

Deploying an Intelligent Pairing Assistant for Air Operation Centers

Jeremy Ludwig¹, Bart Presnell¹, Les Loomis², Drew Decker³, Kevin Light³, & Eric Geiselman⁴

¹ Stottler Henke Associates, Inc., San Mateo, CA; ludwig@stottlerhenke.com

² Monterey Technologies Inc., Monterey, CA; lloomis@montereytechnologies.com

³ Intelligent Software Solutions, Inc., Colorado Springs, CO; drew.decker@issinc.com

⁴ Air Force Research Laboratory, 711th HPW/RHCV, WPAFB, OH; Eric.Geiselman@wpafb.af.mil

Abstract

Within an Air Operations Center (AOC), planners make crucial decisions to create the air plan for any given day. They are expected to complete the plan in part by pairing targeting or collection tasks with the available platforms. Any assistance these planners can acquire to help create the plan in a timely manner would make the entire process more efficient and effective. This paper describes the Intelligent Pairing Assistant (IPA) decision aid, which provides pairing recommendations at specific decision points in the planning process. IPA is deployed as a plug-in to the Master Air Attack Plan Toolkit (MAAPTK) software system already in use within AOCs. The primary contributions of this case study are applying artificial intelligence techniques to a novel domain and discussing the software evaluation efforts in moving from a prototype to a deployed system in this high-stakes domain.

Introduction

The role of an Air Operation Center (AOC) in the United States Air Force is to provide command and control of air operations. Simply put, the AOC receives a high-level description of tasks and effects and generates a plan of how to best execute them. Within an AOC, planners make crucial decisions to create the overall air plan for any given day. They are expected to complete the plan in a limited amount of time—in part by pairing collection tasks consisting of intelligent, surveillance, and/or reconnaissance (ISR) requests with the available platforms. Any assistance these planners, especially the less experienced ones, can obtain to help create the air plan in a timely manner would make the entire process more effective.

One major challenge is making the best use of available resources. For instance, rather than assigning an unmanned

aerial vehicle to a collection task, it may be more cost effective and expedient to further task a manned aircraft that is already operating in the area. Hurried human planners often overlook opportunities to take advantage of relationships between tasks, even though combining missions often results in collecting more intelligence with fewer resources.

The Intelligent Pairing Assistant (IPA) concept is to provide a small number of easy-to-understand pairing recommendations at specific decision points. Recommendations are created using encoded expert knowledge that analyzes relevant information available in AOC databases. These recommendations are presented to the user within the framework of the existing Master Air Attack Planning Toolkit (MAAPTK) software. First, MAAPTK is briefly described. Second, we describe the IPA user decision aid, which functions as a plugin to MAAPTK.

MAAPTK is a planning tool used in the AOC to develop missions for inclusion in an Air Tasking Order (ATO). The MAAP Toolkit application provides near real-time battlespace information that enables planners to visualize and generate battle plans that are accurate and appropriate to developing situations. Planners can view key information on tables, timelines, maps, graphs, and grids so that they can quickly understand the essential situational elements and can create the appropriate missions and packages using simple drag-and-drop operations. MAAPTK allows planners to use wizard systems to walk through the planning process. The wizard systems help focus the user on particular parts of the user interface for different planning decisions. However, there is still a wide array of possibilities within each wizard page.

IPA makes use of data already contained in the existing planning system to make recommendations to planners in the context of a particular wizard page. That is, IPA is a decision aid that highlights preferred decisions and certain

types of optimizations. IPA assists planners in making their decisions by narrowing down the possibilities that must be considered in their standard workflows.

The primary contributions of this case study are applying artificial intelligence techniques to a novel domain and discussing the software evaluation efforts in moving from a prototype to a deployed system in this high-stakes domain.

