JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 22, No. 1, January–February 1999

Optimization and Coordination of Multiagent Systems Using Principled Negotiation

John P. Wangermann* and Robert F. Stengel†
Princeton University, Princeton, New Jersey 08544

Principled negotiation coordinates the actions of agents with different interests, allowing distributed optimization. In principled negotiation, agents search for and propose options for mutual gain. If the other agents agree to the proposal, it is implemented. Under certain conditions, an agent can search for options for individual gain without impacting other agents. In these cases, the agent can negotiate with a coordinator, rather than obtain agreement from all other agents. The tenets of principled negotiation are outlined and stated mathematically. Two examples are formulated to test principled negotiation performance. The first has no coupling between the agent actions if coordination criteria are met. Principled negotiation allows the agents to achieve a solution as good as that achieved by a centralized controller with perfect knowledge. The second problem, based on the air traffic management problem of negotiating arrival slots, is highly coupled, constraining each agent's available set of actions. Principled negotiation allows agents to search options that would not be available otherwise, improving the utility function of all agents. Applied to air traffic operations, principled negotiation allows much greater freedom for optimization by system users while maintaining safety.

Introduction

THE ground-based air traffic control (ATC) system was created to ensure the safety of flights operating in controlled airspace. Aircraft are separated by a combination of procedures and tactical maneuvering instructions. As air traffic has grown, the ATC system has increasingly depended on computer systems. Computers now not only process radar and flight plan data, but also help controllers to manage flow, avert conflicts, and maneuver traffic.[1,2]

Today's ATC system has many problems that are characteristic of traditional control systems for large-scale industrial systems.[3] To manage growing air traffic, ATC computer systems are becoming more complex, increasing expense and making new systems more difficult to introduce. In addition, the aircraft/airspace system (AAS) is not responsive to the desires of users (aircraft and operators). The procedures of the ATC system prevent users from dynamically optimizing their operations and cause many hours of delays.

Distributed artificial intelligence deals with small, simple systems working cooperatively to better control large-scale systems. In a multiagent system (MAS), each agent has its own goals, and it must anticipate the actions of other agents and coordinate actions to meet these goals. The AAS is an MAS. Agents include aircraft, operators, and traffic management agents (TrMAs, a generic term for any air traffic control unit). Each agent makes decisions and takes actions that affect the air traffic process. Their actions interact because aircraft must stay safely separated. Until recently, only the ATC system had sufficient data (on traffic, flight plans, and the weather) and computing power to analyze the situation. Now, airlines and aircraft also have powerful computer systems, and they can access large amounts of data from their sensors and through high-bandwidth communications. They are also capable of making declarative decisions regarding the traffic situation.

Steeb et al.[4] studied whether aircraft alone could resolve conflicts. When a conflict arose, the affected aircraft used a variety of criteria to determine which aircraft was best suited to formulate a resolution plan. The chosen aircraft then calculated the plan and transmitted it to the other aircraft. This was a centralized control system, but the air traffic process was broken down into distributed conflict areas each with a controlling aircraft.

Davis and Smith[5] used a contract net approach to assign surveillance tasks for particular areas to individual aircraft. A manager divided the task and issued a request for bids. The agents then sent in bids, and the manager selected the successful bidders.

Levy and Rosenschein[6] distributed the coordination function using game theory. In the pursuit problem, each pursuer first evaluated the solution of the local game to calculate the total payoff received by all of the agents from their combined actions. Each agent then solved the global game to establish its share of the payoff for each of the combination of actions and chose its Nash equilibrium action.[7,8] This is the action that guarantees an agent the maximum minimum payoff whatever the actions of the other agents (i.e., the agent receives at least this amount). The game-theoretic approach gave each agent some ability to maximize payoff at each move, but the coordination (i.e., the ability of the pursuers to capture the evader) was sometimes compromised.

Proposed Approach

None of the existing coordination methods solve the problems of the AAS. Centralized coordination schemes do not improve the ability of each agent to optimize its own operations. The game-theoretic approach allows better optimization at the expense of poorer coordination. In an AAS, good coordination is critical to avoid accidents. A new method of coordination is required that ensures safety and provides agents with greater freedom to optimize.

The compatibility of the agents' actions can be assured if their actions are defined by a plan that is either explicitly agreed upon by all agents or checked by a coordinating agent. If agents can continually amend the plan, they can dynamically optimize their operations. Such a coordination method must ensure that no agent changes the plan in a way that adversely affects other agents or creates conflicts. If agents want to make conflicting changes to the plan, the method must fairly resolve the differences. Principled negotiation between agents is proposed for this purpose.

Principled Negotiation

Principled negotiation is a technique for efficiently obtaining a favorable and fair outcome from any negotiation situation.[9] The negotiating parties: 1) focus on underlying interests, not the bargaining positions, 2) search for options for mutual gain, and 3) use objective criteria to assess options.

Parties in a negotiation want an agreement that meets their interests. Focusing on the interests of all parties in the negotiation allows

Presented as Paper 96-3853 at the AIAA Guidance, Navigation, and Control Conference, San Diego, CA, July 29–31, 1996; received Aug. 29, 1996; revision received May 28, 1998; accepted for publication June 27, 1998.

*Graduate Student, Department of Mechanical and Aerospace Engineering; currently Lead Consultant, Booz·Allen and Hamilton, Cleveland, OH.

†Professor, Department of Mechanical and Aerospace Engineering. Fellow AIAA.

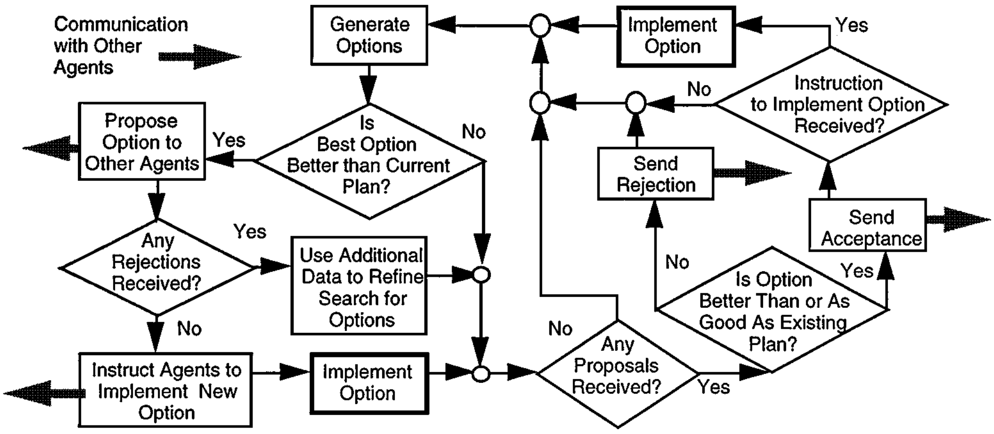
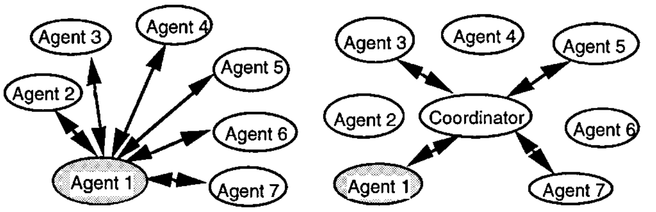


