Ninth USA/Europe Air Traffic Management Research and Development Seminar (ATM2011)

Dynamically Generating Operationally-Acceptable Route Alternatives Using Simulated Annealing

Christine Taylor and Craig Wanke
Center for Advanced Aviation System Development
The MITRE Corporation
McLean, VA, USA
ctaylor@mitre.org, cwanke@mitre.org

***Abstract*—This paper presents a Simulated Annealing methodology for defining operationally-acceptable route alternatives for flights impacted by weather. By dynamically generating route alternatives that inherently possess traits amenable to traffic managers and users, more efficient use of the airspace can be realized. This paper explores the use of Simulated Annealing to provide quality solutions quickly, and to capture additional route alternative options, such as ground delay. For comparison, a k-shortest path approach and an ad-hoc heuristic search approach have also been employed to generate reroutes and the results show that Simulated Annealing indeed provides competitive alternatives to the k-shortest path approach and improved alternatives over the heuristic search procedure. Furthermore, Simulated Annealing can potentially generate these alternatives with less computation effort than the k-shortest path approach and therefore, represents a desirable alternate flight option generation method.**

Keywords- Traffic flow management; decision support; dynamic route generation; weather avoidance

I. INTRODUCTION

Managing congestion, especially during weather events, requires improved methods for assisting decision makers and increasing operational efficiency. Hazardous weather requires traffic managers to reroute flights that plan to pass through the weather, but have limited decision support capabilities to assist in this process. Given the difficulty with manually generating flight-specific reroutes that are operationally-acceptable, National Playbook routes[1] are often used to reroute entire flows of traffic. However, these reroutes are not available for all flights and may not provide alternatives that avoid the weather. As the need to maximize all available airspace capacity is imperative, it is necessary to widen the set of operationally-acceptable reroutes provided to decision makers by defining a real-time decision support capability that can generate flight and weather-event specific operationally-acceptable reroutes.

Including the capability to dynamically generate reroutes provides a larger solution space, but the additional computation expense can be significant. The approach discussed in [2] provides detailed reroutes based on dynamic propagation of weather and trajectory information; however the resulting problem is computationally-intractable and must be simplified to only consider distance as a metric, thereby omitting other traffic management concerns. Other previous research[3],[4],[5] has investigated the definition of reroutes using a static network, derived by overlaying a grid on the area of consideration. Although the reroute generation defined by this network is computationally-efficient, the reroutes themselves may not be operationally-feasible as only distance and weather avoidance are considered, thereby resulting in reroutes that do not necessarily conform to current operational practice.

References [5] and [6] explore the design of a more operationally-feasible network by defining the network components from existing airspace routing structures, but this investigation was limited in scope to the terminal area. Reference [7] examines the extension of this approach to the en-route airspace by defining networks derived from Coded Departure Routes (CDRs) to better respond to the current weather conditions. This approach is limited, since CDRs do not exist for every airport pair, and the methodology does not extend to flights already en route.

Terrestrial rerouting research [8] inspires a compromise approach, as the network is derived from existing roadways and therefore a reroute that utilizes these roadways is operationally-feasible. Extending this methodology to air traffic rerouting would imply that a previously-flown segment can be considered as a component of a feasible reroute, even if it is not normally utilized for traffic between the origin and destination pair of the flight.

The research presented in this paper is part of a greater research effort to bridge this divide by defining operationally-acceptable route alternatives in real time to effectively aid decision makers in managing air traffic. The proposed approach for generating reroutes is based on the definition of route segments derived from historically-flown connections between existing fixes[9] and evaluated using metrics of operational acceptability, derived from analyses based upon subject matter expert advice[10]. Reference [9] presents a heuristic approach for generating reroutes, and details the collection requirements for these fix-pair segments as well as defines an ad-hoc method for generating reroutes. References [11] and [12] describe the construction of a network of these fix-pair segments and employ a k-shortest path (KSP) algorithm to generate multiple reroute options that best capture the operational acceptability metrics. In addition, [12] shows that the reroutes generated from the KSP approach provide more operationally-acceptable options than the reroutes generated from the heuristic approach defined in [9], albeit with significant computation effort. Reference [13] corroborates this finding for a multi-flight problem.

To improve the computational performance associated with dynamically generating flight specific reroutes, the research presented in this paper employs a Simulated Annealing[14] (SA) methodology to generate route alternatives. By using SA to define alternatives consisting of both reroutes and departure delays, and to directly evaluate the operational acceptability of the alternatives, different and potentially better solutions can be generated than through the heuristic search or the KSP approaches. Furthermore, as a heuristic optimization approach, computation effort required for SA can be significantly less than that required for the KSP approach.

This paper is organized as follows. Section II provides a description of the operational acceptability metrics considered. Section III presents the problem formulation and the description of the SA implementation used. Section IV presents a set of rerouting problems, and compares the reroute alternatives generated using the heuristic search and KSP approach, which are briefly described, to the route alternatives generated using SA. Section V presents the conclusions drawn from this research and the ongoing work in this area.

II. Defining Operational Acceptability

The goal of this research is to dynamically define operationally-acceptable flight-specific alternatives for decision makers. However, defining operational acceptability can be challenging. The approach described in detail in [10] involves the extraction of the essence of quality route design, as understood by subject matter experts[1] (SMEs), into quantifiable and generic evaluation metrics or constraints. The remainder of this section provides a description of the operational acceptability metrics included as evaluation criteria in this research.

A. Route Distance

The most frequently-considered metric of reroute quality is the distance of the reroute as compared to the original route. In this paper, we define the distance of a route to be the sum of the distances between each consecutive pair of fixes. The distance of a fix pair can be interpreted as the air distance, assuming both the wind and aircraft velocity profiles are provided; however, for the purposes of this research, we use the great circle path distance between consecutive fix pairs in the route.

