A Multi-Decompositional Approach

to Integration of Planning and Scheduling - An Applied Perspective

Alexander Kott
Victor Saks

Carnegie Group, Inc.
5 PPG Place
Pittsburgh, Pennsylvania 15222
akott@cgi.com, saks@cgi.com

Abstract

This paper proposes a multi-decompositional approach to integrating planning and scheduling. In many practical domains, planning and scheduling problems are tightly intertwined: the right decomposition for a given goal cannot be determined without considering scheduling. Commonly used approaches suffer from limited look-ahead. The key steps of the proposed approach are: (a) perform and store ***multiple decompositions*** for each goal or activity that require decomposition and then (b) identify the best selections among the combined set of alternative decompositions. To accomplish the latter – selection and scheduling within the space of multiple alternative decompositions -- we propose a novel extension of Constraint-Directed Heuristic Search. Several applications of the approach to practical large-scale systems in domains such as logistics and transportation are described.

1. Introduction

In this paper, we use term ***planning*** to denote the problem of identifying activities and relations between the activities, required to accomplish a given set of goals. In the context of this paper, we limit our attention to the ***decompositional*** approaches to planning – the family of approaches in which goals are decomposed into networks of activities which satisfy the goals, and in which aggregate activities are decomposed into sub-activities. By ***scheduling*** we mean the assignment of specific resources and time windows to the activities of the plan.

In our experience with constructing applied planning/scheduling systems for large-scale, real-world problems, we observe that the two sub-problems – planning and scheduling – are usually rather tightly intertwined. There are at least two important aspects in this tight interdependency between the two problems.

First, when the planner (either computerized or human) selects a suitable activity decomposition for a given goal, the appropriate selection often depends on time-dependent availability of resources and on time-dependent state of the environment; these in turn are dependent on the decompositions (and scheduling of sub-activities) for other goals. Thus, it is often difficult or impossible to make a "good" selection of a suitable decomposition without having assigned time and resources of the higher-level goal or activity, as well as without considering scheduling of other activities of the overall plan. In short, it is difficult to perform the key element of planning — decomposition of activities — without performing scheduling at the same time. We discuss specific examples later in this paper.

Second, when the planner performs scheduling (assignment of resources and time windows to the activities), it is important for the planner to have a global view of the required set of activities. Ideally, all decompositions should be performed before the scheduling decision is made, yet it is difficult to accomplish, for the reasons we just discussed above.

One can think of a number of approaches to solving this dilemma. We name several common approaches below, and then discuss their strengths and shortcomings:

- Perform decomposition for all goals and/or aggregate activities based on heuristic rules, without detailed scheduling considerations, but possibly taking into account gross availability of resources and resource assignment preferences. This results in a complete plan which is then submitted to a scheduler. Let us call this family of approached CDPS (Complete Decomposition Prior to Scheduling).
- Perform decomposition on a subset of goals (perhaps only one goal at a time) and then schedule the currently known set of activities, without awaiting decompositions of all goals. This sequence of partial decomposition and partial scheduling is repeated until a complete plan/schedule is produced. Let us call this family of approached IDIS (Incremental Decomposition and Incremental Scheduling).

- Combine IDIS with a search technique, usually involving a significant amount of backtracking: when either the planning or scheduling increment is found to be infeasible, revise an earlier decision, either regarding the choice of decomposition or the scheduling choice. Let us call this family of approached IDISB (Incremental Decomposition and Incremental Scheduling with Backtracking).

From the applied perspective, we observe the following strengths and weaknesses in each of the families of approaches:

- CDPS offers the "cleanest" separation between the planning and scheduling sub-problems. Both the planning and the scheduling components can be implemented with simpler, more efficient and robust algorithms. The key disadvantage is the limited ability of the planner to foresee the situations where the chosen decomposition cannot be satisfied with the available resources. This disadvantage is particularly acute when (a) there are multiple decompositions available for each goal and (b) there are bottleneck resources that could be scheduled more efficiently if different activity decompositions were chosen.
- IDIS provides the opportunity for both the scheduler and the planner to act in a well-informed manner: the planner knows all scheduling decisions made for the completed portion of the plan, and the scheduler knows all activities planned up to the given point in the solution process. As a side benefit of this approach, we find that it is well suited for mixed-initiative planning and scheduling (unlike CDPS): when the human user reviews the newly proposed incremental set of activities, both planning and scheduling decisions are available for review and modification. However, IDIS suffers from limited look-ahead, particularly in common real-world problems with over-constrained resources.

