Differential Anti-Chain Algorithms for the Generalized Resource-Envelope Problem

T. K. Satish Kumar
Knowledge Systems Laboratory
Stanford University
tksk@ksl.stanford.edu

Abstract.
Interleaved planning and scheduling employs the idea of extending partial plans by regularly heeding to the scheduling constraints during search. One of the techniques used to analyze scheduling and resource consumption constraints is to compute the so-called *resource-envelopes*. These envelopes can then be used in a variety of ways to guide the search for a good plan. In this paper, we will define a generalized version of the resource-envelope problem, and provide efficient algorithms for solving it.

1 Introduction

Interleaved planning and scheduling employs the idea of extending partial plans by regularly heeding to the scheduling constraints during search. One of the techniques used to analyze scheduling and resource consumption constraints is to compute the so-called *maximum (upper)* and *minimum (lower) resource consumption envelopes*. These envelopes can be used to the guide search for a good plan in a variety of ways (see [3], [5] and [6]). First, they provide sanity checks for early backtracking when it is possible to examine the lower envelope and determine that no consistent schedule for the current set of constraints could possibly satisfy all the resource conflicts. Second, they provide a heuristic value for estimating the "constrainedness" of a partial plan. Third, they provide a search termination criterion when it is possible to examine the envelopes and determine that any consistent schedule for the current set of constraints would succeed in satisfying the resource conflicts. Fourth, they provide potential subroutines in designing approximation algorithms for optimal plan dispatching.

The *generalized resource-envelope* problem is as follows. A directed graph $\mathcal{G} = \langle \mathcal{X}, \mathcal{E} \rangle$ has $\mathcal{X} = \{X_0, X_1 \ldots X_n\}$ as the set of nodes corresponding to events (X_0 is the "beginning of the world" node and is assumed to be set to 0), and $\mathcal{E}$ as the set of directed edges between them. A directed edge $e = \langle X_i, X_j \rangle$ in $\mathcal{E}$ is annotated with the simple temporal information $[LB(e), UB(e)]$ indicating that a consistent schedule must have X_j scheduled between $LB(e)$ and $UB(e)$ ($LB(e) \leq UB(e)$) seconds after X_i is scheduled. Some edges (called *action edges*) correspond physically to actions and have $LB(e), UB(e) > 0$. An action edge A can consume a resource in a variety of ways: (Type 1) A claims w_A amount of the resource at the beginning of its execution and returns it at the end, (Type 2) A claims w_A amount of the resource at the beginning of its execution and does not return it at the end, or (Type 3) A consumes the resource according to a positive rate function

$u_A(t)$. Given a consistent schedule s for all the events, the amount of resource consumed by time t is denoted by $C_s(t)$. The goal is to build the upper envelope function $g(t)$ and the lower envelope function $h(t)$ such that $g(t) = \max_{\{s \text{ is a consistent schedule}\}} C_s(t)$ and $h(t) = \min_{\{s \text{ is a consistent schedule}\}} C_s(t)$.

Some attempts for estimating $g(t)$ have been made in [3], [5] and [6]. This paper improves upon them in a number of ways. First, the estimation of $g(t)$ provided in [5] is conservative, while it is tight in [3], [6] and in the algorithm provided in this paper. Tightness in the estimates of $g(t)$ and $h(t)$ is extremely important because a tight bound can save a potentially exponential amount of search through early backtracking and solution detection when compared to a looser bound. Tight bounds also provide better heuristic estimates for the "constrainedness" of a problem during search. Second, [3], [5] and [6] deal only with producer-consumer models, whereas the algorithm presented in this paper efficiently handles the most general case where some actions can be of Type 1, others of Type 2 and still others of Type 3. We note that although it is unlikely that any particular resource is consumed by actions of all possible types (Types 1, 2 and 3), the chosen model is extremely important in reasoning about plans (schedules) that require multiple resources to execute. Building the resource envelopes separately for every resource does not ensure that the schedules that achieve $g(t)$ or $h(t)$ (for different resources) are the same (for any given t). Instead, converting the resource requirements to *costs* and building a single profile function is often more useful. This would however, then require us to reason about all possible ways of consuming all possible resources, which the chosen model justifiably allows. Third, we provide algorithms for the computation of both $g(t)$ and $h(t)$ (even when there are Type 3 actions). Fourth, our algorithms are constructive in the sense that we can determine a flexible schedule s that achieves $g(t)$ or $h(t)$. This is better than determining an arbitrary fixed schedule (as in [6]) because flexible schedules tend to be robust in dealing with uncertainty of execution.[1]

We assume that the constraints specified in $\mathcal{E}$ are consistent, and throughout the paper, we will refer to a Type 1 action A being *active* when it holds w_A amount of resource, a Type 2 action B being *active* when it has consumed w_B amount of resource, and a Type 3 action C being *active to a degree* d when it has consumed d amount of resource. Moreover, for a Type 3 action C, we will assume that $LB(C) = UB(C)$.

2 Computing $g(t)$ for Type 1 Action Edges

In this section, we will restrict all actions to be of Type 1. Figure 1 presents the algorithm for the computation of $g(t)$ and a flexible schedule s that achieves it for a given time instant t. Figure 2 presents the algorithm for computing $g(t)$ for all t.

Lemma 1: A consistent schedule exists for $X_0, X_1 \dots X_n$ in $\mathcal{G} = \langle \mathcal{X}, \mathcal{E} \rangle$ if and only if the distance graph $D(\mathcal{G})$ does not contain any negative cycles (see Figure 1).

Proof: (see [2]).

Lemma 2: A Type 1 action $A = \langle X_i, X_j \rangle$ can be made active at time t if it is possible to add the two edges $\langle X_0, X_i \rangle$ and $\langle X_j, X_0 \rangle$ annotated with t with $-t$ respectively to $D(\mathcal{G})$ without introducing any negative cycles.

Proof: A Type 1 action edge $\langle X_i, X_j \rangle$ is active at time t if and only if X_i is scheduled before t and X_j is scheduled after t. Using the semantics of the distance graph—that a constraint $X_b - X_a \leq w$ is specified as the edge $\langle X_a, X_b \rangle$ annotated with w—this corresponds to the addition of the two edges $\langle X_0, X_i \rangle$ and $\langle X_j, X_0 \rangle$

[1][3] provides constructive and incremental algorithms for determining flexible consistent schedules in producer-consumer models (in addition to computing the envelopes themselves).

