

Costs of Limiting Route Optimization to Published Waypoints in the Traffic Aware Planner

David A. Karr* and Robert A. Vivona†

Engility Corporation, Billerica, MA 01821, USA

David J. Wing‡

NASA Langley Research Center, Hampton, VA 23681, USA

The Traffic Aware Planner (TAP) is an airborne advisory tool that generates optimized, traffic-avoiding routes to support the aircraft crew in making strategic reroute requests to Air Traffic Control (ATC). TAP is derived from a research-prototype self-separation tool, the Autonomous Operations Planner (AOP), in which optimized route modifications that avoid conflicts with traffic and weather, using waypoints at explicit latitudes and longitudes (a technique supported by self-separation concepts), are generated by maneuver patterns applied to the existing route. For use in current-day operations in which trajectory changes must be requested from ATC via voice communication, TAP produces optimized routes described by advisories that use only published waypoints prior to a reconnection waypoint on the existing route. We describe how the relevant algorithms of AOP have been modified to implement this requirement. The modifications include techniques for finding appropriate published waypoints in a maneuver pattern and a method for combining the genetic algorithm of AOP with an exhaustive search of certain types of advisory. We demonstrate methods to investigate the increased computation required by these techniques and to estimate other costs (measured in terms such as time to destination and fuel burned) that may be incurred when only published waypoints are used.

I. Introduction

THE Traffic Aware Planner (TAP) is a pilot advisory tool that monitors for route and altitude changes to improve flight attributes such as flight time or fuel burn while also avoiding traffic aircraft. In computing optimized alternatives to the current trajectory, TAP processes ownship trajectory and flight status information, traffic surveillance data, and other available information on changing wind fields, convective weather hazards, and special use airspace. The resulting routes are optimized for time or fuel, according to user preference, and are displayed to the pilot for reference in making trajectory change requests to ATC. To increase the likelihood of ATC approval, the candidate trajectory changes leverage traffic information received by Automatic Dependent Surveillance Broadcast (ADS-B) to avoid requesting routes that conflict with known traffic aircraft. To this end, the TAP algorithms were derived from the Autonomous Operations Planner (AOP), a research prototype airborne avionics system designed to enable aircrews to self-separate from traffic and thereby achieve significant operational flexibility and route optimization.^{1,2}

This concept of self-separation is illustrated in Figure 1. Because self-separating aircraft select and authorize their own trajectories, rerouting must take into account the types of constraints shown in Figure 1, but otherwise full use of the airspace is available and waypoints may use arbitrary latitude/longitude locations as computed dynamically by on-board algorithms.

The Traffic Aware Strategic Aircrew Requests (TASAR) concept proposes to apply similar airborne computational capabilities for flight optimization in a near-term environment in which trajectory changes

*Principal Computer Scientist, ATRE Department, 300 Concord Road, AIAA senior member. 978-671-1110.

†Chief Research Engineer, ATRE Department, 300 Concord Road, AIAA associate fellow. 978-671-1116.

‡ATM Research Engineer, Crew Systems & Aviation Operations, M.S. 152, AIAA member. 757-864-3006.

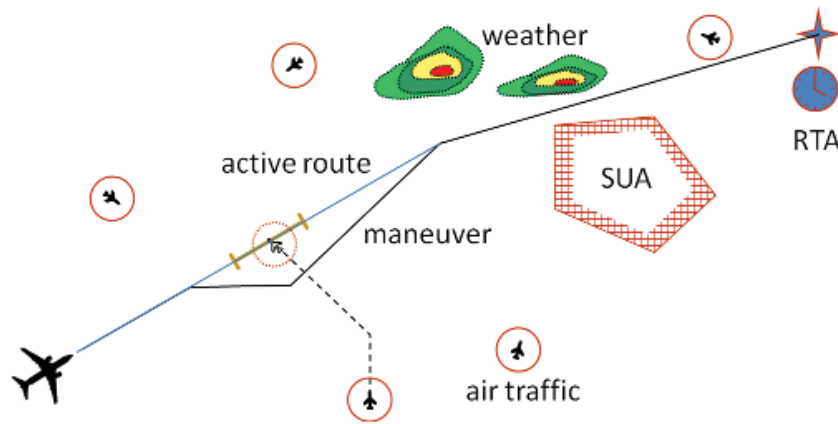


Figure 1. Global trajectory-based self-separation with constraints.

must be requested and approved by ATC.^{3,4} Since aircrew requests over voice communication frequencies are only practical with the use of named waypoints, the algorithm for computing conflict-free, flight-optimizing trajectories must be constrained to a published waypoint database, an adaptation that comes at some cost. This paper will describe the adaptation method and assess this cost in operationally relevant terms.

II. The Pattern-Based Genetic Algorithm

The strategic planning capability of AOP (and now also TAP) is supported by the Pattern-Based Genetic Algorithm (PBGA) as described in previous publications.^{2,5,6} An instance of AOP performs strategic planning and other functions for an aircraft on which it is installed, known as the *ownship* aircraft with respect to that instance of AOP. During the flight of the ownship aircraft, AOP may call PBGA at various times to *resolve* various strategic-planning problems by generating new routes, known generally as *resolution routes*. The algorithm applies a parameterized *pattern* to an existing route—typically the active route in the aircraft’s Flight Management System (FMS)—in order to produce a resolution route consisting of part of the existing route combined with a *maneuver* (usually consisting of one or more new legs) that may be uploaded to the aircraft’s FMS.

The “genetic” aspect of PBGA requires AOP to try multiple instances of patterns in order to address any individual strategic-planning problem. A *chromosome* is an individual instance of a pattern. An essential stage of the algorithm is the evaluation of the *fitness* of each chromosome, which requires AOP to perform several calculations:

1. Determine the new route that would result from applying this chromosome to the existing route.
2. Accurately predict the four-dimensional trajectory that the aircraft would fly if the new route were uploaded and executed in the FMS.
3. Accurately predict the cost (usually measured in terms of time in flight and/or fuel burned) of executing the new route.
4. Accurately predict any *conflicts* with the new route, that is, losses of separation with other aircraft or penetrations of special-use airspace or weather cells.

