United States Military Academy

West Point, New York 10996

On Assigning Long-Endurance Unmanned Aircraft Systems to Theater Combatant Commands

Roger Chapman Burk, Ph.D.
Associate Professor, Department of Systems Engineering

Approved by
Colonel Timothy E. Trainor, Ph.D.
Professor and Head, Department of Systems Engineering

February 2010

On Assigning Long-Endurance Unmanned Aircraft Systems to Theater Combatant Commands

Roger Chapman Burk, Ph.D.

Associate Professor, Department of Systems Engineering

OPERATIONS RESEARCH CENTER OF EXCELLENCE
TECHNICAL REPORT DSE-TR-0914
DTIC #: ADA513753

Approved by

Colonel Timothy E. Trainor, Ph.D.

Professor and Head, Department of Systems Engineering

February 2010

Abstract

We describe, formulate, and propose a heuristic algorithm for Strategic Command's problem of assigning long-endurance unmanned aircraft systems to theater combatant commands (COCOMs). We identify the problem's important characteristics, which include the relative priorities of the different COCOMs, the relative priorities of the different requirements in each COCOM, the different amounts of aircraft time required to work on the different requirements, and the probabilistic results of pursuing a given task. We formulate the problem as a binary nonlinear program (NLP) with a polynomial objective function and linear constraints. We identify the data required to define an instance of the problem. We discuss different approaches to finding solutions to the NLP and recommend a greedy heuristic algorithm, which is given in detail.

About the Author

Dr. Roger Chapman Burk is an associate professor in the Department of Systems Engineering, United States Military Academy, where he has been since 2000. Previously, he was a senior project engineer at The Aerospace Corporation (1998-2000) and a senior scientist at SAIC (1995-98). He is a retired US Air Force officer, with tours in spacecraft mission control (1979-82), spacecraft engineering (1982-84), mission analysis for the National Reconnaissance Office (1986-90), and on the faculty of the Air Force Institute of Technology (AFIT) (1993-95). His academic degrees include an AB from St. John's College in Annapolis, MD, an MS in Space Operations from AFIT (1985) and a PhD in Operations Research from the University of North Carolina at Chapel Hill (1993). He has published technical papers in *Interfaces*, *Military Operations Research*, *Journal of Multi-Criteria Decision Analysis*, and other scholarly journals. His research interests include decision analysis and applications of mathematical modeling to spacecraft and to unmanned aircraft.

Acknowledgements

Table of Contents

List of Tables

Chapter 1: The ISR Asset Assignment Problem

1.1   Problem Statement

United States Strategic Command (USSTRATCOM) presented this problem in the following words:

> In the Joint Enabler mission area ISR Operations, analyze and make recommendations for the use and employment of unmanned aerial systems (UASs) based on COCOM mission requirements.

USSTRATCOM (also called STRATCOM) has many mission areas, some of which are "Joint Enablers," meaning that they comprise missions to support other combatant commands (COCOMs) with operational assets that are managed globally. (Examples of COCOMs include Central Command (CENTCOM) with responsibility for the Middle East and central and southern Asia, Pacific Command (PACOM) with responsibility for the Pacific and eastern Asia, and Southern Command (SOUTHCOM) with responsibility for Latin America.)   One of STRATCOM's mission areas is Intelligence, Surveillance, and Reconnaissance (ISR) Operations, and one of STRATCOM's tasks within that mission area is the allocation of long-endurance UASs to COCOMs.   These UASs include medium- and high-altitude systems with mission endurance of a day or more; examples include the MQ-1 Predator, RQ-4 Global Hawk, and MQ-9 Reaper. Flight operations for these aircraft are based in the US in some cases and overseas in others, and they can operate worldwide, but on a given day they generally operate in one COCOM's area of responsibility and pursue requirements set by that COCOM.   It is STRATCOM's job to decide which COCOM these ISR assets will support.

1.2   Scope

In discussions with the client we confirmed the following limits of the scope of the problem to address:

Our scope is restricted to the use and employment of existing systems. We will not consider the acquisition of new systems.

Our scope is restricted to UASs. There are other aerial ISR assets that are or could be managed at the USSTRATCOM level, including the Army's Guardrail, the Navy's EP-3, and the Air Force's U-2 and RC-135. However, we will only be concerned with the unmanned systems.

We need only consider the ISR requirements that are identified at the COCOM level. ISR requirements can also be developed at the national level, and some of these requirements might be assigned to long-endurance UASs, but we will not modeel this possibility. Also, some requirements for long-endurance UASs might be developed at the unit level, as in an Army or Marine brigade or a Navy task force. We will assume that such unit requirements will be included in the COCOM requirements when necessary.

Our problem is restricted to the assignment of ISR assets to COCOMs. Once such an asset is assigned, it is the responsibility of the COCOM's Joint Forces Air Component Commander (JFACC) to decide how to employ it. This would include target assignment, sortie scheduling, and route planning. This JFACC problem is outside our scope. Also, COCOM intelligence officers have to decide which of their ISR needs to submit to STRATCOM as requirements for long-endurance UASs and which to meet with already-assigned COCOM assets or to submit to national systems. This problem is also outside our scope.