IDISB alleviates the problem of the limited look-ahead, but at the expense of slower and less predictable performance. This disadvantage is especially acute in applications that involve dynamic and near-real-time re-planning and rescheduling, which recently are becoming more common.

2. Overview of Our Approach

Our work in the last decade has focused on constructing large planning and scheduling applications. The planning / scheduling problems addressed in these applications are often characterized by:

- Large number of goals, activities, resources and constraints;
- Capacity constraints are dominant;
- Large fraction of the resources are over-constrained (multiple bottlenecks);
- Significant number of alternative decompositions are possible for many of the goals;
- Rapid and robust capability to re-plan and reschedule is required;
- Mixed-initiative planning and scheduling is a desirable but not a critical concern.

Given these characteristics, we were motivated to find an approach that overcomes the shortcomings we discussed earlier in reference to the CDPS, IDIS and IDISB families of approaches. In particular, we were looking for:

- Ability to deal with multiple over-constrained resources;
- Ability to match activity decompositions with resource availability;
- Rapid and predictable execution times.

The approach we developed in response to the above requirements can be summarized as follows:

(a) perform and store ***multiple decompositions*** for each goal or activity that require decomposition and then
(b) identify the best selections among the combined set of alternative decompositions ***during*** the scheduling process.

Assuming that the first aspect of the approach - production of multiple decompositions – can be addressed by the use of domain-specific heuristics or specialized algorithms, the key challenge of the approach is its second aspect - selection of the subset of alternative decompositions. We elected to use Constraint-Directed Heuristic Search (CDHS) as a technique for addressing the latter challenge and developed a novel extension of CDHS that uses multiple decomposition for computing the constraint textures (section 3).

In the remainder of this paper we focus on (a) discussion of the multiple decompositions approach in more detail (section 3), and (b) describing our practical experiences with the approach (sections 4 and 5).

3. Scheduling within the Space of Multiple Alternative Decompositions

We elected to develop and use a novel extension of the CDHS technique for selecting and scheduling the alternative decompositions[1]. We extend a particular instantiation of Constraint-Directed Heuristic Search [Fox, Sadeh & Baykan 1989], which uses the texture measures

[1] However, one can argue that the key idea of our approach – scheduling within the space of multiple alternative decompositions – is not dependent on the choice of a specific allocation and scheduling method, and that other techniques could be used for this purpose.

of ***contention and reliance*** [Sadeh 1991]. Our approach differs from the CDHS because it (a) uses multiple decompositions in computing the texture measures, and (b) delays the commitment to a particular decomposition until maximum possible information is available about the scheduling aspect of the problem.

Our algorithm for selecting and scheduling within the space of multiple alternative decompositions can be summarized (in a somewhat simplified form) as follows.

Inputs: set of goals {G}; set of resources {R}.
Outputs: for each goal G – a network of activities {A(G)}; for each activity A - a resource R and a time interval (T1, T2).

1. For each goal G in a given set of goals {G}, domain-specific rules or algorithms are used to construct a collection of up to N alternative activity decompositions {D(G)}. Each D(G) is a network of activities {A} that are able to produce G. For example, in a transportation problem, a goal can be "Move Container-X from Port-A to Port-B, subject to time window constraints TW1 and TW2." A collection of alternative decompositions for this goal may consist of several transportation plans, each one of the type "Repackage container into 10 pallets; move 5 of them from Port-A to Port-C by Plane-123, then to Port-B by Train-456... etc." Each D in {D(G)} is assigned a heuristic "goodness score" g(D) that reflects how well D can satisfy G, using domain-specific rules.
2. For each resource R and for each time interval, each activity A expresses its expected demand for R (e.g., the repackaging activity will request a certain amount of capacity of repackaging resources at Port-A), taking into account its domain-specific constraints and preferences. Continuing our transportation example, constraints may include compatibility of cargo and vehicles, capacity of ports and vehicles, lowest-cost preferences, etc.
3. Each resource R (e.g., a vehicle or a port) accumulates its expected demand and calculates its contention c(R) over time as a function of the accumulated demand and available supply or capacity. (Similarly to the technique of [Sadeh 91].)
4. Each activity calculates its reliance on each resource/time (r(A, R) is the reliance of activity A on resource R) as a function of its expected demand, its priority, and the activity decompositions to which it belongs. (Similarly to the technique of [Sadeh 91].)
5. For each goal G, and each resource R, a combined reliance measure r(G, R) is computed by "rolling up" the reliance measures r(A, R) of all activities within {D(G)}.
6. The scheduler selects the resource with highest c(R) (Most Contended Resource – MCR).
7. The scheduler selects a goal with the highest reliance on the MCR (Most Reliant Goal – MRG).
8. The scheduler selects one decomposition D* among {D(MRG)} using a heuristic combination of two criteria: (1) the domain-specific "goodness score" g(D) and (2) the demand of D for MCR. Higher g(D) and lower demand for MCR are preferable.
9. All activities within D* are scheduled against {R}.
10. Constraints are propagated. These usually include temporal and capacity constraints.
11. The propagation of constraints after the assignment may render some of the decompositions infeasible (e.g., no space is left on Train-456). Infeasible decompositions are removed and the planner is requested to construct additional decompositions in order to keep the required size N of {D(G)} for every G. (The planner may not be able to construct additional decompositions.)
12. The process is completed when for each goal (except those for which no feasible decompositions can be found), a decomposition has been selected and each activity of the decomposition is scheduled.