The remainder of the introduction is devoted to describing related work that contributed to building this system. In the next section, we describe the IPA software and how it fits within MAAPTK. Following are sections that describe the methods and results from a usability evaluation with subject matter experts. This was the final in-house evaluation of the software prior to submitting IPA as part of an existing MAAPTK test event. Due to the difficulty, expense, and timing of AOC software updates, IPA could not be changed once submitted for testing with real users. Either it would work as submitted or it would be pulled from the release. The conclusion summarizes the outcome and outlines directions for future work.

Related Work

The work described in this paper relies heavily on existing research. This includes representing expert knowledge (Buchanan & Shortcliffe, 1984), interactive/adaptive recommendation systems (Adomavicius, 2005; Gervasio et al., 2005; Langley, 1999), and research in the area of building trust in recommendation systems so that planners can understand why recommendations were made (Ehrlich et al., 2011; Glass et al., 2008; Gregor & Benbasat, 1999; Pu & Chen, 2006).

There is also a large body of related work that extends well beyond what was carried out in IPA. The DARPA COORDINATORS project includes teams of agents that work together to execute portions of a preexisting global schedule (Smith et al., 2007). This work is related in that it could be used to flexibly execute the ATO that was developed with the assistance of IPA in MAAPTK. More closely related, DARPA's Personal Assistant that Learns (pal.sri.com) has been used to learn workflow decision aids for the US Army that are similar to those manually created for IPA (Myers et al., 2011). The main differences between the two approaches involve the types of tasks automated, the personnel responsible for creating and maintaining the automated tasks, and the training requirements to use the system. The authors describe the advantages of automated tasks that apply to both approaches: reduced stress, ability to manage more tasks, and ability to consider more options, all of which result in better decisions.

IPA Software

The first goal of the Intelligent Pairing Assistant is to help planners make more efficient and effective use of resources. In the specific case discussed in this paper, IPA is intended to help users pair collection requests with existing missions in the MAAPTK NTISR Mission Planning dialog (Figure 1). NTISR stands for Non-Traditional Intelligence, Surveillance, and Reconnaissance. The basic concept for this dialog is to allow the user to assign information-gathering tasks to existing missions that are already planned. IPA first sorts the existing missions to show the most likely candidates for pairing and then provides additional information to help the planner choose among these missions.

The second goal of IPA is to require as little additional training as possible in order to make use of the results. Planners will be trained in the MAAPTK NTISR Mission Planning dialog as they were previously—with little additional training time devoted to the specific columns generated by IPA. This means that the information gathered by IPA must be easily understood by users with a variety of roles in the AOC in order for it to have an

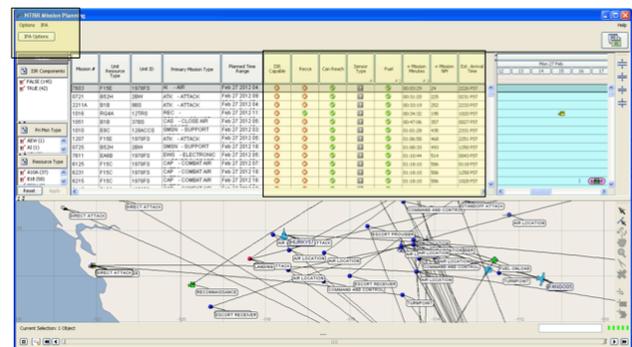


Figure 1. The MAAPTK NTISR Mission Planning dialog with a table, timeline, and map to describe the existing missions. IPA additions to the table and toolbar are highlighted in yellow.

impact on decision making.

IPA is implemented as an OSGI plug-in (<http://www.osgi.org>). This flexibility allows the IPA columns to easily be visible to involved developers without pushing any untested code to core MAAPTK developers. It also allows for IPA to be easily removed from MAAPTK.

While the prototype design and implementation (Ludwig & Geiselman, 2012) made extensive use of Subject Matter Experts (SMEs), drastic changes were still made as the project moved from prototype to deployment. Below we describe the user interface and provide implementation details of the evaluated system.