ALGORITHM: UPPER-ENVELOPE-TYPE-1-AT-T
INPUT: An instance of the resource-envelope problem with all actions of Type 1, and a time instant t.
OUTPUT: $g(t)$ and a flexible schedule s achieving it.
(1) Construct the distance graph $D(\mathcal{G})$ on $X_0, X_1 \ldots X_n$ as:
(a) Every edge $e = \langle X_i, X_j \rangle \in \mathcal{E}$ is compiled to two edges: $\langle X_i, X_j \rangle$ annotated with $UB(e)$, and $\langle X_j, X_i \rangle$ annotated with $-LB(e)$.
(b) Let $dist(X_a, X_b)$ = distance from X_a to X_b in $D(\mathcal{G})$.
(2) FOR actions $A_1 = \langle X_{i_1}, X_{j_1} \rangle$ and $A_2 = \langle X_{i_2}, X_{j_2} \rangle$:
(a) Construct a (directed) size-2 conflict from A_1 to A_2 (denoted $A_1 \rightarrow A_2$) iff $dist(X_{i_2}, X_{j_1}) < 0$.
(3) Construct a node-weighted directed graph $E(\mathcal{G})$ as follows:
(a) The nodes of $E(\mathcal{G})$ correspond to the actions.
(b) The weight on a node corresponding to action A, is w_A.
(c) A directed edge $\langle A_1, A_2 \rangle$ in $E(\mathcal{G})$ encodes a size-2 conflict $A_1 \rightarrow A_2$.
(4) Construct a node-weighted directed graph $M(\mathcal{G})$ from $E(\mathcal{G})$ by removing a node corresponding to action $A = \langle X_i, X_j \rangle$ iff $t + dist(X_i, X_0) < 0$ or $dist(X_0, X_j) - t < 0$ (such actions are referred to as size-1 conflicts).
(5) Compute $Q = \{A_{q_1}, A_{q_2} \ldots A_{q_k}\}$ as the largest weighted anti-chain in $M(\mathcal{G})$.
(6) RETURN:
(a) $g(t) = |Q| = \sum_{p=1}^{k} w_{A_{q_p}}$.
(b) $s = D(\mathcal{G}) \cup \{\langle X_0, X_i^{q_p} \rangle$ annotated with t, and $\langle X_j^{q_p}, X_0 \rangle$ annotated with $-t \mid A_{q_p} = \langle X_i^{q_p}, X_j^{q_p} \rangle \in Q\}$.

Figure 1: $g(t)$ for Type 1 actions for a specified t.

ALGORITHM: UPPER-ENVELOPE-TYPE-1-ALL-T
INPUT: An instance of the resource-envelope problem with all actions of Type 1.
RESULT: $g(t)$ for all t.
(1) FOR all action edges $A = \langle X_i, X_j \rangle$:
(a) Insert $-dist(X_i, X_0)$ into list L.
(b) Insert $dist(X_0, X_j)$ into list L.
(2) Sort L in ascending order $\langle d_1, d_2 \ldots d_{|L|} \rangle$.
(3) FOR $i = 1, 2 \ldots |L| - 1$:
(a) Compute $g(d_i)$ = UPPER-ENVELOPE-TYPE-1-AT-T at time d_i.
(b) Set $g(t) = g(d_i)$ for all t in the interval $[d_i, d_{i+1})$.
(4) Set $g(t) = 0$ for all t in the intervals $[-\infty, d_1)$ and $(d_{|L|}, +\infty]$.

Figure 2: $g(t)$ for Type 1 actions for all t.

annotated with t with $-t$ respectively. Further, since any inconsistency in the constraints is reflected by the presence of a negative cycle in $D(\mathcal{G})$ (Lemma 1), the truth of the Lemma follows.

Definition 1: A *conflict* is a set of action edges all of which cannot be simultaneously active at a given time t. A *minimal conflict* is a conflict no proper subset of which is also a conflict.

Lemma 3: A set of action edges can be made simultaneously active at time t if and only if there is no subset of them that constitutes a minimal conflict.

Proof: A set of action edges can be made simultaneously active at time t if and only if there is no subset of them that constitutes a conflict. Further, the truth of the Lemma follows from the fact that there exists a subset of actions that constitutes a conflict if and only if there exists a subset of actions that constitutes a minimal conflict.

Lemma 4: The size of a minimal conflict is ≤ 2.

Proof: A set of action edges $A_1 = \langle X_{i_1}, X_{j_1} \rangle, A_2 = \langle X_{i_2}, X_{j_2} \rangle \ldots A_k = \langle X_{i_k}, X_{j_k} \rangle$ can be attempted to be made simultaneously active at time t by the addition of the edges $\langle X_0, X_{i_m} \rangle$ and $\langle X_{j_m}, X_0 \rangle$ (for $m = 1, 2 \ldots k$) annotated with t and $-t$ respectively to the distance graph $D(\mathcal{G})$. Let these edges be referred to as *special* edges and let $D'(\mathcal{G})$ refer to the resulting distance graph. Knowing that $D(\mathcal{G})$ does not contain any negative cycles (because $\mathcal{E}$ is consistent), a negative cycle can occur in $D'(\mathcal{G})$ only if it involves a special edge. Since all special edges have X_0 as an end point, a negative cycle must involve X_0. Further, since a fundamental cycle can have any node repeated at most once, at most 2 special edges can be present in a negative cycle in $D'(\mathcal{G})$. Finally, since special edges correspond to the activation of Type 1 actions, the size of a minimal conflict is ≤ 2.

Lemma 5: A size-2 conflict is independent of t.

Proof: Continuing the above arguments, when the size of a minimal conflict is 2, the negative cycle in $D'(\mathcal{G})$ must involve an incoming special edge to X_0—say $\langle X_{j_d}, X_0 \rangle$—with weight $-t$, and an outgoing special edge from X_0—say $\langle X_0, X_{i_c} \rangle$—with weight t. The weight of the negative cycle containing exactly these two special edges is therefore $t + dist(X_{i_c}, X_{j_d}) - t$. This is independent of t and is < 0 if and only if $dist(X_{i_c}, X_{j_d}) < 0$.

Lemma 6: The relation $\succeq$ defined on Type 1 actions as follows forms a POSET (partially ordered set)—$A_1 \succeq A_2$ if and only if $A_1 = A_2$ or there is a size-2 conflict $A_1 \rightarrow A_2$ (see Figure 1).

Proof: We will show that the relation $\succeq$ is *reflexive*, *antisymmetric* and *transitive*. By definition, $\succeq$ is reflexive since $A_1 \succeq A_1$. For the antisymmetry property, we will show that when $A_1 \neq A_2$, $A_1 \succeq A_2$ and $A_2 \succeq A_1$

