakest one. Although less myoptic than maximin, SE optimality is limited to a subset of the Pareto frontier, and in any case has not been applied to DTPPs.

Situations similar to our example require a mixture of objectives, demanding both the sensitivity of utilitarianism and the worst-case safeguards of maximin. This drives us to seek a more expressive extension of the DTPP formulation to capture multiple criteria and their interactions.

4.2 The Multi-Criteria DTPP

We define a new temporal constraint optimization formalism, in which the aforementioned limitations of the DTPP are resolved by incorporating higher-level criteria. This is achieved by a representation that explicitly models criteria as subsets of soft constraints, and employs an aggregation function in the MAUT (multi-utility decision theory) fashion to combine and trade-off the criteria.

Definition 1. A *Multi-Criteria DTPP* (MC-DTPP) is defined by $\langle X, C, \mathcal{S}, A \rangle$, where X and C are as in a DTPP, $\mathcal{S}$ is a set of m criteria $\{\mathcal{C}_1, \mathcal{C}_2, ..., \mathcal{C}_m\}$ where each element $\mathcal{C}_i$ is a subset of C, and A is a triangular matrix of coefficients in the range $[-1, 1]$.

The intuition behind this augmented formalism is that each criterion $\mathcal{C}_i$ captures a particular *feature* of the problem. For instance, $\mathcal{C}_i$ might correspond to all the constraints involved in a single user i's schedule, or to the set of all constraints over the durations of activities. In effect, any feature of the constraint problem that involves a coalition of constraints can be regarded as a separate criterion.[2]

For an object-level assignment S to an MC-DTPP D, we define the *utility* u_i of a criterion $\mathcal{C}_i$ as the proportion of the combined preference achieved by the constraints in its scope:

$$u_i = \frac{\sum_{C_j \in \mathcal{C}_i} val_D(S, C_j)}{\sum_{C_j \in \mathcal{C}_i} max_D(C_j)} \tag{5}$$

Hence, u_i is a *fuzzy measure* [15] taking the range $[0, 1]$. If all constraints are satisfied at their lowest preference levels, this results in a utility of 0; conversely, a utility of 1 indicates that all constraints are maximally satisfied.

The global value of an MC-DTPP D for a given assignment S is (as expected) an aggregation of the values of the individual criteria. More specifically, we define it as a function parameterized by the coefficients $\{a_{ij}\}$ in A:

$$val_D(S) = \sum_i a_{ii} u_i + \sum_{i=1}^{m} \sum_{j=i+1}^{m} a_{ij} min(u_i, u_j) \tag{6}$$

Since the criteria values are normalized to $[0, 1]$, this aggregation (6) of values from the object-level solution S to global value $val_D(S)$ corresponds to a form known in MAUT as the *two-additive Choquet integral* [13].[3]

The coefficients in A form a triangular matrix. A diagonal coefficient a_{ii} expresses the relative importance of the value of criterion $\mathcal{C}_i$ in relation to the values of the other criteria. The non-diagonal coefficients a_{ij} express the interaction (positive or negative correlation) of the values of criteria i and j in terms of the overall value ascribed to solutions.

The importance coefficients a_{ii} take values in $[0, 1]$. A value of 0 indicates that criterion $\mathcal{C}_i$ contributes nothing toward the global value (i.e., feature i has no significance in the

[2]The MC-DTPP can be defined more generally, where the level of participation of a constraint in a criterion is specified as a numeric weight. For simplicity, we focus on the case of binary membership.

[3]We refer to [15] for details on the derivation. Note that, since we define u_i in terms of the satisfaction of the constraints in the scope of criterion $\mathcal{C}_i$, we automatically have an ordering on u_i, i.e. higher values correspond to more preferred solutions.

overall preference aggregation), while a value of 1 indicates that $\mathcal{C}_i$ contributes with relatively maximal importance.