The fitness values in the “solution space” to be explored for a given chromosome can be a highly non-linear function of the chromosome’s genetic parameters. In particular, a small variation in the route can make the difference between an efficient conflict-free trajectory (with a good fitness) and a trajectory that has one or more conflicts with other aircraft, weather cells, and so forth, resulting in a much worse fitness. Sometimes, however, the position of a waypoint might range freely within a region of many square miles without much impact on the chromosome’s fitness. Such properties make exhaustive-search approaches difficult at best, and favor the use of an algorithm with “genetic” features.

Another implication of the “genetic” nature of PBGA is that it is desirable to evaluate more than a few chromosomes during the resolution of a single strategic-planning problem. In a typical application, a few hundred chromosomes (belonging to an evolving *population*) will be evaluated within a few seconds.⁵ Only one of these chromosomes, namely the one with the best fitness, can be returned by the algorithm to be used as a resolution route. If no chromosome within the population has acceptable properties when PBGA ends that population, the attempt to obtain a resolution route from that population fails.

III. Maneuver Patterns for the Traffic Aware Planner

The Traffic Aware Planner generates advisories in sets of three: one advisory with a lateral-only maneuver, obtained from a population of lateral maneuver patterns; one vertical advisory with no change in the lateral path of the route, obtained from a population of vertical maneuver patterns; and one hybrid advisory that combines lateral and vertical maneuvers, obtained from a population of hybrid maneuver patterns.

A. Lateral Patterns

The most general lateral maneuver pattern developed for TAP is the Two-Point Reconnect Pattern illustrated in Figure 2. The new route generated by this pattern consists of the following waypoints:

- The aircraft’s current location.
- A *turn-out* waypoint at a fixed distance ahead of the aircraft.
- Two *off-route* waypoints at published waypoint locations selected by the pattern.
- A *reconnection* waypoint that is part of the existing route.
- All the remaining waypoints of the existing route after the reconnecting waypoint.

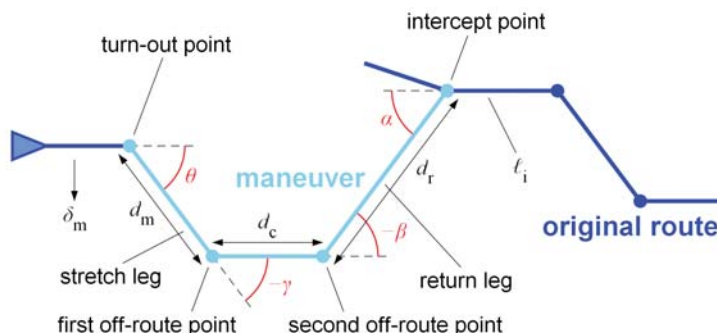


Figure 2. The Two-Point Reconnect Pattern.

The distance to the turn-out point is a predetermined function of the aircraft’s ground speed, estimated so as to give sufficient time to calculate, approve, and upload the new route before the aircraft has flown too far to make the desired turn. The remaining waypoints of the maneuver are determined by a combination of *discrete* parameters (selecting among a small number of alternatives) and *continuous* parameters (selecting within a range of numeric values not restricted to whole numbers). The first of these parameters, the integer ℓ_i , selects a waypoint on the existing route from a list of waypoints that have been deemed to be reachable by this pattern from the present position.^a The full set of genetic parameters of this pattern are listed in Table 1, using the same symbols as in Figure 2. The intercept angle need not have the same sign as the turn-out angle.^b

Another lateral pattern developed for TAP is the One-Point Reconnect Pattern illustrated in Figure 3. This pattern has only one off-route point, which corresponds to the first off-route point of the Two-Point

^aIn patterns developed for the separation-assurance function, the parameter ℓ_i selects an entire leg of the existing route that is to be intercepted. In TAP, however, maneuvers are not permitted to insert new waypoints along existing route legs.

^bAll angles in the TAP patterns are measured in the clockwise direction. That is, left turns have negative turn angles.

Table 1. Parameters of the Two-Point Reconnect Pattern

Symbol	Type	Description
ℓ_i	discrete	choice of reconnection waypoint
δ_m	discrete	sign of the turn-out angle θ (left or right)
$ \theta $	continuous	magnitude of the turn-out angle
d_m	continuous	length of the stretch leg
α	continuous	intercept angle
d_r	continuous	length of the return leg

Reconnect Pattern but connects directly to the reconnection waypoint via the return leg. The genetic parameters of this pattern are listed in Table 2.

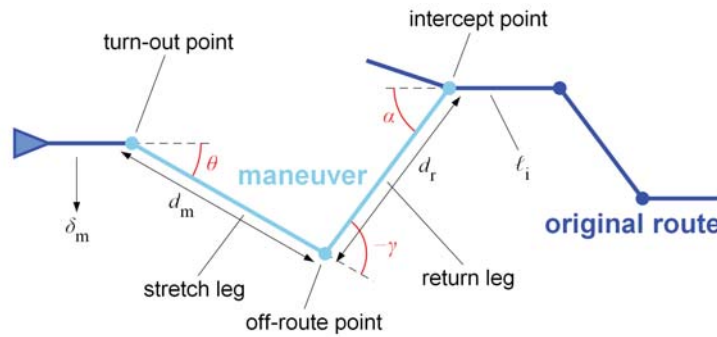


Figure 3. The One-Point Reconnect Pattern.

Table 2. Parameters of the One-Point Reconnect Pattern

Symbol	Type	Description
ℓ_i	discrete	choice of reconnection waypoint
δ_m	discrete	sign of the turn-out angle θ (left or right)
$ \theta $	continuous	magnitude of the turn-out angle
d_m	continuous	length of the stretch leg

The final lateral pattern developed for TAP is the Direct-to-Waypoint Pattern illustrated in Figure 4. This pattern has only one genetic parameter, the discrete parameter ℓ_i , which selects the reconnection waypoint as it does in the other two lateral patterns.

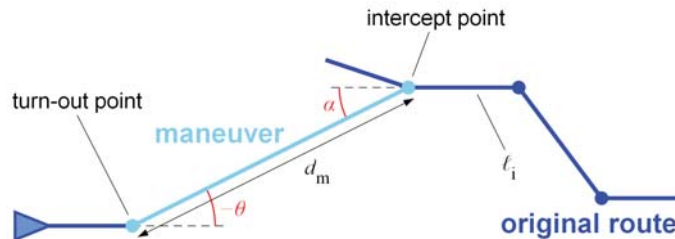


Figure 4. The Direct-to-Waypoint Pattern.

