

An Approach for Finding Multiple Area of Specialization Configuration Advisories

Michael Bloem*

NASA Ames Research Center, Moffett Field, CA 94035, USA

Nicholas Bambos†

Stanford University, Stanford, CA 94305, USA

Area of specialization supervisors dynamically configure a set of air traffic control resources so that air traffic in the area can operate safely and efficiently. These resources include airspace sectors, air traffic control positions staffed by controllers, and physical air traffic control equipment. In this paper, we motivate and demonstrate an approach for finding multiple advisories that can assist area supervisors as they accomplish this task. The first motivating factor is that a preference for multiple good and also distinct advisories has been documented in similar contexts, including some air traffic management problems. The second factor that motivates our approach is that the model, problem statement, and algorithm used to generate a single advisory are incomplete and do not perfectly represent reality. The third factor, which we speculate is primarily a result of the second factor, is that area supervisors have indicated a preference for multiple (usually two or three) advisories over a single advisory. Area supervisors have further indicated that each proposed advisory should be different from the other proposed advisories. We investigate the set of advisories that perform best according to a particular objective function for some realistic problem instances. The best few advisories are typically not meaningfully different and therefore should not be presented together to supervisors, and this is the fourth and final factor that motivates our approach. Based on these motivating factors, we define a problem statement which requests multiple good advisories that are all sufficiently different from each other. We briefly describe a heuristic algorithm that was developed for this problem. To more concretely illustrate and motivate the proposed approach, we present the advisories provided by this algorithm for a sample problem instance. We also demonstrate that the proposed heuristic can find feasible second advisories for as many realistic problem instances (15 of 18) as a nearly-exhaustive search. When executed on a desktop workstation computer, the proposed heuristic returns advisories for these realistic problem instances in less than one second per problem instance.

Nomenclature

\mathcal{C}	Set of all valid configuration schedules
$\mathcal{C}_{\text{CSA}}^*(\mathcal{C}, T)$	Set of solutions to a Configuration Schedule Advisory problem instance with constraints \mathcal{C} and traffic situation data T
\mathcal{C}^M	Set of M advisories that make up a solution to the M ε -Optimal d -Distinct Configuration Schedule Advisories problem
\mathcal{C}_k	Set of valid configurations at configuration time step k
C	Configuration schedule advisory
C_k	Configuration advisory at configuration time step k
C_k^A	Airspace configuration

*Research Aerospace Engineer, Systems Modeling and Optimization Branch, MS 210-15; member, AIAA; michael.bloem@nasa.gov.

†Professor of Electrical Engineering and Management Science & Engineering; bambos@stanford.edu.

C_k^{OP}	Operating position configuration
C_k^{W}	Workstation configuration
d	Advisory difference threshold
g	Cost
g_k	Cost at configuration time step k
g_k^{R}	Reconfiguration cost at configuration time step k
g_k^{S}	Static cost at configuration time step k
K	Number of configuration time steps
k	Configuration time step
M	Number of advisories
S	Set of sectors
s	A sector
T	Traffic situation data
t	Traffic time step
T_t^s	Traffic situation data in sector s at traffic time step t
T_k	Traffic situation data at configuration time step k
W	Set of workstations
w	A workstation
β^{R}	Reconfiguration weight
Δ	Length of configuration time steps [minutes]
δ	Length of traffic time steps [minutes]
ε	Fraction of optimality threshold
λ	Sequential Distinct A^* algorithm parameter
Φ	Advisory difference metric
Φ_{max}	Maximum possible advisory difference metric value
ϕ	Configuration difference metric
ϕ_{max}	Maximum possible configuration difference metric value
σ	An open sector; a set of one or more sectors
$\tau(k)$	Set of traffic time steps in configuration time step k

I. Introduction

IN current air traffic management operations, a set of resources that make up an *Area of Specialization* (*AoS* or just *area*) is configured by an area supervisor so that the air traffic in the area's airspace can operate safely and efficiently.¹ An area configuration specifies how airspace, air traffic controller personnel, and physical air traffic control equipment will be used to control air traffic. Some configurations allocate these resources in a way that facilitates safe and efficient operations. For example, a safe and efficient configuration would not require a single controller to control too many aircraft at once, as this might make the control task difficult. Neither would it ask a controller to control just a few aircraft at once, as this might make it difficult for the controller to remain engaged in the control task. Configurations can be changed, but such changes require that additional tasks be performed by controller personnel, which may degrade the safety and efficiency of the traffic operations in the area for a period of time. It is difficult to model the relationship between area configurations and the safety and efficiency of traffic operations in an area. In spite of these difficulties, some algorithms and tools have been proposed to help supervisors select area configurations.²⁻⁸ All of these algorithms and tools only suggest part of the area configuration and leave the remaining components for the supervisor to determine. For example, most algorithms do not suggest which physical workstation should be used to control each region of airspace. Furthermore, none of these algorithms provide multiple configuration advisory options for the supervisor to consider. Providing only a single advisory may be problematic because it is difficult for a model to capture all of the components of area configurations and because any objective function will imperfectly quantify the relationship between a sequence of area configurations, predicted traffic, and safe and efficient operations in the area airspace.

In this paper, we motivate and then demonstrate an approach for generating multiple area configuration advisories. The motivation is based on an investigation of related problems in other domains, the difficulty of modeling area configurations and defining an appropriate objective function, subject-matter expert comments, and an investigation of the set of near-optimal advisories according to one objective function for some

realistic problem instances. By presenting the supervisor with a set of diverse configuration advisories that all perform relatively well according to an objective function, there is an increased likelihood that one of the advisories will also perform well enough with respect to the objective that really matters: the supervisor's judgment of the safety and efficiency of the area. The approach presented in this paper is an extension of the approach for finding a single advisory proposed in Ref. 8.

The next section provides background on area of specialization configurations and the model and objective function that make up an approach for finding a single advisory.⁸ This approach will serve as the starting point for the new approach proposed in this paper. Next, Section III motivates the new approach by discussing related research, relevant modeling issues, subject-matter expert comments, and characteristics of near-optimal advisories for realistic problem instances. The new approach, which involves finding a set of near-optimal and distinct advisories, is specified in Section IV. Section V briefly describes the heuristic algorithm that was used to generate sample results. Sample results that further illustrate the approach are in Section VI and some results that demonstrate the heuristic algorithm's ability to find appropriate solutions for realistic problem instances are in Section VII. The paper finishes with conclusions in Section VIII.

II. Background

A. Area of Specialization Configurations

Airspace is partitioned into predefined volumes called *sectors* to facilitate the division of responsibilities between air traffic controllers. An *airspace configuration* maps a set of sectors to a set of *open sectors* such that each sector is assigned to exactly one open sector. A team of air traffic controllers, staffing one to three *operating positions*, monitors each open sector. At least a radar (also known as R-side) operating position is allocated to each open sector. A radar associate or data (also known as D-side) operating position can also be allocated to an open sector. Although rare, a third operating position can be allocated to an open sector. When more operating positions are allocated to an open sector, the tasks associated with controlling traffic in the open sector are divided among more controllers. An *operating position configuration* specifies how many operating positions are allocated to each open sector in a corresponding airspace configuration. Furthermore, each open sector is monitored from a particular *workstation* consisting of seats for air traffic controllers, a radar scope, plugs for headsets, and other equipment used by controllers to monitor traffic. Which workstation is utilized to monitor an open sector can influence how much work is involved when the open sector is changed by adding or removing sectors from it. A *workstation configuration* specifies which workstation is utilized for monitoring each open sector in a corresponding airspace configuration. Together, corresponding airspace, operating position, and workstation configurations make up how we model an area configuration (see sub-section II.B for details). Although this model is more comprehensive than any we are aware of, we have chosen not to model some relevant aspects of area configurations, such as the assignment of controllers to operating positions (see sub-section III.B).

An example area and area configuration is now provided to clarify these definitions. The shapes of the five sectors in AoS 4 of Cleveland Air Route Traffic Control Center (ZOB) as of 20 October 2011 are shown in Fig. 1 (a). The shapes of the open sectors in a sample airspace configuration are shown in Fig. 1 (b) and the floor layout of corresponding operating position and workstation configurations is shown in Fig. 1 (c). The airspace configuration contains four open sectors. Three of these open sectors each consist of airspace from only a single sector (ZOB45, ZOB46, and ZOB48). These three open sectors are each allocated two operating positions (indicated by the number in parentheses in Figs. 1 (b) and (c)). The fourth open sector consists of the combined airspace of sectors ZOB47 and ZOB49 and it is controlled by a single operating position. In Fig. 1 (c), each workstation is depicted as a rectangle with spots for R- and D-side operating positions and operating positions allocated to open sectors are depicted by stick figures. This figure shows that the two workstations on the left side are used for the four operating positions corresponding to the open sectors consisting of ZOB45 and ZOB46. The workstation at the top of the right side is used by the R- and D-side operating positions controlling the open sector consisting of ZOB48. Finally, the single R-side operating position controlling the open sector consisting of ZOB47 and ZOB49 is using the bottom workstation on the right side. The empty workstation in the middle on the right side is not used in this configuration, but it is used in other configurations. The locations of the R- and D-side operating positions at each workstation are fixed.