B. Origin-destination flow factor

The flow conformance of a route is a measure of how consistent the route is with historical routing. Although all fix-pairs defined in the network were historically-flown, these fix-pairs may not have been historically-used by flights traveling between a particular pair of regions. As such, we define the origin-destination (O-D) flow factor of a reroute to be the sum of the O-D flow factors on each fix-pair in the reroute. The O-D flow factor for each fix-pair was derived by an analysis presented in [10] and can be summarized as follows.

[1] The subject matter experts queried for this analysis consisted of former traffic managers, air traffic controllers, and airline dispatchers.

Approximately 4000 airports were grouped into 35 geographically-distinct regions, and for each of these region pairs, the historical usage of each fix-pair segment was analyzed. The O-D flow factor assigned to each fix-pair segment is not a count of usage, but a relative comparison of usage ranging between high usage (O-D flow factor approaching zero) and almost no usage (O-D flow factor approaching one). All fix-pair segments not used between a region pair are assigned an O-D flow factor of one. The region pair for a given flight is determined by the flight's departure and arrival airports, which in turn determine the O-D flow factors of the fix-pair segments.

C. Route Blockage

Route blockage measures the forecasted usability of each fix-pair in the reroute to quantify how likely a given reroute is to be weather impacted. The blockage of a fix pair segment is determined using the methodology defined in [15] which can be briefly summarized as follows. A grid of 1km by 1km cells is overlaid on a fix-pair segment and the blockage of each grid cell is forecasted using the Convective Weather Avoidance Model (CWAM)[18] developed by MIT and Lincoln Labs. CWAM determines the flight altitude below which pilots would likely deviate around the cell, based on precipitation and echo top forecasts from the Corridor Integrated Weather System. The "minimum traversable altitude" is computed for each grid cell, for 15-minute time bins over a look-ahead time (LAT) of two hours.

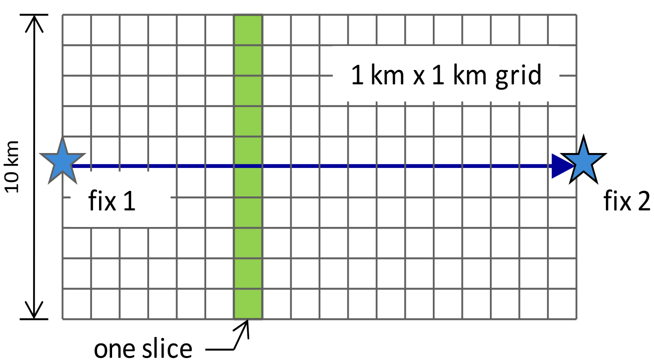

Figure 1. Geometry for Route Blockage Computation

For each LAT, the minimum clear altitude for each cross-section slice of the grid is computed, where it is assumed that 10km of contiguous cells are needed to denote a passable cross-section. Figure 1 illustrates this concept. The clear altitude for the fix-pair is simply the maximum clear altitude for each cross-section. A fix-pair is considered blocked if the flight altitude is less than the clear altitude for the segment during the LAT the flight will traverse the fix-pair; otherwise the fix-pair is considered open and weather-free.

D. Lateral Deviation

The lateral deviation of a reroute is a measure of how differently the reroute will impact sectors as compared to the original route and thus serves as a metric for coordination complexity. To define lateral deviation mathematically, we begin by computing the maximum lateral distance, or cross-track between the two routes. This distance is then scaled using a linear regression, described in [10], to define a zero-one scale metric where a score approaching zero is assigned to reroutes with small lateral deviations and a score approaching one is assigned to reroutes with large lateral deviations.

E. Global Flow Conformance

When large severe weather systems occur, greater deviation from the original route may be required to define a good reroute that avoids the weather. In these cases, the O-D flow conformance may not provide a useful measure of the route's flow conformance, as many of the possibly-useful arcs may never have been flown between the region pair of interest. However as we still prefer to employ reroutes that contain highly-utilized segments, we define the global flow conformance of a reroute to be the sum of the usage fractions on the fix-pairs in the reroute, where the usage fraction of a fix-pair is the usage count of that fix-pair divided by the total usage of all fix-pairs in the network.

F. Sector Congestion

Measuring sector congestion enables reroutes to be evaluated based on their overall impact on the National Airspace System (NAS) performance. The congestion metric determines the maximum probability of sector congestion incurred if the given reroute is utilized. As discussed in [15], forecasted estimates of demand can be used to define estimates of sector congestion by estimating the probability that the demand will exceed the capacity of the sector for a given time period. Using the Monitor/Alert parameter as an estimate of sector capacity, the estimated sector entry times for a reroute are computed to determine the maximum congestion probability incurred for the reroute.

G. Airline Schedule Disruption

The airline schedule disruption metric provides a non-linear evaluation on the impact of arrival time delay incurred by a route alternative. Arrival delay is a commonly-used measure of impact on flight operators, but fails to capture some important effects. First, small delays are unlikely to have much impact on schedules, because schedules are normally built with some padding to absorb small delays. Large delays can cause major impacts, such as missed connections for passengers or freight, and can propagate to later flights which use the same aircraft, flight crew, or cabin crew.

To compute the airline schedule disruption metric, delay propagation multipliers were taken from [16]. In that study, multipliers were given as a function of departure delay and time of day which reflects that if a flight is delayed early in the day, the opportunity for delay propagation is greater than for those delayed later in the day. Next, using Airline Service Quality Performance (ASQP) data for 2008, delay per flight and time of take-off was assessed. For each flight, a table-lookup from the cited source yielded a delay multiplier. For each month of 2008, a distribution of multipliers was constructed. In order to map these values into the desired [0-1] range, the distribution was sorted, and the cumulative probability distribution function (CDF) was created, yielding a value in the range of [0-1]. CDFs for each of the 12 months in 2008 were constructed and July was selected, via visual assessment, as a representative CDF.