B. Vertical Pattern

Only one vertical pattern, the Altitude Change Pattern, is used in TAP. This pattern selects an altitude to which the aircraft might climb or descend in cruise phase. The new altitude is selected from a small set consisting of altitudes at 1000-foot or 2000-foot intervals within a predetermined vertical distance above or below the current altitude. Altitudes too high for the aircraft to reach at its current weight in existing conditions are excluded.

The resulting route places a waypoint along the route at a fixed lateral distance ahead of the aircraft's present position with a STEP constraint that specifies the selected altitude. The distance to the STEP constraint is designed to enable the crew to obtain a clearance at the new altitude and to set that altitude as the new cruise altitude by the time the aircraft reaches that point. The STEP constraint is not reported as part of TAP's advisory, since it is not designed to be part of the clearance request or to be uploaded to the aircraft's FMS; instead, it serves as a proxy for the immediate cruise altitude change that the crew will execute if and when the advisory is approved.

C. Hybrid Patterns

Three hybrid patterns are used in TAP. Each of these patterns—the Hybrid Two-Point Reconnect Pattern, the Hybrid One-Point Reconnect Pattern, and the Hybrid Direct-to-Waypoint Pattern—combines a new altitude (specified the same way as in the Altitude Change Pattern, and similarly modeled by a STEP waypoint at a fixed distance ahead along the route) with a lateral route change specified by the same parameters as the Two-Point Reconnect Pattern, the One-Point Reconnect Pattern, or the Direct-to-Waypoint Pattern, respectively.

D. Limiting the Number of Waypoints in Advisories

The interface to TAP provides an option to limit the number of waypoints that will be included in any lateral or hybrid advisory. When this limit is set to two waypoints (not counting the reconnection waypoint), any of the lateral or hybrid maneuver patterns may be included in their respective populations. When the limit is set to one waypoint, the Two-Point Reconnect Pattern and Hybrid Two-Point Reconnect Pattern are excluded. When the limit is set to zero, only the Direct-to-Waypoint or Hybrid Direct-to-Waypoint Patterns may be used.

In other words, the patterns introduced into the lateral and hybrid populations are selected so that no chromosome in the population can produce more than the desired number of additional advisory waypoints prior to the reconnection waypoint.

E. Geometric Constraints of Maneuvers

While the choices of reconnection waypoint and off-route points in the lateral and hybrid maneuver patterns are randomized, the resulting route must satisfy certain geometric constraints. The purpose of these constraints is to avoid extreme maneuvers and to ensure that the FMS can construct a well-behaved trajectory along the resulting route.

In any of the lateral or hybrid patterns, the turn angle θ at the turn-out point and the turn angle α at the reconnection waypoint must satisfy the conditions

$$\begin{aligned} |\theta| &\leq 60 \text{ deg and} \\ |\alpha| &\leq 60 \text{ deg.} \end{aligned}$$

Moreover, the turn angle at the (first) off-route point of the One-Point and Two-Point Reconnect Patterns (and of the corresponding hybrid patterns) must satisfy the condition

$$|\gamma| \leq 90 \text{ deg,}$$

and the second off-route point of the Two-Point and Hybrid Two-Point Reconnect Patterns must satisfy

$$|\beta| \leq 90 \text{ deg.}$$

The lateral patterns must also satisfy certain minimum-distance criteria. The distance d_m from the turn-out point to the reconnect point of the Direct-to-Waypoint Pattern must satisfy the condition

$$d_m \geq 12 \text{ nmi.}$$

The lengths d_m and d_r in the One-Point Reconnect Pattern, as well as the great-circle distance d_{capt} from that pattern's turn-out point directly to its reconnect waypoint, must satisfy the conditions

$$\begin{aligned} d_m &\geq 15 \text{ nmi,} \\ d_r &\geq 15 \text{ nmi, and} \\ d_{\text{capt}} &\geq 30 \text{ nmi.} \end{aligned}$$

The corresponding distances in the Two-Point Reconnect Pattern, as well as the distance d_c between its two off-route points, must satisfy the conditions

$$\begin{aligned} d_m &\geq 25 \text{ nmi,} \\ d_c &\geq 25 \text{ nmi,} \\ d_r &\geq 25 \text{ nmi, and} \\ d_{\text{capt}} &\geq 100 \text{ nmi.} \end{aligned}$$

The hybrid versions of each lateral pattern must satisfy the corresponding distance constraints.

In the implementation of PBGA, the preceding conditions are treated merely as preliminary limits on the ranges of the genetic parameters. These limits can never be violated, but when a genetic parameter is assigned a value in a new chromosome, the value may be limited to a more restricted range than the preceding conditions imply, because assigning that parameter to some values within those preliminary limits would make it impossible to construct the desired maneuver from the turn-out point to the reconnection point while satisfying all the other conditions. The implementation of each maneuver pattern computes and enforces these additional restrictions on the ranges of genetic parameters during the construction of each chromosome.

IV. Use of Published Waypoints in Patterns

Previous experiments with AOP assumed that the aircraft had been equipped to upload route modifications directly from AOP to the FMS, and that it was not necessary to transmit the route modifications to controllers or systems on the ground. Route modifications therefore were permitted to specify new waypoints by latitude and longitude to several digits of precision. The design of TAP, however, assumes that the crew must transmit route modifications to ground controllers by voice communication; and if the modifications are accepted, the crew must key them into the FMS by hand. For this application, a more concise specification of the waypoints is desired. In particular, it is desirable to specify the off-route waypoints in the One-Point and Two-Point Reconnect patterns by the names by which they are known in the published database. (The turn-out point, which is merely a proxy for an immediate turn toward the next waypoint when the route is finally executed, is expected neither to be part of the clearance requested nor the route modification keyed into the FMS, and therefore need not be at a published location.)

The new waypoints formerly added to AOP routes were found by projecting points at random distances (and sometimes along random radials) from other waypoints. In principle, any arbitrary point in the airspace might be used as one of these waypoints. In practice, since only a limited number of digits of precision are uploaded for each latitude and longitude, the range of possible waypoints is a lattice or grid of points, in which the distance between adjacent grid points is determined by the number of digits of precision with which the latitudes and longitudes are expressed. This causes the locations of the new waypoints to be "rounded off" or "snapped" to this grid. It takes only a few significant digits, however, to locate such a grid point very close to any new waypoint that might be generated by one of the AOP patterns. Under these conditions, it is not difficult to accurately predict trajectories, times, fuel costs, and conflicts as required for the calculation of chromosome fitness in PBGA.