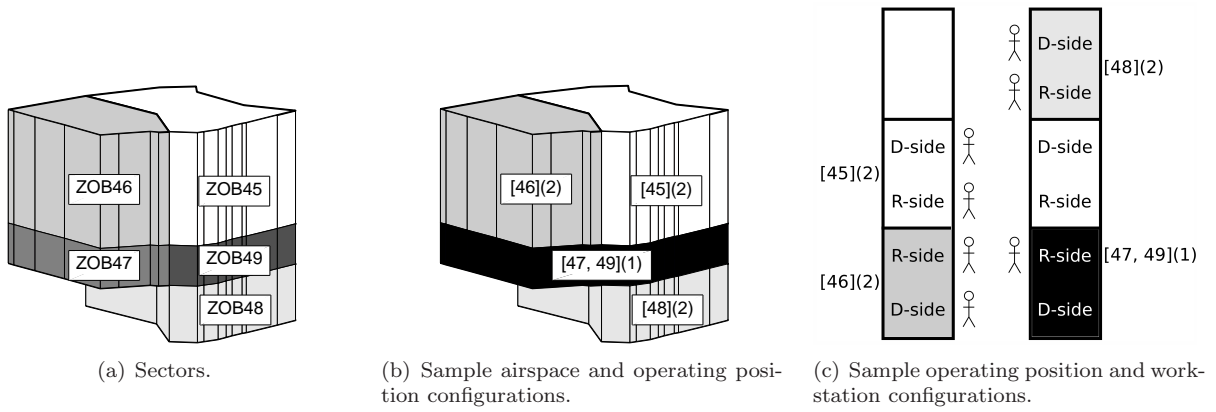


Figure 1. Sectors and sample configuration of ZOB AoS 4.

B. An Approach for Finding a Single Advisory

The problem of finding multiple configuration schedule advisories will be posed as an extension to the problem statement found in Ref. 8 for finding a single advisory. This problem statement will be referred to as the *Configuration Schedule Advisory (CSA)* problem.

1. Decision Variables

The time horizon of the schedule is broken into $K + 1$ discrete configuration time steps $k = 0, 1, 2, \dots, K$, each of length Δ minutes. The configuration time step $k = 0$ is used for data describing the state of the area at the time an advisory is generated.

The decision variables C that make up a configuration schedule advisory are C_k for $k \in \{0, 1, 2, \dots, K\}$, where C_k is the advised configuration at configuration time step k . More concretely, a configuration advisory for configuration time step k is $C_k = \{C_k^A, C_k^{OP}, C_k^W\}$ and it consists of an airspace configuration C_k^A , a corresponding operating position configuration C_k^{OP} , and a corresponding workstation configuration C_k^W . For a given set of sectors $S = \{s_1, s_2, \dots, s_{|S|}\}$ under consideration, an airspace configuration consists of a set of open sectors $C_k^A = \{\sigma_1, \sigma_2, \dots, \sigma_{|C_k^A|}\}$. Each open sector $\sigma_i \in C_k^A$ is itself a set consisting of at least one sector from S . An operating position configuration C_k^{OP} is a function that specifies whether one or two operating positions are allocated to each open sector in the corresponding airspace configuration.^a Finally, a workstation configuration C_k^W is a mapping from open sectors in C_k^A to the set of available workstations W .

2. Data

The traffic situation data T is a set consisting of a configuration time step traffic situation data element T_k for each configuration time step $k \in \{0, 1, \dots, K\}$. Generally, this traffic situation data must contain any predicted air traffic data required to compute the problem objective function. Although many other cost function formulations are possible, the function specified in sub-section II.B.4 for use in this paper requires that T_k contain a unique identifier for each flight in each sector at each *traffic time step* during configuration time step k . Since air traffic characteristics and their impact on controller workload often change faster than airspace configurations, time is discretized into finer traffic time steps of length δ minutes. Let $\tau(k)$ be the set of D traffic time steps in configuration time step k . Then T_k contains the traffic situation data for each sector during configuration time step k : $T_k = \{T_k^{s_1}, T_k^{s_2}, \dots, T_k^{s_{|S|}}\}$. Here each T_k^s is itself a set containing the traffic situation data in sector s at each $t \in \tau(k)$: $T_k^s = \{T_t^s\}_{t \in \tau(k)}$. Finally, each T_t^s contains a unique identifier for each aircraft located within s during traffic time step t . Then $|T_t^s|$ is the *sector count* for s at t : the number of aircraft located within sector s at traffic time step t .

^aWith this constraint, the formulation does not allow three operating positions to be allocated to a single open sector. This constraint can be relaxed to allow three operating positions to be allocated to an open sector. Open sectors with three operating positions are relatively rare in operations, so they are excluded for now.

Another piece of data required for the problem objective used in this paper is open sector capacities, expressed in terms of a maximum number of aircraft at a time that can safely be within each open sector when the open sector is allocated two operating positions. An open sector *Monitor Alert Parameter (MAP)* is used as a capacity bound in current air traffic operations and the MAP values for each open sector are the required sector capacity data.

3. Constraints

The configuration schedule advisory C must be in the set \mathcal{C} of all valid configuration schedules. Although \mathcal{C} could be defined more generally, for this paper it is specified as a set of valid configurations at each configuration time step: $\mathcal{C} = \{C_k\}_{k=0}^K$.

Valid configurations in \mathcal{C}_k must fulfill several fundamental requirements that apply to any problem instance and any configuration time step. Open sectors must be spatially contiguous, for example. Airspace configurations at each configuration time step must assign each sector to an open sector. A sector can be assigned to only one open sector. Only one or two operating positions can be allocated to any open sector. Each open sector must be assigned to a single workstation. A workstation cannot be assigned to multiple open sectors.

Valid configurations can also be specific to certain problem instances and may apply for all or only a subset of configuration time steps. Configurations containing certain open sectors might be denoted as invalid because they are geographically too large to be displayed clearly to controllers. Other configurations might be invalid for some period of time due to temporary workstation equipment outages. More permanent technological limitations, such as radio frequency coverage issues, may also limit the set of valid configurations. Training sessions may require that certain open sectors be a part of any configuration utilized for certain configuration time steps. The number of available controller personnel can impose an upper bound on the number of operating positions that can be used in a configuration. This list is not exhaustive; any configuration can be removed from consideration during any configuration time step and for any reason.

4. Objective

The problem objective is to minimize a cost $g(C, T)$. The cost for a configuration schedule advisory is a sum of the costs incurred by the configuration at each configuration time step in the time horizon:

$$g(C, T) = \sum_{k=1}^K g_k(C_{k-1}, T_{k-1}, C_k, T_k). \quad (1)$$

For a single configuration time step, the cost is a weighted sum of a single configuration time step *static cost* and a single configuration time step *reconfiguration cost*:

$$g_k(C_{k-1}, T_{k-1}, C_k, T_k) = g_k^S(C_k, T_k) + \beta^R g_k^R(C_{k-1}, T_{k-1}, C_k, T_k), \quad (2)$$

where β^R is the *reconfiguration weight*. The static cost penalizes configurations that do not facilitate safe and efficient operations during the configuration time step, such as those that require controllers to control excessively high or low amounts of traffic. The reconfiguration cost penalizes changes in configurations that occur between configuration time steps, particularly those that require controllers to transfer control of many aircraft. These cost functions are complex and involve many parameters; a detailed specification of these costs and suggested parameter values can be found in Ref. 8. The reconfiguration weight determines the relative importance of these competing types of cost.

5. Single-Advisory Problem Statement Summary

The Configuration Schedule Advisory (CSA) problem is

$$\underset{C=\{C_0, C_1, \dots, C_K\}}{\text{minimize}} \quad \sum_{k=1}^K g_k(C_{k-1}, T_{k-1}, C_k, T_k) \quad (3)$$

$$\text{subject to } C_k \in \mathcal{C}_k, \quad k = 0, 1, 2, \dots, K. \quad (4)$$

The set $\mathcal{C}_{\text{CSA}}^*(\mathcal{C}, T) \subseteq \mathcal{C}$ is the set of solutions to a CSA problem instance with constraints \mathcal{C} and traffic situation data T .

Ref. 8 describes how the CSA problem can be mapped to a shortest-path problem on a time-expanded graph. The origin for the path is dictated by the initial configuration C_0 . Any configuration in \mathcal{C}_K is a valid destination for the path. Each configuration at a time step is a node in the graph and transitions between configurations in one time step to configurations in the next time step are edges. Node costs correspond to the static cost and edge costs correspond to the reconfiguration cost.