User Interface

The IPA user interface for NTISR Mission Planning consists of a number of additional columns added to an existing table that describes all of the available missions. A close-up of the columns is shown in Figure 2. Each column consists of a header, a header tooltip that explains the column, the values for each mission in each column, and a corresponding tooltip that explains how the decision was reached. The tooltips are designed to present the right amount of detail on how the decision was made in order to foster understanding and trust of the results. When appropriate, icons provide a quick summary of the details, where the various icons indicate:  a likely match,  not a likely match,  unknown (e.g., missing data), and  a warning. Empty cells indicate that there was no information available. For example, if a mission cannot reach a collection, then there will not be information available on estimated arrival time. Each column and tooltip is briefly described based on the result given for each mission.

ISR Capable. At least one of these is true for the Mission: contains ISR Component, is NTISR capable, or is type Reconnaissance.

Recce. (Reconnaissance) The mission is primarily an ISR mission; that is, the type belongs to the set of roughly 25 ISR mission types.

Can Reach. A match is shown if the mission can reach the collection during the time window of the collection request. This is determined by examining all of the pairs of route points, e.g., flying from A → B and analyzing at what time the mission would reach the collection point if the mission instead flew from A → Collection Request → B. The can reach tooltip describes where the mission would leave from, and return to, the existing route.

Sensor Type. A match is shown if the aircraft in the mission contain the same type of sensor specified in the collection request. The value is listed as unknown if the sensor cannot be found in the mission aircraft. This is because it is equally likely that ‘no match’ indicates that the database is missing information or that the aircraft does not have the desired sensor. The tooltip lists the sensors found in the mission with the match (if any) highlighted.

Fuel. A match is shown if the mission should have enough fuel to fly to and from the collection, and ‘unknown’ the result if the mission requires refueling, due to the possible complications of refueling. The tooltip indicates how much fuel would be used without and with the collection task.

+ Mission Minutes. The value of this cell is the additional minutes that would be added to the mission were it necessary to go from the existing route to the collection point and then back to the existing route. Specifically, this is:

ISR Capable	Recce	Can Reach	Sensor Type	Fuel	+ Mission Minutes	+ Mission NM	Est. Arrival Time
					01:19:29	556	0913 PST
					01:36:19	674	0048 PST
					00:03:29	24	2226 PST
					00:31:20	225	0231 PST
					00:33:19	252	2233 PST
					00:34:32	195	1020 PST
					00:47:06	357	0027 PST
					00:50:39	354	0735 PST
					01:01:28	435	2331 PST

Figure 2. IPA columns in the MAAPTK NTISR Mission Planning dialog .

(existing route X → collection → existing route Y) →
(existing route X → existing route Y).

+ Mission NM. The value of this cell is the additional nautical miles (NM) that would be added to the mission to go from the existing route to the collection point and then back to the existing route.

Est. Arrival Time. The value of this cell is the estimated time at which the mission would arrive at the collection request. The tooltip indicates what the request window of the collection was and when in the mission the collection task would be added (e.g., After Primary Tasks).

User Interface Options

One of the interesting aspects of the design of a decision aid in this domain is that planning is very much an art form. We worked with numerous subject matter experts, starting with the prototype design and through the current implementation. Some of the SMEs followed the whole project, while other were only included in the prototype or deployed system development. SMEs had diverse backgrounds, including operational and instructional AOC experience. While there was general agreement on what data IPA should present, every individual—based on their way of looking at the problem—had different ideas on how they would use this information. This resulted in a number of options that allow the user flexibility in how information is displayed and calculated.

First, users generally expect the best results to be at the top of the table, but which attributes constitute the best results differ among users. To support this, IPA presents a default three-level sort, first based on Sensor Type, second on Fuel, and third on + Mission Minutes. This was a compromise hammered out with a number of SMEs after discussing the prototype effort. However, the table supports sortable headers so users can create their own single- or multi-level sorts. This combination satisfied all of the SMEs.