Very few of the possible new waypoints in a resolution route are within a negligible distance of any published waypoint. This is partly due to the limited number of published waypoints, and partly due to the distribution of these waypoints within the airspace. For example, Figure 5 shows the distribution of published

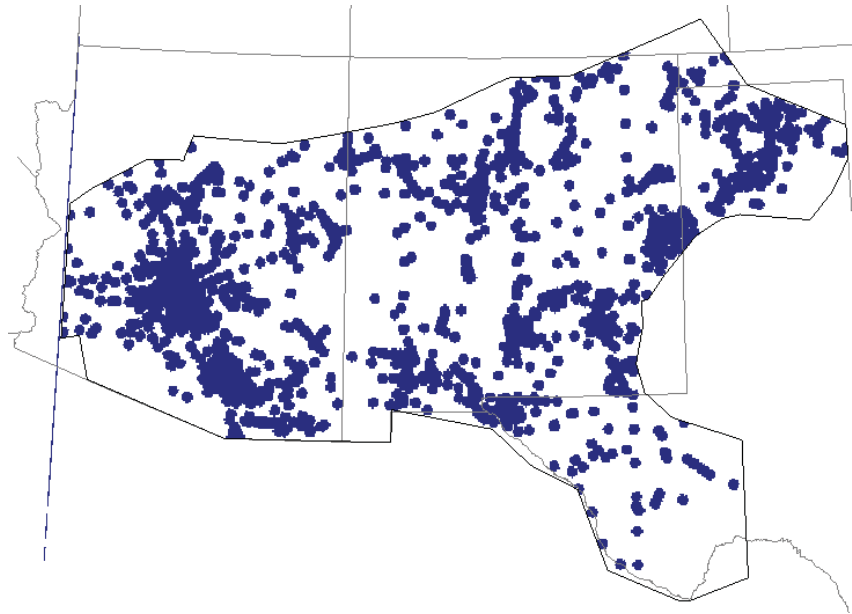


Figure 5. Published waypoints in the Albuquerque ARTCC.

waypoints within the Albuquerque Air Route Traffic Control Center (ARTCC). The figure depicts a solid blue disc of five nautical miles radius around each of the waypoints. It is evident that this ARTCC contains clusters of waypoints within which no point is more than five nautical miles from a published waypoint, and many other points that are far from the nearest published waypoint. The percentage of the ARTCC within the solid blue regions is the coverage provided by the published waypoints within that ARTCC for a five-nautical-mile radius. The percentage of the ARTCC covered at that radius is an approximate estimate of the probability that a randomly-projected waypoint of a new route can be replaced by a published waypoint no more than five nautical miles from the originally selected location of that waypoint. The percentage not covered is approximately the probability that such a waypoint will have to be “moved” more than five nautical miles in order to produce a route using only published waypoints.

Similar patterns arise elsewhere. Figure 6 illustrates the coverage of the published waypoints within each ARTCC for various radii around each waypoint. It is clear that in all but a few of the ARTCCs, there is a substantial probability that a randomly-projected waypoint will “miss” the nearest published waypoint by at least five nautical miles. Even at twenty nautical miles, there is still a substantial probability of a “miss” in several of the ARTCCs. Bearing in mind that a resolution route may include multiple new waypoints defined by arbitrary distances and/or radials from other waypoints, this chart suggests that the routes generated by the chromosomes of PBGA will often have to be altered significantly in order to convert them from routes between points at arbitrary latitudes and longitudes to routes through published waypoints.

A. Implications of Converting Routes to Published Waypoints

The use of published waypoints in TAP reduces the potential equipment costs and workload costs that might be incurred by using explicit latitudes and longitudes to describe route modifications. This practice, however, raises several questions:

- How should waypoints be “moved” to published locations? Is this movement likely to cause the route to violate one of the geometric constraints of the chromosome’s pattern?
- What is the computational cost (the additional elapsed time required to compute a strategic planning resolution) incurred by the requirement to replace arbitrary latitudes and longitudes by published waypoints?
- How much is the final outcome of a resolution (including costs such as flight time and fuel) likely to be affected by the requirement to use only published waypoints?

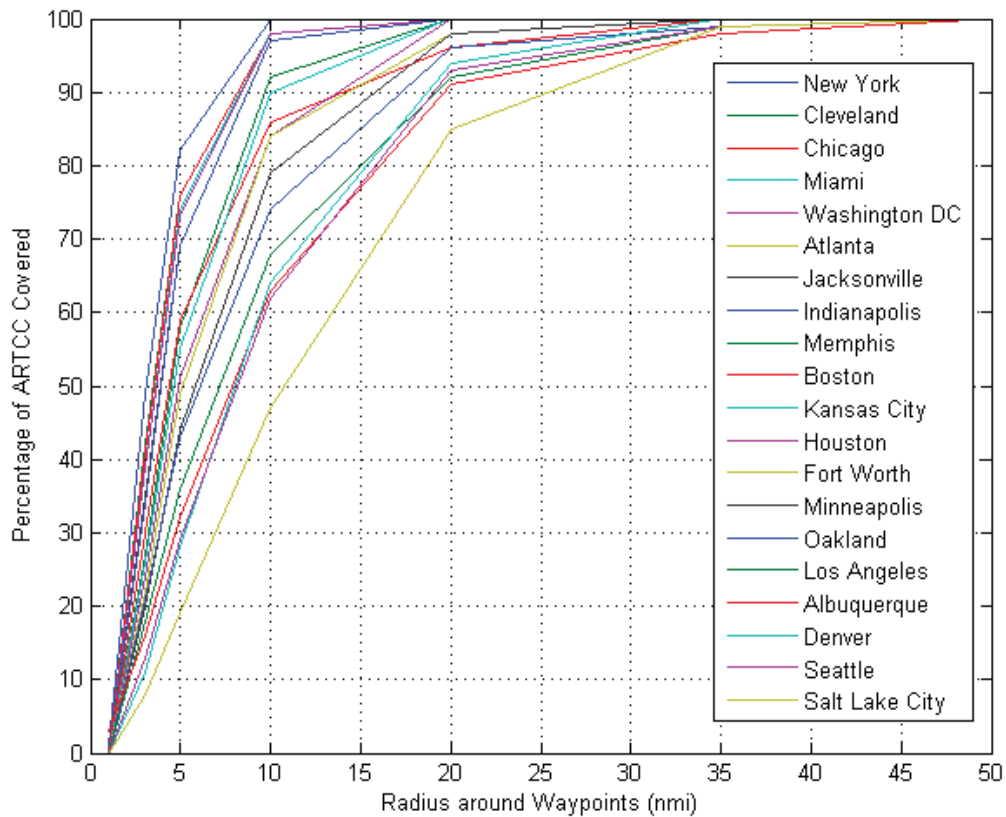


Figure 6. Coverage of ARTCC as a function of maximum distance to the waypoint.

