Externalizing Planning Constraints for More Effective Joint Human -Automation Planning

Ronald SCOTT[a], Emilie ROTH[b], Beth DEPASS[a] and Jeffrey WAMPLER[c]
[a]*Raytheon BBN Technologies*
[b]*Roth Cognitive Engineering*
[c]*Air Force Research Laboratory, Human Performance Directorate, 711HPW/RH*

ABSTRACT

We describe a prototype cognitive work aid for airlift mission allocation and scheduling. A key design challenge was how to capture, represent, and utilize human-generated planning constraints that need to be respected when automatically replanning across multiple missions in response to situational changes. User feedback solicited via a formal user evaluation confirmed that the visualization and control mechanisms provided enabled them to understand and control the plans generated by the automated scheduler, and that the prototype aid allowed them to better assess and respond to large situational changes with across mission impacts. The work provides concrete illustrations of methods for externalising planning constraints so that they can be recognized and respected across distributed planning agents both human (airlift planners) and machine (automated scheduler). It serves to extend the range of available techniques for design of more effective joint cognitive systems.

KEYWORDS

Command and control; Transportation; Common Ground; Joint Cognitive Systems; Human-Automation Integration; Planning;

INTRODUCTION

This paper presents our most recent work leveraging automated support for airlift mission planning and scheduling. Our approach combines visualization and user-interaction principles to foster more effective joint performance between automated planners and schedulers and human users that together constitute the *joint cognitive system* (Woods & Hollnagel, 2006). Over the past decade we explored a variety of concrete techniques for design of joint cognitive systems across multiple decision-support development projects (DePass, Roth, Scott, Wampler, Truxler & Guin, 2011; Scott, Roth, Truxler, Ostwald & Wampler, 2009; Truxler, Roth, Scott, Smith and Wampler, 2012). These have included techniques to enhance *observability,* by visually representing the problem to be solved and candidate solutions in a form that users can easily understand, evaluate, and contribute to; and techniques to enhance *directability,* through control mechanisms that enable users to bound and direct the automated solution generation process (Roth, DePass, Scott, Truxler, Smith, and Wampler, in preparation). The goal is to enable the joint system to perform better than either the person or the automation could on its own.

The present application posed new design challenges that served to extend our cadre of techniques for observability and directability. In particular it required planning constraints that were implicit in the air mission plans generated by human planners to be exposed and explicitly represented so as to enable the automated planner to respect these constraints when situational changes dictated the need for dynamic replanning. In this paper we describe the design challenges, an initial prototype we developed and tested that begins to address these design challenges, as well as future directions.

BACKGROUND

Our work concerns the joint process of planning, scheduling, and execution of air missions by two cooperating military organizations. USTRANSCOM is charged with directing and executing the overall transportation needs for deployment of troops and distribution of goods via air, sea, and ground movements. The Air Mobility Command (AMC) is charged with executing the air movements. The first stage, referred to as *early planning,* begins at the point at which air movement requirements are locked in to the enterprise three weeks before movement. At that stage, planners at USTRANSCOM examine the entire known set of requirements, and match those against the limited resources they have to plan with – the number of aircraft that are scheduled to be available to AMC day by day, and the available throughput capacities of the airfields to be used to deliver those requirements. The second stage, detailed planning (also known as scheduling) is the focus of the current work.

Once requirements are released by USTRANSCOM, detailed planners at AMC produce schedules of air missions to be flown, and maintain those schedules in an executable state (as changes happen) until the schedules are turned over to the AMC execution cell twenty-four hours before the air missions take off. The third stage, execution, covers the period from 24 hours before takeoff until the end of each air mission, and is handled by a group of Duty Officers (DOs) in AMC. In previous projects we developed systems to support both early planning (DePass et al, 2011 ; Truxler et al., 2012) as well as dynamic rescheduling during execution (Scott et al., 2009). The work described in this paper is aimed at improving detailed planning at AMC, both in terms of ease of producing the schedules and the quality of the schedules turned over to the AMC execution cell.

CURRENT PRACTICE

The scope of the decisions that must be made by AMC planners, both in terms of numbers of decisions and complexity of decisions is far greater than either of the other two stages. Further the quality of decisions made in this stage has the largest effect on the stated enterprise goals of effectiveness (delivering requirements on time) and efficiency (at a minimal cost). To transition from individual air movement requirements to executable mission schedules, there are four types of decisions to be made, that are split between two offices within AMC.