The correlation coefficients a_{ij} take values in $[-1, 1]$. The import of their magnitude $|a_{ij}|$ is similar to that of the importance coefficients: $|a_{ij}| = 0$ indicates no correlation between $\mathcal{C}_i$ and $\mathcal{C}_j$ (features i and j are independent in the overall preference aggregation), while $|a_{ij}| = 1$ indicates relatively maximal correlation (features i and j have maximal influence on one another). The sign of the correlation coefficient indicates complementary (if $a_{ij} > 0$) or substitutive (if $a_{ij} < 0$) behavior. That is, if $a_{ij} > 0$, a solution S must be "good" on both criteria to be considered good (i.e. more highly preferred) overall, while if $a_{ij} < 0$, S is considered good overall as soon as it is good on either one criterion.

The aggregation of the multiple criteria defined by (6) employs a subset of the full Choquet integral formalism [13]. The two-additive Choquet integral we adopt has been found to provide a practical compromise between versatility and representational compactness [14]. In fact, there exists a theory and practical methods for eliciting the coefficients A, and for explaining the significance of optimal decisions. For details of these methods, we refer to [14; 15].

The MC-DTPP can express any of the DTPP objective functions that have been considered in prior work. For example, utilitarian optimality (3) is modelled by $a_{ii} = 1\ \forall i$ and $a_{ij} = 0\ \forall i, j$. A local approximation of maximin optimality (2) is modeled by the two-additive Choquet integral when $a_{ii} = 0\ \forall i$ and $a_{ij} = 1\ \forall i, j$. Stratified egalitarian optimality is then captured by the subset of these maximin solutions that intersect the Pareto frontier [14]. Finally, exact maximin optimality is achieved using the full Choquet integral.

Looking back to our scheduling example, we can view the constraints needed to represent the problem as belonging to one or more of the criteria. For example, all temporal constraints related to Bob's meeting time preferences are mapped to $\mathcal{C}_2$. The preference values for these constraints are aggregated and normalized onto a $[0, 1]$ scale by (5), producing the u_2 value in the Choquet function. In this simple example, we have only one constraint for each of our criteria. In a real-world scheduling application, however, we have many constraints mapping to some criteria [1].

The quantitiative example given Section 4.1 assumed that $a_{11} = a_{22} = a_{33} = 1$, producing weighted sums of 2.2 and 2.3 for the 12:30 and 15:00 time slots, respectively. Within the Choquet objective function (6), the preference for balance is modelled by the a_{ij} coefficients. When all are set to 1.0, it indicates each pair of criteria are complementary, and contributes to the $a_{ij} min(u_i, u_j)$ terms. For the 15:00 slot, the terms in the Choquet function are $a_{11}u_1 = 1.0, a_{22}u_2 = 0.3, a_{33}u_3 = 1.0$, $a_{12}min(u_1, u_2) = 0.3, a_{13}min(u_1, u_3) = 1.0$, and $a_{23}min(u_2, u_3) = 0.3$, which sum to 3.9. For the 12:30 slot, all three interaction terms evaluate to 0.6, producing a Choquet value of 4.0. Thus, the interaction terms are critical in identifying the optimal solution.

5 Algorithms for Solving MC-DTPPs

The expressive power of the MC-DTPP is of no value unless we can compute relevant solutions. We present two preliminary algorithms for performing optimization of the MC-DTPP, which rely on the notion of *preference projections* [11]. An DTPP preference projection essentially "slices" an DTPP constraint into a set of intervals that produce a preference value greater than or equal to some specified level l.

Definition 2. Given a DTPP constraint $C_i = c_{i1} \vee c_{i2} \vee \cdots \vee c_{in}$, the *preference projection at level* l for C_i is $\mathcal{P}_i[l] = \bigcup_{j=1}^{n} \mathcal{P}_{ij}[l]$, where $\mathcal{P}_{ij}[l] = \{c_{ij1}, c_{ij2}, \ldots, c_{ijn}\}$ with $c_{ijk} = \langle a_{ijk} \leq x_{ij} - y_{ij} \leq b_{ijk} \rangle$, $b_{ijk} < a_{ij(k+1)}$ for $0 \leq k < n$ and $\bigcup_{k=1}^{n} [a_{ijk}, b_{ijk}] = \{t | f_{ij}(t) \geq l\}$.