Several comments about this algorithm:

1. It differs from the CDHS of [Sadeh 91] mainly in steps 5, 7, 8, 9 and 11, i.e., where multiple decompositions are used in computing and utilizing the contention and reliance measures. When N=1 (i.e., only a single decomposition is used), the algorithm reduces approximately to the CDHS of [Sadeh 91].
2. Steps 1, 2 and 8 involve an extensive use of domain-specific information, procedures and rules. This is motivated by our focus on real-world problems which usually require taking into account large amounts of domain-specific preferences, etc.
3. There is no backtracking in this algorithm. Our preference for no-backtracking algorithms is also motivated by the pragmatic demands of developing large applied systems: relative ease of development, debugging and controlling the system without the complications of backtracking. Adding backtracking would be a worthwhile research direction.
4. There are no claims about the completeness of this algorithm. When some goals deem infeasible, it is assumed that there is no solution that satisfies all the goals. Our reason for not pursuing this issue with greater vigor is also pragmatic – most of our real-world problems are over-constrained.

4. Implementation and Practical Experience

We explored applications of this approach—fully or partially—in several problem domains: logistics distribution, medical evacuation, transportation, and maneuver planning. In the following brief descriptions, we focus on the domain-specific instantiations of the key aspects of the multi-decompositional approach: goals, decompositions, resources.

4.1 Logistics Distribution

The Knowledge-Based Logistics Planning Shell (KBLPS) (Saks, Kepner, and Johnson 1992; Saks, Johnson, and Fox 1993) was developed for US Army. Among other functions, KBLPS automatically generates a distribution plan for an Army corps. Such a plan involves routing and scheduling of large number (on the order of 5000) goals (of the type "deliver product P to the point X within time window W"). The plan strives to maximize fulfillment of the goals in an environment where resources (such as inventory of supplies, storage facilities and transportation assets) are usually over-constrained.

For a given goal, a decomposition consists of activities to move product through the distribution network, from rear supply points to forward supply points. The origin supply point is not given and the product may be moved from any supply point where the product is available. Different origin points give rise to different activity decompositions. As product or a transportation asset is consumed (allocated) during the scheduling process, its unavailability renders some of the corresponding activity decomposition infeasible, and triggers the construction of more alternatives. Similarly, using different transportation modes to move the product between supply points requires different activities, leading to multiple alternative activity decompositions.

Chronologically, KBLPS was the first of the systems where we recognized and explored the need for integrated planning and scheduling. In KBLPS we explored only some of the aspects of the multi-decompositional approach presented in this paper: the algorithm of KBLPS originally generates only one alternative; then additional alternatives are generated when the solution process leads to the infeasibility of the available decomposition. KBLPS produced good solutions without backtracking, demonstrated excellent performance (complete plans generated in a few minutes) and is currently used by U.S. Army logisticians at military sites in the US, Korea and Europe.

4.2 Medical Evacuation

TRAC2ES (US TRANSCOM Regulating and Command & Control Evacuation System) includes the capability to plan the process of evacuating patients on air-medical missions to hospitals that has the capabilities to treat them (Kott and Saks 1996; Saks et al. 1997). A typical goal is of the type "move Patient-X from Location-L to any suitable hospital as soon as possible after earliest pick-up Time-T." A number of such goals within a given plan may be on the order of several thousands. Resources, frequently over-constrained, include hospital beds, aero-medical evacuation missions, cargo missions, intermediate staging facilities, etc.

For each goal, a decomposition consists of activities to transport patient from the current location to the destination hospital. Different destination hospitals will give rise to different activity decompositions. Similarly, different routes (often consisting of several legs and several missions) produce different decompositions.