1.3   Context.

Table 1 shows the high-level context of this problem. There are rows for the different organizational categories of ISR collection systems as they relate to the ISR asset assignment problem. At the top are national systems, such as the National Reconnaissance Office, National Security Agency, and the Defense Attaché System. The next group includes aircraft systems we call "deployable," meaning that they are deployed individually to different COCOMs rather than being organic to tactical or operational units that might be so deployed. The last group comprises theater assets that are organic to units in the COCOM, first the aircraft systems and then all other ISR systems. The columns in Table 1 are for four different types of ISR platforms as they relate to this problem. The first is for multipurpose tactical aircraft, which can do ISR missions as well as other missions. The final three columns are for more or less dedicated ISR assets. (Some UASs, including Predator and Reaper, have a strike capability, but we are treating this as armed reconnaissance and so part of ISR.) Two columns are for ISR aircraft, first unmanned and then manned. The final column is for the great variety of other ISR systems: human intelligence (HUMINT), open-source intelligence, and technical collection carried out from land, from the sea, or from space.

ISR Systems

Multi-purpose Tacair | UASs | ISR Aircraft | Non-aerial ISR — HUMINT; Open-Source; Land/Sea/Space-based Technical Collection

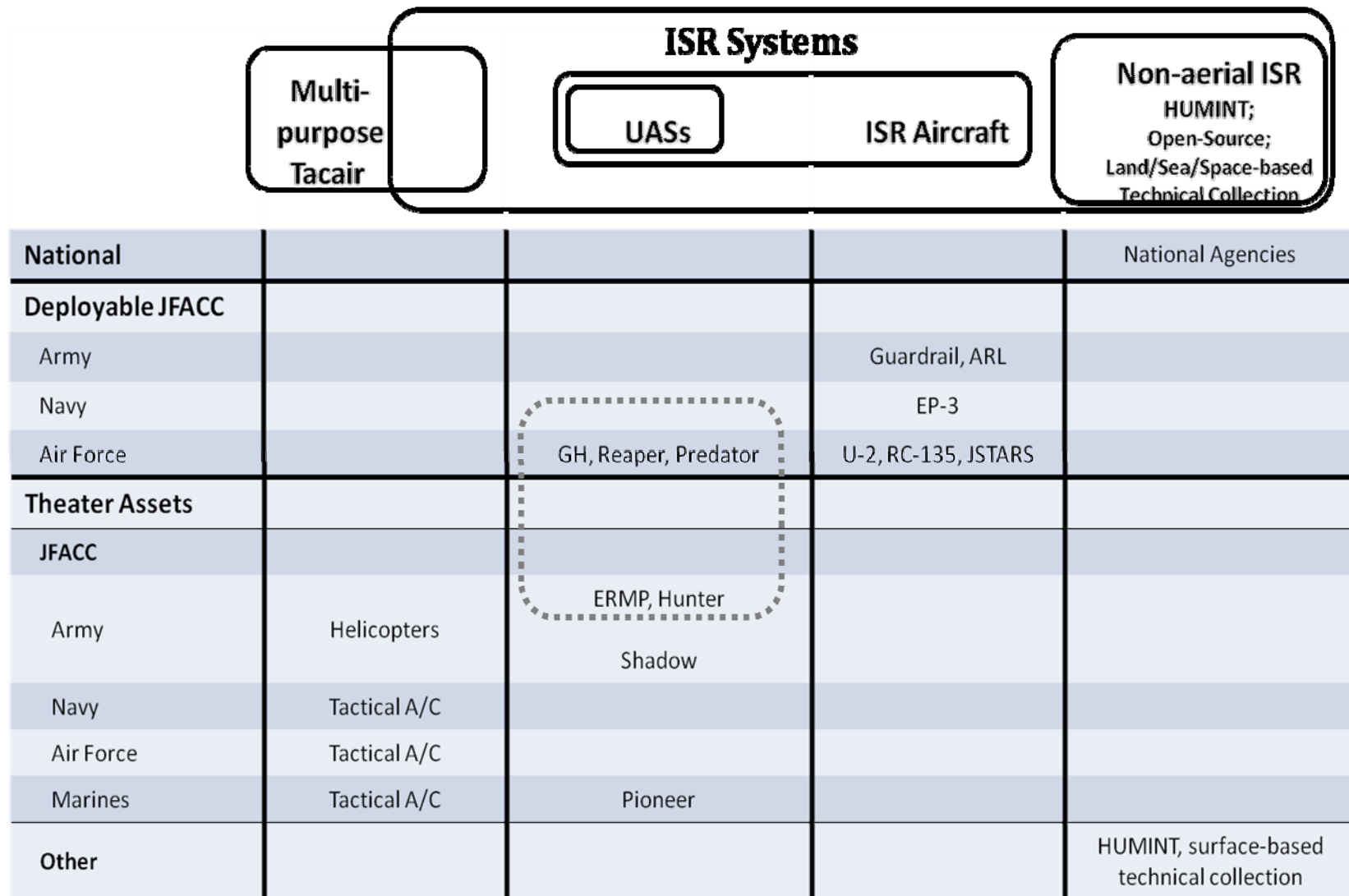|	Multi-purpose Tacair	UASs	ISR Aircraft	Non-aerial ISR
National				National Agencies
Deployable JFACC				
Army			Guardrail, ARL	
Navy			EP-3	
Air Force		GH, Reaper, Predator	U-2, RC-135, JSTARS	
Theater Assets				
JFACC				
Army	Helicopters	ERMP, Hunter / Shadow		
Navy	Tactical A/C			
Air Force	Tactical A/C			
Marines	Tactical A/C	Pioneer		
Other				HUMINT, surface-based technical collection