1. The first decision, known as *aggregation*, identifies a set of air movement requirements (or the pieces of a single air movement requirement) to occupy a single aircraft.
2. The aircraft resources available to planners are divided into wings, groups of like aircraft which share a common home airfield base. The second decision, known as *allocation*, determines the wing from which to source an aircraft for a particular aggregation of requirements as well as the allocation interval – the set of days for which an aircraft from that wing will be available for this use.
3. Air movement requirements specify cargo or passengers to be picked up at one location and dropped off at a second location. While not always possible, planners will try to link two or more air movement requirements together to be serviced by the same aircraft, in order to reduce the number of hours an aircraft will fly empty. The decision on how to combine movements together into a more efficient home-base to home-base route is referred to as *chaining*.
4. Finally, the creation of a detailed schedule for an entire home-base to home-base route is referred to as *scheduling*. Scheduling requires selection of en route stops to be made for refueling or crew rest, as well as determining takeoff and landing times for each of the flight legs (sorties). A host of details affecting timing must be taken into account in scheduling, including operating hours for each airfield to be visited, limited airfield parking capacity, and regulations on crew duty day lengths.

The four decisions above cannot be made independently. Making the aggregation decision entails knowing what kind of aircraft is available, in order to know how much cargo can be put on the aircraft. The aggregation decision may need to be revisited if aircraft of that type turn out not to be available. Scheduling requires having aggregation, allocation, and chaining decisions already made – but difficulties in scheduling arising from overuse of enroute port capacity, for example, will require increasing the allocation interval for an aircraft, possibly leading to the aircraft not having suitable availability at all.

Making the decision process more complicated is the fact that the underlying planning state is not static. Both the numbers of aircraft scheduled to be available and the operating hours of the airfields can change with little notice. The air movement requirements that in theory were locked in three weeks before movement, in fact, are not static either. Requirements may be cancelled, they may change their schedule (as the availability of the cargo changes), they may change the amount of cargo to be carried. And, of course, as the enterprise reacts to military contingencies, to disaster relief and humanitarian assistance needs, late requirements (which tend to be high priority) can and will be added in to the mix at any stage.

Adding further complication, these decisions are made by two independent, but cooperating, AMC offices. *Mission planners* are responsible for receiving the requirements from USTRANSCOM and creating the executable mission schedules that are eventually published to the AMC execution group. *Barrels*, who liaise between AMC and the wings decide which wing (and type of aircraft) a mission will be assigned.

In the current practice, the process followed is roughly linear and time consuming. First the aggregation decision is made by a mission planner. The mission planner may then ask a barrel planner for an allocation decision, or may wait a day or two while he looks for an appropriate second aggregation to pair it with (effectively making the chaining decision) before asking the barrel for an allocation.The barrel planner will make the allocation decision, and may suggest a chaining solution if the mission planner has not already done so. The mission planner will then prepare the detailed schedule, respecting aggregation, allocation, and chaining decisions.

While this process works, it leads to overall slates of mission schedules that are neither as efficient (cost-wise) nor as effective (in terms of delivering requirements on time) as they could be. These deficiencies stem from two factors. First is the sequential nature of the current process. When the barrel planner is asked for an allocation decision for a particular mission, he makes the best decision he can at that time, with the information he has available at that time. It might be that the very next allocation decision he is asked for would make better use of the aircraft he has just allocated to this mission, and an overall better slate of mission schedules would result if he changed his earlier allocation decision. Second, in the current process, decisions are unlikely to be revisited. Both mission planners and barrel planners are reluctant to remake a decision, for three reasons :

- Lack of time. Both mission planners and barrel planners are kept quite busy.
- Lack of visibility into how a change one planner would make would affect other planners. Either type of planner is reluctant to make a change that would cause potentially time-consuming rework for the other type of planner.
- Lack of cuing as to what the value of changing a decision might be. Even given their lack of time, if the planners could see what the value of a change would be, they would make time for the more valuable changes.