cannot hold simultaneously. Assuming the contrary, let $A_1 \neq A_2$, and let the size-2 conflicts $A_1 \rightarrow A_2$ and $A_2 \rightarrow A_1$ hold. If $A_1 = \langle X_{i_1}, X_{j_1} \rangle$ and $A_2 = \langle X_{i_2}, X_{j_2} \rangle$, $dist(X_{i_2}, X_{j_1}) < 0$ and $dist(X_{i_1}, X_{j_2}) < 0$. Further since $D(\mathcal{G})$ contains the edges $\langle X_{j_1}, X_{i_1} \rangle$ and $\langle X_{j_2}, X_{i_2} \rangle$ annotated with $-LB(A_1)$ and $-LB(A_2)$ respectively (for $LB(A_i)$ known to be positive for all actions A_i), we would have the negative cycle $dist(X_{j_1}, X_{i_1}) + dist(X_{i_1}, X_{j_2}) + dist(X_{j_2}, X_{i_2}) + dist(X_{i_2}, X_{j_1})$. This contradicts that $D(\mathcal{G})$ does not contain any negative cycle, hence establishing the antisymmetry property. To prove the transitivity property, suppose $A_1 = \langle X_{i_1}, X_{j_1} \rangle$, $A_2 = \langle X_{i_2}, X_{j_2} \rangle$ and $A_3 = \langle X_{i_3}, X_{j_3} \rangle$ are 3 different actions such that $A_1 \rightarrow A_2$ and $A_2 \rightarrow A_3$. This means that $dist(X_{i_2}, X_{j_1}) < 0$ and $dist(X_{i_3}, X_{j_2}) < 0$. Since $dist(X_{j_2}, X_{i_2}) \leq -LB(A_2)$ (for $LB(A_2)$ known to be positive), $dist(X_{i_3}, X_{j_1}) \leq dist(X_{i_3}, X_{j_2}) + dist(X_{j_2}, X_{i_2}) + dist(X_{i_2}, X_{j_1}) < 0$. This makes $dist(X_{i_3}, X_{j_1}) < 0$, hence establishing the conflict $A_1 \rightarrow A_3$ and proving the transitivity property.

Lemma 7: An action edge $A = \langle X_i, X_j \rangle$ constitutes a size-1 conflict at time t when $t + dist(X_i, X_0) < 0$ or $dist(X_0, X_j) - t < 0$.

Proof: Continuing the arguments in the proof of Lemma 4, a negative cycle in $D'(\mathcal{G})$ can also involve just one special edge (referred to as a size-1 conflict). In the case that it involves an incoming special edge to X_0—say $\langle X_j, X_0 \rangle$ for the action $A = \langle X_i, X_j \rangle$—$dist(X_0, X_j) - t$ needs to be < 0 to create a negative cycle. In the case that it involves an outgoing special edge from X_0—say $\langle X_0, X_i \rangle$ for the action $A = \langle X_i, X_j \rangle$—$t + dist(X_i, X_0)$ needs to be < 0 to create a negative cycle.

Lemma 8: If $Q =$ largest weighted anti-chain in $M(\mathcal{G})$, $g(t) = |Q|$ and $s = \ D(\mathcal{G}) \cup \{\langle X_0, X_i^{q_p} \rangle$ annotated with $t,$ and $\langle X_j^{q_p}, X_0 \rangle$ annotated with $-t \mid A_{q_p} = \langle X_i^{q_p}, X_j^{q_p} \rangle \in Q\}$.

Proof: By construction (step 4 in Figure 1), $M(\mathcal{G})$ incorporates the deletion of all size-1 conflicts. Further, by definition, an anti-chain in $M(\mathcal{G})$ chooses a set of actions no two distinct ones of which are comparable using $\succeq$. This means that no two distinct actions exhibit a size-2 conflict. Together, all minimal conflicts are removed in any anti-chain of $M(\mathcal{G})$ and the maximum weighted anti-chain targets the maximum possible resource demand at time t over all consistent schedules. The required flexible schedule s corresponds to the addition of edges required to activate the qualifying actions, and is appropriately given as above.

Lemma 9: $g(t)$ is piecewise constant and changes only at a polynomial number of time points.

Proof: By the previous Lemma, $g(t)$ is the largest weighted anti-chain in the graph $M(\mathcal{G})$. Since $M(\mathcal{G})$ is computed from $E(\mathcal{G})$ by deleting all size-1 conflicts at time t, the number of times $g(t)$ changes is equal to the number of times the set of size-1 conflicts changes. Further, since $dist(X_0, X_j)$ and $-dist(X_i, X_0)$ mark the membership of an action $\langle X_i, X_j \rangle$ in this set, the potential number of transition points for the piecewise constant function $g(t)$ is $O(K)$ where K is the number of Type 1 actions.

The largest weighted anti-chain in a POSET having B nodes can be efficiently computed in time $O(B^{2.5})$ using *max-flow* techniques (see [1] and [4]).

The complexity of the steps in UPPER-ENVELOPE-TYPE-1-AT-T that are independent of t (and can therefore be done just once) is referred to as its *static complexity (*$|sc|$*)*. The time dependent complexity is referred to as its *dynamic complexity (*$|dc|$*)*. The analysis of these two complexities is subsumed in the analysis presented for the most general version of the problem. The complexity of UPPER-ENVELOPE-TYPE-1-ALL-T (by Lemma 9) is equal to $O(|sc| + K|dc|)$, where K is the number of Type 1 actions.

3 Computing $g(t)$ for Type 2 Action Edges

In this section, we will restrict all actions to be of Type 2. Figure 3 presents the algorithm for the computation of $g(t)$ and a flexible schedule s that achieves it for a given time instant t. Figure 4 presents the algorithm for computing $g(t)$ for all t.

Lemma 10: A Type 2 action edge $A = \langle X_i, X_j \rangle$ can be made active at time t if and only if it is possible to add

ALGORITHM: UPPER-ENVELOPE-TYPE-2-AT-T
INPUT: An instance of the resource-envelope problem with all actions of Type 2, and a time instant t.
OUTPUT: $g(t)$ and a flexible schedule s achieving it.
(1) FOR all actions $A = \langle X_i, X_j \rangle$:
(a) Compute $dist(X_i, X_0)$ in the distance graph $D(\mathcal{G})$.
(2) RETURN:
(a) $g(t) = \sum_A \{w_A$ (s.t. $A = \langle X_i, X_j \rangle$ has $dist(X_i, X_0) + t \geq 0)\}$.
(b) $s = D(\mathcal{G}) \cup \{\langle X_0, X_i \rangle$ annotated with $t \mid A = \langle X_i, X_j \rangle$ and has $dist(X_i, X_0) + t \geq 0\}$.

Figure 3: $g(t)$ for Type 2 actions for a specified t.

ALGORITHM: UPPER-ENVELOPE-TYPE-2-ALL-T
INPUT: An instance of the resource-envelope problem with all actions of Type 2.
RESULT: $g(t)$ for all t.
(1) FOR all action edges $A = \langle X_i, X_j \rangle$:
(a) Insert $-dist(X_i, X_0)$ into list L.
(2) Sort L in ascending order $\langle d_1, d_2 \ldots d_{|L|} \rangle$.
(3) FOR $i = 1, 2 \ldots |L| - 1$:
(a) Compute $g(d_i)$ = UPPER-ENVELOPE-TYPE-2-AT-T at time d_i.
(b) Set $g(t) = g(d_i)$ for all t in the interval $[d_i, d_{i+1})$.
(4) Set $g(t) = 0$ for all t in the interval $[-\infty, d_1)$.
(5) Set $g(t) = \sum_{\text{all actions } A} w_A$ for all t in the interval $[d_{|L|}, +\infty]$.