H. ATC Facility Cost

The ATC facility cost represents the number of facility changes required by a given reroute. As facility changes incur increased workload for controllers, it is desirable to keep these at the minimum necessary to reach the destination point.

I. ATC Point-Out Cost

A point-out is required if a reroute transits a sector for less than 120 seconds, which increases the workload for controllers and is therefore undesirable. As such, we seek to minimize the number of point-outs involved in a reroute.

III. GENERATING OPERATIONALLY-ACCEPTABLE SOLUTIONS WITH SIMULATED ANNEALING

This research investigates the performance of Simulated Annealing for generating operationally-acceptable flight specific route alternatives. The generation of a single or multiple routes through a network can be accomplished by a variety of solution methods, each yielding different performance in reroute generation and computation effort. Dijkstra's Algorithm[17] is the classic means of defining a path through a network, as shown in [3]. Dynamic Programming, a related technique, is used by [18] for path construction. An A* search method is implemented to construct the paths defined by the Flow-Based-Route-Planner, described in [6], and utilized in [5] and [7]. Modifications to the A* search that include a heuristic estimation function to speed the shortest path search are employed by [4] and [19]. Reference [8] uses a heuristic method known as multi-agent systems to search and update path performance in dynamic networks.

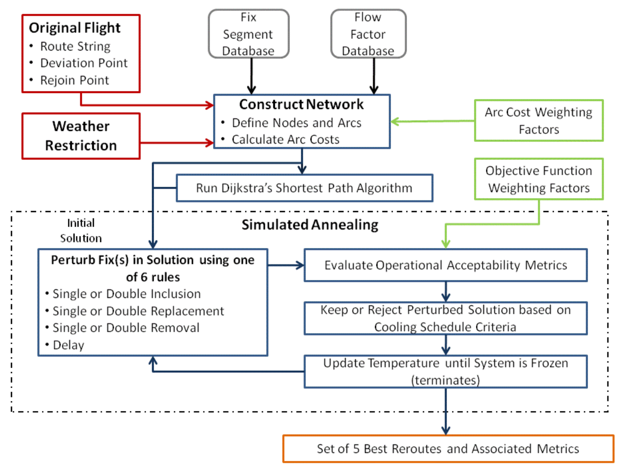

Figure 2. Flow Diagram for Computation of Operationally-Acceptable Reroutes Using SA

SA is an appealing alternative to traditional optimization approaches as it provides a computationally-efficient methodology for globally optimizing the routing problem, though no guarantees are made on optimality. Furthermore, given the flexibility inherent in heuristic optimization procedures, additional routing options such as ground delay can be easily added into the decision space for generating route alternatives. Figure 2 shows the overall flow of the route generation methodology employed by SA and the remainder of this section describes each component of Figure 2.

A. Problem Formulation

The dynamic routing problem considered in this research begins with the identification of a flight either pre-departure or en-route to the destination airport that must deviate from its route because of a weather event. As the flight can be anywhere between the departure airport and the arrival airport when a reroute is initiated, the deviation point and the rejoin point of the original route must be specified. The deviation point is the fix along the original route where the reroute can begin, which is any fix including the departure airport (when the flight is pre-departure), that occurs before the weather intersects the original route. Similarly the rejoin point is the final fix along the original route where the reroute reconnects to the original route and can be any fix along the original route after the weather event, including the destination airport.

B. Defining the Flight-Specific Network

Reroutes are generated as paths through a network that is constructed for the specific flight under consideration and the network is derived from a database of historically-flown fix-pair segments[9]. As such, the nodes of the network comprise the set of fixes identified in the historical fix-pair segment database and the connections between these nodes, or directed arcs, are taken from the connections that have previously been flown.

The fix segment database is defined for the entire NAS, but for a given flight, only a subset of these segments is useful in defining the network. As such, the search area of the network is scoped by an ellipse containing the deviation and rejoin points of the reroute. Specifically, the semi-major axis of the ellipse is defined by the distance between these two points plus a buffer distance added to each point to ensure that all feasible and desirable connections are included. The buffer distance of the deviation point is 25 nm if the flight is en-route, and 100 nm if the flight is pre-departure. The buffer distance of the rejoin point is 100 nm. The semi-minor axis of the ellipse is defined as the maximum of half the semi-major axis or 100 nm greater than the maximum lateral distance of the original route from the great circle connecting the origin and destination airports.

C. Generating an Initial Solution

As shown in Figure 2, an initial feasible solution is provided to SA through the implementation of a Dijkstra's[17] shortest path algorithm, which requires that costs be computed for each arc in the network. Based upon previous research[12], we define three operational acceptability metrics to represent the arc cost components, namely distance, O-D flow factor and route blockage.

The distance of an arc (d_{ij}) is defined as the great circle distance from the starting fix (node i) to the end fix (node j). The distance cost of an arc from node i to node j is normalized by the total distance of the original route from the deviation to the rejoin points (d^R), as shown in Equation 1.

$$c_{i,j}^d = \frac{d_{i,j}}{d^R} \quad (1)$$

The flow factor of an arc represents its usage when travelling between the region pair of the flight, as described in Section II. The flow cost of the arc ($c_{i,j}^f$) is defined as the flow factor of the arc from node i to node j between the departure airport cluster and the arrival airport cluster ($f_{i,j}^{DA}$) multiplied by the normalized distance of the arc ($c_{i,j}^d$), as shown in Equation 2.

$$c_{i,j}^f = f_{i,j}^{DA} * c_{i,j}^d \quad (2)$$

The flow factor is scaled in this manner to emphasize that the longer the arc, the more important it is for the arc to have a low flow factor.

Defining the route blockage of an arc is similar to the methodology described in Section II with the exception that, for an individual arc, the current flight altitude and time of arrival to the fix-pair is unknown. As such, we estimate the flight altitude by assuming (1) the cruise altitude, if the fix-pair is outside the transition radius of either airport, or (2) zero ft, if within the transition radius. The transition radius is defined as the distance it takes a flight to reach cruise altitude, which is assumed as 3 miles for every 1000 ft of cruise altitude. If the flight altitude is above the minimum clear altitude, the segment is unblocked during this time period; otherwise the segment is considered blocked.