We will use the preference projection to convert an MC-DTPP into a *Projected MC-DTPP*. The basic idea is to create multiple (weighted) DTP constraints for each individual DTPP constraint: one for each distinct preference level.[4] Let D' be a MC-DTPP with constraints $\{C'_1, \ldots, C'_n\}$. Create Projected MC-DTPP D by, for each constraint C'_i in D':

- Create a constraint $C_{<i,0>} = \bigvee \mathcal{P}_i[0]$ in D, and assign it a weight $w_{i,0} = \infty$. Set l to zero.
- Find the smallest $l' > l$ such that $\mathcal{P}_i[l'] \neq \mathcal{P}_i[l]$.
- Create a constraint $C_{<i,l'>} = \bigvee \mathcal{P}_i[l']$ in D, and assign it a weight $w_{i,l'} = (l' - l)$. Set l to l'.
- Iterate until a level l' is reached such that $\mathcal{P}_i[l'] = \emptyset$.

The transition to a Projected MC-DTPP allows us to exploit the meta-CSP search space commonly used in disjunctive temporal reasoning. Specifically, we can create a meta-variable for each projected constraint $C_{<i,l>}$, whose domain is the associated set of disjuncts or intervals. Any assignment to this variable can be viewed as satisfying the original constraint C'_i at preference level l (whereas the absence of an assignment can be regarded as its explicit violation).

```
Solve-MC-DTPP(A, U)
If (s* ≠ nil & val(A) ≤ val(s*)) then return
If (U = ∅) then s* ← A ; return
C_i ← select-variable(U), U' ← U − {C_i}
For each disjunct c_ij of D(C_i)
   A' ← A ∪ {C_i ← c_ij}
   If (consistent(A')) then Solve-MC-DTPP(A', U')
EndFor
A' ← A ∪ {C_i ← ε}
Solve-MC-DTPP(A', U')
```

Figure 3: A branch-and-bound algorithm for solving MC-DTPPs

5.1 A Branch-and-Bound Method

Using this Projected MC-DTPP reformulation, we first describe a branch-and-bound algorithm for computing optimal solutions; pseudocode is given in Fig. 3. The input variable A is the current set of assignments to meta-variables, and is initially $\emptyset$; variable U is the set of unassigned meta-variables (initially, the entire set C); a global variable *s** stores the best solution that has been found, and is initially *nil*.

The algorithm, which takes an approach similar to that in [17], resembles the backtracking search commonly used for solving traditional DTPs, but with two notable differences. First, backtracking occurs only when the value of the partial assignment $val(A)$ is worse than that of the best known solution; in a standard DTP solver, all constraints must be satisfied, and thus backtracking occurs whenever any constraint is violated in the partial assignment. Second, in addition to the values in the original domains of the meta-variables, there is the possibility of an empty assignment ('ϵ'), and so the branching factor increases by exactly one.

Due to its strong resemblance to a standard DTP search, this branch-and-bound algorithm for solving MC-DTPPs allows the direct incorporation of several powerful techniques

[4] As in prior DTPP literature [19; 17], we assume that the preference functions have been discretized into piecewise-constant form.