Fig. 1 Principled negotiation process for an agent.



a) Negotiation without a coordinator **b) Negotiation with a coordinator**

Fig. 2 Communications with a coordinator; Agent 1 proposes an option that does not conflict but that affects Agents 3, 5, and 7.

the parties to identify common and separate interests. The parties create options that can provide mutual gain. Most negotiation situations are not zero-sum games because the interests are not usually directly opposed. Finally, options are more easily agreed upon if the parties use common criteria for assessing them, as the parties will agree upon the relative value of any options. If the criteria are not agreed upon, knowledge of the criteria being used allows parties to assess how others will react to a proposal.

Principled negotiation motivates an iterative optimization method for MASs (Fig. 1). An initial master plan is formulated, specifying the actions of each agent. Agents repeatedly search for options (i.e., alternative master plans) that provide mutual gain. When an agent finds an option that better meets its interests, it proposes the option. The other agents evaluate the option and accept or reject it. If there are no rejections, the option is implemented. If an agent rejects the option, it sends a message to the proposer explaining the reasons for rejection. The proposing agent uses this additional information to improve its search.

Agent interaction through principled negotiation ensures coordination while giving freedom to optimize. Agent actions stay coordinated, as any proposals that caused conflict are rejected by other agents during negotiation. Agents can optimize their actions by amending the master plan, limited only by their ability to find options for mutual gain.

In large MASs, agent actions may not interact all of the time. Proposals might change only the actions of the proposing agent and leave the actions of other agents unchanged. Needless communication is avoided if the system has a coordinating agent (Fig. 2). The coordinator assesses proposals to ensure that the actions of the agents do not conflict. If a proposal ensures coordination but involves changes to the actions of other agents, the coordinator passes the proposal on to the affected agents. The coordinator requires no ability to optimize the actions of agents, only the information to assess how the agents interact.

Fundamentals of Principled Negotiation

An MAS has a set of agents $N = \{i\}$, $i = 1, \ldots, n$, whose actions are governed by their action plans $a_i(t)$ defined up to the agent's planning horizon. The planning horizon is the time at which the payoff is determined or a task is completed. In the AAS, the planning horizon of most agents is of the order of hours.

The dynamics of an MAS depend on the actions of all of the agents, the action profile:

$$\boldsymbol{a} = (a_1, a_2, \ldots, a_n) \tag{1}$$

This is the subject of principled negotiation. The consequences $\boldsymbol{c}$ of an action profile depend on the action profile, initial state ω, and disturbances $\boldsymbol{v}$:

$$\boldsymbol{c} = \boldsymbol{g}(\boldsymbol{a}, \omega, \boldsymbol{v}) \tag{2}$$

The consequences of an action profile determine how well each agent's goals and interests are met. The goals and interests of an agent are represented by its preference relation $\succ_i$. If Agent i prefers the consequences of Action Profile $\boldsymbol{b}$ to those of $\boldsymbol{a}$, then

$$\boldsymbol{g}(\boldsymbol{b}, \omega, \boldsymbol{v}) \succ_i \boldsymbol{g}(\boldsymbol{a}, \omega, \boldsymbol{v}) \tag{3}$$

Each agent has a feasible set of plans A_i that it could follow. The action plan it is pursuing a_i is a member of this set. If Agent i has a set of discrete choices, A_i is simply the list of choices. In the pursuit game,[6] A_i is {North, South, East, West}. If the action plan of an agent can be represented by m parameters, then A_i is a volume in m-dimensional space.

In the AAS, feasible sets are extremely large. Consider the choice of airport runway configuration (i.e., which runways are used for landings, takeoffs, or both). The number of configurations is finite. The TrMA looks at the predicted traffic and weather conditions to decide which configuration at which time maximizes capacity. Even if an airport has only four runway configurations and the day is discretized into 15-min portions, the airport TrMA has a choice of $4^{96} = 6.28 \times 10^{57}$ possible action plans for the next 24 h. An aircraft has an infinite choice of flyable trajectories to its destination. The large feasible sets make an exhaustive search of all of the options impossible.

Types of Agent

Agents are either "maximizers" or "satisficers." Maximizers try to maximize a function of the consequences. A utility function u_i can be defined on the set of consequences, whose value represents how well an action plan meets the interests of Agent i. The utility function depends on the desired final state ω_{des}, as well as on the consequences. Given a choice between two action profiles, $\boldsymbol{a}$ and $\boldsymbol{b}$, maximizing Agent i prefers $\boldsymbol{b}$ if

$$u_i[\boldsymbol{g}(\boldsymbol{b}), \omega_{\text{des}}] > u_i[\boldsymbol{g}(\boldsymbol{a}), \omega_{\text{des}}] \tag{4}$$

The utility function u_i is a way of implementing the preference function [Eq. (3)]. By converting the consequences into a scalar, a simple $>$ test can be used.

Satisficers want the action plan consequences to be satisfactory. For example, TrMAs want to ensure that aircraft always maintain a safe separation. The satisficing function $\boldsymbol{S}_i$ may be vector-valued if the satisficer is interested in several quantities. An action profile $\boldsymbol{a}$ is acceptable to a satisficer if

$$\boldsymbol{S}_i[\boldsymbol{g}(\boldsymbol{a})] > \boldsymbol{S}_{\min} \tag{5}$$

where $\boldsymbol{S}_{\min}$ is the vector of minimum acceptable values.