Figure 4: $g(t)$ for Type 2 actions for all t.

the edge $\langle X_0, X_i \rangle$ annotated with t to $D(\mathcal{G})$ without introducing any negative cycles.

Proof: (similar to that of Lemma 2).

Lemma 11: The size of a minimal conflict is $= 1$.

Proof: (similar to that of Lemma 4).

Lemma 12: An action edge $A = \langle X_i, X_j \rangle$ constitutes a size-1 conflict at time t when $t + dist(X_i, X_0) < 0$.

Proof: (similar to that of Lemma 7).

Lemma 13: $g(t) = \sum_A \{w_A$ (such that $A = \langle X_i, X_j \rangle$ has $dist(X_i, X_0) + t \geq 0)\}$ and $s = D(\mathcal{G}) \cup \{\langle X_0, X_i \rangle$ annotated with $t \mid A = \langle X_i, X_j \rangle$ and has $dist(X_i, X_0) + t \geq 0\}$.

Proof: The only minimal conflicts are of size 1 and are removed in the summation $\sum_A \{w_A$ (such that $A = \langle X_i, X_j \rangle$ has $dist(X_i, X_0) + t \geq 0)\}$—which is the required $g(t)$ since all w_As are known to be positive. The required flexible schedule s corresponds to the addition of edges required to activate the qualifying actions.

Lemma 14: $g(t)$ is piecewise constant and changes only at a polynomial number of time points.

Proof: (similar to that of Lemma 9).

The analysis of $|sc|$ and $|dc|$ of UPPER-ENVELOPE-TYPE-2-AT-T is subsumed in the analysis presented for the most general version of the problem. The complexity of UPPER-ENVELOPE-TYPE-2-ALL-T (by Lemma 14) is equal to $O(|sc| + K|dc|)$, where K is the number of Type 2 actions.

4 Computing $g(t)$ for Type 3 Action Edges

In this section, we will restrict all actions to be of Type 3. Figure 5 presents the algorithm for the computation of $g(t)$ and a flexible schedule s that achieves it for a given time instant t. Figure 6 presents the algorithm for computing $g(t)$ for all t.

Lemma 15: A Type 3 action edge $A = \langle X_i, X_j \rangle$ can be made active at time t to a degree $\geq d$ if and only if it is possible to add the edge $\langle X_0, X_i \rangle$ to $D(\mathcal{G})$ without introducing any negative cycles—where $\langle X_0, X_i \rangle$ should be annotated with $t' \leq t$ and t' is such that $\int_0^{t-t'} u_A(x)dx = d$.

Proof: A Type 3 edge contributes a resource consumption of $\int_0^{t-X_i} u_A(x)dx$ by time t when X_i is scheduled before t. To ensure that A consumes $\geq d$ of resource by time t, we must have $\int_0^{t-X_i} u_A(x)dx \geq d$. Since $u_A(x) > 0$, and t' is such that $\int_0^{t-t'} u_A(x)dx = d$, we require $X_i \leq t'$. Using the semantics of the distance graph, this corresponds to the addition of the edge $\langle X_0, X_i \rangle$ annotated with t'. Also, since $d > 0$, $t' \leq t$.

Lemma 16: The size of a minimal conflict is $= 1$.

Proof: (similar to that of Lemma 4).

Lemma 17: An action edge $A = \langle X_i, X_j \rangle$ constitutes a size-1 conflict at time t when attempted to be activated

ALGORITHM: UPPER-ENVELOPE-TYPE-3-AT-T
INPUT: An instance of the resource-envelope problem with all actions of Type 3, and a time instant t.
OUTPUT: $g(t)$, the set of actions Q constituting it, and a flexible schedule s achieving it.
(1) FOR all actions $A = \langle X_i, X_j \rangle$:
(a) Compute $dist(X_i, X_0)$ in the distance graph $D(\mathcal{G})$.
(2) RETURN:
(a) $Q = \{A \mid A = \langle X_i, X_j \rangle$ and $dist(X_i, X_0) + t \geq 0\}$.
(b) $g(t) = \sum_{A \in Q} \{\int_0^{t+dist(X_i,X_0)} u_A(x)dx\}$.
(c) $s = D(\mathcal{G}) \cup \{\langle X_0, X_i \rangle$ annotated with $-\, dist(X_i, X_0) \mid A = \langle X_i, X_j \rangle \in Q\}$.

Figure 5: $g(t)$ for Type 3 actions for a specified t.

ALGORITHM: UPPER-ENVELOPE-TYPE-3-ALL-T
INPUT: An instance of the resource-envelope problem with all actions of Type 3.
RESULT: $g(t)$ for all t.
(1) FOR all action edges $A = \langle X_i, X_j \rangle$:
(a) Insert $-dist(X_i, X_0)$ into list L.
(2) Sort L in ascending order $\langle d_1, d_2 \ldots d_{|L|} \rangle$.
(3) FOR $i = 1, 2 \ldots |L|$:
(a) Compute $Q =$ UPPER-ENVELOPE-TYPE-3-AT-T at time d_i.
(b) Set $g(t) = \sum_{A \in Q} \{\int_0^{t+dist(X_i,X_0)} u_A(x)dx\}$ for all t in the interval $[d_i, d_{i+1})$ (treat $d_{|L|+1} = +\infty$).
(4) Set $g(t) = 0$ for all t in the interval $[-\infty, d_1)$.

Figure 6: $g(t)$ for Type 3 actions for all t.

to a degree $\geq d$ if $t' + dist(X_i, X_0) < 0$ where t' is such that $\int_0^{t-t'} u_A(x)dx = d$.
Proof: (similar to that of Lemma 7).
Lemma 18: $g(t) = \sum_A \{\int_0^{t+dist(X_i,X_0)} u_A(x)dx$ (such that $A = \langle X_i, X_j \rangle$ has $dist(X_i, X_0) + t \geq 0)\}$ and $s = D(\mathcal{G}) \cup \{\langle X_0, X_i \rangle$ annotated with $-\, dist(X_i, X_0) \mid A = \langle X_i, X_j \rangle \in Q\}$ (see Figure 5).
Proof: The only minimal conflicts are of size 1 and are appropriately removed in $\sum_A \{\int_0^{t-t'} u_A(x)dx$ (s.t. $A = \langle X_i, X_j \rangle$ has $dist(X_i, X_0) + t' \geq 0)\}$. Since each term in this summation is maximized with decreasing t', the smallest value of t' that does not introduce a negative cycle, together with the conditions that $t' + dist(X_i, X_0) \geq 0$ and $t' \leq t$, is $-dist(X_i, X_0)$ provided that $t + dist(X_i, X_0) \geq 0$. Put together, this means that $g(t) = \sum_A \{\int_0^{t+dist(X_i,X_0)} u_A(x)dx$ (such that $A = \langle X_i, X_j \rangle$ has $dist(X_i, X_0) + t \geq 0)\}$. The required flexible schedule s corresponds to the addition of edges required to activate the qualifying actions.
Lemma 19: $g(t)$ is piecewise continuous and is discontinuous only at a polynomial number of time points.
Proof: By the previous Lemmas, $g(t)$ is the summation of integrals over all actions $\langle X_i, X_j \rangle$ that have $t + dist(X_i, X_0) \geq 0$. Assuming that $u_A(x)$ is continuous, all integrals of the form $\int_0^{t+dist(X_i,X_0)} u_A(x)dx$ are continuous and the number of times $g(t)$ becomes discontinuous is bounded by the number of times the set of actions having $t + dist(X_i, X_0) \geq 0$ changes. Since $-dist(X_i, X_0)$ marks the membership of an action $\langle X_i, X_j \rangle$ in this set, the number of discontinuities in $g(t)$ is $O(K)$ where K is the number of Type 3 actions.