Table 1. Context of the ISR Asset Assignment Problem

The scope of the ISR asset assignment problem is shown by the dotted line. To some extent this definition of scope is tentative. Global Hawk (GH), Reaper, and Predator are certainly within it. However, the Army regards its new Extended Range Multi-Purpose (ERMP) UAS and its older Hunter UAS as organic at the corps or division level, so they might or might not be available for redeployment under STRATCOM direction. The lower-altitude tactical UASs Shadow and Pioneer are shown outside the scope, but they could be drawn within it if they were managed as independent swing units to be moved from theater to theater as need arose. For that matter, the manned ISR aircraft like the Army's Airborne Reconnaissance Low (ARL) and the Air Force's Joint Surveillance and Target Attack System (JSTARS) could be drawn within the scope of the ISR asset assignment problem without affecting the problem's structure. The exact scope of which systems are included does not affect the nature of the problem.

The primary lesson to be drawn from Table 1 is that the ISR asset assignment problem exists in a context of many different ISR requirements that can be addressed by a wide variety of different systems managed at different levels. The deployable long-endurance unmanned ISR systems are just part of the problem, if a very important part.

1.4   Problem Characteristics.

This section highlights characteristics of the ISR asset assignment problem that give it its natural form. The basic structure of the problem is a many-to-many assignment problem of ISR assets and COCOMs. The problem is complicated by the fact that the desirability of assigning a given asset to a given COCOM is mediated by the individual ISR requirements against which that asset would fly in that COCOM. Also, the benefit of a given assignment is probabilistic, since assigning the asset to a requirement does not necessarily mean that the intelligence need behind the requirement will be satisfied. In addition, we note that many of the important parameters of this problem, such as the probabilities, can only be roughly estimated, and that other parameters, such as priorities, are likely to change from month to month or even day to day as operational conditions change.

Each COCOM has a set of ISR requirements that it would like to address with one or more deployable ISR assets. The practice of the ISR community is to assign priorities to the different requirements, so that all Priority 1 requirements are roughly equal in importance, and are all more important than all Priority 2 requirements. The Priority 2 requirements in turn are

more important than the Priority 3 requirements, and so on. These priorities provide rank order only, not a ratio or interval scale of importance. Similarly, the COCOMs have their rank-order priorities: there can be a Priority 1 COCOM, a Priority 2 COCOM, etc.

In general, an ISR asset can fly against requirements in only one COCOM, because of their geographical separation. But even if an asset flies against a requirement, it may be that no benefit is realized, i.e. the intelligence need underlying the requirement may not be satisfied. Some assets may be unsuited to some requirements, as an imagery platform would be unsuited to an electronic intelligence requirement, so the probability of satisfying the need would be zero. But even if the asset is suited to the requirement, the result may still be probabilistic. The requirement may be to find some deployed weapon, and the asset may be the one best suited to find it, but the search might still fail. On the other hand, there could be a significant probability of some requirements being met by a means other than the assigned UAS, e.g. if the location of the deployed weapon were found by a national asset such as a reconnaissance satellite, or human intelligence, or a tactical reconnaissance asset like a short-range UAS. The ISR asset allocation problem has a fundamentally probabilistic nature. A procedure for solving it needs to take this into account.

If probabilities are important to the problem, they will most likely have to be more or less roughly estimated by subject matter experts. For instance, the probability of finding something that is trying to hide depends on the tactical situation, weather, and adversary courses of action. There is little hope of describing these factors completely. If we are going to address the problem quantitatively, we will have to be satisfied with more or less rough estimates of such probabilities. Also, any estimate of the amount of time it will take an asset to work on a given requirement will be subject to revision based on actual experience. Thus, once an asset is assigned to one COCOM's JFACC, its pattern of use is likely to change as necessary based on changing circumstances. This unavoidable imperfection in the precision of some of the key parameters of the problem means that it is probably futile to attempt to solve it to mathematical optimality.

Finally, we should note that the problem parameters are likely to change, but not too fast. Since at least some of the deployment decisions will take days or weeks to execute (e.g. moving Predator flight operations from one theater to another), we should not expect the user to search for new assignments on an hourly or daily basis. On the other hand, since operational conditions

and needs will change as time progresses, we should not expect the problem to be solved once for all time. This has implications for the kind of solution procedure we should look for.

1.5 Initial Assumptions.

In order to formalize this problem, we need to make some assumptions about its structure. We propose the following, which seem to match the situation reasonably well:

One-to-Many Assignment of COCOMs to Assets. We will assume that an ISR asset can be assigned to only one COCOM, but a COCOM can be assigned any number of assets.

Many-to-Many Assignment of Requirements to Assets. Within a COCOM, an ISR asset can be assigned any number of requirements in that COCOM, and a requirement can be assigned to any number of assets.

One Requirement at a Time. We will assume that an ISR asset can be engaged in satisfying only one requirement at a time, and we will neglect the possibility that an asset can satisfy two requirements at once, as by satisfying them both in a single image, or with simultaneous imagery and SIGINT collection. If an asset is assigned to more than one requirement, we will assume that the JFACC will be able to schedule different times for them, providing the total time needed for the assigned requirements does not exceed total time available for the asset.

Probabilistic Independence. We will assume that attempts to satisfy a given requirement by more than one asset are probabilistically independent. For example, if assets 1 and 2 are assigned to the same requirement and will satisfy the underlying intelligence need with probability p_1 and p_2 respectively, then the resulting probability of satisfying the need will be $1 - (1-p_1)(1-p_2)$. This is a modeling approximation and may be a rough one, but it seems impractical to estimate parameters to cover all the possible interactions.