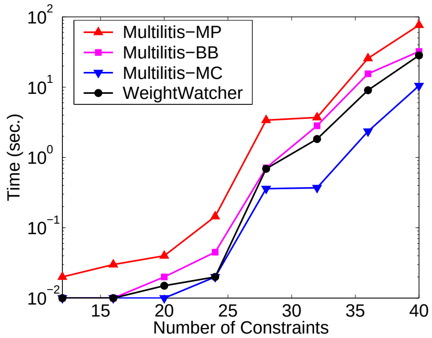

Figure 4: Median running times as the number of constraints increases

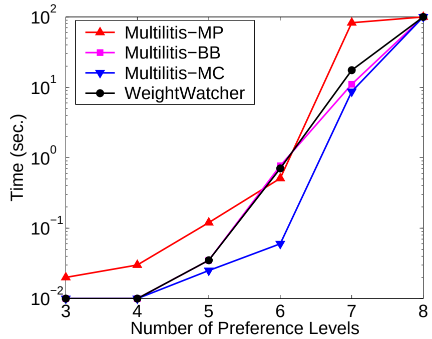

Figure 5: Median running times as the number of preference levels increases

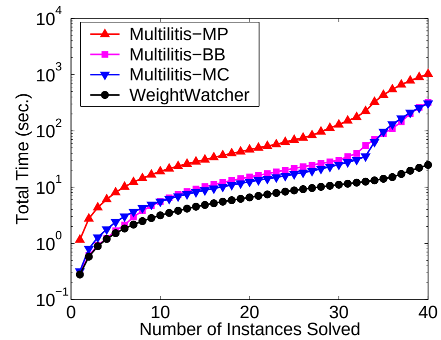

Figure 6: Cumulative runtimes on a set of scheduling MC-DTPPs

previously developed in the DTP literature. As mentioned earlier, a recent variation on this algorithm for solving DTPPs [17] has proven extremely effective.

5.2 An Multi-Criteria Search Method

The success of the branch-and-bound algorithm for DTPPs makes it an attractive candidate for MC-DTPP optimization; however, the presence of multiple criteria presents unique challenges and opportunities that do not arise when solving the DTPP. While it may be relatively straightforward to define a search strategy (i.e., variable and value ordering heuristics) for an aggregation of preferences within a single criterion, such a strategy may not necessarily be appropriate when there exist several criteria competing to contribute to the global objective function.

In response, we turn to the Multi-Criteria Search (MCS) framework [8] introduced in Section 2. Rather than employing a single branch-and-bound search to maximize the global objective function, MCS executes a series of mono-criterion searches, each fine-tuned to maximize the utility of a single feature of the problem. For each mono-criterion search, a set of *local constraints* is imposed temporarily to guide the search toward promising solutions. These constraints often require strict improvement on the selected criterion, as compared to its value in the best known solution.

One of the key advantages of the MCS approach is its ability to exploit certain properties of multi-criteria decision making to ensure that the final solution s^* is optimal with respect to the global objective function. In fact, this guarantee can be made even if the individual searches are not complete, and instead explore only a portion of the search space. This, in combination with the pruning power of individual search strategies, makes MCS an extremely flexible framework for multi-criteria optimization. Due to space limitations, we omit a detailed description of the MCS algorithm, referring the reader to [8] for a complete description.

6 Preliminary Experimental Results

We conducted three experiments to evaluate our proposed algorithms for MC-DTPP optimization. Our goal is twofold: (1) to determine if reasoning with MC-DTPPs is practical; and, in particular, whether the complex preference aggregation scheme of the two-additive Choquet integral incurs larger computational penalties than utilitarian optimization; and (2) to determine whether application of the MCS framework is competitive in performance with our extension of the branch-and-bound algorithm that has proven effective on DTPPs. However, it should be made clear that these tests are far from exhaustive, due to the wide range of possible heuristics, test parameters, and algorithm variations. All experiments were run on a 2.26GHz Pentium 4 with 1 GB of RAM.

In our experiments, we employ four different algorithms. Three of these are variations on a solver we name MULTILITIS, with each variation specifying which search strategy to use: (1) MULTILITIS-BB (the branch-and-bound version); (2) MULTILITIS-MC (an MCS version that performs complete mono-criterion searches); and (3) MULTILITIS-MP (an MCS version that performs partial mono-criterion searches)[5]. All are descendants of the state-of-the-art DTPP solver WEIGHTWATCHER [17], which is the fourth constraint engine we include. To empirically compare our algorithms to WEIGHTWATCHER is not entirely fair, since it performs utilitarian optimization, ignoring the multiple high-level criteria of the MC-DTPP and thus establishing an entirely different ranking of solutions. Even so, this comparison will provide insight into the additional computational expense (if any) of multi-criteria optimization over utilitarian optimization.[6]

There remains an unfortunate absence of real-world benchmarks in temporal preference literature with which to provide an empirical comparison of solvers. Consequently, we begin by employing a problem generator that has been used in recent DTPP studies [17]. The DTPPs produced by this generator are characterized by the parameters $\langle E, C, k, L\rangle$, where E is the number events (or time points), C is the number of constraints, k is the number of disjuncts per constraint, and L is the number of preference levels. We augment this generator with two additional parameters, $\langle m, p\rangle$: m is the number of criteria, and p is the probability that any constraint C_i will appear in the scope of a particular criterion $\mathcal{C}_i$.