The analysis of $|sc|$ and $|dc|$ of UPPER-ENVELOPE-TYPE-3-AT-T is subsumed in the analysis presented for the most general version of the problem. The complexity of UPPER-ENVELOPE-TYPE-3-ALL-T (by Lemma 19) is equal to $O(|sc| + K\Im|dc|)$, where K is the number of Type 3 actions and $\Im$ is the complexity of symbolic integration.

5 Computing $g(t)$ for Hybrid Action Edges

We will now deal with the most general version of the resource-envelope problem—allowing for Type 1, Type 2 and Type 3 actions. Figure 9 presents an algorithm that discretizes a Type 3 action into a series of Type 2 actions such that a solution for the latter can be easily translated to a solution for the former. Interestingly, the discretization depends on the Type 1 actions present in the problem instance, and the time instant t. Figure 7 presents the algorithm for the discretization, and Figure 8 presents the algorithm for solving the modified problem. A series of Lemmas are presented that prove the correctness of these algorithms. The correctness of Figure 8 follows directly from the arguments presented in previous sections.
Lemma 20: The size of a minimal conflict is ≤ 2 (in attempting to make various actions active).

ALGORITHM: DISCRETIZE-TYPE-3-TO-TYPE-2
INPUT: An instance of the generalized resource-envelope problem, and a time instant t.
RESULT: Discretization of all Type 3 actions.
(1) FOR all Type 3 action edges $C = \langle X_i^c, X_j^c \rangle$:
(a) FOR all Type 1 action edges $A = \langle X_i^a, X_j^a \rangle$:
(i) Insert $dist(X_i^c, X_j^a)$ into List D iff it is ≥ 0.
(b) Insert $t + dist(X_i^c, X_0)$ into List D iff it is ≥ 0.
(c) Insert $UB(C)$ into List D.
(d) Sort D in ascending order $\langle d_1, d_2 \ldots d_{|D|} \rangle$.
(e) FOR $h = 1, 2 \ldots |D|$:
(i) Create a Type 2 action $B_h = \langle X_i^{B_h}, X_j^{B_h} \rangle$ so that:
(A) $X_i^{B_h}$ and $X_j^{B_h}$ are new nodes in $D(\mathcal{G})$.
(B) $\langle X_i^{B_h}, X_j^{B_h} \rangle$ and $\langle X_j^{B_h}, X_i^{B_h} \rangle$ are edges annotated with ϵ and $-\epsilon$ respectively (for any $\epsilon > 0$).
(C) $\langle X_i^c, X_i^{B_h} \rangle$ and $\langle X_i^{B_h}, X_i^c \rangle$ are edges annotated with d_h and $-d_h$ respectively.
(D) Set $w_{B_h} = \int_{d_{h-1}}^{d_h} u_C(t)dt$ (use $d_0 = 0$ when $h = 1$).
(f) Set $u_C(t) = 0$ (i.e. remove action C from further consideration).

Figure 7: Discretizing Type 3 actions to Type 2 actions in the context of the Type 1 actions and t.

ALGORITHM: UPPER-ENVELOPE-TYPE-1-2-AT-T
INPUT: An instance of the resource envelope problem with all actions of Type 1 or Type 2, and a time instant t.
OUTPUT: $g(t)$ and a flexible schedule s achieving it.
(1) FOR all pairs of Type 1 actions $A = \langle X_{i_1}, X_{j_1} \rangle$ and Type 1 or Type 2 actions $B = \langle X_{i_2}, X_{j_2} \rangle$:
(a) Construct a (directed) size-2 conflict $A \rightarrow B$ iff $dist(X_{i_2}, X_{j_1})$ in the distance graph $D(\mathcal{G})$, is < 0.
(2) Construct a node-weighted directed graph $E(\mathcal{G})$ as follows:
(a) The nodes of $E(\mathcal{G})$ correspond to actions.
(b) The weight on a node corresponding to action A, is w_A.
(c) A directed edge $\langle A, B \rangle$ in $E(\mathcal{G})$ encodes a size-2 conflict $A \rightarrow B$.
(3) Construct a node-weighted directed graph $M(\mathcal{G})$ from $E(\mathcal{G})$ as follows:
(a) Remove a node $A = \langle X_i, X_j \rangle$ if $\{t + dist(X_i, X_0) < 0 \vee dist(X_0, X_j) - t < 0\}$ and A is of Type 1.
(b) Remove a node $A = \langle X_i, X_j \rangle$ if $t + dist(X_i, X_0) < 0$ and A is of Type 2.
(4) Compute $Q = \{A_{q_1}, A_{q_2} \ldots A_{q_k}\}$ as the largest weighted anti-chain in $M(\mathcal{G})$.
(5) RETURN:
(a) $g(t) = |Q| = \sum_{p=1}^{k} w_{A_{q_p}}$.
(b) $s = D(\mathcal{G}) \cup \{\langle X_0, X_i^{q_p} \rangle$ annotated with t and $\langle X_j^{q_p}, X_0 \rangle$ annotated with $-t \mid \langle X_i^{q_p}, X_j^{q_p} \rangle$ is of Type 1 and is in $Q\}$ $\cup$ $\{\langle X_0, X_i^{q_p} \rangle$ annotated with $t \mid \langle X_i^{q_p}, X_j^{q_p} \rangle$ is of Type 2 and is in $Q\}$.

Figure 8: $g(t)$ for Type (1,2) actions for a specified t.

Proof: (similar to that of Lemma 4).

Lemma 21: The size-2 conflicts may not be independent of t.

Proof: A size-2 conflict should involve an incoming edge (to X_0) of a Type 1 action and an outgoing edge (from X_0) of either a Type 1, Type 2 or Type 3 action. It is only in the final case that the weights on the incoming and outgoing edges do not cancel each other, but are $-t$ and t' (required to activate a Type 3 action to a certain degree) respectively. In such a case, a size-2 conflict arises between a Type 1 action $A = \langle X_i^a, X_j^a \rangle$ and a Type 3 edge $C = \langle X_i^c, X_j^c \rangle$ when $t' + dist(X_i^c, X_j^a) - t < 0$.