Specifically, we define the minimum cleared altitude of a segment at time t as $h_{i,j}^t$. For each time t, we determine if the segment is considered blocked by comparing the cleared altitude to the flight altitude, where

$$b_{i,j}^t = \begin{cases} 1, & h_{i,j}^t \le h^f \\ 0, & h_{i,j}^t > h^f \end{cases} \quad (3)$$

where h^f is the flight altitude assumed for the segment.

The blockage cost of the arc, as expressed in Equation 4, is defined using a weighted average of blockage values from the time bin containing the earliest possible arrival time (t^e) to a two hour LAT (T)

$$c_{i,j}^b = \frac{\sum_{t \ge t^e} f_b^{\mathrm{T}-t+1} * b_{i,j}^t}{\sum_{t \ge t^e} f_b^{\mathrm{T}-t+1}} \quad (4)$$

where the blockage weighting factors are defined as $f_b^t = [1, 1, 1, 1, 0.5, 0.5, 0.5, 0.5]$ to provide a greater emphasis on blockage values closer to the earliest arrival time to the arc. The earliest arrival time is simply estimated using the cruise velocity and the distance from the original deviation point of the route to the fix-pair. If the earliest possible arrival time is later than two hours, no blockage information is available and the segment is considered unblocked.

The total cost of an arc from node i to node j is then represented as

$$c_{i,j} = k_d * c_{i,j}^d + k_f * c_{i,j}^f + k_b * c_{i,j}^b \quad (5)$$

where k_d, k_f, and k_b are the weighting factors for arc distance cost, arc flow factor cost, and arc blockage cost, respectively.

D. Perturbing the Solution

SA explores the search space by perturbing the current alternative using only existing network connections in order to maintain feasibility. As the reroute is defined by an ordered fix list, we consider three actions to modify this list: including fixes, replacing fixes, or removing fixes. Furthermore, we allow either one or two consecutive fixes to be modified in a single perturbation. For pre-departure flights, we consider a seventh perturbation option that alters the assigned departure delay for the current reroute. In a given iteration, one perturbation option is chosen randomly, from a uniform distribution, and the perturbed reroute can only be considered if it does not alter the first and last nodes in the reroute.

1) Perturbation by Inclusion

Perturbing the current reroute by single inclusion or double inclusion adds one or two fixes to the route string, respectively. For a single inclusion, the first step is to select, uniformly at random, the fix in the current reroute, denoted p_i, that will immediately proceed the first included fix We then identify the set of fixes (S_1) that p_i connects to and the set of fixes (S_2) that connect to the *i+1*th fix in the current reroute (p_{i+1}). Therefore S_1 and S_2 are defined as

$$S_1 = \{j_1, \dots, j_n\}\ s.t\ A(p_i, j_i) = 1 \quad (6)$$

$$S_2 = \{j_1, \dots, j_n\}\ s.t\ A(j_i, p_{i+1}) = 1$$

where A is the adjacency matrix of the network, and $A(i,j) = 1$ defines that a connection exists from node *i* to node *j*. The set of nodes eligible for inclusion into the reroute (S) is defined by the intersection of S_1 and S_2. If there exists more than one fix in S, the included fix will be selected uniformly at random. For the inclusion of two fixes, this process is simply extended to search for two connecting fixes.

2) Perturbation by Replacement

Perturbing the current reroute by single replacement or double replacement exchanges one or two fixes, respectively. For a single replacement, the first step is to select, uniformly at random, the fix in the current reroute to be replaced (p_i). We then define the set of fixes (S_1) that p_{i-1} connects to and the set of fixes (S_2) that connect to p_{i+1}. Therefore S_1 and S_2 are defined as

$$S_1 = \{j_1, \dots, j_n\}\ s.t\ A(p_{i-1}, j_i) = 1 \quad (7)$$

$$S_2 = \{j_1, \dots, j_n\}\ s.t\ A(j_i, p_{i+1}) = 1$$

The set of nodes eligible for replacement (S) into the reroute is defined by the intersection of S_1 and S_2. If after removal p_i from S, the resulting set contains more than one fix, the replacement fix will be selected uniformly at random. For the replacement of two fixes, this process is simply extended to search for two connecting fixes.

3) Perturbation by Removal

Perturbation by removal, which removes a single or two consecutive fixes from the current reroute. For a single removal, the first step is to select, uniformly at random, the fix for removal, which we denote as p_i. We then determine if the path is feasible without this fix, by ensuring $A(p_{i-1}, p_{i+1}) = 1$. For a double removal, we evaluate if $A(p_{i-1}, p_{i+2}) = 1$.

4) Perturbation by Delay

For pre-departure flights, an additional perturbation option is offered that randomly increases or decreases the delay of the flight by five minutes.

5) Constraint Checking

Once a perturbed reroute is generated, it is evaluated against a set of operational constraints to determine if it is a feasible alternative. The first constraint eliminates any reroutes that have cycles, or repeated nodes in the path. The second constraint imposed is a turn-angle constraint that prohibits reroutes with large turn angles, as these paths are often not operationally-feasible, and are highly undesirable. Specifically, for every fix-pair connection in the reroute, we evaluate the change in heading between two subsequent connections to ensure that the absolute angle change is less than the maximum angle change permitted.

E. Evaluating Operational Acceptability

To determine the quality of the perturbed reroute, a multi-metric path objective function is defined from the operational-acceptability metrics discussed in Section II. The distance metric (D^p) is defined as the accumulated scaled arc distance cost from the deviation point to the rejoin point as shown in Equation 8

$$D^p = \sum_{(i,j)\in p} c_{i,j}^d \quad (8)$$

where, the arc distance cost $c_{i,j}^d$ is as defined in Equation 1 and p is the reroute defined by the set of arcs.