Second, users had different opinions on when in the mission they would consider adding the collection task. Most SMEs only wanted to add collections after the primary task, so this was set as the default. However, an option dialog reached via a toolbar button allows the user

to select ‘Before,’ ‘After,’ or ‘Before or After’ the primary tasks as well as anytime during the mission.

Third, some users said they would add collection tasks to existing recce missions and some said they would not. By default, IPA considers recce missions, but this option can be turned off by the user.

Implementation

The reasoning engine is the system that is responsible for filling in the IPA columns by applying a model of expert knowledge to a data model that contains all of the required details of the collection request and of the available mission. For this project, we decided to use the representation of production (IF-THEN) rules to meet the project requirements. These rules were implemented directly in Java code, based on the specifications developed by SMEs with different points of view. That is, we solve the underlying problem of generating recommendations by automating common strategies already used by the planners. The decision to take this approach in IPA was motivated by the focus on ease of use and trust and because the identified strategies are unlikely to change over time.

In some cases, these rules are extremely simple. For example, in the *Recce* column, if the mission type is an element of the set of ISR missions, then a match is shown. In other cases, the rules perform data analysis that would be relatively time-consuming for a human user. For example, ascertaining whether a mission can reach the collection involves first determining which part of the mission route is acceptable. Then for each leg, the system calculates the time at which the mission could arrive at the collection point, using a speed customized for the mission. If the estimated arrival time is within the collection window, then a match is shown. This is a simplified version. The actual calculation, with all of the caveats, is quite complex. The *Can Reach* calculations drive several other columns such as *added Mission Minutes*, *added Mission NM*, and *Est. Arrival Time*. *Fuel* is another example of a complicated calculation; determining fuel use is a surprisingly difficult problem.

We determined that production rules were a reasonable representation given that most of the calculations performed will not change over time. That is, rules pull parameters from MAAPTK such as speeds, sensors, and locations to calculate values such as distances and arrival times. While the parameters might change with aircraft, collection types, or theatres, the underlying rule about distances or arrival time will not change. In instances where the model may change, such as fuel burn models, we rely on MAAPTK libraries to perform the underlying calculations. Whenever MAAPTK is updated, IPA will reflect the latest model. Because of this, we expect that

rules in IPA will not need to be updated over time. Additionally, what the rules are doing is performing a large number of simple calculations combined with a significant amount of searching. These calculations do not contain uncertainty or other considerations that would make it easier to use a different representation than rules.

User Evaluation Methods

An evaluation of both the functionality and utility of the IPA system and its integration with MAAPTK was performed at the development halfway mark. The objective of this evaluation was to generate feedback to perform course corrections prior to final evaluation with actual users (itself a small part of a much larger MAAPTK upgrade test event). We describe the specific objectives of the evaluation, the participants, and the general process followed during the evaluation.

Objective

The evaluation reviewed the IPA plug-in in the area of NTISR pairing for collections. The objective of this evaluation was to answer the following questions about the user interface and functionality of IPA:

- After initial training and a minimal amount of hands-on use, do users understand the information being provided by the IPA columns?
- What information and format would users prefer for each column?
- What other information would the users find useful?
- Are the information content and format useful in supporting users for more efficient and accurate pairing?

Participants

Participants in the evaluation consisted of three SMEs employed at Intelligent Software Solutions (ISS). The participants had expertise developing plans for targeting and/or collection in the MAAPTK software developed by ISS. Two participants had previously participated in the evaluation of an IPA prototype. One subject had no prior knowledge of IPA. Participants read and signed an informed consent form outlining the purpose and nature of the study as well as their rights as participants, including the right to not participate with no consequences. It is generally preferable to evaluate with actual users, as opposed to SMEs—or at least with SMEs not previously associated with the project. For a variety of reasons, our options were limited for this evaluation.