Subsection B, below, discusses the technique for selecting published waypoints for route modifications. Section VI discusses the potential costs of doing so (computational costs as well as impact on the quality of the outcome) in more detail and evaluates those costs quantitatively.

B. Technique for Converting Routes to Published Waypoints

There are several choices to make when designing an algorithm to convert chromosomes' routes from arbitrary waypoints to published waypoints.

One possible implementation would first run PBGA to completion without restricting new waypoints to published locations, obtaining a resolution route whose waypoints are specified by explicit latitudes and longitudes. This implementation would then convert the resolution route to one that used only published waypoints, which is the route it would present to the crew.

This approach has some drawbacks. The optimal route using published waypoints may not be the one closest to the optimal route using arbitrary waypoints. In that case, since PBGA would attempt to optimize the route using waypoints at arbitrary latitudes and longitudes, it would be biased away from the optimal route that uses only published waypoints.

Moreover, the resulting route through published waypoints might not even be acceptable; it might cause conflicts or might violate geometric constraints of the pattern that produced it. Re-evaluation of the route after the conversion to published waypoints may enable such problems to be detected, but these problems are difficult to correct after PBGA has finished.

These drawbacks can be avoided by letting PBGA evaluate only routes that already pass through published waypoints. The maneuver patterns developed for TAP enforce this condition by “moving” each off-route waypoint to a published location as soon as the desired location of that waypoint (specified by a course and distance from another waypoint) has been determined. This means that the fitness assigned to each chromosome in PBGA is computed from the route through published waypoints that would actually result from that chromosome.

Moreover, the patterns take their own geometric constraints into account before “moving” the new waypoint. As most lateral maneuver patterns in AOP do, the One-Point and Two-Point Reconnect patterns, before setting the value of a genetic parameter, estimate the range of values of that parameter that permit the maneuver to satisfy all its geometric constraints. For example, the estimated upper bound of the magnitude of the turn-out angle, $|\theta|$, may be less than the previously-given upper bound, and the estimated lower bound may be greater than the previously-given lower bound. Once the turn-out angle has been selected, the lower bound of the stretch distance may be estimated to be larger than the previous bound, and an estimated upper bound is computed as well. The published waypoint is required to be at a location such that the resulting turn angle and distance are within these bounds.

The application of these rules to the off-route waypoint of the One-Point Reconnect Pattern is illustrated in Figure 7. The minimum and maximum turn angles and distances define the annular sector labeled “search space” in this figure. The waypoint selected by the pattern will be the waypoint within this search space that is closest to the projected waypoint (the waypoint at the turn angle and distance selected by the genetic parameters $|\theta|$ and d_m). Waypoints outside the search space are excluded, even if they are closer to the projected waypoint than any published waypoint within the search space. The objective of this exclusion is to minimize the probability that the resulting route will violate a geometric constraint.

The Two-Point Reconnect Pattern applies the same algorithm to select a published waypoint as its first off-route point. This procedure is applied before the genetic parameters of the second off-route point (α and d_r) are determined, because the estimated minimum and maximum values of the genetic parameters for the second off-route point may be affected by the exact location of the first off-route point.

These techniques are a logical extension of the techniques already used in other lateral patterns in AOP, which take in account of the locations of waypoints determined by higher-priority parameters before selecting values of lower-priority parameters.

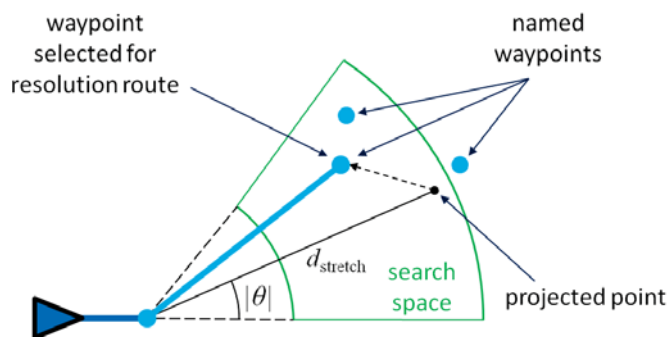


Figure 7. Converting genetic parameters to a published waypoint.

V. Hybrid Genetic and Deterministic Strategic Planning

Unlike previous maneuver patterns used in AOP, the Direct-to-Waypoint pattern applied to a typical route has only one possible instantiation for each waypoint that can be selected as the reconnection waypoint. Since the number of waypoints in a route in the ownship’s FMS typically will not be very large, it is practical to perform an exhaustive search over this portion of the solution space. An exhaustive search can easily be performed over all possible instantiations of the Altitude Change Pattern as well, since the number of possible new altitudes is limited by the maximum allowed altitude change.

The number of possible instantiations of the Hybrid Direct-to-Waypoint Pattern is the product of the number of reconnection waypoints that it can select and the number of new altitudes that it can select. This product is typically much larger than the number of choices for either parameter but still much smaller than the number of chromosomes examined in one population within a call to PBGA. Hence it is deemed preferable to perform an exhaustive search over this pattern rather than apply an evolutionary algorithm.

The implementation of PBGA has been extended to account for patterns with such characteristics. An initial population is formed by enumerating the discrete chromosomes deterministically (rather than initializing a population randomly). When TAP is allowed to produce other advisory waypoints prior to the reconnection waypoint, the best chromosome from each of the initial lateral and hybrid populations is then introduced into a new lateral or hybrid population, respectively, in which the remaining chromosomes are generated by the other permitted lateral or hybrid patterns. The resulting lateral and hybrid populations are then allowed to evolve according in the usual way in PBGA.

As long as no other chromosome is found in the lateral or hybrid population that is better than the deterministic chromosome that was introduced at the start of the population, PBGA will never eliminate

the deterministic chromosome from the population. This guarantees that the optimal deterministic resolution will be returned as the advisory unless a better resolution is found using other patterns.