The O-D flow factor metric (F^p) is the accumulated distance-weighted flow factor defined in Equation 9

$$F^p = \sum_{(i,j)\in p} c_{i,j}^f \quad (9)$$

where $c_{i,j}^f$is the O-D arc cost defined in Equation 2.

Unlike the route blockage calculation for the arc costs, we know precisely which arcs are blocked in the reroute, as these can be determined by analyzing the resulting trajectory of the reroute. The route blockage metric (B^p) is the number of blocked arcs in the reroute, as defined in Equation 10.

$$B^p = \sum_{(i,j)\in p} b_{i,j} \quad (10)$$

The lateral deviation (L^p) of the reroute is the fourth metric considered. The computation of the reroute lateral deviation and translation into the zero-one scale is as described in Section II.

The global flow conformance (G^p) is the distance-weighted usage fraction of the arc within the network. If we define the usage of the arc as $u_{i,j}$, where usage is computed as frequency of use for the historical time period evaluated in [9], then the global flow conformance can be defined as shown in Equation 11.

$$G^p = \sum_{(i,j)\in p} \left(1 - \frac{u_{i,j}}{u_{max}}\right) * c_{i,j}^d \quad (11)$$

Here, $c_{i,j}^d$ is the arc distance cost and u_{max} is the maximum usage factor in the fix-pair database.

The sector congestion metric (S^p) provides the maximum probability of congestion over the entire reroute. The airline schedule disruption metric (A^p) provides the non-linear impact of arrival delay.

The two workload metrics, ATC facility crossing (X^p) and ATC point-out (O^p) provide the final two metrics considered. Specifically, the ATC facility crossing metric is simply the number of facility crossings required for a reroute divided by five, to better scale with the remaining objectives. The ATC point-out metric simply counts the number of point-outs required for a reroute, given all sector transits that are less than 120 seconds.

Combining these performance metrics into an overall objective function for the reroute operational acceptability yields the expression in Equation 12

$$C^p = w_d * D^p + w_f * F^p + w_b * B^p + w_l * L^p + w_g * G^p + w_s * S^p + w_a * A^p + w_x * X^p + w_o * O^p \quad (12)$$

where w_d, w_f, w_b, w_l, and w_g are the relative weighting factors for reroute distance, weighted average O-D flow factor, route blockage, scaled lateral deviation, and global flow factor, respectively.

F. Optimization Using Simulated Annealing

Simulated Annealing[14] is a heuristic optimization approach inspired by the metallurgical process of annealing, used to change the crystalline structure of metals to improve their working properties. By simulating the annealing process, SA is first able to search broadly (when the system is still heated), by moving to areas of the design space even if initially less optimal, but then search more locally (when the system is cooler), aiming to successively improve the current design, until the termination criteria is reached and the system is deemed frozen or cooled. The best answer found during the search is returned, regardless of the final design configuration. Figure 3 illustrates this process.

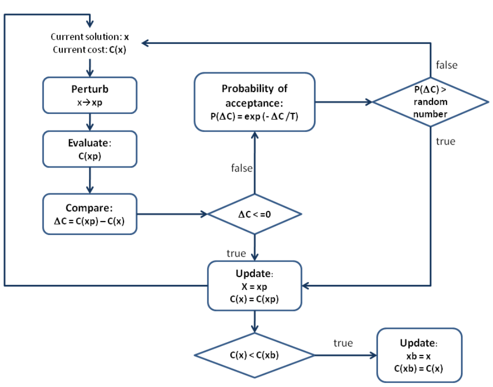

Figure 3. Flow Diagram of SA Design Search Process

As with most heuristic optimization approaches, the progression of SA is guided by a number of user-defined parameters, specifically the initial temperature (T_o), temperature increment (ΔT), the cooling schedule, equilibrium condition (neq), and frozen condition (nfz). The initial temperature is calculated from the objective function value of the initial solution (p_o), as shown in Equation 13

$$e^{\frac{C^{p_o}}{T_o}} > 0.99 \quad (13)$$

where C^{p_o} is the corresponding objective function value of the initial solution. The temperature increment determines how the temperature is decreased once an equilibrium state is reached. The cooling schedule determines how this temperature decrement is applied. In this research, we choose an exponential cooling schedule, where the temperature of a given step (k) is defined as shown Equation 14.

$$T_k = \Delta T^k * T_o \quad (14)$$

The equilibrium and frozen conditions define the number of attempts at a given temperature to reach equilibrium and the number of successive temperature steps where equilibrium was not reached, respectively.

IV. Simulation Results

The research presented in this paper compares the performance of the SA approach with a KSP approach[11],[12] and a heuristic reroute generation approach[9], using the metrics of operational acceptability defined in Section II as well as a evaluating the computational performance of each algorithm.

A. Heuristic Approach

The heuristic generation approach, described in [9], provides a fast-time ad-hoc approach for generating reroute alternatives which consist of pre-defined routes from the departure airport to the arrival airport, for pre-departure flights and from en-route fixes to the arrival airport for active flights. These alternatives are augmented by reroutes constructed incrementally using the same fix-pair segments defined in the network. We note that the operational acceptability metrics are not used when generating the heuristic alternatives and simply provide a post-generation analysis of the options.

B. KSP approach

All network-optimized reroutes are defined using the k-shortest path approach[20] where 200 paths are generated using the same arc costs used to generate the initial solution for SA. From the 200 paths generated, all infeasible paths are culled and the remaining paths are evaluated against the operational acceptability metrics defined in Equation 12. The five reroutes with the lowest objective values are returned.

C. SA implementation

The SA implementation described in the previous section is implemented for the example problems considered using the SA parameter values listed in Table 1. In addition, as multiple reroutes are desired, the SA algorithm is run 15 times and the best five reroutes generated are returned. We note that it is not guaranteed that SA will produce five unique reroutes, and therefore the number of alternatives provided may be less.