DESIGN OBJECTIVES AND CHALLENGES

The design objective was to leverage automated scheduling software to enable more rapid and efficient mission allocation and scheduling. The goal was to make more effective use of the limited airlift assets (e.g., reduce empty flying hours) and reduce overall costs (e.g., reduce overall flying hours which equates to reduced operating and fuel costs). More particularly, the objective was to facilitate replanning across multiple missions (mission reallocation and scheduling) when situational changes (e.g., an airfield closure ; a new, high priority emerging requirement) necessitated revisiting prior allocation and scheduling decisions to make most efficient use of assets, meet as many requirements as possible, and minimize overall costs.

To achieve these goals, we needed to address the factors we'd identified as obstacles to efficient replanning in current operation. We needed to design a system that allowed barrels and mission planners to more rapidly plan (and replan) missions than is possible with today's tools . The system also needed to make clear the changes made to missions, particularly changes that affected prior decisions made by others. Finally, the system needed to provide clear indication of the impact of the replan on high level efficiency (e.g., total flying hours ; empty flying hours) and effectiveness (e.g., number of missions delivering late) metrics. The system description below explains how all these design objectives were met.

One of the most significant technical challenges we faced was how to capture, represent, and utilize planning constraints that Barrels and Mission Planners consider in allocating assets and developing detailed schedules that were not explicitly represented in a system. Examples include cases where a mission was assigned to a particular wing because it required a plane type that was only available at that wing; and cases where a mission had to stop and refuel at a particular location because other refueling locations were temporarily dedicated to a different mission type (e.g., Ebola humanitarian aid missions). These detailed, mission-specific, planning constraints are not currently formally captured (appearing in comments sections if anywhere). The consequence is a lack of common ground with respect to relevant mission constraints across the distributed planning team (i.e., the Barrel, the Mission Planner, and the DO on the execution floor). The lack of common ground results in inefficiencies and scheduling errors, particularly when dynamic replanning, by someone other than the original planner, is required. The consequences of lack of common ground is amplified when one of the elements of the distributed planning team is an automated scheduler that is being relied upon to reschedule across multiple missions in response to situational events that create plan perturbations. In order to enable robust automated replanning that respects important (currently implicit) constraints it became necessary to develop mechanisms to allow Barrels and Planners to explicitly communicate planning constraints. These could be then be externally represented and used by the automated scheduler, effectively creating common ground across the broader distributed team that includes the automated scheduler, for more effective joint human automation planning.

In the next section we describe the prototype that was built and illustrate how it enabled users to communicate constraints that were then respected by the automated scheduler when a large-scale replan was required.

DESIGN FEATURES

We designed and implemented a prototype cognitive work aid to be used by both Mission Planners and Barrels. Each is supported with dynamic visualizations specifically designed for their needs. Mission planner visualizations display full details of individual mission schedules, and offer the ability to drag pieces of mission schedules with immediate feedback as to constraint violations. Barrel visualizations are organized around how the aircraft assets for particular wings are currently allocated, again offering Barrels direct graphical manipulation of elements of the plan with immediate alerting to conflicts. Our prototype offers a wide range of capabilities – we cannot discuss the full set of features in this paper. Here we concentrate on the design features that enable users to communicate planning constraints that are then explicitly represented in shared visualizations and respected by the automation when called on to replan.

In the simplest case a user can request the automation to schedule a new mission given a requirement (what to move, starting location, destination and by when it needs to arrive) and the scheduler will build a mission taking into account the current missions, wing allocations and airport constraints, possibly aggregating with other existing missions that match well in time and location. The resulting detailed mission can then be modified by the user; for example, a user may drag a sortie (a flight betwen two ports) in a timeline view to manually indicate when a sortie

should arrive or depart. The system will adjust the other sorties accordingly making sure to immediately present any violations caused by the new schedule.
Importantly, the user can explicitly indicate a number of constraints that the scheduler needs to respect before asking the automation to schedule or reschedule one or more missions. Such constraints become directives and restrictions communicated to the scheduler which must be incorporated into the mission schedules being generated. For example, a constraint may specify that the mission must use an allocation from a certain wing. Another constraint is to specify that a sortie must arrive at a port within a certain time window (e.g., to account for customs and overflight clearances) ; or that it can only remain on the ground at the airbase for a limited amount of time. Other constraints include that a mission may not refuel at a certain location or must refuel at a certain location. Similarly they can specify a particular location for a rest stop or indicate that a particular location cannot serve as a rest stop.
In all of the above cases, it should be noted that the user can always manually adjust schedules output by the automation or even specify additional constraints and retask the automation to incorporate those changes into a revised mission schedule.