III. Motivation of Multiple-Advisory Problem Statement

Before introducing the multiple-advisory problem statement we have defined in Section IV, the various factors that motivated this problem statement will be explained. The first motivating factor is that a preference for multiple good (*i.e.* near-minimum-cost) and also distinct advisories has been documented in similar contexts. The second factor is that the model, problem statement, and algorithm used to generate a single advisory are incomplete and imperfect. The third factor, which we speculate is primarily a result of the second factor, is that area supervisors have indicated a preference for multiple (usually two or three) advisories over a single advisory. Area supervisors have further indicated that each proposed advisory should be different from the other proposed advisories. We investigate the set of advisories that perform best according to a particular objective function for some realistic problem instances. The best few advisories are typically not meaningfully different and therefore should not be presented together to supervisors, and this is the fourth and final factor that motivates our approach.

A. Related Research

When an objective function is difficult to specify or the system model is incomplete and imperfect, researchers have proposed presenting decision-makers with multiple distinct advisories. We will show in the next sub-section that both of these challenges are encountered when finding area configuration advisories. In this sub-section, we review how researchers working on problems in artificial intelligence, operations research, and air traffic management have addressed these challenges by finding multiple distinct advisories. Although this related research provides motivation for providing users with a diverse set of good advisories, none of the approaches discussed here can be applied directly to the problem under consideration in this paper.

In the context of “planning” in the artificial intelligence domain, Nguyen et al. investigate how to generate a diverse set of plans when user preferences are unknown or when only a distribution over user objective function parameter values is provided.⁹ Neither of these cases exactly describe the situation we investigate in this paper, but their work still helps motivate our approach. In general, Nguyen et al. propose presenting the user with options and then allowing the user to resolve uncertainty in the objective function by selecting a solution from the set of options. Similarly, we hope that the user will resolve the impact of unmodeled or imperfectly-modeled components of the area configuration problem by selecting an advisory option that performs well enough when these components are considered.

More specifically, when user preferences are completely unknown, Nguyen et al. suggest presenting the user with a *diverse* set of plans. If diversity is defined appropriately, diverse plans are less likely to be equally preferred by users, so a diverse set of plans increases the chances that one of the plans in the set will be acceptable to the user. In the area configuration context, an objective function has been defined⁸ but is also known to be incomplete and imperfect. Therefore, rather than striving only for diversity, we propose finding set of two to three advisories that is both diverse and in which each advisory performs relatively well with respect to the objective function. The hope is that the diverse advisories will achieve a variety of levels of performance with respect to unmodeled or imperfectly-modeled components of the problem so that at least one advisory will perform well enough when every component of the problem (well-modeled or not) is taken into account.

The planning problems studied by Nguyen et al. are much more general than the shortest-path problems considered in this paper. The algorithms they investigate fail to take advantage of the special structure of shortest-path problems.

However, the search for multiple short and diverse paths has been investigated in the operations research literature. Motivated by the “compromises” or “approximations” inherent in any mathematical model, Bellman studied the problem of finding the k shortest (or lowest cost) paths.¹⁰ This k -shortest paths

problem has received considerable attention in the literature,^{10–14} but simply searching for the k shortest paths without considering the diversity of the paths might lead to k similar paths. For the reasons discussed earlier, we seek a set of low-cost advisories (this corresponds to a set of short paths) that are also diverse in a particular way. Furthermore, the investigation in sub-section III.D suggests that the k lowest-cost advisories will often not be sufficiently diverse. Therefore, we are not facing a k -shortest paths problem.

For certain special definitions of diversity, the problem of finding a low-cost set of diverse paths has been studied.^{15–21} For example, Ref. 15 involves finding low-cost paths that share no nodes. The motivation provided in this body of research is often related to robustness—a path from the set should remain available even if some set of nodes or links fails. Unfortunately, none of the definitions of diversity used in this body of research correspond to the type of diversity that matters for area configurations advisories (see sub-section III.C). This is not surprising because we are not concerned with finding a robust set of area configuration advisories. Therefore, this research also does not apply directly.

The need for multiple and diverse advisories has arisen in other areas of air traffic management, typically as a way to resolve the tension between competing objectives. In Ref. 22, Montoya et al. propose a generalized dynamic programming algorithm that finds the set of Pareto-optimal solutions for a runway scheduling problem. Each solution in the Pareto set achieves a certain trade-off between throughput and delay; no other solution can do better with respect to *both* throughput and delay than a solution that is in the Pareto set. Since both high throughput and low delay are desirable, this work finds a set of solutions that achieve a variety of trade-offs between these objectives. Similarly, Marceau et al. develop an approach that finds the set of Pareto-optimal solutions for a strategic traffic flow management problem.²³ In this case, the relevant trade-off is between an expected congestion cost and an expected delay cost. In each of these situations, a decision-maker is expected to select a solution from among the Pareto-optimal set, thereby determining an appropriate trade-off between the competing objectives. A tool that suggests Ground Delay Program parameters for San Francisco International Airport provides a primary and two alternative solutions by increasing and decreasing an important program parameter: the program end time.²⁴ The alternative that is longer than the primary proposed program would have more delay but less airborne holding risk and the alternative that is shorter would have less delay but greater airborne holding risk. This approach is reasonable because of the simple relationship between this single program parameter and the two competing objectives of low delay and low airborne holding risk. When configuring an area, static cost and reconfiguration cost are competing objectives. Presenting the user with multiple solutions that achieve different trade-offs between these objectives would allow the user to account for their relative importance in each situation, but factors other than the tension between such objectives also motivate the need for multiple solutions (such as the modeling issues described in sub-section III.B). Therefore, generating multiple solutions primarily to allow the user to resolve this tension may not be an appropriate choice in this context.

B. Model, Problem Statement, and Algorithm Issues

Any model is only an approximation of reality and no mathematical optimization problem statement perfectly captures the problem faced in reality. Compromises must always be made to arrive at a tractable problem for which a solution can be found in a reasonable amount of time. There are three major known issues related to our configuration model, the CSA problem, and the algorithm used to solve this problem. The configuration model is incomplete, the CSA problem objective function is imperfect, and algorithmic considerations imply that enforcing a certain desirable type of constraint on advisories would reduce the likelihood of finding a solution in a reasonable amount of time. Each of these issues will be discussed in detail.

1. Incomplete Model

First, as was alluded to in sub-section II.A, our model of area configurations is incomplete. The main missing component is a mapping of available controllers to operating positions. This component of configurations is excluded from the problem statement because the influence of this component on the safety and efficiency of operations, which depends on controller skill, fatigue, preferences, and personality, is particularly difficult to quantify. Furthermore, controller breaks and controller tasks that do not involve controlling traffic (such as certain training activities) play an important role in this component of configurations. Such constraints would further complicate the model and require additional inputs. Finally, considering this component of configurations would increase the number of possible configurations considerably, which would potentially

make finding a good advisory in a reasonable amount of time more difficult. For these reasons, this component of the configuration is left for the supervisor to determine without the assistance of an advisory.

2. *Imperfect Objective Function*

Next, the CSA objective function (3) is lacking in a number of ways. First of all, it utilizes an incomplete configuration model and therefore cannot account for relevant components of area operations that are not captured in the model. For example, the model does not capture which human controller is assigned to which operating position. Secondly, the difficulty of quantifying controller workload²⁵ also makes it difficult to determine when traffic operations in an area would be safe and efficient. The static cost in the CSA problem objective function uses aircraft count divided by open sector MAP as a measure of the workload associated with controlling traffic. This measure of workload does not explicitly take into account many important factors, such as climbing and descending aircraft.²⁶ The reconfiguration cost penalizes a different sort of controller workload: the effort associated with transitioning from one configuration to another. Relatively little is known about this type of workload.^{27,28} Furthermore, the number of operating positions allocated to each open sector impacts workload, but it is not obvious how to quantify this impact.^{29,30} Thirdly, even if controller workload were known exactly, it is not clear what levels of controller workload facilitate safe and efficient traffic operations. Although the work associated with changing configurations seems to always hinder safe and efficient operations, too much *or* too little workload might lead to degraded safety and efficiency when controlling traffic in between configuration changes. The relative impact of the two types of controller workload on safe and efficient traffic operations, quantified in the CSA objective function by the β^R parameter, is not known. Finally, this β^R parameter and many others in the CSA objective function have only been crudely tuned.⁸

3. *Inability to Handle an Important Constraint*

Lastly, although the CSA problem statement is capable of handling many constraints that arise when configuring an area, the algorithm currently in use to solve the problem does not easily handle one commonly-requested type of constraint: once an open sector has been generated by a configuration change, users often request that this open sector not be changed again for at least some period of time (approximately 15–30 minutes). When such a constraint spans multiple configuration time steps, either the configuration model must be changed to keep track of this open sector duration or a different solution algorithm must be utilized. Changing the model would lead to a much larger set of possible solutions and longer algorithm run times, and we have not found a suitable algorithm to handle such constraints when they are enforced but the model is not changed. While appropriate tuning of the β^R parameter can ensure that this constraint is usually met, there may still be some proposed advisories that violate it.