If the system is deterministic and agents have perfect information, then the action plans can be calculated just once. However, most MASs are stochastic, and agents have only partial information. In addition, agents may have freedom of action within the restrictions defined by their action plans. Therefore, an agent cannot determine exactly the consequences of a particular action profile and must estimate the consequences and the value of the utility or satisficing functions, too. Thus, Maximizer i prefers Action Profile $\boldsymbol{b}$ to $\boldsymbol{a}$ if

$$E_i[u_i(\boldsymbol{g}(\boldsymbol{b}))] > E_i[u_i(\boldsymbol{g}(\boldsymbol{a}))] \tag{6}$$

and Satisficer j finds Action Profile $\boldsymbol{d}$ acceptable if

$$E_j[\boldsymbol{S}_j(\boldsymbol{g}(d))] > \boldsymbol{S}_{\min} \tag{7}$$

where $E_j[\,]$ is the expectation operator for Agent j.

Options for Mutual Gain

Repeated rejections of proposals wastes communications bandwidth and has no beneficial effect to the system. Principled negotiation minimizes the chances of fruitless negotiation because agents search for options for mutual gain. A maximizing agent still searches for options that provide it with a higher utility, but it proposes an option only if it believes that all other agents benefit or are not affected by the change.

The form of option that an agent examines depends on its knowledge of the system. If Maximizing Agent i has no knowledge about the other agents, its search space is limited to A_i, i.e., it looks only at changes in its own action plan and assumes that other agents continue their present action plans. If the existing action profile is $\boldsymbol{a}$ [as in Eq. (1)], then Agent i tests options of the form

$$\begin{aligned}\boldsymbol{b} &= (b_1, b_2, \ldots, b_i, \ldots, b_n)\\ &= (a_1, a_2, \ldots, a_{i-1}, b_i, a_{i+1}, \ldots, a_n)\end{aligned} \tag{8}$$

Agent i proposes an option if it finds b_i such that

$$E_i[u_i(\boldsymbol{g}(\boldsymbol{b}))] > E_i[u_i(\boldsymbol{g}(\boldsymbol{a}))] \tag{9}$$

There is no guarantee that an option of this form [Eq. (8)] will not decrease the utility function of the other maximizers. For acceptance, not only must Eq. (9) be true, but also

$$E_j[u_j(\boldsymbol{g}(\boldsymbol{b}))] \geq E_j[u_j(\boldsymbol{g}(\boldsymbol{a}))] \forall j \in N\backslash\{i\} \tag{10}$$

where $N\backslash\{i\}$ signifies all agents other than i.

If Agent i has no knowledge of the other agents, a coordinating agent is needed. The coordinating agent ensures that any agents affected by a new action profile have the chance to approve or reject it. If an agent does have knowledge about the other agents, it can increase the size of its search space by examining options that change the action plans of other agents, i.e., Eq. (8) no longer holds. To propose an option, the agent must expect some gain [Eq. (9)], and it must also expect other agents to benefit or not be affected:

$$E_i[u_j(\boldsymbol{g}(\boldsymbol{b}))] \geq E_i[u_j(\boldsymbol{g}(\boldsymbol{a}))] \forall j \in N\backslash\{i\} \tag{11}$$

Negotiation must still occur because Agent i may not correctly estimate the assessment of the option by the other agents j [Eq. (10)].

Application to AAS

Principled negotiation with a coordinator is particularly effective for an AAS because if separation criteria are met, the aircraft action plans are decoupled. The action plan of Aircraft a_i is its future trajectory. The separation criteria cover both en route separation and the separation of arrival and departure slots. The utility of a flight to an aircraft or operator depends on attributes of the outcome and the desired final state $\boldsymbol{\omega}_{\text{des}}$. A general form of aircraft utility function is

$$u_i = \boldsymbol{w}_i^T(\boldsymbol{\omega})\boldsymbol{y}_i[\boldsymbol{g}(\boldsymbol{a}, \boldsymbol{\omega}, \boldsymbol{v}), \boldsymbol{\omega}_{\text{des}}] \tag{12}$$

where

$$\boldsymbol{y}_i = [f_i \quad t_i \quad ls_i \quad vs_i \quad e_i]^T \tag{13}$$

The components of $\boldsymbol{y}$ are the fuel usage, timeliness, lateral aircraft separation, vertical aircraft separation, and environmental hazard avoidance measures. Here, $\boldsymbol{w}$ is the vector of weights for each attribute, whose components change to reflect the importance of each attribute given the aircraft's state.

The actions of other aircraft affect the utility of Agent i only through the two aircraft separation measures. A plausible form for the aircraft lateral separation measure (Fig. 3a) is

$$\begin{aligned} ls_i &= \left(D_{\min} + \Delta D - \min_{t,j}(|\hat{\boldsymbol{r}}_i(t) - \hat{\boldsymbol{r}}_j(t)|)\right)^k \\ &= (D_{\text{marg}} - \hat{d}_{\min})^k \qquad \hat{d}_{\min} < D_{\text{marg}} \\ &= 0 \qquad \hat{d}_{\min} \geq D_{\text{marg}} \\ &\qquad D_{\text{marg}} \equiv D_{\min} + \Delta D \end{aligned} \tag{14}$$

$D_{\min}$ is the minimum desired lateral separation, $\hat{d}_{\min}$ is the estimate of the minimum lateral separation that will arise between Aircraft i and any other aircraft, k is a positive even integer, and ΔD is a small margin added so that a predicted minimum lateral separation of $D_{\min}$ incurs a high cost and so is unlikely to be chosen.

Consider an AAS with two aircraft. Aircraft 1 is choosing between two options, $\boldsymbol{a}$ and $\boldsymbol{b}$, where $\boldsymbol{b}$ has the form of Eq. (8), i.e., Aircraft 1 is looking only at changes to its own trajectory. Both options incur no lateral separation cost penalty for Aircraft 1, as the predicted minimum separation from Aircraft 2 is greater than D_{marg} in both cases (Fig. 3a). Assume option $\boldsymbol{b}$ has a higher total utility to

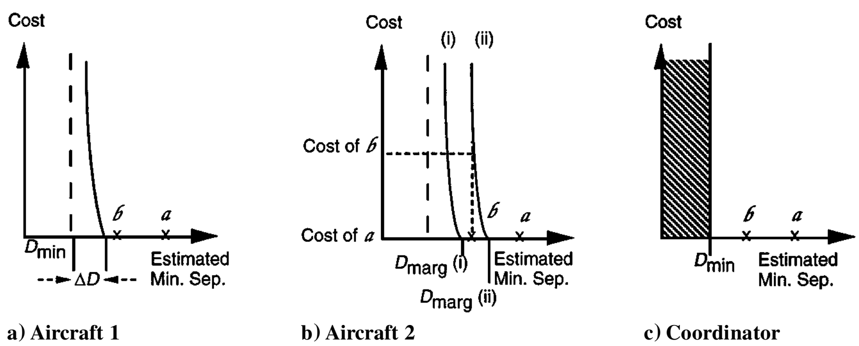


Fig. 3 Costs incurred by options with different lateral separation cost functions.