Lemma 22: For a Type 3 action $C = \langle X_i^c, X_j^c \rangle$, the schedule achieving $g(t)$ must either have $t + dist(X_i^c, X_0) < 0$ or incorporate the addition of the edge $\langle X_0, X_i^c \rangle$ annotated with t'_{opt} such that $t - t'_{opt} \in D$ (see Figure 7).

Proof: When $t + dist(X_i^c, X_0) < 0$, C cannot contribute to the total amount of resource consumed by time t because the addition of $\langle X_0, X_i^c \rangle$ annotated with t'_{opt} constrained to be $\leq t$ causes a size-1 conflict. When this does not happen, we will show that $t'_{opt} \in D'$ where $d'_i \in D'$ is $t - d_i$. Suppose $t'_{opt} \notin D'$. Let L be in D' such that it is closest to t'_{opt} and smaller than it. L exists because D is known to contain $t + dist(X_i^c, X_0)$ and it cannot be the case that $t'_{opt} < t - (t + dist(X_i^c, X_0))$ as this would cause the size-1 conflict $t'_{opt} + dist(X_i^c, X_0) < 0$. The set of Type 1 actions that C conflicts with are the same whether $\langle X_0, X_i^c \rangle$ is annotated with t'_{opt} or L. This is because if there is any Type 1 action $A = \langle X_i^a, X_j^a \rangle$ such that $t'_{opt} + dist(X_i^c, X_j^a) - t \geq 0$ but $L + dist(X_i^c, X_j^a) - t < 0$, then $t - dist(X_i^c, X_j^a)$ is $> L$ but $\leq t'_{opt}$. Further, since $t'_{opt} \leq t$, $dist(X_i^c, X_j^a) \geq 0$, and by construction, $t - dist(X_i^c, X_j^a)$ belongs to D'. This contradicts that L is the closest element in D' that is lesser than t'_{opt}. It also cannot be the case that annotating $\langle X_0, X_i^c \rangle$ with L causes a size-1 conflict when annotating it with t'_{opt} does not cause one. Suppose this were possible, then $dist(X_i^c, X_0) + L < 0$ and $dist(X_i^c, X_0) + t'_{opt} \geq 0$. Since $t \geq t'_{opt}$, $dist(X_i^c, X_0) + t \geq 0$ and belongs to D. This implies that $-dist(X_i^c, X_0)$ is present in D' and is required to be $> L$ but $\leq t'_{opt}$—contradicting the construction of L. Finally, since the conflicts remain the same in both cases, the POSET remains the same except that in the case where $\langle X_0, X_i^c \rangle$ is annotated with L, the contribution of C is $\int_0^{t-L} u_C(x)dx$, which is greater than $\int_0^{t-t'_{opt}} u_C(x)dx$. The size of the largest weighted anti-chain will therefore be necessarily greater in the case of L, proving that $t'_{opt} \in D'$.

ALGORITHM: UPPER-ENVELOPE-HYBRID-AT-T
INPUT: An instance P of the generalized resource envelope problem, and a time instant t.
OUTPUT: $g(t)$ and a schedule s that achieves it.
(1) DISCRETIZE-TYPE-3-TO-TYPE-2 to obtain instance P'.
(2) RETURN:
(a) $g(t)_{[P]} = g(t)_{[P']}$.
(b) $s_{[P]} = s_{[P']}$ with the modification that if the series of Type 2 actions $B_1, B_2 \dots B_{|D|}$ ($B_k = \langle X_i^{B_k}, X_j^{B_k} \rangle$) in P' corresponds to the discretization of the Type 3 action $C = \langle X_i^c, X_j^c \rangle$ in P, then $s_{[P]}$ contains $\langle X_0, X_i^c \rangle$ annotated with $t - d_h$, where h is the highest number such that $\langle X_0, X_i^{B_h} \rangle$ is annotated with t in $s_{[P']}$.

Figure 9: $g(t)$ for the most general version of the problem.