C. **Acceptability Among Users**

We consulted area supervisors, the target users of configuration schedule advisories, during the development of the model and problem statement. In May of 2012, nine supervisors or former supervisors provided input and feedback in a two-day workshop at NASA Ames Research Center. We have also made five visits to three FAA Air Route Traffic Control Centers, including three visits to Cleveland Center.

Area supervisors requested that the lowest-cost advisory be presented along with the advisories achieving the second- and perhaps third-lowest costs. The reason for this request may be that they are aware of the issues with the model and problem statement listed in sub-section III.B. However, the supervisors did trust the model and problem statement enough to request that the advisories be ranked according to the CSA objective function and to request that sufficiently sub-optimal advisories not be displayed. Considerably sub-optimal advisories would rarely be selected over the optimal advisory.

Area supervisors have also requested that the multiple advisories be different. They indicated that two advisories are different if they use different airspace configurations for at least 30 minutes of a two-hour time period. The reason for this request may be that, given two similar advisories, it is unlikely that users would strongly prefer one over the other.

D. Investigation of Sets of Near-Optimal Advisories

We investigated the sets of near-optimal advisories for some realistic problem instances to gain insight into the relationship between the cost of an advisory and its difference from the minimum-cost advisory. If the lowest-cost advisories are typically sufficiently different from the minimum cost advisory, then a constraint on the difference between advisories may not need to be enforced and some existing algorithms for finding a certain number of lowest-cost paths could be used directly.¹⁰⁻¹⁴ If this is not the case, then the difference constraint will need to be explicitly enforced and a new solution technique may be needed.

1. Finding Sets of Near-Optimal Advisories

The objective in this sub-section is to find all the configuration advisories that achieve a cost within some fraction of the minimum cost such that each returned advisory uses a different airspace configuration than the other returned advisories during at least one time step. Advisories that only use different operating position or workstation configurations are not meaningfully different and are therefore not considered. To attempt to find these advisories, we used an adaptation of the dynamic programming-based algorithm proposed by Byers and Waterman in Refs. 31 and 32. From each configuration, the algorithm always explores at least one next minimum-cost-to-go configuration (if it meets the cost bound), as well as any other next configuration that meets the cost bound and uses a different airspace configuration than all other explored next configurations. It will return a minimum-cost advisory. The algorithm ensures that each returned advisory will use a different airspace configuration than all other returned advisories in at least one configuration time step, and that each returned advisory achieves the cost bound. It does not ensure that all advisories meeting these criteria will be returned. It also does not guarantee that it will return the lowest-cost option from among all the advisories that use the same airspace configurations in the same time steps (with the exception of the minimum-cost airspace configurations). Although this algorithm allows us to explore this set of advisories, the set can be large and the algorithm can take tens of minutes to execute, making it unacceptable for use in a decision-support tool.

2. Problem Instances

All of the problem instances involved the five sectors in ZOB AoS 4 (see Fig. 1(a)). The algorithm was restricted to select from the eight airspace configurations that were used operationally during the last couple of months of 2011. There were 86 configurations of airspace and operating positions that could be generated from these airspace configurations.

The configuration time step was set to five minutes ($\Delta = 5$) and the traffic time step was set to one minute ($\delta = 1$). The problem instances were two hours long, so they include 24 configuration time steps ($K = 24$). The cost function parameters were set to values very similar to the default values in Ref. 8. The value of β^R was set to 1.

The algorithm searched for a set of near-optimal advisories for 18 CSA problem instances. These problem instances span two relatively high-traffic-volume dates (Thursday 01 December 2011 and Tuesday 06 December 2011) and nine two-hour time periods (6 am–8 am, . . . , 10 pm–midnight local time). The advisories were required to achieve a cost within 25% of the minimum cost (*i.e.* advisory cost \leq minimum cost \times 1.25). Other than the constraint that the initial configuration be the configuration that was in use historically at that time, no other constraints were imposed.

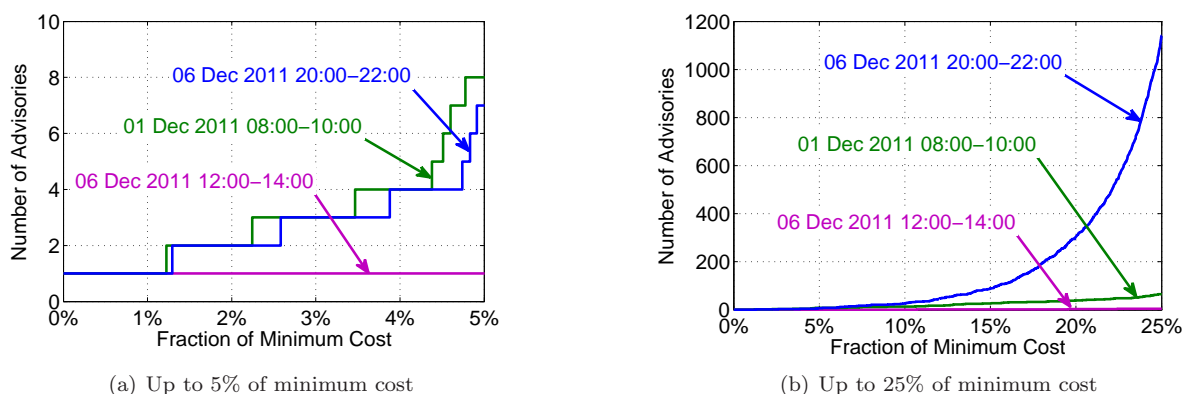
3. Results

The number of advisories returned by the algorithm for each problem instance is shown in Table 1. There is considerable variation from problem instance to problem instance in the number of returned near-optimal advisories. For three problem instances, only a single near-optimal advisory was found (the minimum-cost advisory) but from 20:00–22:00 on 06 December 2011 there were 1140 such advisories returned by the algorithm. There does not appear to be an obvious relationship between the time of day and the number of returned near-optimal advisories.

Figure 2 shows the cumulative distribution of the costs of the returned advisories, expressed as a fraction of the minimum cost, for three representative problem instances. The number of advisories appears to increase exponentially as the fraction of minimum cost increases, but the exponent that best captures the relationship varies considerably from instance to instance.

Table 1. Number of Near-Optimal Distinct Airspace Configuration Advisories

Local Time	Thursday 01 December 2011	Tuesday 06 December 2011
06:00–08:00	179	19
08:00–10:00	65	21
10:00–12:00	47	273
12:00–14:00	4	4
14:00–16:00	21	65
16:00–18:00	1	14
18:00–20:00	34	22
20:00–22:00	76	1140
22:00–24:00	1	1

**Figure 2. Cumulative cost distribution of returned near-minimum-cost configuration advisories for three problem instances.**

We are not only interested in the size of the set of near-optimal advisories but also in the degree of difference between advisories in this set. To get a feel for the diversity of the set for each problem instance, we investigated the distribution of the difference between each advisory in the set and the minimum-cost advisory. Difference is quantified as the number of minutes in which the advisories use different airspace configurations. Figure 3 shows this distribution for each of the instances. Instances are numbered chronologically. The area of each point is proportional to the number of advisories achieving a particular level of difference from the minimum-cost advisory.

Although the near-minimum cost advisories are more likely to achieve lower levels of difference from the minimum-cost advisory than higher levels of difference, there is no obvious pattern in the distributions across instances. The distributions tend to be relatively uniform over their support, which can exceed 100 minutes but may also be less than 20 minutes. The algorithm was able to find at least one near-optimal advisory that used different airspace configurations than the minimum-cost advisory for 30 minutes or more in 15 of the 18 problem instances.

Finally, we investigated the relationship between the cost of second-lowest-cost advisories and how different they are from the minimum-cost advisory. In Fig. 4, each point shows the fraction of optimality and difference achieved by the second-lowest-cost returned advisory. Different colors and/or symbols are used for different problem instances. In only five of the 18 instances is the second-lowest-cost advisory different enough from the minimum-cost advisory to meet the 30-minute threshold suggested by area supervisors. The second-lowest-cost advisory would be even less likely to be sufficiently different from the minimum-cost advisory if we did not search only through advisories that use different airspace configurations in at least one time step.

Overall, this investigation indicates that it will not be sufficient to simply return the lowest-cost advisories.