FUNCTIONAL EXAMPLE

The following series of screenshots illustrate how users would interact with the prototype to respond to a new emerging requirement that creates an over demand for available airlift assets requiring a broad across-mission reschedule. They illustrate how visualizations enable users to assess repercussions of situational changes, use the automated scheduler to revisit and repair previously scheduled missions, and direct which missions can be changed and how, by placing explicit constraints that are then respected by the automated scheduler.

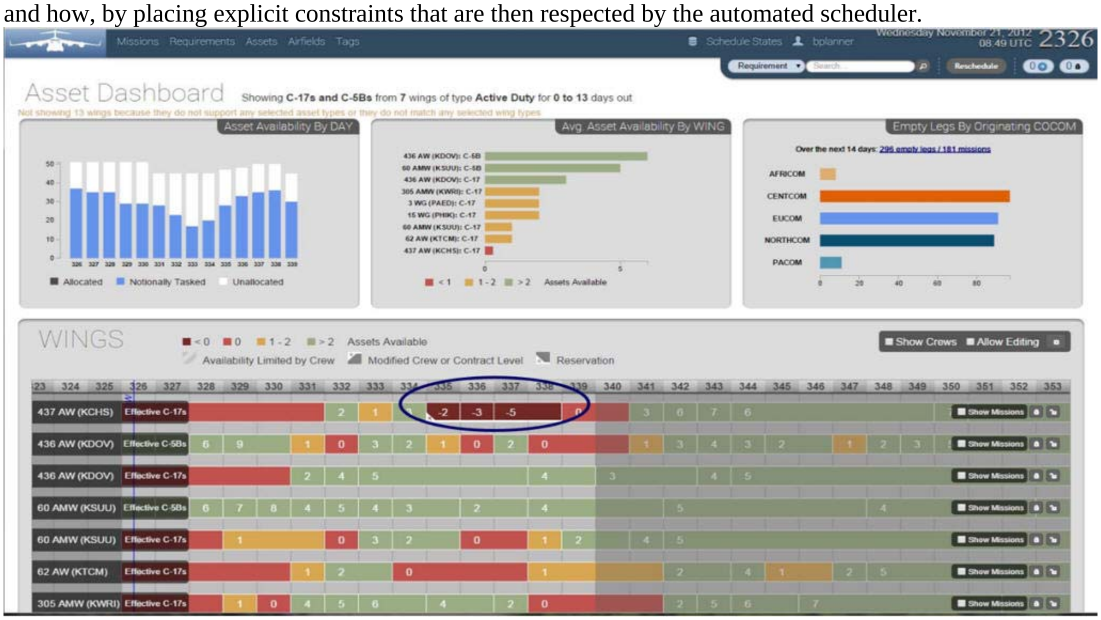
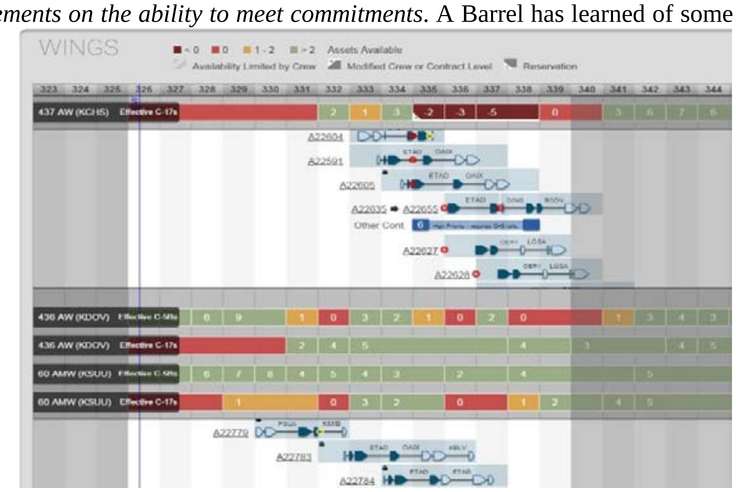


Figure 1: Asset Dashboard with Aircraft Overcommtment at the CHS Wing

Visualizing impacts of emerging requirements on the ability to meet commitments. A Barrel has learned of some high priority missions about to be planned that will require 6 aircraft from the Charleston (KCHS) wing. To understand the intial implications, the barrel adds a reservation for 6 aircraft for the allocation window he believes will be needed to support the missions in the KCHS wing and looks at the initial impact. As shown in Figure 1, the prototype visualization will immediately show that this will overcommit tails at KCHS on days 335 – 338 (depicted as red boxes with negative numbers).