Process

The following was the planned process of this informal UI evaluation. The process was closely followed for the first SME to participate in the evaluation—the individual who was unfamiliar with IPA. Due to time constraints, the remote location of one participant, and the nature of participant responses, the process was followed less closely for subsequent participants, and ultimately there was a group discussion of the general approach for this function.

- Participants were shown a series of PowerPoint training slides describing and explaining each of the columns and Graphical User Interface (GUI) features of the NTISR feature of IPA. Participants were encouraged to ask questions.
- Participants were next asked to perform NTISR mission-pairing exercises and comment on their experiences as they went through the exercises.
- Participants were asked subsequently to describe what information they thought each of the columns and features of the NTISR mission pairing was providing and what it meant to them.
- Participants were finally asked about their thoughts regarding the utility of the IPA NTISR functionality and UI and how it might be improved.

Results

The user evaluation resulted in eight major feedback items.

1. **Reduce Processing Time:** The perceived amount of time required to process all of the missions for any particular collection request was perceived as being too long. Users stated that they would like to be able to see the results as soon as the window is available.
2. **Realistic Data Set:** Inconsistencies and errors in the dataset prevented calculations from working correctly and did not highlight the utility of the IPA decision aid. Users recommended that a more realistic targeting plan and corresponding collection requests be used for any future demonstration.
3. **Move IPA Decision Aid to the Collection Pairing Manager:** The Collection Pairing Manager was seen as the preferred location for the information provided by IPA (Figure 3). It has a well-developed user interface and numerous features that make it more useful for actually performing NTISR pairing than the NTISR Mission Planning dialog. For example, the Collection Pairing Manager already displays relevant information about each collection and allows planners to quickly go through multiple collection requests. This dialog also contains a toolbar button to actually assign the mission to the collection.
4. **Refactor IPA Columns:** Participants requested we remove redundant columns and make better use of tooltips to reduce the overall number of columns.
5. **ISR Capable, Recce:** Participants suggested removing these columns as they can be duplicated using the filter functionality available in MAAPTK for the mission table.
6. **Can Reach:** It was recommended that we update the tooltip to include information that is currently presented in Estimated Arrival time and Request Window and that we add information on when to add into the mission (e.g., After Primary Tasks).
7. **Sensor Type:** It was suggested we remove this column, as SMEs assume that every aircraft has NTISR capabilities and they were not concerned with matching sensors at this point in the planning process. Additionally, sensor-related data is not always reliably entered in AOC databases.
8. **Fuel:** SMEs recommended that we re-verify calculations on a more realistic and complete data set and improve the calculations so they work better when refueling is required. It was also suggested that we update the tooltip to only show remaining fuel after collection.
9. **+ Mission Minutes:** SMEs advised that the tooltip be updated to include original mission length and NM. It was also suggested to include the information currently shown in the + NM column. The reason for this: it is important for planners to factor in the duration of the existing mission and of the new task before adding the task.
10. **+ Mission NM, Est. Arrival Time:** Users advised us to remove these columns.

Acknowledgements

This material is based upon work supported by the United States Air Force Research Laboratory under Contract No. FA8650-11-C-6220. The views, opinions, and/or findings contained in this article/presentation are those of the author/presenter and should not be interpreted as representing the official views or policies, either expressed or implied, of the AFRL.

Distribution A: Cleared for public release; distribution unlimited. 88ABW. Cleared 1/25/2013; 88ABW-2013-0361.

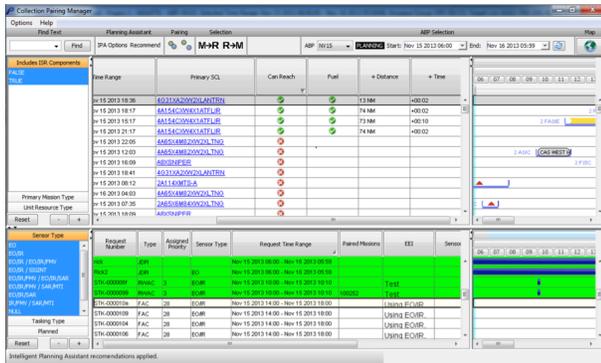


Figure 3. IPA deployed as part of the Collection Pairing Manager in MAAPTK. The Can Reach, Fuel, +Distance, and +Time calculations help the planner more quickly pair tasks with missions.