No Redeployment Cost. We will assume that the time and effort required to redeploy an ISR asset from one COCOM to another is negligible. For some it might in fact be quite small. A Global Hawk can fly out of bases in the continental US, so it might cost little to fly it to one COCOM for one mission and to another for the next. However, Predators and Reapers typically fly out of airfields that are located in-theater, even if mission control is in the US, and redeployment would require a change of base. We will neglect this cost. In other words, we will

search for the best assignment of ISR assets to COCOMs for a given set of requirements, rather than the best sequence of assignments given a changing set of requirements.

No Equitability Requirement. There is no requirement to show equitability between COCOMs, beyond that implied by their relative priorities. If a COCOM is assigned no ISR assets, and that can be justified by its relative priority and/or the low probability of any assigned asset satisfying a high-priority intelligence need, then that is an acceptable result.

1.6 Organization.

We are now ready to propose a formalization of this problem. Chapter 2 provides a formalization as a binary nonlinear programming problem with a polynomial objective function and linear constraints. Chapter 3 discusses different approaches to solving this problem and recommends and presents a straightforward greedy heuristic algorithm. Details for the heuristic are given and some implicit assumptions listed. Chapter 4 concludes with some general remarks on the problem and its solution.

Chapter 2: Problem Formulation

2.1   Formalization.

      The basic problem is to decide which ISR assets to deploy to which theaters. In order to do that with the greatest overall benefit, we need to look at which requirements each asset will be tasked against. We formalize the problem as described in this section.

      We have n theater COCOMs, numbered in priority order from 1 to n (highest to lowest). We will assume that no two COCOMs are tied in priority.

      Each COCOM i has a number n_i of ISR requirements, which we will label R_{i1} through R_{in_i}. Each requirement R_{ij} has a priority $t_{ij} \in \{1,2,3,...\}$, with priority 1 being the highest. Requirements may tie in priority, even within a COCOM.

      We will number the deployable ISR assets with the numbers 1 though a. Each asset must be assigned to exactly one COCOM, but each COCOM may receive multiple assets. An asset can be assigned to a requirement R_{ij} only if it is assigned to COCOM i.

      Assigning an asset to a requirement does not necessarily guarantee any intelligence benefit. Instead, there is some probability of benefit (possibly 0 if the asset is unsuited to the requirement, possibly 1 in some circumstances). Thus, if ISR asset k is assigned to requirement R_{ij}, then there is a probability p_{ijk} of meeting the intelligence need underlying the requirement. We will assume that these probabilities are independent, and we will not model partial satisfaction of requirements.

      An asset can be assigned more than one requirement in a COCOM, and a requirement can have more than one asset assigned to it. However, there is a limit to the number of requirements that an asset can accept, because each requirement takes a certain amount of time and the asset only has a certain amount of time available. Furthermore, some requirements will take up more of an asset's time than others.

      We will take the following as the objective of this ISR assignment problem: maximize the total expected satisfaction of intelligence needs, considering the relative priorities of the COCOMs and the relative priorities of the requirements. (A method to relate these priorities will be developed in the following section.) This implies additivity of expected satisfaction. For instance, satisfying an intelligence need with certainty will have the same score as satisfying two

intelligence needs of the same priority in the same COCOM, each with probability 0.5. This is justified by the fairly large number of intelligence needs. The ultimate total proportion of intelligence needs satisfied will be close to the expected number divided by the total number, so it is appropriate to maximize the expectation without explicitly taking into account attitudes towards risk.

On the other hand, the size of the ISR assignment problem is not too large compared to the capabilities of modern software and computers. The number of COCOMs n, the number of requirements in each COCOM n_i, and the number of assets a are all in the tens or hundreds, not in the thousands or millions. The dimensionality of the problem will not be a bar to quick solution, though its combinatorial nature will require a heuristic method rather than a solution to formal optimality.

2.2 Required Data

Most of the data in the previous section is assumed to be given when the problem is posed: we assume that the ISR assets have been identified and the COCOMs and their associated requirements have been enumerated and their relative priorities set. This assumption is reasonable because these data are already in use for intelligence planning. However, the probabilities p_{ijk} are not immediately available. They will have to be found, derived, or estimated as well as possible in order to make a quantitative estimate of the expected benefit of assigning an asset to a requirement. This may not be an easy task, and it may require subjective probability estimates from intelligence experts. It will be hard to be certain that such estimates are correct. Nevertheless, the probabilistic nature of the satisfaction of intelligence requirements seems to be fundamental to this problem. We believe that it is better to make the best possible estimate of the probabilities than to forego any attempt at quantifying the problem.

We will also need the fraction f_{ijk} of asset k's capacity that would be used up if that asset is assigned to R_{ij} ($0 \leq f_{ijk} \leq 1$), for all i, j and k. For instance, if asset k's hours per day on station is H_k, and R_{ij} would require h_{ijk} hours per day if assigned to asset k, then we would calculate $f_{ijk} = h_{ijk} / H_k$. As stated above, we will assume that an asset can be engaged in satisfying only one requirement at a time.

We will need an estimate of the probability r_{ij} that the intelligence need underlying R_{ij} will be satisfied by some means other than a deployable ISR asset, for all i and j. These

probabilities are required to ensure an efficient overall allocation of assets. Without them, we might send deployable ISR assets against high-priority requirements that will probably be satisfied by other means anyway, rather than after other requirements that only deployable ISR assets can satisfy.