Assessing impacts on particular missions. As shown in Figure 2, the user can expand a given wing to view the currently allocated missions against the wing in a timeline form. From this timeline view he can observe which missions are impacted by the overcommitment (i.e., the missions with red dots under the KCHS wing) and he can look at more detailed mission information via tooltips and other gestures on the timeline such as the cargo pickup locations for the missions, their allocation windows, their priority, and required delivery dates.

Revising mission schedules to resolve over commitment. The barrel could start to manually resolve the overcommitment by sliding mission times off to the right to avoid the overcommitment time, or rewing specific missions, noting violations that will appear as he modifies schedules. However, given so many missions are affected the Barrel may prefer to rely on the automated scheduler to rapidly generate a new across-mission schedule that minimizes mission delays.

Defining constraints to be respected by the automated scheduler. The barrel may choose to define constraints to be respected by the scheduler before calling it to resolve the overcommitment. For example, he can specify that some missions out of KCHS must remain where they were originally scheduled (locked in place) even if they coincide with the new high priority missions. Figure 2 provides some examples where the Barrel specified that missions be locked in place. These are indicated by the lock icon next to the mission schematic. As an alternative constraint, the Barrel can specify that a mission needs to continue to come from a particular wing (e.g., because only that wing has appropriately configured tails), but that it can slide in time.