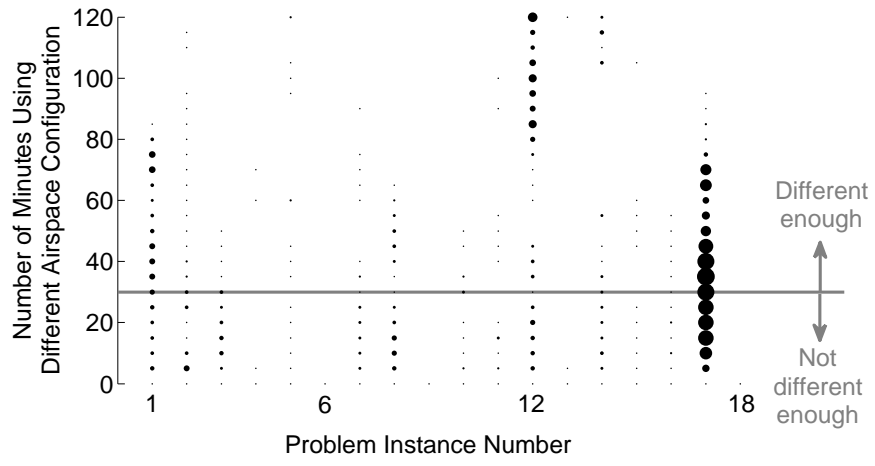


Figure 3. Distribution of diversity in the returned sets of advisories within 25% of the minimum cost.

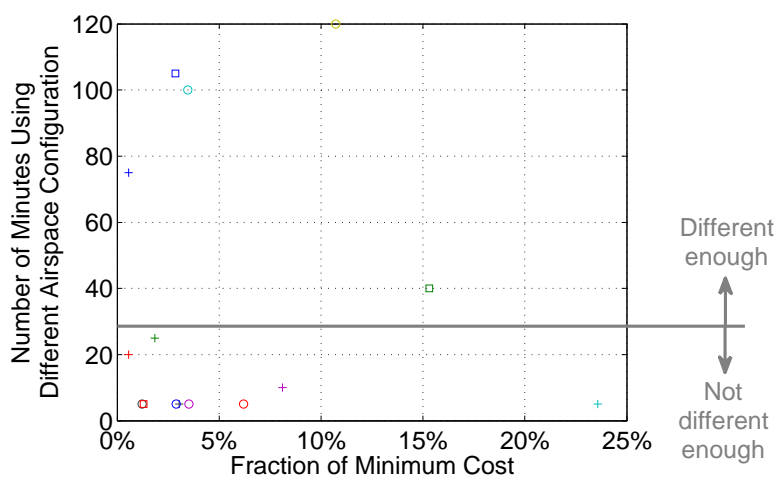


Figure 4. Fraction of minimum cost versus difference from minimum-cost advisory achieved by the second-lowest-cost returned advisory.

These advisories will frequently be too similar to be worth the supervisor's attention individually. However, this analysis also indicates that it is not uncommon to find an advisory with a cost that is within 25% of the minimum cost and that also uses different airspace configurations than the minimum-cost advisory for 30 minutes. This is encouraging, assuming such advisories could be found in a computationally-efficient manner.

IV. Multiple Advisories Problem Statement

Motivated by the factors discussed in Section III, the CSA problem is extended to require that the solution consist of a set of M advisories. Each advisory must be valid and the first returned advisory must be a minimum-cost advisory. Each other returned advisory must achieve a cost value that is within some fraction of the minimum cost value. Finally, each pair of returned advisories must be sufficiently different according to a difference metric.

The problem statement is

$$\underset{\mathcal{C}^M = \{C^1, \dots, C^M\}}{\text{minimize}} \sum_{m=1}^M \sum_{k=1}^K g_k(C_{k-1}^m, T_{k-1}, C_k^m, T_k) \quad (5)$$

$$\text{subject to } |\mathcal{C}^M| = M \quad (6)$$

$$C_k^m \in \mathcal{C}_k, \quad k = 0, 1, 2, \dots, K, \quad m = 1, 2, \dots, M \quad (7)$$

$$C^1 \in \mathcal{C}_{\text{CSA}}^*(\mathcal{C}, T) \quad (8)$$

$$\frac{g(C^m, T)}{g(C^1, T)} - 1 \leq \varepsilon \quad m = 2, 3, \dots, M \quad (9)$$

$$\Phi(C^m, C^{m'}) \geq d \quad \forall m, m' \in \{1, 2, \dots, M\}, m \neq m'. \quad (10)$$

Here $\mathcal{C}^M = \{C^1, \dots, C^M\}$ is the set of M advisories that make up a solution to the problem. Constraint (6) enforces that there be M advisories. The objective (5) is to minimize the sum of the costs of the M advisories. Constraint (7) ensures that each advisory is valid. Constraint (8) requires that the first advisory be an optimal advisory for a CSA problem instance with configuration constraints \mathcal{C} and traffic situation data T . Constraint (9) requires that all of the additional advisories achieve a cost value within a fraction ε of the minimum cost. Finally, constraint (10) requires that each pair of advisories be sufficiently different according to an advisory difference metric $\Phi : \mathcal{C} \times \mathcal{C} \rightarrow \mathbf{R}_+$. This is referred to as the M ε -Optimal d -Distinct Configuration Schedule Advisories (M - ε - d -CSAs) problem.

4. Advisory Difference Metric

The advisory difference metric Φ maps a pair of configurations from \mathcal{C} to a non-negative real number. It could take many forms, but in this paper it will be defined as

$$\Phi(C^m, C^{m'}) = \sum_{k=1}^K \phi(C_k^m, C_k^{m'}), \quad (11)$$

where $\phi : \mathcal{C}_k \times \mathcal{C}_k \rightarrow \mathbf{R}_+$ is a configuration difference metric. It defines a difference between valid configurations.

For this paper, the configuration difference metric is

$$\phi(C_k^m, C_k^{m'}) = \begin{cases} 1 & C_k^{m,A} \neq C_k^{m',A} \\ 0 & \text{else.} \end{cases} \quad (12)$$

Pairs of configurations that use different airspace configurations (sets of open sectors) in the same time step achieve a configuration difference of 1, while all other pairs of configurations achieve a configuration difference of 0. Therefore, when using this advisory difference metric, constraint (10) requires that the two advisories in each pair of returned advisories utilize different airspace configurations for d time steps. Sub-section III.C describes feedback from potential advisory users that motivates this metric. Differences in operating position or workstation configurations corresponding to the same airspace configuration were not regarded as significant.

V. Sequential Distinct A^* Algorithm

We developed a heuristic algorithm to solve the M - ε - d -CSAs problem. It runs the well-known A^* algorithm for computing shortest paths^{33,34} M times, so it is referred to as the Sequential Distinct A^* (SDA^{*}) heuristic. The computational complexity of the SDA^{*} heuristic is on the order of M times the computational complexity of the A^* algorithm.

A high-level description of the SDA^{*} heuristic is provided in Algorithm 1. A different single-time step configuration cost function is used within the A^* algorithm each time it is called. When finding the first advisory, A^* uses the CSA problem single-time step configuration cost g_k specified in eq. (2). When searching for the second through M^{th} advisories, SDA^{*} adjusts the cost used within A^* so that the cost rewards configurations with low g_k values as well as those that are different from the corresponding configurations in the $m-1$ previously-discovered advisories (see eq. (13)). An algorithm weighting parameter $\lambda \geq 0$ determines the relative importance of the two terms in this adjusted cost. The adjusted cost also makes use of Φ_{\max} and ϕ_{\max} , which are the maximum possible advisory and configuration difference metric values, respectively (see eq. (11) and (12)). Terms in the adjusted cost are normalized by J^* , $(m-1)$, and $(\Phi_{\max} - d + 1)$ to increase the likelihood that the first term and the term multiplied by λ are roughly the same magnitude across problem instances and while searching for subsequent advisories. Hopefully, this in turn increases the likelihood that a particular λ value can consistently perform well.

Algorithm 1 Sequential Distinct A^* (SDA^{*})

Require: C, T, M, ε, d $\{M$ - ε - d -CSAs problem instance specification $\}$

Require: $\lambda \geq 0$ $\{\text{Algorithm parameter}\}$

Compute C^1 with A^* using $g_k(C_{k-1}, T_{k-1}, C_k, T_k)$ as the single-time step configuration cost function

$J^* \leftarrow g(C^1, T)$

for $m = 2, \dots, M$ **do**

 Compute C^m with A^* using

$$\underbrace{\frac{g_k(C_{k-1}, T_{k-1}, C_k, T_k)}{J^*}}_{\text{term encouraging low-cost advisories}} + \lambda \underbrace{\frac{1}{m-1} \sum_{m'=1}^{m-1} \frac{\phi_{\max} - \phi(C_k, C_k^{m'})}{\Phi_{\max} - d + 1}}_{\text{term encouraging different advisories}} \quad (13)$$

as the single-time step configuration cost function

if C^m does not satisfy M - ε - d -CSAs problem instance constraints **then**

 Skip remainder of **for** loop $\{\text{Do not attempt to construct the remaining } M - m \text{ advisories}\}$

return Advisories that satisfy M - ε - d -CSAs problem instance constraints (C^1 through up to C^M)

VI. Sample Illustrative Problem Instance

An example problem instance was designed to illustrate and motivate the problem statement. This problem instance was designed such that appropriate advisories would be obvious, enabling a straightforward discussion of the advisories requested by the problem statement. Except for the traffic, this problem instance is identical to the instances in Ref. 8, which were designed with the help of subject-matter experts to illustrate the nature of the CSA problem and its solutions.