In our first experiment, we test the ability of these algorithms to scale with problem size. The ability to perform well on this set of tests is especially important, since unlike other problem parameters (such as the number of preference levels), the size of the problem is often difficult for a knowledge engineer to control directly. We hold fixed the following parameters: $\{L = 6, k = 2, m = 6, p = 0.25\}$, and vary the number of constraints C from 12 to 40, keeping the number of events E at $\frac{3}{4}C$ to maintain a constant constraint density.

Fig. 4 displays the results of this experiment. The number of constraints in the problem is shown on the x-axis, and on

[5] In our implementation, each partial search returns once a single improved solution is discovered. However, as noted earlier, optimality and thus solution quality is unaffected.

[6] The choice to compare with utilitarian optimization (as opposed to maximin) is due to its strong resemblance to the manner in which our criteria are evaluated individually.

the y-axis is the median running time over 30 instances (note the logarithmic scale). As expected, the computation time required by all algorithms grows with problem size. As can be seen by comparing MULTILITIS-BB and WEIGHTWATCHER, the branch-and-bound approach used for multi-criteria optimization does not appear to be noticeably more expensive than when it is used for the simpler case of utilitarian optimization. Further, the application of MCS using complete mono-criterion searches is effective, as MULTILITIS-MC outperforms all other algorithms. However, the runtime of MULTILITIS-MP indicates that much of this improvement is lost when MCS is applied with only partial mono-criterion searches. Similar results are observed in our second experiment, where the number of preference levels is varied (see Fig. 5).

In our final experiment, we sought to explore the characteristics of these solvers on more realistic types of problems, since randomly generated test cases typically offer little insight into the practical performance of an algorithm. As there appears to be no such set of realistic benchmarks in the literature, we have developed an entirely new suite of test cases drawn from our motivating domain of meeting scheduling [16; 1]. Each of these 40 benchmarks corresponds to a complex scheduling request involving many of the issues reflected in our earlier example (e.g., non-overlap constraints, flexible durations, several participants, and competing criteria).

Fig. 6 displays the results. Since we are not varying problems across any particular test parameter, we show the cumulative time required to solve the 40 instances when sorted in order of increasing difficulty (rather than reporting the median). The branch-and-bound and complete MCS incarnations of MULTILITIS both appear rather competitive on the new set of problems. Further, the utilitarian optimization that WEIGHTWATCHER performs more efficiently than any of the multi-criteria approaches (though, for most problems, by only a narrow margin). This is something that was not observed in the randomly generated problems.

In summary, our preliminary experiments address our goals by showing that (1) the time required to optimally solve MC-DTPPs is comparable to that of optimally solving DTPPs; and (2) the MCS approach for solving MC-DTPPs can compete with (and often outperform) the branch-and-bound approach.

7 Conclusion and Future Work

We have developed a new formalism for temporal reasoning with preferences and multiple optimization criteria, the *Multi-Criteria Disjunctive Temporal Problem with Preferences* (MC-DTPP). To our knowledge, this is the first time that the approach of Multi-Attribute Utility Theory has been specifically applied to constraint-based temporal optimization problems.

The expressive power of the MC-DTPP goes beyond any of the simple aggregation methods implicit in the DTPP objective functions studied in the literature, or their combination. The MC-DTPP allows us broad freedom to express complex preferential relationships and consider a wide spectrum of balancing, combination, and trade-off between multiple optimality criteria. Such an expressive power is necessary in domains such as calendar management; indeed, MC-DTPPs have been adopted in a deployed scheduling system [1].