Aircraft 1. If Aircraft 2 has the same D_{marg} as Aircraft 1 [Fig. 3b (i)], then it accepts a proposal to switch to Action Profile $\boldsymbol{b}$, as its utility is unchanged. However, if Aircraft 2 has a different D_{marg}, then it rejects the option, as it incurs a cost penalty [Fig. 3b (ii)]. Agreement on $D_{\min}$ ensures that no aircraft waste computational time searching for options that will be rejected as unsafe by other aircraft.

The use of objective separation criteria allows decoupling. Each agent searches for changes in its own trajectory, i.e., options of the form of Eq. (8). Because the other aircraft experience no change in their utility functions from such an action profile if the separation criteria are met, a coordinating agent, the TrMA, can eliminate the need for Aircraft 1 to obtain agreement from all other aircraft. [In an AAS, there are many TrMAs, each responsible for a different volume of airspace. A flight usually passes through the airspace of many TrMAs. This presents a coordination problem in itself (see Ref. 10)]. The coordinator's satisficing function should reflect the agreed separation criterion $D_{\min}$ (Fig. 3c). The coordinator must be able to predict the future separations at least as well as the aircraft to carry out this role. If an aircraft proposes an option that changes the action plans of other aircraft, then once the option has been checked by the TrMA for separation, it is passed on to the affected agents for approval. The use of a coordinator ensures that $D_{\min}$ is never violated.

Performance of Principled Negotiation

Two optimization examples are presented to answer the following questions: How good is the action profile developed by principled negotiation compared to that created by a centralized system? What is the effect of agent knowledge on negotiation? How does principled negotiation perform in constrained and unconstrained situations?

Agent Coordination Example

Consider an n-agent MAS in which each agent's action plan is specified by two parameters, x and y, with values between 0 and 10:

$$a_i = (x_i \quad y_i) \qquad i \in N = \{1, \ldots, n\} \tag{15}$$

Each agent tries to maximize a utility function by appropriate choice of its own action plan. An agent's utility function does not depend on the action plans of the other agents:

$$u_i = u_i[g(a_i)] \tag{16}$$

The action plans of the agents must not conflict. The example coordination criterion used is

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} > \mathcal{D} \qquad \forall i, j \in N \tag{17}$$

For this example, Agent i is aware of the action plan of an other Agent j if

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \le r \tag{18}$$

The utility functions of the agents are decoupled in the same way as for aircraft [Eqs. (13) and (14)]. The utility function of the agents used in this example is

$$u_i = -\left[c_1\left(x_i - x_i^{\mathrm{opt}}\right)^2 + c_2\left(y_i - y_i^{\mathrm{opt}}\right)^2\right] \tag{19}$$

On each run, x^{opt} and y^{opt} are randomly assigned for each agent and may conflict. Compared to the general utility function given in Eq. (12), this function has no state-dependent weighting vector.

The optimal solution for a single agent is trivial by analytical means. However, in many MASs, a closed-form analytical optimization may not be possible. The evaluation of Eq. (6) alone can involve significant computational expense. It is often not possible to get a simple relation such as Eq. (19) between the action plan parameters and the utilities. Typical optimization procedures, such as gradient methods or genetic algorithms, iteratively search the feasible set. In this study, random search is used to generate options because its performance is independent of the form of the utility function being maximized. Any utility function could be used in place of Eq. (19), and similar results would be obtained.

Several optimization schemes are studied: distributed optimization with no coordination (Scheme A), three types of centralized optimization (Schemes B1, B2, and B3), and distributed optimization using principled negotiation (Scheme C). The first two optimization schemes provide baselines against which to compare the performance of principled negotiation. A single run of each optimization scheme consists of 40 iterations; averages are compiled from 100 runs. The coordination criterion used is $\mathcal{D} = 1$ [Eq. (17)].

Distributed Optimization with No Coordination: Scheme A

Consider the performance of the system if there is no coordination between agents, and the agents have no knowledge of the action plans of other agents [$r = 0$ in Eq. (18)]. As each agent has no knowledge of the other agents' action plans, it searches the entire action plan parameter space for better options. If it finds a better option, it switches to that option; otherwise its action plan remains unchanged. The probability of success in a random search by Agent i is proportional to the ratio of the area of better options to the total search area (Fig. 4)

$$P(\text{better option for } i) = \frac{|\boldsymbol{B}_i|}{|\boldsymbol{A}_i \cup \boldsymbol{B}_i|} = \frac{|\boldsymbol{B}_i|}{|A_i|} \tag{20}$$

The numerator decreases as the option improves, while the denominator remains constant. (In this case, the option space A_i is the same for all agents. In general, this will not be the case.) Hence, each agent's utility function should monotonically improve, with the improvement rate decreasing as the agent's option improves. The probability that an option fails to meet separation criteria is given by

$$P(\text{conflict}) = \frac{\left|A_i \cap \bigcup_{j \in N \setminus \{i\}} \boldsymbol{C}_j\right|}{|A_i|} \tag{21}$$

where the numerator is the area of the feasible set of Agent i that is in conflict with other agents. As there is no coordination mechanism, conflicts will arise.

The results show the expected monotonic decrease in cost (Fig. 5). Cost is the negative of utility. However, the agents were in conflict during a number of iterations. The probability of conflict increases as the number of agents increases [Eq. (21)]. Whereas average start and end cost for each agent stay almost constant (Fig. 6), the number of violations increases quadratically (Fig. 7).

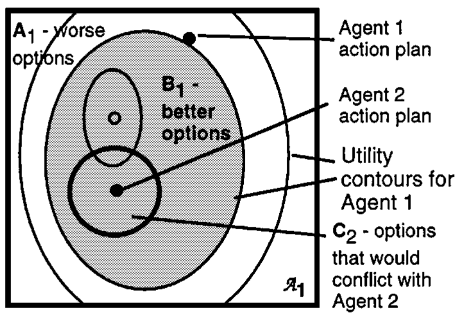


Fig. 4 Regions of worse, better, and conflicting options for Agent 1.