Lemma 23: For a Type 3 action C and its discretization into the Type 2 actions $B_1, B_2 \dots B_{|D|}$ (see Figure 7), $\int_0^{UB(C)} u_C(x)dx = \sum_{h=1}^{|D|} w_{B_h}$.
Proof: We know that $\sum_{h=1}^{|D|} w_{B_h} = \int_0^{d_1} u_C(x)dx + \int_{d_1}^{d_2} u_C(x)dx \dots \int_{d_{|D|-1}}^{d_{|D|}} u_C(x)dx = \int_0^{d_{|D|}} u_C(x)dx$. Since D is known to contain $UB(C)$, $\sum_{h=1}^{|D|} w_{B_h} = \int_0^{UB(C)} u_C(x)dx$.
Lemma 24: For a given time instant t, the discretization of the Type 3 action C to the Type 2 actions $B_1, B_2 \dots B_{|D|}$ produces equivalent results for the computation of $g(t)$ (see Figure 7).
Proof: We have to prove the equivalence of C and $B_1, B_2 \dots B_{|D|}$ with respect to both size-1 conflicts and size-2 conflicts. By previous Lemmas, no higher order conflicts need to be considered. First, consider the equivalence with respect to size-1 conflicts. We have to prove that for the given time t, the sum of the weights on B_h ($h = 1, 2 \dots |D|$) that do not exhibit a size-1 conflict is equal to the maximum degree to which we can activate C without allowing it to exhibit a size-1 conflict. By the previous Lemma, this would also prove that the sum of weights on B_h ($h = 1, 2 \dots |D|$) that exhibit a size-1 conflict at time t is equal to the conflicting part of C. Consider the sum of all w_{B_h} such that $B_h = \langle X_i^{B_h}, X_j^{B_h} \rangle$ can have the successful addition of $\langle X_0, X_i^{B_h} \rangle$ annotated with t. This is possible only when $dist(X_i^{B_h}, X_0) + t \geq 0$. But we know that $dist(X_i^{B_h}, X_0) = dist(X_i^{B_h}, X_i^c) + dist(X_i^c, X_0)$ because the only node that $X_i^{B_h}$ is connected to is X_i^c. By construction, $dist(X_i^{B_h}, X_i^c)$ is $-d_h$. Therefore, we need to qualify all B_h such that $-d_h + dist(X_i^c, X_0) + t \geq 0$—i.e. $d_h \leq t + dist(X_i^c, X_0)$. For the activation of C, we require the lowest t' such that $t' \geq -dist(X_i^c, X_0)$. The corresponding resource consumption by C would be $\int_0^{t-t'} u_C(x)dx = \int_0^{t+dist(X_i^c, X_0)} u_C(x)dx$. By construction, $t + dist(X_i^c, X_0)$ is present in D (if $t + dist(X_i^c, X_0) < 0$, C and $B_1, B_2 \dots B_{|D|}$ are trivially equivalent). Let this be d_g. The required weight of all w_{B_h} such that $d_h \leq d_g$ is $\int_0^{d_1} u_C(x)dx + \int_{d_1}^{d_2} u_C(x)dx \dots \int_{d_{g-1}}^{d_g} u_C(x)dx = \int_0^{d_g} u_C(x)dx = \int_0^{t+dist(X_i^c, X_0)} u_C(x)dx$ as required. Now consider the equivalence of C and $B_1, B_2 \dots B_{|D|}$ with respect to size-2 conflicts. It suffices to prove that for any Type 1 action edge $A = \langle X_i^a, X_j^a \rangle$, and a given time t, the sum of the weights of w_{B_h} ($h = 1, 2 \dots |D|$) that do not conflict with A is equal to the maximum value of $\int_0^{t-t'} u_C(x)dx$ such that $t' - t + dist(X_i^c, X_j^a) \geq 0$. By the previous Lemma, this would also prove that the sum of weights on B_h ($h = 1, 2 \dots |D|$) that exhibit a size-2 conflict with A at time t is equal to the conflicting part of C with A. Consider the sum of all w_{B_h} that do not exhibit a size-2 conflict with A. This happens when $B_h = \langle X_i^{B_h}, X_j^{B_h} \rangle$ is such that we can have a successful addition of the edge $\langle X_0, X_i^{B_h} \rangle$ annotated with t satisfying $t + dist(X_i^{B_h}, X_j^a) - t \geq 0$. We have $dist(X_i^{B_h}, X_j^a) = dist(X_i^{B_h}, X_i^c) + dist(X_i^c, X_j^a)$ because the only node that $X_i^{B_h}$ is connected to is X_i^c. Also, by construction, $dist(X_i^{B_h}, X_i^c) = -d_h$. Therefore, we require to qualify all w_{B_h} such that $dist(X_i^c, X_j^a) - d_h \geq 0$—i.e. $d_h \leq dist(X_i^c, X_j^a)$. Now consider the value of t' that allows for the maximum resource consumption made by C without conflicting with A. This happens when we have the least possible value of t' such that $t' - t + dist(X_i^c, X_j^a) \geq 0$—i.e., $t - t' = dist(X_i^c, X_j^a)$. By construction, $dist(X_i^c, X_j^a)$ is present in D (if $dist(X_i^c, X_j^a) < 0$, C and $B_1, B_2 \dots B_{|D|}$ are trivially equivalent). Let this be d_g. The maximum resource consumption made by C before it conflicts with A is then $\int_0^{t-t'} u_C(x)dx = \int_0^{dist(X_i^c, X_j^a)} u_C(x)dx$. The required weight of all w_{B_h} such that $d_h \leq d_g$ is $\int_0^{d_1} u_C(x)dx + \int_{d_1}^{d_2} u_C(x)dx \dots \int_{d_{g-1}}^{d_g} u_C(x)dx = \int_0^{d_g} u_C(x)dx = \int_0^{dist(X_i^c, X_j^a)} u_C(x)dx$, hence establishing the truth of the Lemma.
Lemma 25: UPPER-ENVELOPE-HYBRID-AT-T is correct (see Figure 9).

ALGORITHM: LOWER-ENVELOPE-HYBRID
INPUT: An instance of the generalized resource-envelope problem P.
OUTPUT: $h(t)$ and a flexible schedule s achieving it.
(1) Construct P_1 from P as follows:
(a) FOR all Type 3 action edges $C = \langle X_i, X_j \rangle$:
(i) Compute $w_C = \int_0^{UB(C)} u_C(t)dt$.
(b) FOR all action edges (Type 1, 2 or 3) $A = \langle X_i, X_j \rangle$:
(i) Add the two complementary edges $\langle X_{-\infty}, X_i \rangle$ and $\langle X_j, X_{+\infty} \rangle$ ($X_{-\infty} = -\infty$ and $X_{+\infty} = +\infty$).
(ii) Make $\langle X_{-\infty}, X_i \rangle$ a Type 1 action with weight w_A.
(iii) Make $\langle X_j, X_{+\infty} \rangle$ a Type 1 action edge with weight w_A only if A is of Type 1.
(iv) Set the weight of A to 0 if it is of Type 1 or Type 2 and assign the rate function $u_A^{P_1}(t)$ if it is of Type 3. $u_A^{P_1}(t)$ satisfies $\int_0^t u_A^{P_1}(x)dx = \int_0^{UB(A)} u_A(x)dx - \int_0^t u_A(x)dx$.
(2) Construct P_2 from P_1 as follows:
(a) FOR all temporal edges $\langle X_i, X_j \rangle$ in P_1 annotated with t:
(i) Reverse $\langle X_i, X_j \rangle$ in P_2 to get $\langle X'_j, X'_i \rangle$ annotated with t.
(b) FOR a Type 3 action edge $C = \langle X'_j, X'_i \rangle$ (originally $\langle X_i, X_j \rangle$ in P_1):
(i) Reassign rate function $u_C^{P_2}(t) = u_C(UB(C) - t)$.
(3) RETURN:
(a) $h(t)_{[P]} = \sum_{\text{all actions } A} w_A - g(-t)_{[P_2]}$.
(b) $s_{[P]} = -s_{[P_2]}$.

Figure 10: $h(t)$ for the most general version of the problem.

Proof: The previous Lemma proves that $g(t)_{[P]} = g(t)_{[P']}$ and it remains to prove the correctness of $s_{[P]}$. If $B_h = \langle X_i^{B_h}, X_j^{B_h} \rangle$ is active in P', it must incorporate the addition of the edge $\langle X_0, X_i^{B_h} \rangle$ annotated with t without causing a negative cycle. This means that $dist(X_i^{B_h}, X_0) \geq -t$. Since X_i^c is the only node that $X_i^{B_h}$ is connected to, and this edge is annotated with $-d_h$, we have $dist(X_i^{B_h}, X_i^c) + dist(X_i^c, X_0) \geq -t$ or equivalently $dist(X_i^c, X_0) \geq -t + d_h$. This implies that C incorporates the addition of the edge $\langle X_0, X_i^c \rangle$ annotated with $t - d_h$ without creating any negative cycle. Since this is true for any active B_h, the maximum activation of C is obtained by considering the largest such h (assuming increasing order of $d_1, d_2 \ldots d_{|D|}$).

The $|sc|$ of UPPER-ENVELOPE-HYBRID-AT-T is dominated by the computation of integrals and the computation of shortest paths in $D(\mathcal{G})$. If K_1, K_2 and K_3 are respectively the number of Type 1, Type 2 and Type 3 actions, integrals are computed $O((K_1{+}1)K_3)$ times and there are $O((K_1 + 1)(K_1 + K_2 + K_3))$ pairs of nodes between which shortest paths are computed. Since $D(\mathcal{G})$ can have negative edges, the complexity of shortest path computation using the Bellman-Ford algorithm is $O(|\mathcal{X}||\mathcal{E}|)$. Hence, $|sc| = O((K_1 + 1)K_3\Im + (K_1 + 1)(K_1 + K_2 + K_3)|\mathcal{X}||\mathcal{E}|)$. Here, $\Im$ is the complexity of symbolic integration. The $|dc|$ of UPPER-ENVELOPE-HYBRID-AT-T is dominated by the time dependent discretization of Type 3 actions, and the computation of the largest weighted anti-chain in a POSET having at most $O(K_1 + K_2 + K_1K_3)$ nodes. Since the latter is related to a staged *max-flow* in a bipartite graph (see [1] and [4]), $|dc| = O(K_3\Im + (K_1 + K_2 + K_1K_3)^{2.5})$.