We will need a way to relate the value of satisfying requirements of different priority. If we regard the priorities as truly absolute, it could result in assigning many assets against a requirement that they have an extremely small probability of satisfying, when they could be assigned instead to satisfy many lower-priority (but still important) requirements with certainty. This would surely be a poor assignment of resources. We propose to relate priorities by eliciting or estimating a probability q_p such that satisfying a priority n requirement with probability q_p is equally preferred to satisfying a priority $n+1$ requirement with probability 1. Logically, there must be some finite value of q_p that satisfies this condition. If $q_p=1$, then clearly we would prefer a priority n requirement with probability q_p over a priority $n+1$ requirement with probability 1, since both are certain and by definition lower priority numbers are preferred. On the other hand, if $q_p=0$, we would prefer the priority $n+1$ requirement with probability 1 over the priority n requirement with probability q_p. For some $0 < q_p < 1$, equal preference must hold. (We are making an implicit assumption that each priority has the same relationship to the next lowest; see section 3.3 for a discussion.)

Similarly, we need a way to relate the value of satisfying requirements of different COCOMs other than the naïve method of regarding lower-numbered COCOMs as having absolute priority over higher-numbered. We propose to elicit or estimate a probability q_c such that satisfying a priority n requirement for COCOM m with probability q_c is equally preferred to satisfying a priority n requirement for COCOM $m+1$ with probability 1. Logically, there must be some finite value of q_c that satisfies this condition, by the same reasoning that was used to show the existence of q_p. (Again, see section 3.3 for a discussion of the implicit assumption.)

2.3 Decision Variables

This problem has a total of $a\left(n + \sum_{i=1}^{n} n_i\right)$ decision variables, all binary, where a is the number of deployable assets, n is the number of COCOMs, and n_i is the number of intelligence requirements in COCOM i:

$$
y_{ik} = \begin{cases} 1 & \text{if asset } k \text{ is assigned to COCOM } i \\ 0 & \text{otherwise} \end{cases}
$$

$$
x_{ijk} = \begin{cases} 1 & \text{if asset } k \text{ is assigned to requirement } R_{ij} \\ 0 & \text{otherwise} \end{cases} \tag{1}
$$

2.4 Objective Function

We build the function to be optimized as follows:

$p_{ijk}x_{ijk}$ is the probability that R_{ij} is satisfied by asset k, where p_{ijk} is the probability that k will satisfy the requirement given that it is assigned to it.

$1 - p_{ijk}x_{ijk}$ is the probability that R_{ij} is *not* satisfied by asset k.

The probability that R_{ij} is not satisfied by any of the deployable ISR assets under consideration, assuming probabilistic independence, is:

$$
\prod_{k=1}^{a} \left(1 - p_{ijk}x_{ijk} \right) \tag{2}
$$

The probability that R_{ij} is not satisfied at all, by a deployable ISR asset or any other means, where r_{ij} is its probability of being satisfied by some other means, is:

$$
(1 - r_{ij}) \prod_{k=1}^{a} \left(1 - p_{ijk}x_{ijk} \right) \tag{3}
$$

The probability that R_{ij} *is* satisfied by one means or another is:

$$
1 - (1 - r_{ij}) \prod_{k=1}^{a} \left(1 - p_{ijk}x_{ijk} \right) \tag{4}
$$

The relative benefit of the probability that R_{ij} will be satisfied, considering its COCOM i and its priority t_{ij} within that COCOM, should be decreased by a factor q_c for every unit by which i exceeds 1, and by a factor q_p for every unit by which t_{ij} exceeds 1. Thus, the relative benefit of this requirement's satisfaction is:

$$
q_c^{i-1} q_p^{t_{ij}-1} \left[1 - (1 - r_{ij}) \prod_{k=1}^{a} \left(1 - p_{ijk}x_{ijk} \right) \right] \tag{5}
$$

Finally, the total benefit of the probabilities of satisfying all collection requirements, and the objective function to be maximized by the deployable ISR assignment problem, is:

$$Z = \sum_{i=1}^{n} q_c^{i-1} \sum_{j=1}^{n_i} q_p^{t_{ij}-1} \left[1 - (1 - r_{ij}) \prod_{k=1}^{a} \left(1 - p_{ijk} x_{ijk} \right) \right] \tag{6}$$

This is the objective function for the ISR asset assignment problem. Note that it does not contain the variables y_{ik}. It is polynomial of order a in the decision variables x_{ijk}, but x_{ijk} does not multiply itself and multiplies $x_{i'j'k'}$ if and only if $i=i'$ and $j=j'$.

2.5 Constraints

As stated above, the decision variables x_{ijk} and y_{ik} are constrained to be binary. In addition, the optimization problem needs three other sets of constraints.

We need a constraint to ensure that each of the a assets is assigned to no more than one COCOM:

$$\sum_{i=1}^{n} y_{ik} \leq 1 \quad \text{for } k = 1 \ldots a \tag{7}$$

We need a constraint to ensure that each of the a assets is assigned to none of the n_i requirements in any COCOM i unless the asset is assigned to that COCOM:

$$x_{ijk} \leq y_{ik} \quad \text{for } i = 1 \ldots n, \ j = 1 \ldots n_i, \text{and } k = 1 \ldots a \tag{8}$$

Finally, we need a constraint to ensure that none of the a assets is over-tasked:

$$\sum_{i=1}^{n} \sum_{j=1}^{n_i} f_{ijk} x_{ijk} \leq 1 \quad \text{for } k = 1 \ldots a \tag{9}$$

where again f_{ijk} is the fraction of asset k's collection capability that will be used if that asset is assigned to requirement R_{ij}.