For the vertical advisory, and for lateral and hybrid advisories that are not permitted to include waypoints prior to the reconnection waypoint, the second population of chromosomes is not required. Instead, the deterministically-selected chromosome is immediately returned as the advisory of that type.

VI. Evaluating the Cost of Using Published Waypoints

It is possible to make a quantitative comparison of the outcomes of the TAP maneuver patterns (which require advisories to use published waypoints) relative to the outcomes that would be obtained from a set of maneuver patterns (called the “alternative patterns” for the purpose of this discussion) that permit advisories to use unpublished waypoints but work similarly to the TAP patterns in all other respects. The potential costs of using the TAP patterns instead of the alternative patterns include the following:

- The elapsed time required to generate an advisory might be lengthened due to increased computation.
- The fitness of the advisory may be inferior to the advisory that would have been produced by the alternative patterns. In other words, TAP’s advisory might take more time and/or fuel than the alternative patterns’ advisory.
- The algorithm might be unable to produce any advisory in some circumstances where the alternative patterns would have produced an advisory. This can occur when the patterns fail to find any maneuvers that meet the criteria for acceptability.

In principle, all of these costs can be measured quantitatively.

These costs may apply to the One-Point, Hybrid One-Point, Two-Point, and Hybrid Two-Point Reconnect Patterns, since the obvious alternatives to those patterns would permit the off-route waypoints to be placed at non-published locations. Although the Direct-to-Waypoint and Hybrid Direct-to-Waypoint Patterns include a waypoint in their advisories, that waypoint is the reconnect waypoint selected from the existing route, and therefore does not incur any cost for substituting a published waypoint for an unpublished one. These costs also do not apply to the Altitude Change Pattern, since the advisory in that case is one of a finite number of possible new altitudes, and no waypoint is included in the advisory.

VII. Experimental Method

The costs of using published waypoints were evaluated by applying the maneuver patterns to a series of optimization requests and analyzing the results.

It was desirable that the optimization requests should result primarily in advisories using patterns that could be affected by the requirement to use published waypoints. The requests were generated by a preliminary run that simulated the flight of a TAP-equipped aircraft along a route within the custom-built scenario illustrated in Figure 8. In this scenario, the aircraft flew along the marked route at a cruise altitude of 36000 feet from the PPOS waypoint toward KDEN. The wind at 36000 feet was 11.85 knots toward the

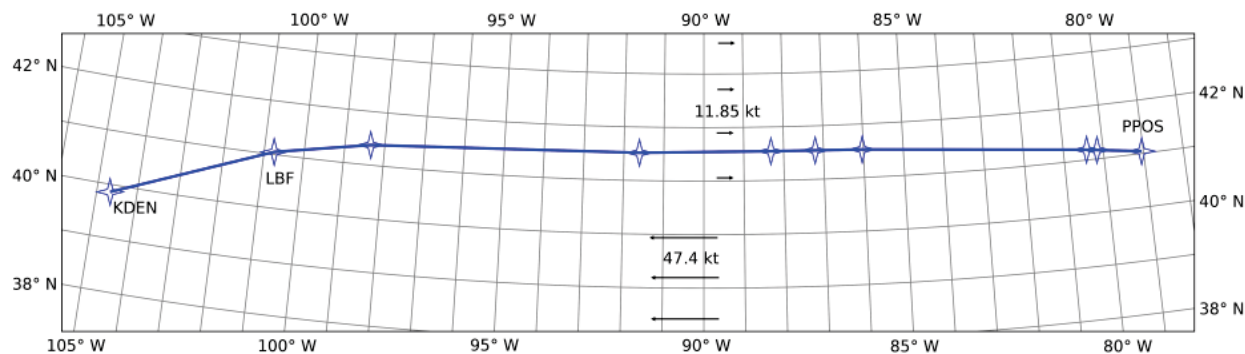


Figure 8. Evaluation scenario.

east (a headwind for this flight) at latitudes 41 deg north or higher, and 47.4 knots toward the west at latitudes 40 deg north or lower. Between the latitudes of 40 deg and 41 deg north, the wind transitioned from 11.85 knots easterly to 47.4 knots westerly.

The scenario included 145 additional aircraft, of which 13 cruised at or near 36000 feet, and 3 of which climbed through 36000 feet during the scenario. The remaining traffic are assumed not to have affected the fitness of lateral optimization; they would still cause computational time to be expended during evaluation of routes, but less than a similar number of aircraft at the ownship's cruise altitude would.

There were 36 optimization attempts altogether during the chosen scenario. The preliminary run of the scenario recorded the precise set of ownship, traffic, and atmospheric data that were provided to each request. The lateral optimization attempts resulted in numerous advisories based on the One-Point and Two-Point Reconnect Patterns.

The experiment used three different implementations of route optimization:

1. An implementation of the TAP optimization algorithm, using the maneuver patterns described earlier in this paper. For convenience in the discussion below, this is called the “always published” algorithm.
2. A modified implementation of TAP optimization that used an alternative set of lateral maneuver patterns in which waypoints of the advisory were not required to be at published locations. In the discussion below, this is called the “any coordinates” algorithm.
3. A modified implementation of TAP optimization that used the alternative set of lateral maneuver patterns to select a chromosome encoding the “winning” optimization route, but that then converted that chromosome to the corresponding TAP maneuver pattern, thereby forcing its off-route point(s) to be at published locations. In the discussion below, this is called the “convert to published” algorithm.

The third implementation did not take the naïve approach of simply moving each off-route point to the nearest published waypoint, but took the pattern's geometric constraints into account. For the Two-Point Reconnect Pattern, this implementation not only kept the first off-route point within certain geometric bounds, but adjusted those bounds for the second off-route point as needed to account for any effects of the change in the position of the first off-route point.

The experiment executed the series of 36 optimization requests in six different ways. Each of the three implementations was given the input data of the 36 requests exactly as they were recorded during the preliminary run, and performed the optimization algorithm to completion. This procedure was repeated for each implementation using the data from the preliminary run, but with all traffic aircraft removed (the “no traffic” scenario).