In this system, we completely implemented the multi-decompositional approach. Multiple decompositions were produced for each goal, followed by the constraint-directed scheduling process in which a subset of decompositions was selected. We found that determination of the best decomposition for a given goal cannot be done without close integration with the scheduling process.

4.3 Transportation

Our work on a large-scale transportation problem generalized the experiences with KBLPS and TRAC2ES systems. This problem involves planning and scheduling multiple (on the order of 10,000) movements of cargo (bulk, pallets, containers, large pieces of equipment) and groups of passengers, by air, sea and ground, over the planning horizons on the order of 180 days. Each goal is of the type "Move entity X from Location-A to Location-B within time windows TW1 and TW2". An entity may be as large as to require multiple planes or ships.

A decomposition for a given goal consists of activities to transport the cargo from its current location to the destination, usually combining transportation via several different transport modes. Activities include splitting, combining, repackaging, loading, moving, unloading, storing, etc. Multiple decompositions arise naturally by varying the transport modes and the intermediate ports. Other differences in decompositions for a given goal may arise from selecting different splitting schemes.

For this problem we also used the multi-decompositional approach. Section 5 below summarizes the results of experiments we conducted with this implementation.

4.4 Domain-Independent Generalization of the Technology

The approach we are describing here is a part of the technology suite called COREPLEX (COnstraint-directed REasoning for rePLanning and EXecution). A collection of reusable software libraries and designs, COREPLEX technology is a generalization of the designs and software we developed in the projects such as KBLPS and TRAC^{2}ES (sections 4.1 and 4.2 above).

Within the framework of COREPLEX, we have added four principal innovations to the basis of Constraint-Directed Heuristic Search:

- Efficient mechanisms for storing, updating and retrieving constraint metrics, critical for solving very large-scale problems.
- Model complex constraints, such as non-capacity soft constraints by combining them in a unified fashion with the basic constraint metrics -- contention and reliance.
- Continuous, dynamic replanning with minimal plan perturbation based on our Continuity-Guided Regeneration technique [Kott & Saks 1996], in which the currently executing plan is used as an explicit preferential constraint.
- Ability to schedule in the space of multiple alternative activity decompositions -- the topic of this paper.

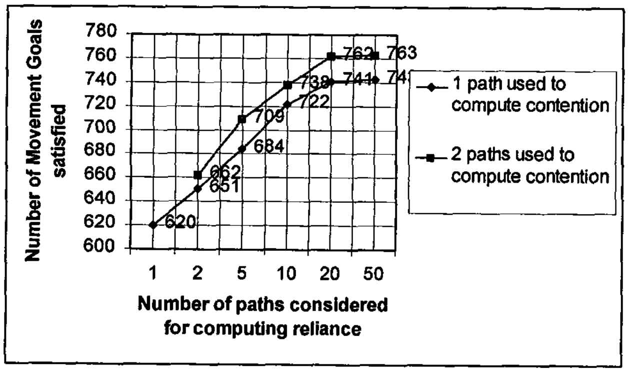

Figure 1. Experimental results – number of goals solved as a function of number of decompositions used.

5. Experimental Results

In this section we describe the series of experiments conducted with one of the instantiations of the multi-decompositional approach for a transportation problem (section 4.3). We used a problem scenario with approximately 30 ports (origin and destination points), about 500 air missions, 60 sea voyages, a total of about 805 movement goals (MG), and scheduling horizon of about 240 days. The algorithm was considering simultaneously the space of up to 40,000 decompositions (up to 50 decompositions per each of 805 Movement Goals). We varied two main search control parameters:

1. The total number of paths (routes) generated for each MG, i.e., the N of the algorithm step 1, in Section 3. All paths generated were used in computing reliance in steps 4 and 5.
2. The number of paths per MG considered for computing contention in step 2 and 3. Computation of contention is relatively expensive, proportionally to the number of paths considered.

The key results of the experiments are summarized in Fig.1. They suggest the following observations:

1. By considering a larger number of alternative decompositions, the algorithm is able to produce a plan of higher quality: greater number of Movement Goals are satisfied with the given resources and under the specified constraints. (We do not discuss the run time numbers because they are not particularly instructive – they reflect mainly the peculiarities of the database used in conjunction with this software implementation.)
2. The data point with a single decomposition (path) approximates the single-decomposition CDHS. Only 620 MGs are solved as compared to 762 MGs in the best of multi-decompositional results.
3. There was no improvement when we increased the number of paths used for computing contention beyond 2. This is a valuable result – the expense of computing contention is proportional to the number of paths considered; apparently there is no need to consider more than 2. (We used 2 "best" paths, i.e., the ones with highest g(D).)
4. The effect of number of paths does not fall off quickly – it continues to produce significant improvement even when the number is increased beyond 10. Note that the real effect of the larger number of paths is the more accurate (or informed) computation of reliance. Much greater influence of multiple decompositions on reliance computation as compared to contention computation is an intriguing and useful result.
5. Perhaps, the difference between the single-decomposition and multi-decomposition cases would disappear if we were to use backtracking in both cases? It is not impossible and deserves further research.
6. Admittedly, our single-decomposition case is an imperfect approximation of CDHS per [Sadeh 91]. Perhaps a more rigorous implementation of CDHS would perform better. This is another topic worthy further investigation.