Note that constraints (7), (8), and (9) are all linear, so the region defined by them is a polytope, and the feasible region of the optimization problem is the binary-valued subset of this

polytope. Also, the right-hand side of each of these constraints is either 0 or 1. The total number of these constraints is $a\left(2 + \sum_{i=1}^{n} n_i\right)$.

With decision variables, objective function, and constraints all defined, the definition of the ISR asset assignment problem is complete.

Chapter 3: A Solution Heuristic

This chapter describes a simple heuristic that should be practical for finding good solutions quickly for this optimization problem. We do not recommend attempting to solve this problem to formal optimality for four reasons. First, many of the problem parameters, such as the probabilities p_{ijk}, cannot be known with precision, so the formal optimal solution is unlikely to be the "true" optimal anyway. Second, optimal solutions in a binary program like this tend to be fragile, in the sense that the optimal values of the decision variables x_{ijk} and y_{ik} could change drastically after only a small change in the problem parameters, perhaps with only a small improvement in the objective function value as well. Such a drastic change in recommended course of action (i.e., which assets to deploy to which COCOM) after a small change in the problem, with perhaps small improvement in performance as well, is hard to explain to stakeholders and tends to erode confidence in the optimization model. Third, the problem will have to be solved regularly as requirements and available assets change, with a frequency on the order of the inverse of the time it takes to redeploy an ISR asset from one theater to another, so the problem cannot be solved once for all time. A good answer produced quickly is likely to be more useful than the best possible answer after a long wait. Fourth and last, there are technical reasons for expecting the problem to be time-consuming to solve to optimality, as will be described in the following section.

3.1 Technical Difficulties in Solving to Optimality

Because of the binary restriction on x_{ijk} and the over-tasking constraints (9), the ISR asset assignment problem is a generalization of the knapsack problem, a classic problem in combinatorial optimization that is well-known to be NP-complete (Garey and Johnson, 1979, p. 65). This means that a polynomial-time solution method is not to be expected. Furthermore, even the real-valued relaxation of the problem is likely to be difficult to solve, despite the fact that its feasible region is a simple polytope. The reason is that the objective function (6) is not concave, as shown in the following proposition, so the relaxed ISR asset assignment problem is not a convex programming problem, and nonconvex programming methods would need to be used for it (Hillier and Lieberman, 2001, section 13.3). In nonconvex programs local maxima are not necessarily global maxima, so time-consuming search techniques are required.

Proposition. The objective function (6) for the ISR asset assignment problem is not a concave function over the feasible region of its real-valued relaxation (defined by (7), (8), (9), and non-negativity), provided that q_c and q_p are nonzero, and provided that there is at least one i, j, k, k' for which p_{ijk} and $p_{ijk'}$ are both nonzero and r_{ij} is not 1.

Proof. We will show this by showing that $-Z$ is not convex over this region. A twice-differentiable function is convex over a region if and only if all the principle minors of its Hessian are non-negative (Winston and Venkataramanan, 2003, p. 677). Taking partial derivatives of $-Z$,

$$\frac{\partial(-Z)}{\partial x_{IJK}} = q_c^{I-1} q_p^{t_{IJ}-1} p_{IJK}(1-r_{IJ})\prod_{k\neq K}(1-p_{IJk}x_{IJk})$$

$$\frac{\partial^2(-Z)}{\partial x_{IJK}^2} = 0 \tag{10}$$

$$\frac{\partial^2(-Z)}{\partial x_{IJK}\partial x_{IJK'}} = -q_c^{I-1} q_p^{t_{IJ}-1} p_{IJK}p_{IJK'}(1-r_{IJ})\prod_{k\neq K,K'}(1-p_{IJk}x_{IJk})$$

Thus the diagonal elements of the Hessian of $-Z$ are zero. Furthermore, if we pick an x_{IJK} and $x_{IJK'}$ such that $r_{IJ}<1$ and p_{IJK} and $p_{IJK'}$ are both nonzero (with $K\neq K'$), the second partial derivative will be strictly negative over the interior of the given region. The corresponding second principle minor will be the determinant of a 2×2 matrix with zeros on the diagonal and strictly positive elements off the diagonal, so the minor will be strictly negative. So $-Z$ is not convex, and Z is not concave.

The objective function (6) does have the form of a generalized positive polynomial ("posynomial"), but posynomial geometric programming cannot be applied because it is to be maximized rather than minimized (Beightler and Philips, 1976, Chapter 3). In principle signomial geometric programming could be applied (ibid., Chapter 5), but we did not pursue this because of the great increase in complexity, the fact that the solution could not be guaranteed to be a global maximum, and the poor prospects for quickly solving the underlying binary program in any case.

Because of these difficulties, we do not recommend attempting solution to analytic optimality via branch and bound. Also, because of the time required and the small marginal benefit we do not recommend a zero-order search technique like a genetic algorithm, tabu search, or simulated annealing. Instead, we propose a straightforward greedy heuristic to find a good and acceptable solution to the ISR asset assignment problem, given in the following section.

3.2 Heuristics for the ISR Asset Assignment Problem

The following is the top-level heuristic. For each step, a narrative description is given in italics, followed by a mathematical formalization in Roman type.