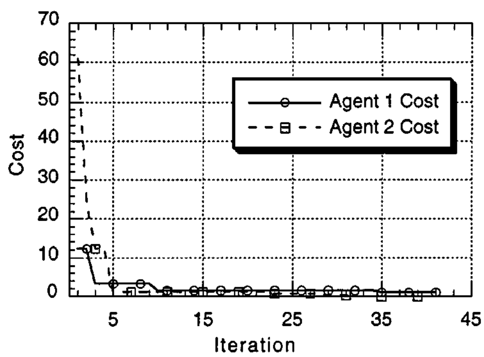


Fig. 5 Optimization performance of a two-agent system with no coordination, single run.

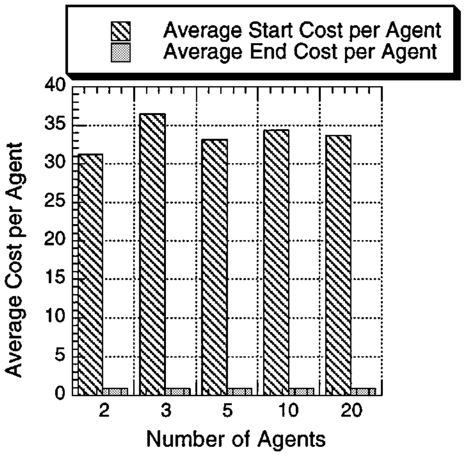


Fig. 6 Average start costs and end costs from 100 runs for independent random searches (Scheme A) with no coordination.

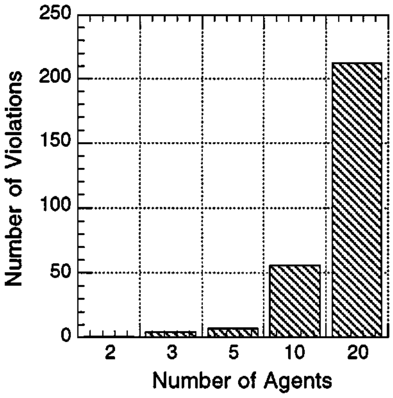


Fig. 7 Number of coordination violations with varying numbers of agents.

Centralized Optimization: Scheme B

Centralized optimization provides the other extreme. A coordinating agent has responsibility for assigning action plans to all of the agents and ensures that they are coordinated. As an ideal case, the coordinating agent has perfect knowledge of the utility functions of each agent. Random search is used by the coordinating agent to generate new options for the other agents. The coordinating agent uses three optimization schemes:

1) Scheme B1 generates a new action profile: Agents change to the option if net utility over all agents improves and the new action profile meets the coordination criteria.

2) Scheme B2 also generates a new action profile: Agents change action plan only if net utility improves, no individual utility worsens, and the action profile meets the coordination criteria.

3) Scheme B3 generates a new action plan for each agent in turn: Agents change action plan if their individual utility improves, and the option is coordinated.

In Schemes B1 and B2, n new (x, y) pairs are generated at each iteration (i.e., a complete action profile is generated). If this option does not meet the coordination criteria, a new set of pairs is generated. In Scheme B3, new options are developed by generating a new action plan for each agent one at a time. Each new action plan is checked for conflicts with the other agents' existing action plans. Only if the option meets the coordination criteria is an agent's action plan changed to the new option. All of these optimization schemes start from an initial feasible action profile (i.e., one that meets the coordination criteria), and coordination conflicts cannot occur under any of these schemes.

Scheme B1 results worsen as the number of agents increases (Fig. 8) because increasing the number of agents reduces the probability of generating an option that meets separation criteria and improves utility. With 20 agents, there is almost no improvement in cost after 40 iterations. Scheme B1 also causes some agents to switch to action plans with worse utility because the net utility has improved. Even averaged over 100 trials, smooth advances cannot be assured (Fig. 9).

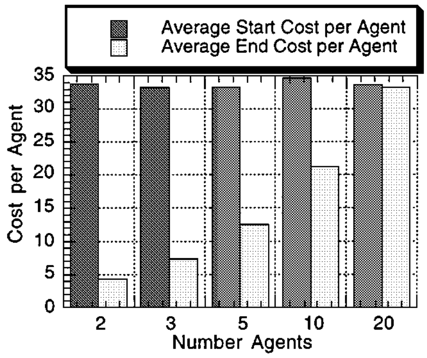


Fig. 8 Centralized optimization using net utility (Scheme B1).

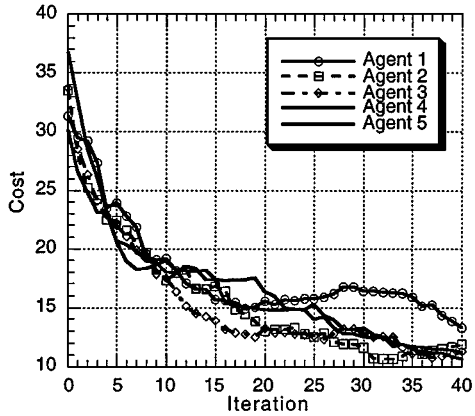


Fig. 9 Agent costs on each iteration averaged over 100 games using Scheme B1.

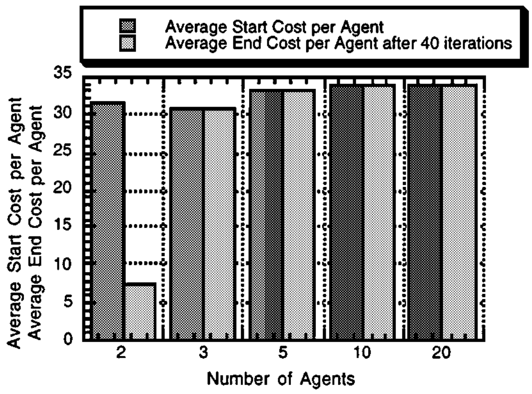


Fig. 10 Scheme B2 optimization performance averaged from 100 runs.

Scheme B2 tackles the problem of worsening individual agent costs but uses the same method for generating options. The performance of this method is worse than Scheme B1 (Fig. 10). The stricter criteria for adopting an option reduce the probability of finding a better option on each iteration. With as few as three agents, no improvement is found.