The optimization algorithms were executed on an otherwise unloaded Dell Optiplex 755 with an Intel® Core™ 2 Duo CPU E8300 with a clock rate of 2.83 GHz and with 2.96 GB of RAM. The algorithms were implemented in Visual C++ on Microsoft Windows XP. Because optimization was executed in “standalone” mode rather than on a thread of a fully operational execution of AOP, it was not interrupted by other processing performed by AOP, and therefore is assumed to have completed in less elapsed time than the same requests would take in the deployed system. Since the purpose of the experiment was to compare the *relative* performance of the three algorithms, however, and since the technique employed by the experiment allowed the conditions under which each algorithm was executed to be replicated almost exactly for each other algorithm, this was considered the preferred approach.

In all optimization requests, the algorithm was configured to minimize the amount of fuel burned. The results of each series of optimization attempts included the success or failure of each optimization attempt, the amount of fuel predicted to be saved by route optimization, the elapsed time, and other measurable outcomes.

VIII. Route-Optimization Outcomes Observed

In general, throughout the experiment the resulting optimization was either a One-Point Reconnect Pattern maneuver with an off-route point at almost exactly 40 degrees north latitude or a Two-Point Reconnect Pattern maneuver with a long leg between the off-route points, both of which were at about 40 degrees north latitude. These characteristics of the optimized routes may be partially attributable to the fact that the transition zone between the two main wind fields in this scenario ended at that latitude.

In the absence of traffic, the Two-Point maneuver was very rarely given as an advisory, although when a Two-Point advisory like the one described in the paragraph above was given, it provided much greater fuel savings than the previous or following One-Point advisory. The infrequency of Two-Point advisories may be due partially to a set of fitness penalties that implied that in order to select a Two-Point maneuver in preference to a One-Point maneuver, the fuel savings of the Two-Point maneuver had to be more than 75 pounds greater. It may also be due to slower convergence of the Two-Point Reconnect Pattern to its optimum value due to the pattern’s complexity.

In the presence of traffic aircraft, the Two-Point Reconnect Pattern was chosen with increased frequency (though still not as often as the One-Point pattern), and in several cases no optimized route was found. This result is consistent with the effect of traffic aircraft in other scenarios, since the routes that would otherwise have been favored by PBGA are often rejected due to conflicts with the traffic.

In all cases, fuel savings for later optimization requests tended to be less than for earlier requests, since the distance over which the optimization could be applied was steadily decreasing.

Table 3 summarizes the outcomes of the six series of attempts at optimization, aggregated over each series of attempts at optimization. The columns of this table show, respectively: the algorithm used; the percentage of optimization attempts that resulted in advisories saving more than 15 pounds of fuel; the number of times each pattern was selected for an advisory; and the mean and standard deviation of the difference between the predicted fuel burned by the advisory route vs. the existing route. The mean difference in fuel burned is negative when the advisory resulted in lower fuel consumption than the existing route.

Table 3. Aggregated results of route optimization by three algorithms

type of optimization	% success	advised maneuver		Δ fuel burn (lb)	
		One-Point	Two-Point	mean	S.D.
without traffic					
any coordinates	100.0	35	1	-471.609	141.8
convert to published	100.0	35	1	-454.923	142.589
always published	100.0	35	1	-467.442	129.063
with traffic					
any coordinates	63.89	18	5	-467.891	159.793
convert to published	63.89	18	5	-452.932	160.259
always published	66.67	20	4	-484.214	179.424

Table 4 summarizes the observed difference in optimized fuel burn when two different algorithms are used under the same conditions. To compile the data in this table, for each optimization request, the change in fuel burn resulting from the algorithm listed in the column of the table under the heading “control” was subtracted from the change in fuel burn resulting from the algorithm listed under the heading “test” when both algorithms produced advisories for that optimization request. Optimization requests where either or both of the algorithms failed to produce any fuel savings were excluded from this analysis. The figures under the heading “all pairs” show the number of matching pairs observed, the mean difference in fuel burn between the algorithms (in which a negative value indicates that the test algorithm obtained a better outcome than the control), the standard deviation of the difference, and a measure of statistical significance (which is described further in Section C). The figures under the heading “One-Point pattern only” are similarly derived, except that paired advisories where either or both advisories used the Two-Point Reconnect Pattern were excluded from the analysis, since the paired data seemed to be skewed in some cases by a small number of Two-Point maneuvers with large fuel savings.

A. Distance to the Published Waypoint

Table 5 summarizes the distances between the “projected” off-route waypoints of the maneuvers and the published waypoints that are used in the “convert to published” and “always published” algorithms. The

Table 4. Cost of using published waypoints for comparable optimizations

algorithms compared		Δ fuel burn from paired advisories (lb)							
		all pairs				One-Point pattern only			
test	control	<i>N</i>	mean	S.D.	<i>p</i>	<i>N</i>	mean	S.D.	<i>p</i>
without traffic									
any coordinates	convert to published	36	-16.686	22.202	< 0.001	35	-16.687	22.527	< 0.001
any coordinates	always published	36	-4.167	58.566	0.566	34	-5.72	40.36	0.5
always published	convert to published	36	-12.519	58.035	0.066	34	-11.317	41.337	0.105
with traffic									
any coordinates	convert to published	23	-14.959	16.181	< 0.001	18	-14.892	14.141	< 0.001
any coordinates	always published	23	29.778	111.196	0.568	16	-9.168	43.317	0.105
always published	convert to published	23	-44.737	111.907	0.061	16	-6.989	47.128	0.402

Table 5. Distance between genetically projected and published waypoints

algorithm	distance to published waypoint (nmi)					
	first off-route point			second off-route point		
	<i>N</i>	mean	S.D.	<i>N</i>	mean	S.D.
without traffic						
convert to published	36	5.893	4.592	1	10.016	—
always published (advisory)	36	6.48	3.091	1	6.993	—
always published (all routes)	17690	6.292	4.48	1369	8.074	4.703
with traffic						
convert to published	29	4.639	2.475	9	7.331	2.68
always published (advisory)	29	5.992	4.123	5	7.252	3.254
always published (all routes)	19600	6.396	5.549	7289	8.081	4.816

data for the off-route point of the One-Point Reconnect Pattern are included under the heading “first off-route point.”

The data for “always published (advisory)” describe the distances that apply only to the chromosome that is finally selected as an advisory, whereas “always published (all routes)” refers to the distances between projected and published waypoints for all chromosomes generated by the TAP algorithm for which published waypoints were found. (This excludes some chromosomes whose geometries were found to be unacceptable before the off-route point(s) were replaced by published points.) In the “convert to published” algorithm, of course, distances to published waypoints are defined only for advisory routes.