ISR Asset Assignment Heuristic

1. *Start with no assets assigned to any COCOM or requirement. Consider asset 1.*

   Initialize: $y_{ij} := 0 \ \forall \ i, j; \ x_{ijk} := 0 \ \forall \ i, j, k; \ \gamma := 1$

2. *Try assigning the asset under consideration to each of the n COCOMs. In each case, use the ISR Requirement Assignment Heuristic given below to find an assignment of the asset under consideration to requirements in that COCOM that is feasible and gives a good improvement in Z, holding all other assignments unchanged. Record the n improved values of Z and the assignments that produced them.*

   For $i := 1,..n$, calculate Z using the ISR Requirement Assignment Heuristic and set $Z_{i\gamma}$ equal to that Z.

3. *Find the best of the n Z's. Assign the asset under consideration to the corresponding COCOM, and assign it to the requirements that produced that Z.*

   Let α be an index such that $Z_{\alpha\gamma} = \max_i Z_{i\gamma}$. Set $y_{\alpha\gamma} := 1$. For all j, set $x_{\alpha j\gamma}$ to the value given by the ISR Requirement Assignment Heuristic.

4. *If asset a is the one under consideration, the heuristic is finished. Otherwise, consider the next asset and go to step 2.*

   If $\gamma = a$, end. Otherwise set $\gamma := \gamma+1$ and go to step 2, with the new asset γ being the one under consideration.

This algorithm will loop *a* times, once for each asset. In each loop, step 2 requires *n* executions of the ISR Requirement Assignment Heuristic described in the next paragraph, one for each COCOM.

Step 2 of the heuristic includes a subproblem: given an asset γ, a COCOM α, and assignments of some (not necessarily proper) subset of the other *a*–1 assets to COCOMs and

requirements, find the feasible assignment of asset γ to requirements $R_{\alpha j}$ that gives the greatest increase in Z. We propose to address this subproblem using another "bang-for-buck" greedy algorithm. The algorithm starts with asset γ unassigned (i.e. $x_{\alpha j \gamma} = 0 \; \forall j$) but possibly some or all of the other assets assigned.

ISR Requirement Assignment Heuristic

1. *Start with none of asset γ's capacity used and Z calculated based on the given assignments of the other assets.*

   Initialize: Set $\xi_j = 0$ for $j = 1,..,n_\alpha$. Set $Z^* := Z$ as calculated with initial values of x_{ijk}. Set $f := 0$.

2. *For every requirement in COCOM α for which asset γ has enough unassigned capacity, calculate the objective function Z assuming that that one requirement is added to asset γ's assignments, without changing any other assignments from what they were when this step was entered.*

   For $j := 1,..n_\alpha$: if $f_{\alpha j \gamma} > 1-f$ or $\xi_j = 1$ set $Z_j := Z^*$; otherwise set $Z_j := Z$ recalculated with $x_{\alpha \beta \gamma} = \xi_\beta$ for $\beta := 1,..n_\alpha$, except that $x_{\alpha j \gamma} = 1$.

3. *If the remaining capacity of asset γ is insufficient for any of the unassigned requirements in this COCOM, the heuristic is finished. Otherwise, for each requirement find the ratio of improvement in Z to fraction of capacity used.*

   If $Z_j := Z^* \; \forall j$, end; the values of ξ_j represent the optimal allocation of asset γ against requirements in COCOM α and Z^* represents the corresponding objective function value. Otherwise, set $B_j := (Z_j - Z^*) / f_{\alpha j \gamma} \; \forall j$ and go to the next step.

4. *Find a requirement that has the largest ratio, i.e. the greatest "bang-for-buck."*

   Find an index β such that $B_\beta = \max_j B_j$.

5. *Assign that requirement to asset γ, and decrement the asset's capacity by the appropriate amount. Repeat from step 2.*

   Set $\xi_\beta := 1$. Set $f := f + f_{\alpha \beta \gamma}$. Go to step 2.

This algorithm will loop a maximum of n_α times, once for each requirement in COCOM α. Each loop requires the calculation of Z a maximum of n_α times in step 2. Thus, the total number of calculations in the ISR Asset Assignment Heuristic is $O(ann_i^2)$. It should finish quickly for realistic values of the parameters.

The resulting assignment of assets to requirements should be good, but of course it is not guaranteed to be mathematically optimal. Some elaborations are possible to increase the

probability of finding optimality, though it is questionable whether the improvement in actual assignments would be worth the increased complexity:

- Consider the assets in each of the $a!$ possible orders and pick the best result.
- Use the results of the heuristic to start an integer NLP solver.
- Use the results to seed a genetic algorithm or other zero-order searcher.

3.3 Additional Implicit Assumptions

Primarily for ease of exposition, we have implicitly made the following additional assumptions when constructing the heuristic. These assumptions can be relaxed at the cost of some small increase in complexity.

Relative Priorities Are Constant. As a modeling simplification, we assumed that q_p *and* q_c are constant. This implies that the same relationship of relative importance applies to all requirements across all levels of priority (priority 1 vs. priority 2, priority 2 vs. priority 3, etc.) and to all COCOMs. Also, this means that satisfying a priority n requirement with probability $q_p{}^2$ is equivalent to satisfying a priority $n+2$ requirement with certainty. This implies that the different priority COCOMs are related in the same way: other things being equal, a requirement from a priority 1 COCOM is worth q_c times one from a priority 2 COCOM, and one from a priority 2 COCOM is worth q_c times one from a priority 3 (and consequently, one from a priority 1 is worth $q_c{}^2$ times one form a priority 3).