Scheme B3 considers each agent in turn. A single action plan is generated and checked for conflict and utility for each agent in turn. Scheme B3 shows significantly better results than Schemes B1 or B2 (Fig. 11). The average cost per agent achieved after 40 iterations is almost zero with as many as 20 agents, in contrast to Schemes B1 and B2. The end costs match those achieved by individual optimization with no coordination (Fig. 6), although this scheme ensures coordination of action plans. These results show that an iterative optimization method considering each agent individually is effective even when the system is constrained by the need to coordinate many agents.

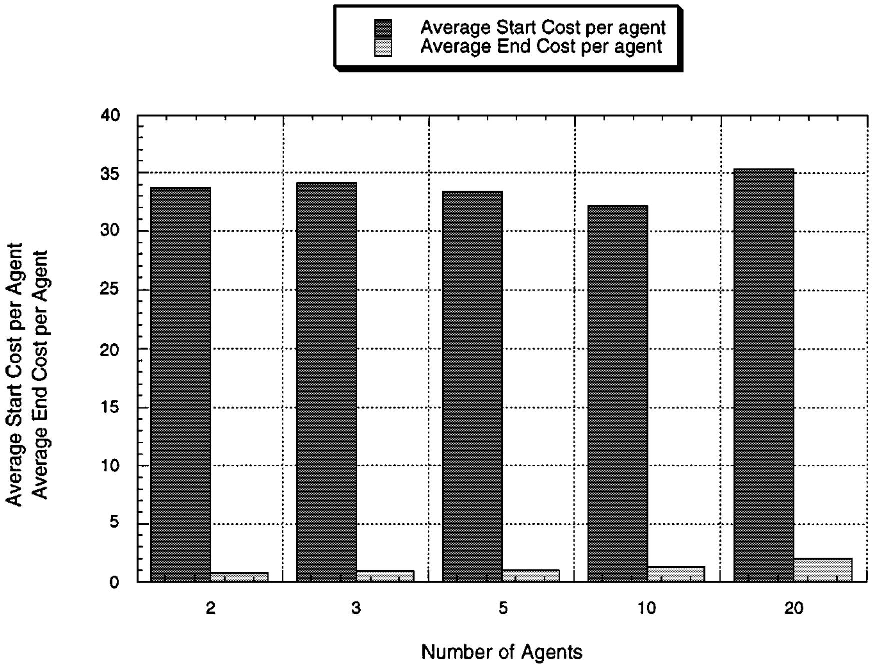


Fig. 11 Optimization performance for Scheme B3 averaged over 100 runs.

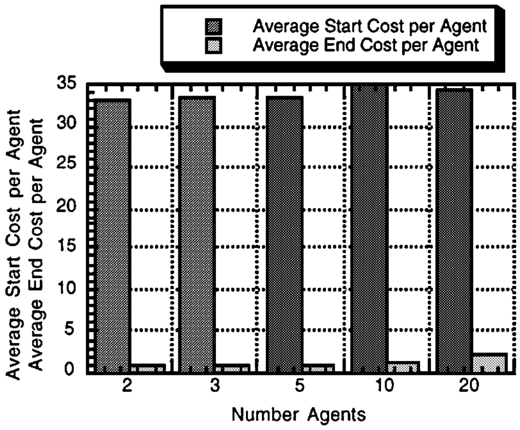


Fig. 12 Distributed optimization, no knowledge of other agents (r = 0).

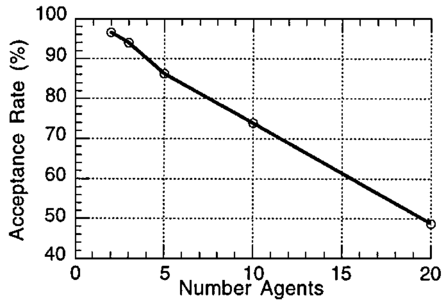


Fig. 13 Variation of acceptance rate with number of agents (r = 0).

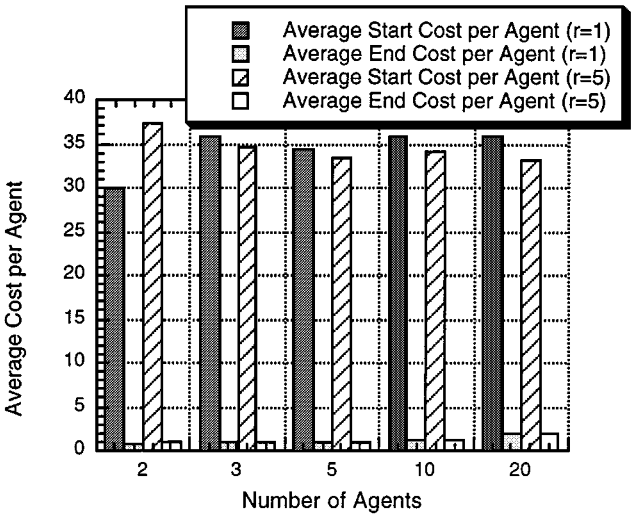


Fig. 14 Optimization performance with varying agent knowledge (r = 1 and 5).

Distributed Optimization Using Principled Negotiation: Scheme C

Scheme B3, pointwise iteration, is the most efficient of the three centralized optimization schemes. It also is suited to implementation as a distributed optimization scheme. Each agent's trajectory, its utility, and new options could be calculated by each agent individually. The coordinating agent checks only the options for coordination. If an agent generates an option with a lower cost, it submits the option to the coordinator. If the coordinator clears the option, it is implemented.

To begin, all agents (except the coordinating agent) are unaware of the action plans of other agents ($r = 0$). The average end cost per agent after 40 iterations is the same as that obtained with centralized optimization using Scheme B3 and is close to the end cost achieved by Scheme A, distributed optimization without coordination (Fig. 12). Principled negotiation is preventing conflicts but with minimal effect on the agents' ability to optimize their action plans.

Because the agents have no knowledge of the other agents' action plans, their proposals may be rejected by a coordinator. Ideally, the acceptance rate, the proportion of proposals accepted, should be maximized, as rejected proposals represent wasted communications bandwidth. With no knowledge of other agents, the acceptance rate falls as the number of agents increases because the probability of generating a conflicting option increases (Fig. 13).

In many MASs, agents have some knowledge of the action plans of other agents. Increasing agent awareness does not significantly change the final costs achieved (Fig. 14), but it increases the acceptance rate (Fig. 15). If an agent generates an unacceptable option, it is discarded and never proposed.