Figure 2: Expanded Mission Timeline View Depicting Locks

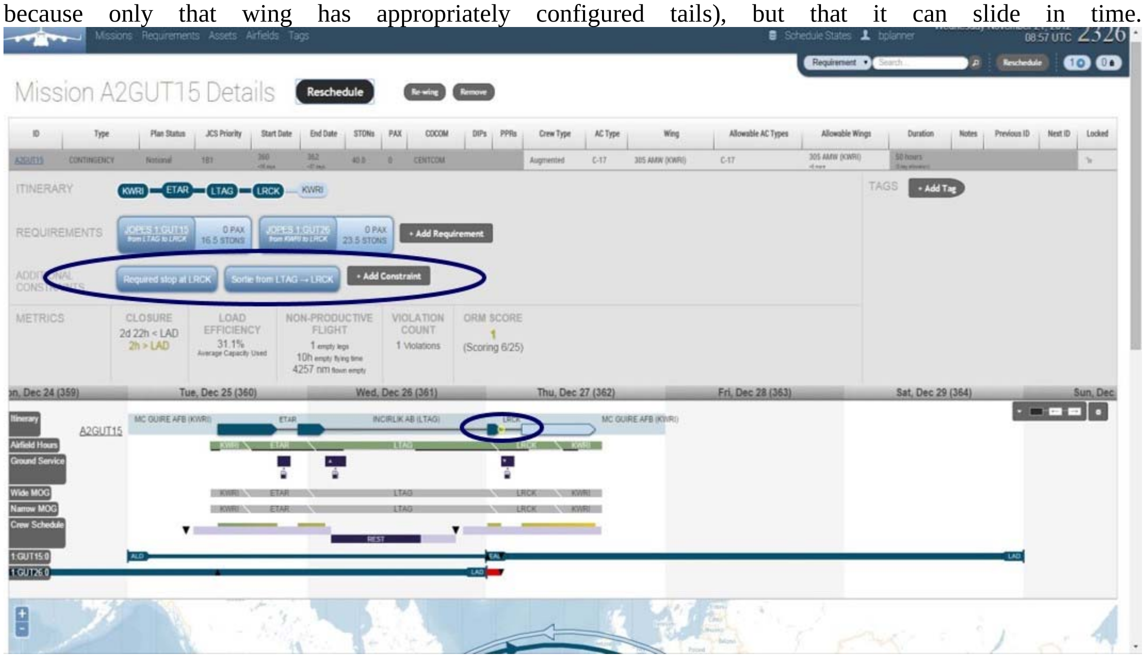


Figure 3: Detailed Mission Planning View with Mission Constraints

In addition to imposing constraints on wing allocation, the Barrel (or a Mission Planner) can impose constraints on individual mission schedules. Figure 3 shows a screenshot of a Detailed Mission Planning view for a single mission with multiple constraints : a timing constraint related to flight time of a specific sortie (LTAG to LRCK), and timing constraints related to a required mission stop (at LRCK). If the shifting of missions to accomodate the additional demand on aircraft out of KCHS starts to affect missions out of other East Coast wings such as McGuire (KWRI), this mission could be rescheduled, but the constraints entered by the original mission planner would be respected in the new schedule generated by the automated scheduler.

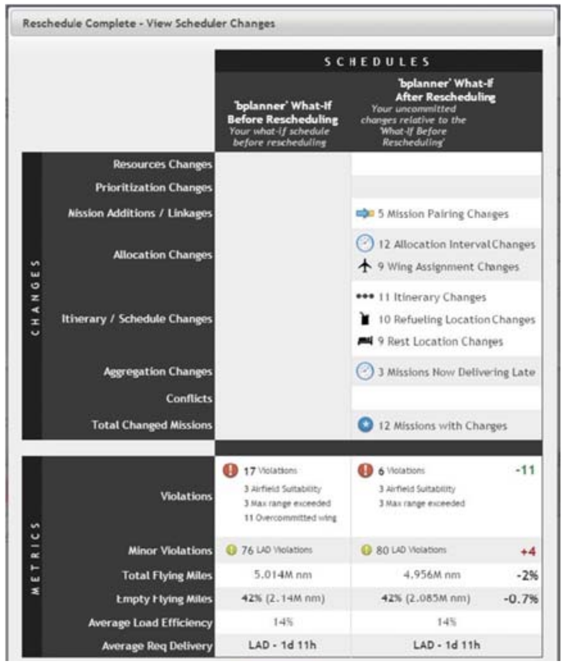


Figure 4 Change Summary View

Inspecting and evaluating the revised plan produced by the scheduler. Once the user(s) have defined all constraints, the automation can be invoked to reschedule missions to accomodate the overcommitment of aircraft out of the Charleston wing. The automation will shift missions to other wings where possible, shift missions later if their required delivery dates allow, and sometimes shift missions such that they are late delivering their requirements, if unavoidable. As illustrated in Figure 4, the Change Summary views available after a reschedule will detail the types of changes made, their impact on overall metrics and the details of each mission change, allowing a planner or barrel to evaluate the solution as a whole or by specific mission changes. As shown in Figure 4, the Before Rescheduling column indicates the base state, that is, *before* the user asked the automation to reschedule. The right column provides details of what was changed when the automation rescheduled missions, taking into account the new reservation and various barrel wing and mission constraints. You can see there were 5 mission pairing changes – these indicate missions were chained together which will increase efficiency, there were 12 allocation interval changes – meaning the allocation duration for which aircraft will be reserved for missions from various wings changed. Looking further down, 9 missions had their wing assignments changed while others had sortie details such as refueling or rest location changes. 3 missions are now delivering past their required delivery dates.

The Change Summary view also summarizes impact on overall efficiency metrics. Note that in this case the metrics at the bottom of the view which consider all missions in the system did not experience large perturbations as a result of the reschedule. In fact, empty flying miles and total miles were reduced likely due to the newly chained missions.

While the summary view provides a broad overview, planners whose missions are impacted would need to inspect and evaluate the mission changes in more detail. To delve into the details the user can click on either the category of interest or the Total Changed Missions row (12 Missions with Changes in the example) to see a multi-mission timeline view of each mission with their before and after schedules. The screenshot below shows details on five missions with changes – 1 row contains the details for each mission changed and if the mission was chained with another the timeline will represent each mission before chaining as well as the resulting chained mission. The left column provides a quick look indication of the categories of change that apply to that mission.