Table 1. SA Parameter Values

SA Parameter	Value
ΔT	0.97
neq	15
nfz	7

D. Example problem

All examples presented in this paper are derived from reroutes generated for scheduled flights on April 20, 2009 at 20:00 GMT. Specifically, we examine the alternatives generated for several flights that plan to utilize a single segment that is blocked by weather, as shown in Figure 4. Table 2 lists the flight IDs used throughout the remainder of this paper, as well as the origin and destination airport, the departure time and the deviation point along the original route.

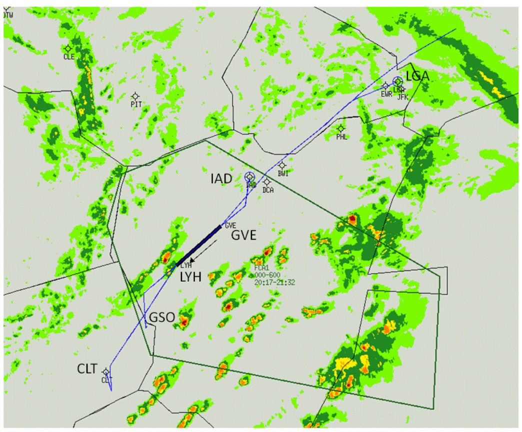

Figure 4. Weather-impacted Flight Routes

The results presented assume that all metrics in Equation 12 are equally weighted and examine how the reroutes generated by the three different approaches perform against each metric for each flight examined.

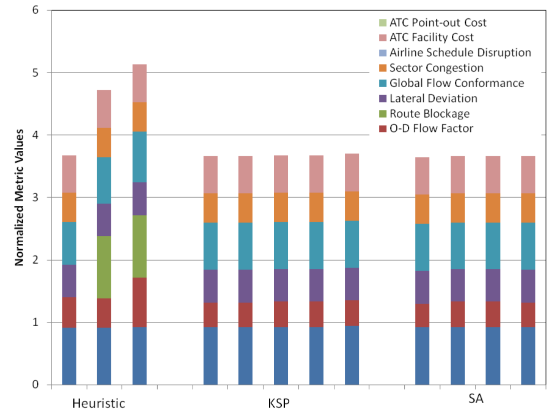

Figure 5. Performance of Route Alternatives for Flight A

Figure 5 shows the performance of the reroute alternatives for Flight A which is already en-route from Logan International Airport (BOS) to Charlotte Douglas International (CLT). Examining Figure 5 we see that of the three heuristically-generated reroutes, only one provides a weather-avoiding reroute; note the green bar segments, arising from weather blockage. Comparing the reroutes for the KSP and SA approaches, we see that SA returns reroutes almost identical in quality to the KSP reroutes, and in fact the last SA reroute is the first KSP reroute.

Figure 6 shows the performance of the reroutes for Flight B, which is scheduled to depart momentarily from LaGuardia Airport (LGA) for Greensboro Airport (GSO) and as such, a fix on the original route (BIGGY) is deemed the starting point for generating reroutes. We again see similar performance when comparing the three algorithms, namely that all but one of the heuristic alternatives is blocked by weather, and that overall the SA and the KSP algorithms perform comparably.

The difference between the SA reroutes and the KSP path reroutes arises from the priority of flow conformance verses sector congestion. As the KSP reroutes are generated by evaluation of the network arc costs, better O-D flow conformance is often observed. SA, in contrast, evaluates all metrics equally and can find reroutes with lower overall cost

Table 2. Weather-impacted Flight Schedules

Flight	Departure Airport	Destination Airport	Departure Time	Deviation Point
Flight A	BOS	CLT	19:45	JERSY
Flight B	LGA	GSO	20:06	BIGGY
Flight C	IAD	CLT	20:08	GVE
Flight D	IAD	GSO	20:16	HAFNR
Flight E	LGA	CLT	20:33	LGA
Flight F	IAD	GSO	21:01	IAD

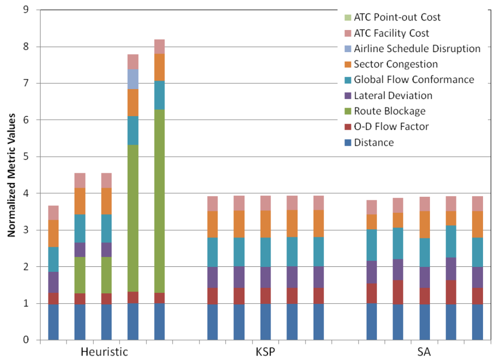

Figure 6. Performance of Reroute Alternatives for Flight B

without preference for a given metric component.

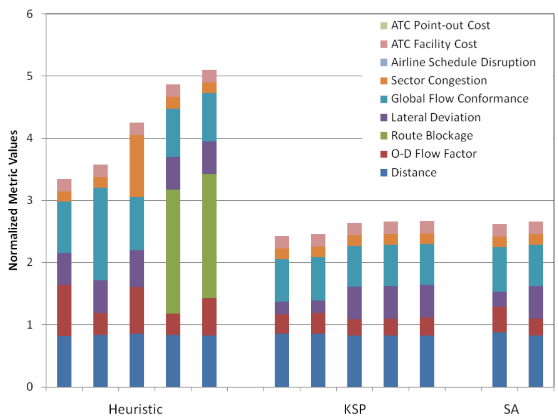

Figure 7. Performance of Route Alternatives for Flight C

Figure 7 shows the performance of the reroutes for Flight C, where again the SA reroutes provide significantly improved alternatives, as compared to the heuristic approach. Comparing the SA reroutes to the KSP reroutes we notice that the first few KSP reroutes have a lower overall objective value, due to the improved flow conformance values; however the second SA alternative generated is the same as the fourth KSP alternative.