A. Setup

The example problem instance is based on ZOB AoS 4. The shapes of the five sectors in AoS 4 of ZOB as of 20 October 2011 are shown in Fig. 1 (a) and a sample configuration of the area is depicted in Figs. 1 (b) and (c).

The two-hour time horizon selected for this instance ran from 13:00 to 15:00 UTC on 1 December 2011, which is 08:00 to 10:00 local time at ZOB. Synthetic constraints specifying the sets of valid configurations C_k for each time step were constructed to demonstrate characteristics of the problem statement. The scheduled range of number of operating positions is shown in Fig. 5. The configuration schedule advisories are required to use seven operating positions for the first 15 minutes of the time horizon, seven or eight operating positions

from 13:15 until 14:00 UTC, and eight operating positions from 14:00 to 15:00 UTC. Constraints required each open sector to be mapped to a particular workstation.

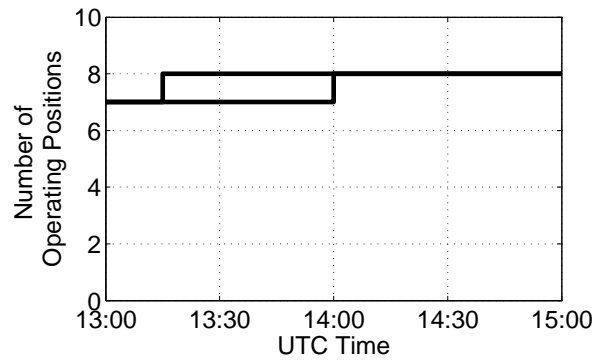


Figure 5. Scheduled range of number of operating positions.

The configuration schedule advisories must satisfy a few other constraints. For the first 15 minutes, the sectors ZOB45, ZOB46, and ZOB48 were required to be open sectors on their own and controlled with two operating positions. Furthermore, during this first 15 minutes, sectors ZOB47 and ZOB49 were required to be combined into a single open sector that was controlled by a single R-side operating position working at the workstation used for ZOB49 when ZOB49 operates as an open sector on its own. This configuration is depicted in Fig. 1 (b) and (c). Other constraints required that sectors ZOB45, ZOB46, and ZOB48 were open sectors on their own and controlled with two operating positions for the entire two-hour period. Taken together, these constraints left only two possible configurations that made use of eight operating positions: one in which ZOB47 and ZOB49 were combined into an open sector controlled with two operating positions and one in which ZOB47 and ZOB49 were each an open sector and each controlled with a single operating position. The MAP value of ZOB47 is 15, the MAP value of ZOB49 is 19, and the MAP value of an open sector consisting of ZOB47 and ZOB49 is 19.

A synthetic traffic scenario was also constructed. The aircraft counts for ZOB45, ZOB46, and ZOB48 are not important because constraints prevent changes in the configuration of these sectors. Figure 6 shows the aircraft counts in ZOB47 and ZOB49 divided by their MAP values, as well as the total aircraft counts in the two sectors divided by the MAP value of an open sector consisting of both of these sectors. Aircraft counts divided by open sector MAP values are referred to as *open sector loads*.

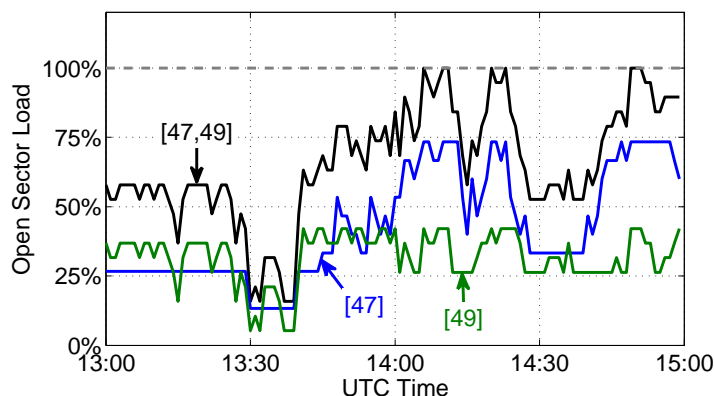


Figure 6. Open sector loads for relevant open sectors involving ZOB47 and ZOB49.

The configuration time step size for these example problem instances was set to $\Delta = 5$ minutes, so there were 24 configuration time steps in the two-hour time horizon. The traffic time step was set to $\delta = 1$ minute. The reconfiguration weight and the other parameters were set to the default values specified in Ref. 8, so $\beta^R = 1.75$. Parameters unique to the M - ε - d -CSAs problem statement request two advisories ($M = 2$) that

use different airspace reconfigurations for at least 30 minutes ($d = 6$ because each time step is five minutes long). The first advisory is required to be a minimum-cost advisory, and the second is required to achieve a cost that is within 25% of the minimum cost ($\varepsilon = 0.25$). The SDA* λ parameter, which determines the relative importance in the adjusted cost (13) of finding low-cost advisories and finding different advisories, was set to 0.11875.

B. Results

The minimum-cost configuration schedule advisory returned by the SDA* heuristic is shown in Fig. 7 and the relevant open sector loads are shown in Fig. 8. The number of operating positions allocated to each open sector is shown in parentheses. The advisory uses the required starting configuration between 13:00 and 13:35 UTC. At 13:35 UTC, the advisory uses the eighth operating position that became available at 13:15 UTC to assign a D-side operating position to the open sector consisting of ZOB47 and ZOB49. This advisory is appropriate because the open sector load is at an appropriate level for two operating positions during this period (see Fig. 8) and because the advisory selects a low-reconfiguration-effort time for the reconfiguration. The static cost of this minimum-cost advisory is 19.1 and the (β^R -weighted) reconfiguration cost is 7.1, so the minimum cost is 26.2.

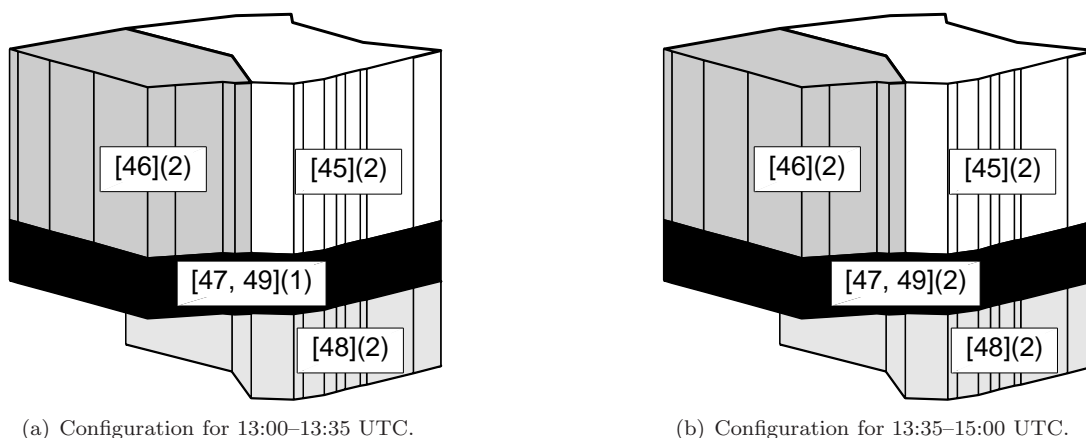


Figure 7. Minimum-cost configuration schedule advisory.

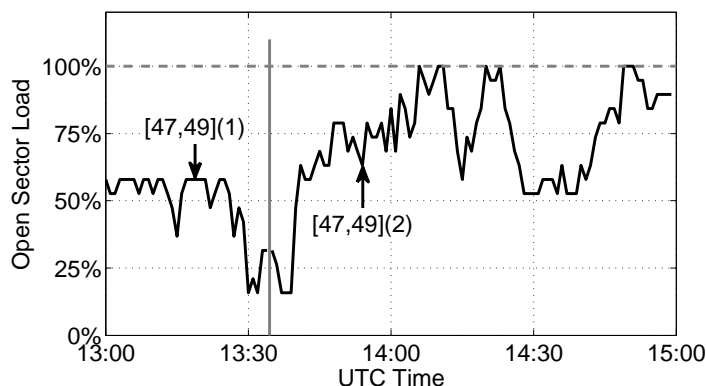


Figure 8. Open sector loads for the minimum-cost advisory.