COCOMs Cannot Tie. We assumed that no two COCOMs have the same priority.

No Aborted Sorties. We assumed that if a given asset is assigned to a given requirement then it will in fact carry out the mission despite weather, mechanical problems, or other chance events. These eventualities could be accounted for in a rough way by changing the right-hand side of the over-tasking constraints (9) to some value less than 1.

No Partial Satisfaction of Requirements. We model all requirements as being either completely satisfied or unsatisfied.

Chapter 4: Conclusion

We have formulated STRATCOM's problem of assigning long-endurance UASs to theater COCOMs as a binary nonlinear programming problem, defining the decision variables, objective function, and constraints. The formulation takes into account quantitatively the important characteristics of the problem, including the relative priorities of the different COCOMs and the different requirements in each COCOM, the different amounts of aircraft time required to work on the different requirements, and the probabilistic results of pursuing a given ISR requirement. Nevertheless, the NLP is simple enough and has enough structure to make it practical to find high-quality solutions relatively quickly. We identified the set of data required to fully define an instance of the problem and proposed a polynomially-bound heuristic algorithm that can be used to find good solutions. In implementing this algorithm, it will likely be more difficult to gather all the required input data than to execute the algorithm and find a good assignment. Nevertheless, the problem parameters as defined seem to be the minimum set necessary to capture the essential features of the ISR asset assignment problem.

References

Beightler, Charles, and Don T. Phillips. *Applied Geometric Programming.* New York: John Wiley & Sons, 1976.

Garey, Michaael R., and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* New York: W. H. Freeman and Company, 1979.

Hillier, Frederick S., and Gerald J. Lieberman. *Introduction to Operations Research.* 7th ed. Boston: McGraw-Hill, 2001.

Winston, Wayne L., and Munirpallan Venkataramanan. *Introduction to Mathematical Programming. Operations Research: Volume One.* 4th ed. Pacific Grove, CA: Brooks/Cole-Thompson Learning, 2003.

Appendix A: List of Abbreviations

A	
A/C	Aircraft
ARL	Airborne Reconnaissance Low
C	
COCOM	Combatant Command
D	
DTIC	Defense Technical Information Center
G	
GH	Global Hawk
H	
HUMINT	Human Intelligence
I	
ISR	Intelligence, Surveillance, and Reconnaissance
J	
JFACC	Joint Forces Air Component Commander
JSTARS	Joint Surveillance and Target Attack Radar System
N	
NLP	Non-Linear Program
O	
ORCEN	Operations Research Center
S	
SE	Systems Engineering
SIGINT	Signals Intelligence
T	
Tacair	Tactical Aircraft
U	
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
USMA	United States Military Academy

Distribution List

The list indicates the complete mailing address of the individuals and organizations receiving copies of the report and the number of copies received. Due to the Privacy Act, only use business addresses; no personal home addresses. Distribution lists provide a permanent record of initial distribution. The distribution information will include the following entries:

NAME/AGENCY	ADDRESS	COPIES
Roger Chapman Burk	Department of Systems Engineering Mahan Hall West Point, NY 10996	2
USSTRATCOM J811	ATTN: LTC Illingworth USSTRATCOM, 901 SAC BLVD STE 1A1, Offutt AFB, NE 68113-6020	1
Dean, USMA	Office of the Dean Building 600 West Point, NY 10996	1
Defense Technical Information Center (DTIC)	ATTN: DTIC-O Defense Technical Information Center 8725 John J. Kingman Rd, Suite 0944 Fort Belvoir, VA 22060-6218	1
Department Head-DSE	Department of Systems Engineering Mahan Hall West Point, NY 10996	1
ORCEN	Department of Systems Engineering Mahan Hall West Point, NY 10996	5
ORCEN Director	Department of Systems Engineering Mahan Hall West Point, NY 10996	1
USMA Library	USMA Library Bldg 757 West Point, NY 10996	1

REPORT DOCUMENTATION PAGE – SF298

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
31-07-2009	Technical Report	1 Oct 08 – 31 May 09

4. TITLE AND SUBTITLE

On Assigning Long-Endurance Unmanned Aircraft Systems to Theater Combatant Commands

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Roger Chapman Burk

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Operations Research Center
Department of Systems Engineering
US Military Academy
West Point, NY 10996

8. PERFORMING ORGANIZATION REPORT NUMBER

DSE-TR-0914

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

STRATCOM
901 SAC BLVD STE 1A1
Offutt AFB, NE 68113-6020

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Distribution A: Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

We describe, formulate, and propose a heuristic algorithm for Strategic Command's problem of assigning long-endurance unmanned aircraft systems to theater combatant commands (COCOMs). We identify the problem's important characteristics, including the relative priorities of the different COCOMs and the different requirements in each COCOM, the different amounts of aircraft time required to work on the different requirements, and the probabilistic results of pursuing a given task. We formulate the problem as a binary nonlinear program (NLP) with a polynomial objective function and linear constraints. We identify the data required to define an instance of the problem. We discuss different approaches to finding solutions to the NLP and recommend a greedy heuristic algorithm, which is given in detail.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF: UNCLASSIFIED			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Roger Chapman Burk
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	None	37	**19b. TELEPHONE NUMBER** *(include area code)* (845) 938-4754