The performance of the reroute alternatives for Flight D, shown in Figure 8, again illustrates how both SA and KSP can dynamically respond to the weather, as compared to the heuristic approach. Comparing the KSP and SA results we again see that the SA results provide comparable alternatives, better than all KSP alternatives except the first. Furthermore, we again notice the priority difference between flow conformance and sector congestion when comparing the two sets of routes.

Figure 9 shows a different performance picture for the reroute alternatives provided for Flight E. By examining Figure 9 we see that the KSP reroutes have weather blockage, which is due to the estimation of route blockage in the network, as described in Section III. SA, however, directly evaluates the route blockage metric and can therefore define unblocked reroute alternatives. In addition, the last SA reroute alternative includes a five minute ground delay which may enable a previously blocked or congested segment to be utilized at lower

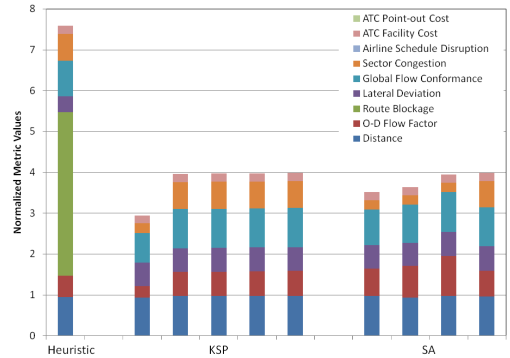

Figure 8. Performance of Reroute Alternatives for Flight D

cost.

The final reroute comparison is provided in Figure 10, corresponding to Flight F. Examining Figure 10 we see a different set of reroutes than generated for Flight D, even though the flight has the same origin and destination airport; however for Flight F we consider reroutes deviating at the origin airport. Specifically, we note that the heuristic approach provides a few weather-free reroutes; albeit many with significant additional distance which generates airline schedule disruption. In contrast, the KSP provides significantly better alternatives by using highly flow conformant alternatives out of

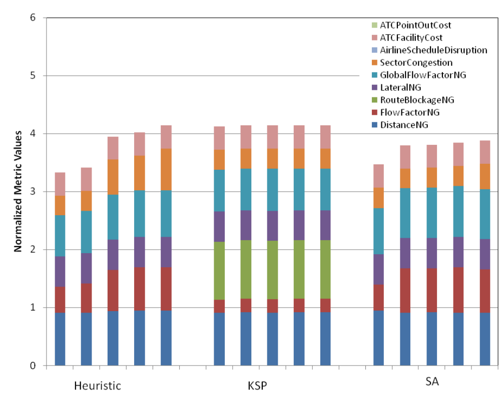

Figure 9. Performance of Reroute Alternatives for Flight E

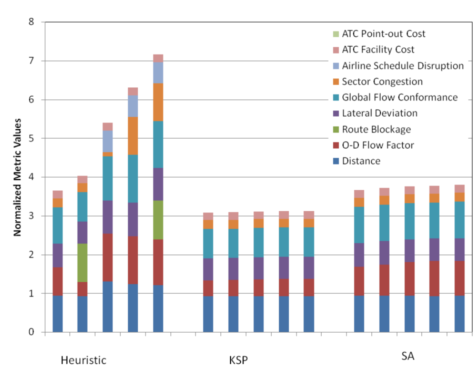

Figure 10. Performance of Reroute Alternatives for Flight F

the departure airport.

E. Computation Comparison

For the examples considered in this paper, the SA-generated reroutes performed as well as the KSP-generated reroutes and significantly better than the heuristically-generated reroutes. However, a major advantage in using Simulated Annealing is the potential reduction in computation effort required to obtain similar quality solutions. As such, we compare the overall computation time for the KSP and SA to generate options for Flight C.

The methodology presented in this paper was implemented in the Java programming language and executed on a non-dedicated server. Furthermore, the code was not optimized for performance. The computation analysis presented here is meant to provide a comparison between the efforts required for both algorithms and not as a measure of the absolute performance possible for either.

The network results, generated by the k-shortest path approach, were obtained through a single iteration of the algorithm. The five best results were then sorted using the overall objective function defined in Equation 12. The computation time required to generate, evaluate, and sort all options in order to define the five best solutions was 22 seconds. As the network optimization approach is deterministic, it is sufficient for the purpose of this analysis to consider this result a performance baseline for the KSP approach.

By comparison, SA is a stochastic heuristic search procedure, where the computation effort required for an iteration can vary based on both the SA tuning parameters as well as the progress through the design space. As such, we define the average computation time required to generate the results presented in this research, which is 48 seconds. This increase is due to the number of objective function calls, which is expensive. However, as the multiple iterations of SA are independent, it is possible to parallelize this process, which would result in significant cost savings as the average time to generate a single SA reroute is 3.3 seconds. The KSP method, on the other hand, cannot be readily parallelized. Note also that SA is searching for solutions with ground delays, a degree of freedom that KSP cannot include without a large increase in computation time.

V. CONCLUSIONS

The research presented in this paper extends previous work on developing a methodology for the computer-based design of operationally-acceptable flight specific reroutes. The purpose of this paper was to analyze how reroute options generated by SA compared to reroute alternatives generated by the KSP approached described in [12] and the heuristic generation approach described in [9]. The methodology presented here aims to improve the reroute alternatives for a given flight by considering all metrics of operational acceptability during the optimization and expand the decision space to enable pre-departure delays. From the analysis presented here, we see that there is significant potential for enhanced reroute development using SA and that the computation benefits incurred warrant further exploration of this algorithm.

Continuing work includes an investigation into additional route alternative parameters that can be easily captured within the SA framework. Options such as altitude deviations are potentially useful, especially when route segments are blocked by weather at lower altitudes and can be utilized at higher altitudes. Capturing additional metrics in a dynamic nature is also a desirable feature of a reroute generation method. Specifically, flow conformance metrics should have elements defining how the fix-pair is currently being used so as not to override the current operational constraints.