Conclusion

The results generated from the user evaluation were surprising in a number of ways. First, users were not willing to overlook issues that were not of concern for the prototype, such as the speed of processing and quality of the dataset. Second, the request to move IPA from the NTISIR Pairing Manager to the Collection Pairing Manager was unexpected given the prior focus on the former dialog. We believe this was in part due to recent experiences that the SMEs had had in operational environments with the latest version of MAAPTK, wherein they refreshed their AOC workflows. Third, we thought we had ironed out the display columns through multiple iterations with many SMEs on the prototype, but there are always more changes involved once they are able to see and critique an interface. Finally, we were reminded to adhere to one of the design principles that was of primary importance in the prototype when the SMEs requested that we show only the n -best recommendations.

Following the evaluation, all of the suggested UI and functionality improvements were made to IPA and it was submitted for formal testing. IPA received high marks during its evaluation as part of Ops RECCE evaluation events conducted using MAAPTK at Langley AFB. Although it was a small part of the overall test, IPA was specifically called out as a capability enabler at the event hot wash. Planners noted that IPA did exactly what it was intended to do by reducing the time associated with identifying Ops Recce candidates to support mission pairing. Due to its success, IPA will be deployed as part of the upcoming MAAPTK 2.1.2 software release. IPA illustrates a solid first step in bringing automated decision aids into the AOC as part of MAAPTK.

References

- Adomavicius, G. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17, 6, 734–749.
- Buchanan, B. and Shortcliffe, E. (1984). *Rule-Based Expert Systems: The Mycin Experiments of the Stanford Heuristic Programming Project*. Addison Wesley Publishing Company, Reading, MA.
- Ehrlich, K., Kirk, S.E., Patterson, J., Rasmussen, J.C., Ross, S.I., & Gruen, D.M. (2011). Taking advice from intelligent systems: the double-edged sword of explanations. In *Proceedings of IUI '11*, 125-134. ACM, New York, NY.
- Gervasio, M.T., Moffitt, M.D., Pollak, M.E., Taylor, J.M., & Uribe, T.E. (2005). Active preference learning for personalized calendar scheduling assistance, in *Proceedings of IUI '05*, ACM, New York, NY.
- Glass, A., McGuinness, D.L., & Wolverton, M. (2008). Toward establishing trust in adaptive agents. In *Proceedings of IUI '08*, 227-236. ACM, New York, NY.
- Gregor, S. and Benbasat, I. (1999). Explanations From Intelligent Systems: Theoretical Foundations and Implications for Practice. *MIS Quarterly* 23, 4 (December 1999), 497-530.
- Langley, P. (1999) User modeling in adaptive interfaces. In *Proceedings of UM '99*, 357–370.
- Ludwig, J. and Geiselman, E. (2012). Intelligent pairing assistant for air operations centers. In *Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces*, 241-244. ACM, New York, NY.
- Myers, K. Kolojechick, J., Angiolillo, C. Cummings, T. Garvey, T. Gervasio, M. Haines, W. Jones, C. Knittel, J. Morley, D. Ommert, W. and Potter, S. (2011) Learning by Demonstration Technology for Military Planning and Decision Making: A Deployment Story. In *Proceedings of the Conference on Innovative Applications of Artificial Intelligence (IAAI-11)*.
- Pu, P. and Chen, L. (2006). Trust building with explanation interfaces. In *Proceedings of IUI '06*, 93-100. ACM, New York, NY.
- Stephen F. Smith, Anthony Gallagher, and Terry Zimmerman. 2007. Distributed management of flexible times schedules. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems (AAMAS '07)*. ACM, New York, NY, USA.