If the minimum-cost advisory were implemented, the open sector load would reach the MAP value during a few minutes after 14:00 UTC in the open sector consisting of ZOB47 and ZOB49. Even with two operating positions, such open sector loads are slightly higher than is typically preferred for safe and efficient traffic

operations (see the static workload cost in Ref. 8 for details). Therefore, it may also be appropriate to split this open sector and operate ZOB47 and ZOB49 as separate open sectors. This is exactly what the second configuration schedule advisory suggests. The second advisory returned by SDA* is shown in Fig. 9 and the relevant open sector loads are shown in Fig. 10. The advisory uses the required starting configuration between 13:00 and 13:35 UTC. From 13:35–15:00 UTC, the advisory uses a configuration with eight operating positions in which ZOB47 and ZOB49 each operate as open sectors and each is allocated a single operating position. This advisory is appropriate because the two new open sectors experience loads that are acceptable when they are monitored by a single R-side operating position (see Fig. 10). The load in the open sector consisting of ZOB49 is slightly lower than is ideal, and the load in the open sector consisting of ZOB47 is sometimes slightly higher than is ideal, but the loads are often in a safe and efficient range during this interval. Furthermore, the advisory selects a relatively low-reconfiguration-effort time to perform the required airspace reconfiguration. The static cost of this advisory is 18.4 and the (β^R -weighted) reconfiguration cost is 14.0, so the total cost is 32.4. This is 23.7% larger than the minimum cost, so the cost constraint (9) is satisfied for $\varepsilon = 0.25$. Also, this advisory uses different airspace configurations than the minimum-cost advisory in 17 of the 24 time steps, which satisfies the difference constraint (10). An area supervisor could further evaluate these advisories based on factors not considered by the problem model and objective to judge which should be selected for implementation.

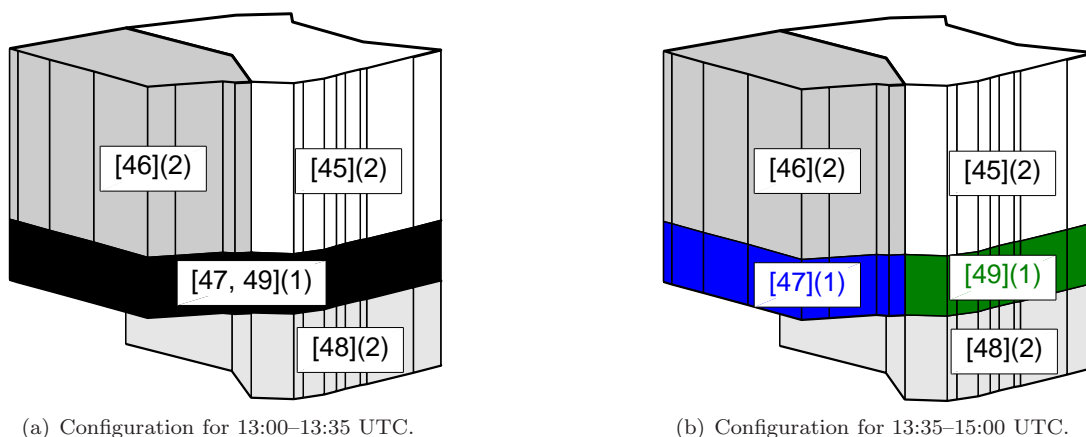


Figure 9. Second returned advisory.

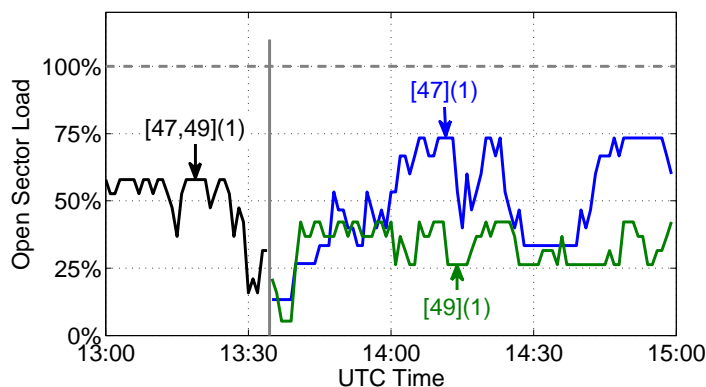


Figure 10. Open sector loads for the second returned advisory.

VII. Problem Instances Based on Historical Data

The heuristic algorithm was also run on problem instances based on the historical data utilized in sub-section III.D. These instances use the same data and cost function parameters as were used in sub-section III.D. The heuristic's solutions to M - ϵ - d -CSAs problem instances based on these historical situations are presented to further motivate the problem statement, as well as to provide further evidence that the heuristic algorithm can provide adequate sets of advisories. For these problem instances, we used the same M - ϵ - d -CSAs problem parameters as used in Section VI ($M = 2$, $\epsilon = 0.25$, $d = 6$). Again, the SDA* λ parameter was set to 0.11875.

Figure 11 plots the fraction of minimum cost and difference from the minimum-cost advisory for the second advisory returned by the SDA* heuristic for each of these problem instances. Figure 4 is the corresponding plot of the results when a more exhaustive search was conducted for near-optimal advisories with a less restrictive constraint on the difference between returned advisories. The results of that more exhaustive search indicated that a feasible second advisory for these M - ϵ - d -CSAs problem instances existed for at least 15 of the 18 instances (see Fig. 3). The heuristic algorithm found a feasible second advisory for all 15 of these instances. As required by the problem instance specification, none of the second advisories use different airspace configurations than the minimum-cost advisory for less than 30 minutes. Eleven of the advisories use different airspace configurations for an hour or more. As required by the problem instance specification, all 15 second advisories achieve costs within 25% of the minimum cost. Ten of the advisories achieve costs that are within 10% of the minimum cost.

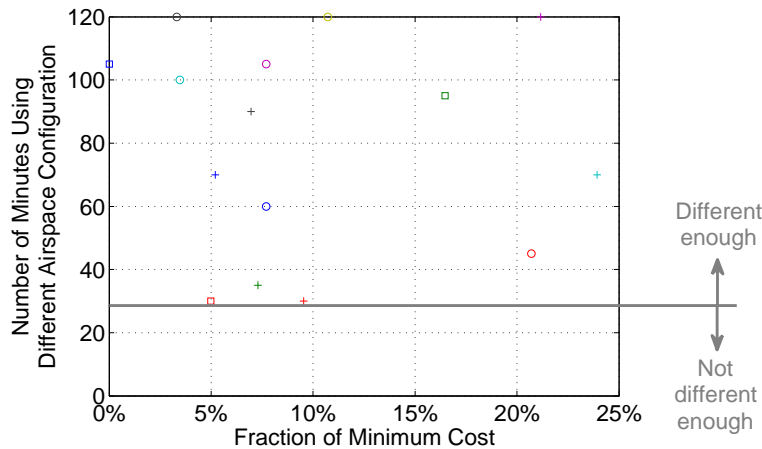


Figure 11. Fraction of minimum cost versus difference from minimum-cost advisory achieved by the second advisory returned by the SDA* heuristic.

The algorithms were coded in Java and executed on a MacPro workstation with a Quad-Core Intel Xeon 2.8 GHz processor and 4 GB of memory. The heuristic algorithm solved each problem instance in less than one second. The more exhaustive search documented in sub-section III.D required up to 20 minutes per problem instance.

VIII. Conclusions

In this paper, we motivate the need for presenting area supervisors with multiple low-cost and distinct area configuration schedule advisories. First, related research from artificial intelligence, operations research, and air traffic management demonstrates the value of presenting decision-makers with multiple distinct advisories. However, no approach in the literature was found that directly applies to the generation of multiple area configuration advisories, so a new problem statement and solution approach are needed. Second, presenting such advisories to a decision-maker is important because any model and problem objective function are incomplete and imperfect. Third, area supervisors themselves requested multiple good and distinct advisories for their consideration. Fourth, an investigation of the set of near-optimal advisories indicates that more often than not the lowest-cost advisories are too similar to be useful, motivating the need for a constraint

enforcing advisory distinctness.

Based on this motivation, the M - ε - d -CSAs problem statement is specified. This problem is an extension of a single-advisory problem statement defined in previous work.⁸ It requests a set of M advisories such that one is optimal and the others are within a fraction ε of the minimum cost. Each pair of advisories must use different airspace configurations for d or more configuration time steps to ensure that each advisory is sufficiently distinct from the others in the set. Hopefully, at least one advisory in the set of returned advisories will perform well enough to be implemented with respect to all relevant factors, whether accurately captured in the problem model and objective or not.

To further illustrate and motivate the new problem statement, we specify a synthetic M - ε - d -CSAs problem instance and solve it with a novel heuristic based on the A^* algorithm and referred to as Sequential Distinct A^* . Since the advisories in the solution are distinct but both achieve relatively low costs (both within 25% of the minimum cost), an area supervisor could further evaluate them based on factors not considered by the problem model and objective to judge which should be selected for implementation. Finally, Sequential Distinct A^* solutions for 18 problem instances based on historical data demonstrate that the algorithm can return feasible advisories in a relatively small amount of time. More concretely, in addition to a minimum-cost advisory, a distinct second advisory with a cost that is within 25% of the minimum cost was returned for 15 of the 18 instances. The algorithm returned a solution in less than one second for each problem instance when executed on a workstation computer.