6 Computing the Lower Envelope $h(t)$

Given an instance P of the generalized resource-envelope problem, we show how to construct an instance Q such that a solution for $g(t)$ (and a flexible schedule that achieves it) on Q can be translated to a solution for $h(t)$ (and a flexible schedule that achieves it) on P. Figure 10 presents this idea, and a series of Lemmas are presented that prove its correctness.

Lemma 26: $h(t)_{[P]} = \sum_{\text{all actions } A} w_A - g(t)_{[P_1]}$ (see Figure 10).

Proof: For every schedule s, if $A = \langle X_i, X_j \rangle$ is a Type 1 action edge that is active in P, then it contributes a weight of 0 in P_1, and if it is not active in P, exactly one of its complementary edges $\langle X_{-\infty}, X_i \rangle$ or $\langle X_j, X_{+\infty} \rangle$ is active in P_1 and contributes a weight of w_A. Similarly, if A is a Type 2 action edge that is active in P, it contributes a weight of 0 in P_1, and if it is not active in P, its complementary edge $\langle X_{-\infty}, X_i \rangle$ contributes w_A in P_1. Again, if A is a Type 3 action edge that is not active in P, it contributes w_C in P_1, and if it is active to the degree d in P, it contributes $w_C - d$ in P_1. The foregoing statements imply that for a given schedule s, $(C_s(t))_{[P]} = \sum_A w_A - (C_s(t))_{[P_1]}$. Also, a schedule is consistent for P if and only if it is so for P_1 (because they have the same temporal constraints). Together, we can conclude that $h(t)_{[P]} = \sum_{\text{all actions } A} w_A - g(t)_{[P_1]}$.

However, $g(t)$ cannot be computed directly in P_1 because $u_A^{P_1}(t)$s defined on its Type 3 edges are negative, and the discretization of Type 3 edges for computing $g(t)$ was performed under the assumption that these rates are positive. P_1 is therefore transformed to P_2 such that $g(t)_{[P_1]} = g(-t)_{[P_2]}$ and yet, $u_C^{P_2}(t)$s defined on the Type 3 edges of P_2 are positive, hence making the application of the discretization procedure possible.

Lemma 27: $g(t)_{[P_1]} = g(-t)_{[P_2]}$ (see Figure 10).

Proof: For a constraint $X - Y \leq r$ in P_1, we would have $Y' - X' \leq r$ in P_2. These two constraints are equivalent if $Y' = -Y$ and $X' = -X$. Hence, s is a consistent schedule in P_1 if and only if $-s$ is a consistent schedule in P_2. If a Type 1 action $\langle X_i, X_j \rangle$ is active at time t in P_1, then the Type 1 action $\langle X_j', X_i' \rangle$ is active at time $-t$ in P_2. This is because the conditions $X_i \leq t$ and $X_j \geq t$ in P_1 and $X_j' \leq -t$ and $X_i' \geq -t$ in P_2 are equivalent when $X_i' = -X_i$ and $X_j' = -X_j$. Similarly, if a Type 3 action $\langle X_i, X_j \rangle$ is active at time t to a degree d in P_1, then the Type 3 action $\langle X_j', X_i' \rangle$ is active at time $-t$ to the same degree d in P_2. This can be established if we can prove that $\int_0^{t-X_i} u_C^{P_1}(x)dx = d$ is equivalent to $\int_0^{-t-X_j'} u_C^{P_2}(x)dx = d$. We have $\int_0^{t-X_i} u_C^{P_1}(x)dx = \int_0^{UB(C)} u_C(x)dx - \int_0^{t-X_i} u_C(x)dx = \int_{t-X_i}^{UB(C)} u_C(x)dx$. Since $X_j = X_i + UB(C)$ (because $LB(C) = UB(C)$ for any Type 3 action C), we have $\int_{t-X_i}^{UB(C)} u_C(x)dx = \int_{t-(X_j-UB(C))}^{UB(C)} u_C(x)dx = \int_{UB(C)-(X_j-t)}^{UB(C)} u_C(x)dx$. In P_2, since $X_j' = -X_j$, we have $\int_0^{-t-X_j'} u_C^{P_2}(x)dx = \int_0^{X_j-t} u_C^{P_2}(x)dx = \int_0^{X_j-t} u_C(UB(C)-x)dx = \int_{UB(C)-(X_j-t)}^{UB(C)} u_C(x)dx$ as required. Finally, since there are no Type 2 actions in the transformation of P to P_1, the truth of the Lemma is established.

Lemma 28: $h(t)_{[P]} = \sum_{\text{all actions } A} w_A - g(-t)_{[P_2]}$ and $s_{[P]} = -s_{[P_2]}$.

Proof: Follows directly from the arguments presented in the proofs of the previous two Lemmas.

The complexity of LOWER-ENVELOPE-HYBRID is dominated by the computation of $g(-t)_{[P_2]}$ and is therefore equal to that of UPPER-ENVELOPE-HYBRID-AT-T.

7 Conclusions and Future Work

We described efficient algorithms for variants of the resource-envelope problem (including the generalized version) by reducing them to largest weighted anti-chain computations in a POSET. The algorithms are constructive—producing flexible schedules that actually achieve the upper or lower bounds. We expect that this will potentially help us to save exponential amounts of work during search, along with providing good heuristic values to guide search, when compared to looser bounds (see [3], [5] and [6]). We are currently working on algorithms for plan dispatching that employ the resource-envelope computations as subroutines.

References

[1] Cormen T. H., Leiserson C. E. and Rivest R. L. 1990. Introduction to Algorithms. *Cambridge, MA, 1990*.

[2] Dechter R., Meiri I. and Pearl J. 1991. Temporal Constraint Networks. *AI Journal 49. 1991*.

[3] Kumar T. K. S. 2003. Incremental Computation of Resource-Envelopes in Producer-Consumer Models. *Proceedings of CP 2003*.

[4] Kumar T. K. S. 2004. A Polynomial-time Algorithm for Simple Temporal Problems with Piecewise Constant Domain Preference Functions. *Proceedings of AAAI 2004*.

[5] Laborie P. 2001. Algorithms for Propagating Resource Constraints in AI Planning and Scheduling: Existing Approaches and New Results. *Proceedings of ECP 2001*.

[6] Muscettola N. 2002. Computing the Envelope for Stepwise-Constant Resource Allocations. *CP 2002*.