6. Strengths and Limitations

Let us summarize the key advantages of the proposed approach:

1. It enables integration of planning and scheduling in those domains where selection of a decomposition for a given goal is strongly dependent on resource scheduling. The need for a time-consuming and hard-

to-control backtracking is alleviated. Also alleviated is the need for accurate (and difficult to obtain) heuristics for apriori selection of one best decomposition.
2. It provides for a more accurate estimate of constraint textures, such as contention and reliance, enabling a more accurate look-ahead and ultimately leading to better plans/schedules. Construction of multiple alternative decompositions for each goal gives a broader, more global view of demand and contention for resources. Reliance of activities on resources can be more accurately estimated.
3. There exists a simple and effective control of search depth; by varying the maximum allowable number of decompositions for each goal, the quality of the resulting plan can be traded against the solution time.

Let us also mention the limitations of the approach:
1. In some cases, it does not fully eliminate the need to have some scheduling information during the production of decompositions. E.g., in order to decide if routing of a shipment through a particular port is meaningful, the planner may need to estimate when the shipment will pass through the port. Our approach does not provide any support for such a need. However, in practical problems this limitation does not appear to be critical because (a) estimates of such nature can be made with inexpensive domain-specific rules and (b) even if poor (scheduling-wise) decompositions are produced, the scheduling process eventually weeds them out.
2. Multi-level hierarchical decomposition can lead to a challenging size of the number of alternative decompositions that need to be produced, stored and handled during the scheduling process. In our examples, we were dealing with problems where the planner was able to produce a complete decomposition from each given goal. This may not be the case in other problem domains.

Conclusion

We observe that in a number of practical domains, planning and scheduling problems are tightly intertwined. Often, the right decomposition for a given goal cannot be determined until scheduling of resources, for this and for other goals, is taken into consideration. In other words, there is a need for mechanism that allows to finalize the selection of a suitable decomposition not before but during the scheduling process. The multi-decompositional technique proposed in this paper provides such a mechanism.

We propose a specific algorithm for selecting and scheduling within the space of multiple decompositions – a novel extension of the Constraint-Directed Heuristic Search in which multiple decompositions are used to compute the texture measures.

We present results of experiments in which the proposed algorithm significantly outperformed a single-decomposition version of CDHS, and discuss directions for further research.

We describe several problems in which we applied the proposed approach successfully. The approach is an element of the COREPLEX technology developed at Carnegie Group, Inc. over the last ten (10) years; it generalizes our experiences with several large-scale, real-world systems.

References

Beck, J.C., Davenport, A.J., Sitarski, E., and Fox, M.S. 1997. Beyond Contention: Extending Texture-Based Scheduling Heuristics. In Proceedings AAAI-97.

Fox, M.S., Sadeh, N. and Baykan, C. 1989. Constrained Heuristic Search. In Proceedings of the Eleventh International Joint Conference on Artificial Intelligence.

Kott, A. and Saks, V. 1996. Continuity-Guided Regeneration: An Approach to Reactive Replanning and Rescheduling, In Proceedings of the Florida Artificial Intelligence Research Symposium, FLAIRS-96, Key West, Florida.

Sadeh, N. 1991. Look-ahead Techniques for Micro-opportunistic Job Shop Scheduling. Ph.D. thesis, School of Computer Science, Carnegie Mellon University.

Saks, V., Braidic, G., Kott, A., and Kirschner, C. 1997. Distributed Medical Evaluation Planning: What Problem Should Each Agent Solve? 14th National Conference on Artificial Intelligence, Workshop on Constraints and Agents.

Saks, V., Kepner, A. and Johnson, I. 1992. Knowledge Based Distribution Planning, In Proceedings of the American Defense Preparedness Association AI Conference.

Saks, V., Johnson, I. Fox, M. S. 1993. Distribution Planning: A Constrained Heuristic Search Approach, In Proceedings of the Canadian DND Workshop on Advanced Technologies in Knowledge-Based Systems.