The results show that principled negotiation allows agents to optimize their action plans, achieving the same cost levels as centralized coordination. Increasing each agent's awareness of the action plans of other agents increases the acceptance rate, but does not affect the final cost achieved. This suggests that the level of optimization that an agent can achieve is limited by the performance of its individual optimization method, not by its awareness of other agents.

Runway Slot Negotiation Example

Principled negotiation also is an effective optimization method in situations where the agents' actions are closely coupled, such as arrivals sequencing at a busy airport. The acceptance rate of an airport (the number of available slots per hour) depends on the runway configuration, the number of departures, and the weather conditions. The Federal Aviation Administration (FAA) monitors the national AAS. If the FAA predicts that arrivals will exceed the arrival rate of an airport, flow control restrictions are placed on aircraft departing for that airport. If a ground delay program is run, departing aircraft are given the time at which they can expect to receive clearance to depart, calculated from their allocated arrival slot time at the constrained airport.

Currently, airlines must accept the arrival slots they are allocated, unless they first cancel a flight. If they cancel a flight, they can substitute another flight into this slot, thus freeing another slot. No swapping of flights between airlines is allowed. The airline incurs a number of costs from a change in arrival time (e.g., extra crew duty time and missed connections of passengers, crews, or aircraft), which airlines want to minimize.

A model of this situation was created using MATLAB™. The Action Profile $\boldsymbol{a}$ is the list of slot times allocated to the arriving flights. The feasible set for an Aircraft A_i is the list of all available slots for the time period. Only one aircraft can occupy each arrival slot. The consequence function $\boldsymbol{g}$ calculates the average velocity required to meet the arrival slot. A large cost penalty is incurred if an aircraft cannot meet a slot time because it must exceed its maximum velocity. Given an originally scheduled arrival time t_s, an allocated slot time t_p, and flight time t_f, a velocity required to meet the slot time V and a preferred velocity V_{pref} a general form of the cost function is

$$
\begin{aligned}
J &= c_f(V - V_{\text{pref}})^2 t_f + c_{\text{apron}}(t_s - 0.25 - t_p) \\
&\quad + c_{\text{crew}}(t_p - t_s - 0.15) + c_{\text{cons}}(t_p - t_s - 0.5) \\
&\quad + c_{V_{\max}}(V - V_{\max}) + c_{V_{\min}}(V - V_{\min})
\end{aligned}
$$

$$
\begin{aligned}
c_{\text{apron}} &= \begin{cases} c_1, & t_p < t_s - 0.25 \\ 0, & t_p \ge t_s - 0.25 \end{cases} \\
c_{\text{crew}} &= \begin{cases} c_2, & t_p > t_s + 0.15 \\ 0, & t_p \le t_s + 0.15 \end{cases} \\
c_{\text{cons}} &= \begin{cases} c_3, & t_p > t_s + 0.5 \\ 0, & t_p \le t_s + 0.5 \end{cases} \\
c_{V_{\max}} &= \begin{cases} c_4, & V > V_{\max} \\ 0, & V \le V_{\max} \end{cases} \\
c_{V_{\min}} &= \begin{cases} c_5, & V < V_{\min} \\ 0, & V \ge V_{\min} \end{cases} \\
& c_1, c_2, c_3, c_4, c_5 > 0
\end{aligned}
\tag{22}
$$

The terms represent the fuel cost from nonpreferred aircraft velocity, extra apron costs from early arrival, extra crew costs from extra duty length, cost of missed connection, penalty for exceeding $V_{\max}$, and penalty for extra path length required when velocity would otherwise drop below $V_{\min}$. The units of time are hours; 0.25 represents 15 min (Fig. 16).

In this model problem, a fixed number of slots are created for a 1-h period. Three airlines have flights scheduled to arrive during that period. The airport TrMA initially calculates the expected arrival sequence (based on the preferred velocity and flight distance for each arrival) and allocates slots in this sequence.

Optimization proceeds in two steps. First, each airline in turn examines the slots it has been allocated and any unallocated slots. It uses an A^* search (a branch-and-bound search with dynamic programming[11]) through the tree of possible valid slot allocations to find the least-cost option. As this slot allocation is acceptable to the airport TrMA, the slot allocation is updated. Then, all airlines enter bilateral negotiation, considering each possible swap of arrival slots between them. If any swaps provide mutual gain (i.e., the cost incurred by both flights decreases), then the slots are exchanged.

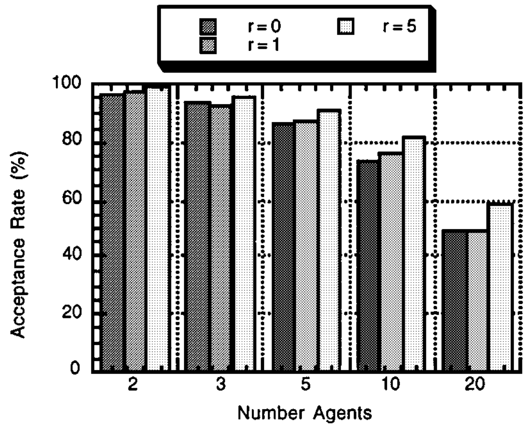


Fig. 15 Variation of acceptance rate with agent knowledge.

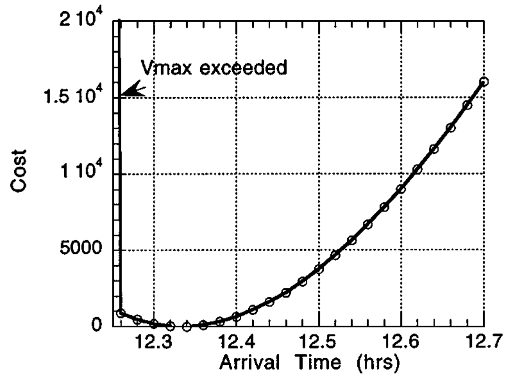


Fig. 16 Example cost function for a single flight.

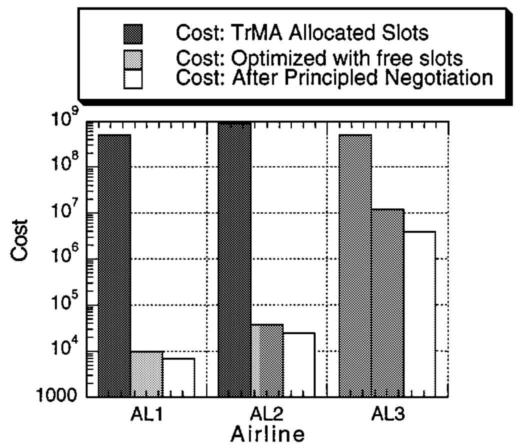


Fig. 17 Results for 35 flights, 3 airlines, 51 slots.