We use a restricted form of the general MAUT framework in that the MC-DTPP expresses features of a DTPP problem solely in terms of subsets of the constraints. The implementation of our current algorithms depends on a second restriction: that the aggregation function is the two-additive Choquet integral rather than the general instance. These two factors enable the effective solving of MC-DTPPs, despite the NP-hardness of DTP solving, the multiple criteria, and the expressive preferences.

Besides demonstrating the computational efficiency of reasoning with MC-DTPP, our empirical results suggest that dedicated multi-criteria optimization algorithms show promise as a means to efficiently handle complex preferences in constraint-based temporal reasoning. In the future we plan to study at greater depth the strengths and weaknesses of the MCS and branch-and-bound algorithms, in order to pursue a hybrid solver that combines the qualities of both.

References

[1] P. Berry, K. Conley, M. Gervasio, B. Peintner, T. Uribe, and N. Yorke-Smith. Deploying a personalized time management agent. In *Proc. of AAMAS'06 Industrial Track*, 2006.

[2] C. Boutilier, R. I. Brafman, C. Domshlak, H. H. Hoos, and D. Poole. Preference-based constrained optimization with CP-Nets. *Computational Intelligence*, 20:137–157, 2004.

[3] R. Dechter, I. Meiri, and J. Pearl. Temporal constraint networks. *Artificial Intelligence*, 49(1-3):61–95, 1991.

[4] M. Ehrgott and X. Gandibleux, editors. *Multiple Criteria Optimization*. Kluwer, Boston, MA, 2002.

[5] F. Focacci and D. Godard. A practical approach to multi-criteria optimization problems in constraint programming. In *Proc. of CP-AI-OR'02*, 2002.

[6] M. Gavanelli. An algorithm for multi-criteria optimization in CSPs. In *Proc. of ECAI-02*, 2002.

[7] C. Gervet, Y. Caseau, and D. Montaut. On refining ill-defined constraint problems: A case study in iterative prototyping. In *Proc. of PACLP-99*, 1999.

[8] F. L. Huede, M. Grabisch, C. Labreuche, and P. Saveant. Multi-criteria search in constraint programming. In *Proc. of CP-AI-OR'03*, 2003.

[9] U. Junker. Preference-based search and multi-criteria optimization. In *Proc. of AAAI'02*, 2002.

[10] R. L. Keeney and H. Raiffa. *Decisions with Multiple Objectives*. Cambridge University Press, Cambridge, UK, 1993.

[11] L. Khatib, P. Morris, R. Morris, and F. Rossi. Temporal constraint reasoning with preferences. In *Proc. of IJCAI'01*, 2001.

[12] L. Khatib, P. Morris, R. Morris, and K. B. Venable. Tractable Pareto optimal optimization of temporal preferences. In *Proc. of IJCAI'03*, 2003.

[13] C. Labreuche and M. Grabisch. The Choquet integral for the aggregation of interval scales in multicriteria decision making. *Fuzzy Sets and Systems*, 137(1):11–26, 2003.

[14] C. Labreuche and F. L. Huédé. MYRIAD: a tool for preference modeling application to multi-objective optimization. In *Proc. of CP'05 Workshop on Preferences and Soft Constraints*, 2005.

[15] J.-L. Marichal and M. Roubens. Determination of weights of interacting criteria from a reference set. *European J. of Operational Research*, 124(3):641–650, 2000.

[16] P. J. Modi, M. M. Veloso, S. F. Smith, and J. Oh. CMRadar: A personal assistant agent for calendar management. In *Proc. of AOIS'04*, 2004.

[17] M. D. Moffitt and M. E. Pollack. Temporal preferece optimization as weighted constraint satisfaction. In *Proc. of AAAI'06*.

[18] P. Morris, R. Morris, L. Khatib, S. Ramakrishnan, and A. Bachmann. Strategies for global optimization of temporal preferences. In *Proc. of CP'04*, 2004.

[19] B. Peintner and M. E. Pollack. Low-cost addition of preferences to DTPs and TCSPs. In *Proc. of AAAI'04*, 2004.

[20] K. Stergiou and M. Koubarakis. Backtracking algorithms for disjunctions of temporal constraints. In *Proc. of AAAI'98*.