Acknowledgments

We are grateful to Karl Bilimoria and Michael Drew for suggestions and feedback on this work. We also appreciate the patient assistance provided by Chok Fung (Jack) Lai as we wrote software to implement algorithms for the problem. Finally, we thank the subject-matter experts who proposed the basic idea of this problem in the first place and also helped refine it. These experts include but are not limited to Mark Madden, Brian Hanlon, Kevin Shelar, Al Mahilo, Tom Roherty, Connie Atlagovich, Bill Hikade, Steve Herbruck, Martin Mielke, Don Lamoreaux, Rick Buentello, Dale Juhl, Todd Wargo, Mike Klupenger, Mark McCurdy, Stephen Hughes, Raphell Taylor, Miguel Anaya, Mark Evans, Brian Holguin, Wayne Bridges, and Bill Preston. The historical airspace configuration data used in this research was provided by ATAC.

References

- ¹Federal Aviation Administration, "Order JO 7210.3X Facility Operation and Administration," http://www.faa.gov/air_traffic/publications/atpubs/FAC/index.htm, February 2012.
- ²Bloem, M. and Kopardekar, P., "Combining Airspace Sectors for the Efficient Use of Air Traffic Control Resources," *Proc. of AIAA Guidance, Navigation, and Control Conference and Exhibit*, Honolulu, HI, August 2008.
- ³Bloem, M., Gupta, P., and Kopardekar, P., "Algorithms for Combining Airspace Sectors," *Air Traffic Control Quarterly*, Vol. 17, No. 3, 2009.
- ⁴Drew, M. C., "A Method of Optimally Combining Sectors," *Proc. of AIAA Aviation Technology, Integration and Operations Conference*, Hilton Head, SC, September 2009.
- ⁵Dorado, M., "ETLM Support tool for Dynamic Airspace Configuration," *NASA Dynamic Airspace Configuration Workshop*, Moffett Field, CA 2009.
- ⁶Bloem, M. and Gupta, P., "Configuring Airspace Sectors with Approximate Dynamic Programming," *Proc. of International Congress of the Aeronautical Sciences*, Nice, France, September 2010.
- ⁷Tien, S.-L., Hoffman, R., and Schonfeld, P., "En Route Sector Combination Scheme to Minimize Air Traffic Controller Staffing," *Proc. of Transportation Res. Board Annual Meeting*, Washington, DC, January 2012.
- ⁸Bloem, M., Drew, M. C., Lai, C. F., and Bilimoria, K., "Advisory Algorithm for Scheduling Open Sectors, Operating Positions, and Workstations," *Proc. of AIAA Aviation Technology, Integration, and Operations Conference*, Indianapolis, IN, September 2012.
- ⁹Nguyen, T., Do, M., Gerevini, A. E., Serina, I., Srivastava, B., and Kambhampati, S., "Generating Diverse Plans to Handle Unknown and Partially Known User Preferences," *Artificial Intelligence*, Vol. 190, October 2012, pp. 1–31.
- ¹⁰Bellman, R. and Kalaba, R., "On k th Best Policies," *J. SIAM*, Vol. 8, No. 4, December 1960, pp. 582–588.
- ¹¹Hoffman, W. and Pavley, R., "A Method for the Solution of the N th Best Path Problem," *J. ACM*, Vol. 6, No. 4, October 1959, pp. 506–514.
- ¹²Dreyfus, S. E., "An Appraisal of Some Shortest-Path Algorithms," Memorandum RM-5433-1-PR, United States Air Force Project RAND, September 1963.
- ¹³Eppstein, D., "Finding the k Shortest Paths," *SIAM J. Comput.*, Vol. 28, No. 2, 1998, pp. 652–673.
- ¹⁴Jiménez, V. M. and Marzal, A., "Computing the K Shortest Paths: A New Algorithm and an Experimental Comparison," *Algorithm Engineering*, edited by J. Vitter and C. Zaroliagis, Vol. 1668 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, 1999, pp. 15–29.

- ¹⁵Suurballe, J. W., "Disjoint Paths in a Network," *Networks*, Vol. 4, 1974, pp. 125–145.
- ¹⁶Suurballe, J. W. and Tarjan, R. E., "A Quick Method for Finding Shortest Pairs of Disjoint Paths," *Networks*, Vol. 14, 1984, pp. 325–336.
- ¹⁷Andreas, A. K. and Smith, J. C., "Exact Algorithms for Robust k -Path Routing Problems," *Proc. of the GO*, 2005.
- ¹⁸Peng, C. and Shen, H., "An Improved Approximation Algorithm for Computing Disjoint QoS-Paths," *Proc. of IEEE International Conference on Systems and Networking*, April 2006.
- ¹⁹Andreas, A. K., Smith, J. C., and Küçükyavuz, S., "Brand-and-Price-and-Cut Algorithms for Solving the Reliable h -Paths Problem," *J. Global Optimization*, Vol. 42, No. 4, December 2008.
- ²⁰Loh, R. C., Soh, S., and Lazarescu, M., "Edge Disjoint Paths with Minimum Delay Subject to Reliability Constraint," *Proc. of IEEE Asia-Pacific Conference on Communications*, Shanghai, October 2009.
- ²¹Loh, R.-C., Soh, S., and Lazarescu, M., "An Approach to Find Maximal Disjoint Paths with Reliability and Delay Constraints," *Proc. of IEEE International Conference on Advanced Information Networking and Applications*, June 2009.
- ²²Montoya, J., Wood, Z., and Rathinam, S., "Runway Scheduling Using Generalized Dynamic Programming," *Proc. of AIAA Guidance, Navigation, and Control Conference*, Portland, OR, August 2011.
- ²³Marceau, G., Savéant, P., and Schoenauer, M., "Strategic Planning in Air Traffic Control as a Multi-objective Stochastic Optimization Problem," *Proc. of USA/Europe Air Traffic Management Research & Development Seminar*, Chicago, IL, June 2013.
- ²⁴Cook, L. S. and Wood, B., "A Model for Determining Ground Delay Program Parameters Using a Probabilistic Forecast of Stratus Clearing," *Proc. of 8th USA/Europe Air Traffic Management Research & Development Seminar*, Napa, CA, June 2009.
- ²⁵Rantanen, E., "Development and Validation of Objective Performance and Workload Measures in Air Traffic Control," Technical Report AHFD-04-19/FAA-04-7, Aviation Human Factors Division, Institute of Aviation, University of Illinois, September 2004.
- ²⁶Kopardekar, P., Schwartz, A., Magyarits, S., and Rhodes, J., "Airspace Complexity Measurement: An Air Traffic Control Simulation Analysis," *Proc. of 7th USA/Europe ATM Seminar*, Barcelona, Spain, July 2007.
- ²⁷Yousefi, A., Hoffman, R., Lowther, M., Khorrani, B., and Hackney, H., "Trigger Metrics for Dynamic Airspace Configuration," *Proc. of AIAA Aviation Technology, Integration, and Operations Conference*, Hilton Head, SC, September 2009.
- ²⁸Jung, J., Lee, P., Kessel, A., Homola, J., and Zelinski, S., "Effect of Dynamic Sector Boundary Changes on Air Traffic Controllers," *Proc. of AIAA Guidance, Navigation, and Control Conference*, Toronto, Canada, August 2010.
- ²⁹Tien, S.-L. A. and Hoffman, R., "Optimizing Airspace Sectors for Varying Demand Patterns using Multi-Controller Staffing," *Proc. of USA/Europe Air Traffic Management Research & Development Seminar*, Napa, CA, June 2009.
- ³⁰Tien, S.-L., *Demand-Responsive Airspace Sectorization and Air Traffic Controller Staffing*, PhD dissertation, University of Maryland, College Park, MD, 2010.
- ³¹Byers, T. H. and Waterman, M. S., "Determining All Optimal and Near-Optimal Solutions When Solving Shortest Path Problems by Dynamic Programming," *Oper. Res.*, Vol. 32, No. 6, 1984, pp. 1381–1384.
- ³²Waterman, M. S. and Byers, T. H., "A Dynamic Programming Algorithm to Find All Solutions in a Neighborhood of the Optimum," *Mathematical Biosciences*, Vol. 77, 1985, pp. 179–188.
- ³³Hart, P. E., Nilsson, N. J., and Raphael, B., "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Trans. Systems Science and Cybernetics*, Vol. 4, No. 2, July 1968, pp. 100–107.
- ³⁴Patel, A., "Amit's A* Pages," <http://theory.stanford.edu/~amitp/GameProgramming/>, January 2013.