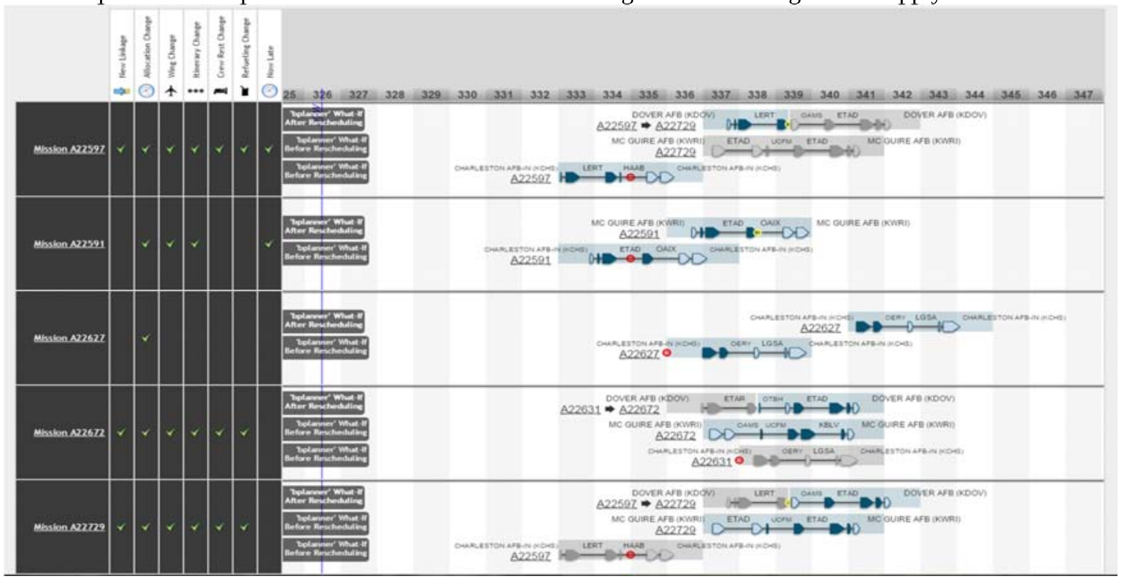


Figure 5 : Changed Mission Details

As the example illustrates the set of visualizations provides the distributed team - Barrels, Mission Planners, and the automated scheduler - with a shared representation of the missions to be scheduled, the constraints that need to be respected, and whether any constraints are violated by the present schedule, supporting common ground. Automated scheduler technology is leveraged to enable more efficient and effective schedules, while providing users control mechanisms to direct the automation via explict representation of constraints to be respected.

USER EVALUATION

A user evaluation was conducted at the completion of this first development cycle. Nine current practitioners, 3 Barrels and 6 detailed Mission Planners participated. A live demonstration of the prototype was presented using

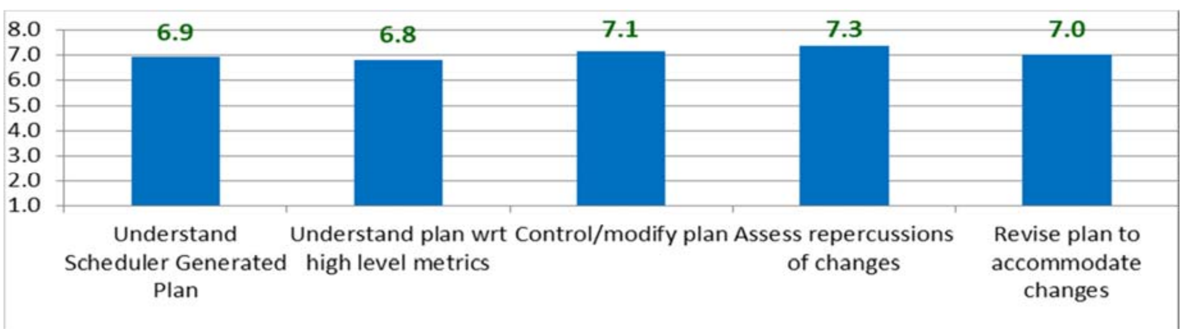


Figure 6: : Mean rating scores on user evaluation questions (8-point scale ; 8 = extremely good)

representative scenarios. Feedback was obtained via a written questionnaire that included 8-point scale rating questions eliciting feedback on the usability and usefulness of the capabilities demonstrated, and open-ended questions soliciting suggestions for additional capabilities to incorporate into future iterations.