The principled negotiation step brings improvements in the cost to each airline (Fig. 17). For the example shown, the further cost reduction due to principled negotiation was 29, 32, and 67%, respectively. Principled negotiation increases the size of the search space available to the agents. Agents can look at options that involve changes to the other agents' action plans; they are not limited to options of the form of Eq. (8). If Airline i has m_i flights, Airline j has m_j flights, and there are p free slots, then without negotiation Airline i has $(m_i + p)!/p!$ possible arrival sequences. If the airlines negotiate, Airline i now has $(m_i + m_j + p)!/(m_j + p)!$. If Airline 1 has 10 slots for its 10 flights and there are no free slots, then it has a total of $10! = 3.63e06$ possible slot allocations. If Airline 2 has 5 slots and the airlines negotiate, Airline 1 now has $1.09e10$ possible slot allocations. The size of the search space has increased by several orders of magnitude.

Implementing Principled Negotiation in an AAS

Principled negotiation can be applied immediately to all AAS interactions. Limited negotiation occurs already (e.g., slot substitution and pilot altitude change requests) but principled negotiation can be used more extensively. It can be used to varying degrees by aircraft of varying equipment levels, as it does not require a particular level of agent intelligence or knowledge. If an agent can decide which action profile it prefers, then principled negotiation can be used. Better-equipped aircraft will be able to assess more options that are possibly more complex and negotiate more often. The more capable the agent, the better the agent can optimize its operations. The requirement that a proposal provide mutual gain ensures that agents with less knowledge or computational power are not disadvantaged.

Increased agent knowledge reduces wasted negotiation by constraining the search space. For AAS agents other than TrMAs to assess traffic coordination, information on the states and action plans of the aircraft in the system are needed. Plans to broadcast global positioning system position data in a traffic alert and collision avoidance system (TCAS) 2000 system will improve the available state data, but knowledge of the action plans of other aircraft would still be limited. Using estimation techniques to predict the future actions of aircraft may prove too inaccurate. (An advantage of the free-flight concept is that it should promote the use of more direct and, therefore, more predictable routings.[12]) Three approaches could be taken. Flight plan information maintained by the ground could be made accessible to queries from aircraft or airlines. Alternatively, a TrMA could supply this information when it rejects a proposal from an agent. The agent could then re-evaluate the options in the light of the new information. Aircraft could transmit concise flight plan data as well as state data in a TCAS 2000-type system.

Today, many airlines are not installing satellite communications systems and other types of data link because of the perceived lack of benefits compared to the installation costs. Principled negotiation lets operators gain maximum benefit from their aircraft systems, encouraging the use of the best possible equipment. If an aircraft has more information and more computational power, it can generate and examine more complex options, more often, and propose them with a greater chance of acceptance. At the same time, principled negotiation guarantees that no operator is adversely affected compared to today's operations if it cannot afford an advanced equipment fit.

To quantify the benefits of principled negotiation on AAS operations, a computer simulation is being developed that models AAS agents developing options for mutual gain and negotiating. The required times of arrival are first negotiated between the airline and the destination airport TrMA, and a flight plan is filed. During the flight, aircraft repeatedly search for trajectory improvements. Aircraft use a rule-based search method that accounts for the scenario. If a conflict arises, both aircraft try to find an avoidance maneuver. If the aircraft fail to find a solution, the TrMA instructs the aircraft to implement a particular avoidance maneuver. Aircraft search for higher utility trajectories if they are not in conflict situations. The weighting function of the aircraft depends on the progress of the flight and the priorities communicated by the airline.

Conclusions

Principled negotiation effectively coordinates distributed optimization in MASs. It works well in both constrained and unconstrained situations, as demonstrated by two example problems. The optimization performance of an MAS using principled negotiation is as good as that of a centralized system that uses the same optimization method and has perfect knowledge about each agent. The proportion of fruitless negotiations decreases as agent knowledge increases. Principled negotiation offers great advantages for AASs because it would increase the freedom of system users to optimize their operations while maintaining safety.

Acknowledgment

This research was funded by the FAA under Grant 92-G-0011.

References

[1] Nolan, M. S., *Fundamentals of Air Traffic Control*, Wadsworth, Belmont, CA, 1990.

[2] Green, S. M., den Braven, W., and Williams, D. H., "Development and Evaluation of a Profile Negotiation Process for Integrating Aircraft and Air Traffic Control Automation," NASA TM 4360, April 1993.

[3] Jennings, N. R., *Cooperation in Industrial Multi-Agent Systems*, World Scientific, Singapore, 1994.

[4] Steeb, R., McArthur, D. J., Cammarata, S. J., Narain, S., and Giarla, W. D., "Distributed Problem Solving for Air Fleet Control: Framework and Implementation," *Expert Systems: Techniques, Tools and Applications*, edited by P. Klahr and D. Waterman, Addison–Wesley, Reading, MA, 1986, pp. 391–432.

[5] Davis, R., and Smith, R. G., "Negotiation as a Metaphor for Distributed Problem Solving," *Artificial Intelligence*, Vol. 20, 1983, pp. 63–109.

[6] Levy, R., and Rosenschein, J. S., "A Game Theoretic Approach to Distributed Artificial Intelligence and the Pursuit Problem," *Decentralized A.I.-3*, edited by Y. Demazeau and J.-P. Müller, Elsevier, New York, 1992, pp. 129–146.

[7] Wang, J., *The Theory of Games*, Oxford Univ. Press, Oxford, England, UK, 1988.

[8] von Neumann, J., and Morgenstern, O., *Theory of Games and Economic Behavior*, Princeton Univ. Press, Princeton, NJ, 1947.

[9] Fisher, R., and Ury, W., *Getting to Yes: Negotiating Agreement Without Giving In*, Penguin, New York, 1981.

[10] Wangermann, J. P., and Stengel, R. F., *Technology Assessment and Baseline Concepts for Intelligent Aircraft/Airspace System*, Dept. of Mechanical and Aerospace Engineering, MAE-2016, Princeton Univ., Princeton, NJ, Feb. 1995.

[11] Winston, P. H., *Artificial Intelligence*, Addison–Wesley, Reading, MA, 1992.

[12] "Report of the RTCA Board of Directors' Select Committee on Free Flight," RTCA, Inc., Washington, DC, Jan. 1995.