As can be seen from Table 5, the requirement to use published waypoints often causes waypoints to be placed several miles from where the “any coordinates” patterns would have placed them. While the off-route

points of advisory routes are found only in a relatively small portion of the airspace near the original route, the off-route points of other routes generated during PBGA, which cover a much larger portion of nearby airspace, do not appear to result in much greater or smaller distances to their corresponding published waypoints.

B. Comparison of Estimated Costs

Using the figures in Tables 3 and 4 to estimate the cost of replacing the “any coordinates” algorithm with either of the other two algorithms, it is apparent that the cost of using the “convert to published” algorithm (comparing it to the “any coordinates” algorithm) is much higher than the cost of the “always published” algorithm. Likewise it appears that a noticeable cost would be incurred by switching from the “always published” algorithm to the “convert to published” algorithm. This supports the decision to implement the “always published” algorithm for TAP. Curiously, when optimizations were attempted in the presence of traffic, the “always published” algorithm appeared to perform better than either of the other two, finding more advisories with a larger mean fuel savings. The apparent advantage over the “any coordinates” algorithm disappears, however, when only pairs of One-Point advisories are considered.

There was no obvious difference in the ability of any of the three algorithms to find optimization routes. The TAP algorithm found an optimized route for one request (including traffic data) for which the other two algorithms found no optimization; the fuel savings for this optimization, however, were much smaller than for either the previous or following successful optimization. In all other cases, an optimized route was found by all three algorithms or by none of them. There was no evidence that the substitution of published waypoints for arbitrary waypoints in the “convert to published” algorithm caused acceptable optimized routes to become unacceptable.

C. Statistical Significance

Because the fuel savings in each series of advisories were trending toward zero, it is not reasonable to assume that these values are identically distributed, and Student’s t -test is not applicable. The Fisher sign test, however, indicates that the difference between the “any coordinates” algorithm and the “convert to published” algorithm was statistically significant ($p < 0.001$) with or without traffic, whether the test considered all paired advisories or only pairs of One-Point advisories. The Fisher sign test also indicated that the advantage of the “always published” algorithm over “convert to published” was significant at the $p < 0.1$ level when all paired advisories were considered, either in the presence or in the absence of traffic. The p -values of this test are listed in Table 4.

IX. Computational Cost Observed

The implementation of PBGA in AOP is instrumented to record the total time elapsed during an attempted route resolution. This instrumentation was enabled during the experiment.

The measured elapsed time included the time required to set up the resolution algorithm and to record certain data in a file during the resolution, in addition to the time taken by the functions of the maneuver patterns themselves. The recording of the distances between genetically selected locations and published waypoints (in order to compute the figures in Table 5) greatly increased the amount of data written to files, so this capability was disabled and each series of optimization attempts was re-run in order to record elapsed times.

The resulting elapsed times are summarized in Table 6. The “relative mean” shows each algorithm’s mean elapsed time as a multiple of the mean elapsed time of the “any coordinates” algorithm (which does not replace any of its maneuvers’ waypoints with published waypoints). As is evident from these results, the elapsed time for the “convert to published” algorithm is nearly identical to that for the “any coordinates” algorithm, as would be expected. The elapsed time for the “always published” algorithm is longer, but only by about 15 to 20 percent. Moreover, there are opportunities to reduce these elapsed times by further optimizing the implementation of the algorithm for finding published waypoints.

Table 6. Elapsed time of route optimization algorithms

algorithm	elapsed time of optimization requests (sec)					
	without traffic			with traffic		
	mean	S.D.	relative mean	mean	S.D.	relative mean
any coordinates	3.107	0.131	1.0	3.765	0.543	1.0
convert to published	3.147	0.164	1.013	3.753	0.544	0.997
always published	3.722	0.503	1.198	4.352	1.063	1.156

X. Conclusion

Due to the limited scope of the data collected (consisting of 36 distinct “optimization problems” all selected from a single artificial scenario), the quantitative analysis in the preceding text may be considered a pilot study for possible future work. For the cases studied, however, there was little or no apparent loss of fuel savings due to using published waypoints in the manner implemented in the TAP maneuver patterns.

The algorithm for forcing TAP advisories to use published waypoints appeared to incur a small but measurable cost in processing time relative to other possible methods of obtaining advisories with published waypoints. The processing cost observed in this experiment may be reduced in the future, but even at the present level an argument can be made that the processing cost is justified by the improvement in the quality of outcomes.

Acknowledgment

The authors thank Jeff Henderson of the Engility Corporation for calculating and plotting the data in Figures 5 and 6.

References

- ¹Wing, D. J. and Cotton, W. B., “Autonomous Flight Rules: A Concept for Self-Separation in U. S. Domestic Airspace,” Technical Publication NASA/TP-2011-217174, NASA, Nov. 2011.
- ²Karr, D. A., Vivona, R. A., Roscoe, D. A., DePascale, S. M., and Wing, D. J., “Autonomous Operations Planner: A Flexible Platform for Research in Flight-Deck Support for Airborne Self-Separation,” *AIAA Aviation Technology, Integration, and Operations Conference*, AIAA-2012-5417, AIAA, Washington, DC, 2012.
- ³Ballin, M. G. and Wing, D. J., “Traffic Aware Strategic Aircrew Requests (TASAR),” *AIAA Aviation Technology, Integration, and Operations Conference*, AIAA-2012-5623, AIAA, Washington, DC, 2012.
- ⁴Woods, S., Vivona, R., Roscoe, D., Lefebvre, B., Wing, D., and Ballin, M., “A Cockpit-based Application for Traffic Aware Trajectory Optimization,” *AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 2013.
- ⁵Vivona, R. A., Karr, D. A., and Roscoe, D. A., “Pattern-Based Genetic Algorithm for Airborne Conflict Resolution,” *AIAA Guidance, Navigation, and Control Conference*, AIAA-2006-6060, AIAA, Washington, DC, 2006.
- ⁶Karr, D. A., Vivona, R. A., Roscoe, D. A., DePascale, S. M., and Consiglio, M., “Experimental Performance of a Genetic Algorithm for Airborne Strategic Conflict Resolution,” *AIAA Guidance, Navigation, and Control Conference*, AIAA-2009-5748, AIAA, Washington, DC, 2009.