Participant feedback, as reflected in both verbal comments and closed-form rating questionnaire scores was highly positive. As shown in Figure 6 participants indicated that they were able to understand and control the plans generated by the automated scheduler, and that the prototype aid would allow them to better assess and respond to large situational changes that had across mission impacts.

SUMMARY AND DIRECTIONS FOR FUTURE WORK

This paper provides an interim point description of our current application. Although our user evaluation results show that we have made significant progress towards delivering a mission scheduling capability to support AMC mission planners and barrel planners, we continue to improve system capabilities and particularly capabilities for capturing, representing, and utilizing mission planning constraints. To ensure the automated schedules meet the constraints known to the Barrels and Planners, it is imperative that Barrels and Planners be able to enter the constraints underlying the detailed planning decisions they consider to be important, sufficient so that any schedule that meets those constraints will be acceptable to the planners. This requires :

1. A sufficiently rich set of constraints for the planners to be able to detail their planning needs to the system.
2. A simple and effective mechanism to enter, visualize, and edit those constraints so that managing these constraints is not an undue burden on the planners.
3. The ability of the automated scheduler to respect these constraints in its production of schedules.

While we have made progress on this path, more research and development is required. Our future work will primarily focus on expanding the set of constraints available for planners to use to describe their planning needs. The first broadening of the space of constraints will concentrate on adding conditions to the constraints. In the current system, for example, a planner might define a constraint that a particular airfield must be used as an enroute stop (for either refueling or crew rest). For the most part this unconditional constraint makes sense to the planners. But there are times when they would like to represent a more complicated pattern : Use airfield A as an enroute stop, as long as we are traveling east out of CONUS (or as long as we are using an aircraft from wings 1 or 2, or as long as we are starting this mission between days 52 and 54, to pick some more examples). And the planner might add an alternative – to use airfield B as the enroute stop if other conditions are met. The set of conditions available to planners clearly (given the examples above) will have to include methods of geographical reasoning, temporal reasoning, as well as simple boolean logic. The challenge in extending our system will be maintaining condition 2 – ensuring the ability of the planners to continue to manage this (expanded) set of constraints.

A second needed expansion of the constraint language arises from the fact that mission planners are not always planning individual air missions. They often schedule entire *movements* of air missions – some number of missions that are going to and from the same set of airfields, spread over some number of days. In the case of scheduling of larger movements, the mission planner can often identify particular constraints that apply to each of the individual missions that make up the movement. This leads to the desire for mission planners to be able to define a single constraint that will be applied to each of a set of missions. Our prototype system already has the notion of tags – one or more text strings attached to each mission. A set of missions, such as a movement, can be identified as the set of missions containing a particular tag. It will be relatively straightforward to allow constraints to be added at the tag level, instead of at the mission level.

While our projects have a primary aim of meeting our customers' needs, our objective is to also contribute to the generic corpus of reusable techniques for fostering more effective joint cognitive systems by making automated planners more observable and directable. In this case we provided a concrete illustration of methods for externalising planning constraints, so they can be recognized and respected across distributed planning agents both human (Airlift Planners) and machine (automated scheduler). It is our hope that the present work serves to extend the range of available techniques for design of collaborative automation.

REFERENCES

DePass, B., Roth, E. M., Scott, R., Wampler, J. L., Truxler, R., and Guin, C. (2011). Designing for collaborative automation: A course of action exploration tool for transportation planning. In *Proceedings of the 10th International Conference on Naturalistic Decision Making*, May 31-June 3, 2011, Orlando, FL. (pp. 95 - 100)

Scott, R., Roth, E. M., Truxler, R., Ostwald, J., Wampler, J. (2009) Techniques for effective collaborative automation for air mission replanning. In *Proceedings of the Human Factors and Ergonomics Society 53rd Annual Meeting*. (pp. 202-206). Santa Monica, CA: HFES.

Truxler, R., Roth, E., Scott, R., Smith, S., and Wampler, J. (2012) Designing collaborative automated planners for agile adaptation to dynamic change. *Proceedings of the Human Factors and Ergonomics Society 56th Annual Meeting (pp.* 223- 227) Santa Monica, CA: HFES.

Woods, D. D. and Hollnagel, E. (2006). *Joint Cognitive Systems: Patterns in Systems Engineering*. Boca Raton, FL: Taylor & Francis.