As was shown in the computation comparison, much of the benefit of SA would be realized if the independent iterations were parallelized. As such, continuing work is needed to improve the computational performance of SA, especially in view of the goal of enlarging the design space for generating route alternatives.

NOTICE

The contents of this material reflect the views of the authors and The MITRE Corporation and do not necessarily reflect the views of the FAA or the DOT. Neither the Federal Aviation Administration nor the Department of Transportation makes any warranty or guarantee, or promise, expressed or implied, concerning the content or accuracy of these views.

ACKNOWLEDGMENT

The authors would like to thank all who helped with this work, especially Jonathon Anderson, James DeArmon, Dan Greenbaum, Joe Nardelli, Laurel Rhodes, and Stephen Zobell.

REFERENCES

[1] *Severe Weather and Route Management 2005,* Federal Aviation Administration - Air Traffic Control System Command Center, http://www.fly.faa.gov/Operations/Strategic_Planning/svrwx_handbook.html

[2] Mayhew, D., and Manikonda, V., "Constrained Airspace Rerouting Planner for ACES", *AIAA Guidance Navigation and Control Conference and Exhibit,* 18-21 August 2008, Honolulu, HI

[3] Grabbe, S., Sridhar, B., and Mukerjee, Av., "Sequential Traffic Flow Optimization with Tactical Flight Control Heuristics", *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 3, May-June 2009, pp. 811-820.

[4] Bokadia, S., and Valasek, J., "Sever Weather Avoidance Using Informed Heuristic Search", *AIAA Guidance Navigation and Control Conference and Exhibit,* 6-9 August 2001, Montreal, CA

[5] Krozel, J., Penny, S., Prete, J., and Mitchell, J., "Automated Route Generation for Avoiding Deterministic Weather in Transition Airspace" *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 1, January-February 2007. pp. 144-153

[6] Prete, J., and Mitchell, J., "Safe Rerouting of Multiple Aircraft Flows in the Presence of Time-Varying Weather Data" *AIAA Guidance Navigation and Control Conference and Exhibit,* 16-19 August 2004, Providence, RI

[7] Krozel, J., Prete, J., Mitchell, J., Smith, P., Andre, A., "Designing On-Demand Coded Departure Reroutes" *AIAA Guidance Navigation and Control Conference and Exhibit,* 21-24 August 2006, Keystone, CO

[8] Weyns, D., Holvoet, T., and Helleboogh, A., "Anticipatory Vehicle Routing using Delegate Multi-Agent System", *Proceedings of the 2007 IEEE Intelligent Transportation Systems Conference*, 30 September -3 October 2007, Seattle, WA

[9] Taber, N., Jackson, C., Masalonis, A., Song, L., Taylor, C, Wanke, C., and Zobell, S., "Advanced TFM Congestion Management Performance Analysis and Research Results" MITRE Technical Report #090220, The MITRE Corporation, September, 2009.

[10] DeArmon, J., Wanke, C., and Nardelli, J., "Quantification of Desirable Air Route Attributes in a Reroute Generation Capability", *AIAA Aviation Technology, Integration, and Operations Conference*, 21-23 September 2009, Hilton Head, SC

[11] Taylor, C., and Wanke, C., "Dynamic Generation of Operationally Acceptable Reroutes" *AIAA Aviation Technology, Integration, and Operations Conference*, 21-23 September 2009, Hilton Head, SC

[12] Taylor, C., and Wanke, C., "Improved Dynamic Generation of Operationally-Acceptable Reroutes Using Network Optimization", *AIAA Guidance Navigation and Control Conference and Exhibit,* 2-5 August 2010, Toronto, Canada

[13] Wanke, C., DeArmon, J., Taylor, C., Greenbaum, D., "Flight Option Generation for Traffic Management Decision Support" *Aviation Technology, Integration, and Operations Conference*, 13-15 September 2010, Fort Worth, TX

[14] Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P., "Optimization by Simulated Annealing" *Science*, Vol. 220, Num. 4498, May 1983.

[15] Wanke, C., Taber, N., Greenbaum, D., Jackson, C., Jacobs, G., Masalonis, A., Rhodes, L., Song, L., Stewart, T., Taylor, C, and Zobell, S., "Research Results for Advanced Congestion Prediction and Resolution Capabilities" MITRE Technical Report #080087, The MITRE Corporation, August, 2008

[16] Beatty, R., Hsu, R., Berry, L., and Rome, J., "Preliminary Evaluation of Flight Delay Propagation Through an Airline Schedule", *2nd USA/Europe ATM R&D Seminar*, Orlando, FL, December 1998.

[17] Ahuja, R., Magnanti, T., Orlin, J., *Network Flows: Theory, Algorithms, and Applications*, Prentice Hall Inc., Englewood Cliffs, NJ, 1993

[18] Ng, H., Grabbe, S., and Mukherjee, A., "Design and Evaluation of a Dynamic Programming Flight Routing Algorithm Using the Convective Weather Avoidance Model" *AIAA Guidance Navigation and Control Conference and Exhibit,* 10-14 August 2009, Chicago, IL

[19] Liu, G., and Ramakrishnan, K., "A*Prune: An Algorithm for Finding K Shortest Paths Subject to Multiple Constraints" *Proceedings of the 2001 IEEE INFOCOM,* Vol. 2, pp. 743-749.

[20] www.jgrpaht.org, Accessed 12/11/2010

AUTHOR BIOGRAPHY

Christine P. Taylor is a Senior Simulation and Modeling Engineer at MITRE, specializing in decision support system development for traffic flow management applications. She holds a B.S. from Cornell University, and M.S. and Ph.D. degrees in aeronautical engineering from the Massachusetts Institute of Technology.

Craig R. Wanke is a Senior Principal Engineer at MITRE, specializing in simulation and modeling. He has worked for the past 18 years on decision support systems for cockpit and air traffic management applications. He holds S.B, S.M., and Ph.D. degrees in aeronautical engineering from the Massachusetts Institute of Technology.

.

.