Methodology for Automated Supersonic Flight Path Routing and Optimization

Bernd Liebhardt[1] and Thorsten Ehlers[2]
German Aerospace Center (DLR), Hamburg, Germany

Rhea Deven Shah[3]
Hamburg Technical University (TUHH), Hamburg, Germany

A methodology is presented for automatically designing and optimizing flight paths for civilian supersonic airplanes on long-haul flights. The paths are rerouted over water to avoid the landfall of emitted sonic booms. The toolchain consists of all-proprietary modules for flight path drafting, mission trajectory simulation, sonic boom carpet computation, and coastline modification. The latter constitutes a novel approach to path finding. The method's realism is enhanced by employing 3-D atmospheres including winds.

I. Introduction

Supersonic overland flight is, explicitly or implicitly, banned from most parts of the world due to the sonic boom that aircraft produce constantly when flying at supersonic speeds. Until "low-boom" technologies [1] become mature (which we assume to happen rather in the long term), future supersonic aircraft will have to fly routes over water to make use of their high-speed, high-altitude capabilities. Over landmasses, they will have to fly at restricted speeds, and they will most likely share flight levels with regular subsonic traffic.

To maximize time savings, supersonic aircraft need flight paths that find a flight time-optimum compromise between maximum overwater share, i.e. supersonic portions, and minimum detours. Operators will aim to accelerate to supersonic speeds early, to circumnavigate coasts closely, and to decelerate to subsonic late. This requires knowing the extent of the "sonic boom carpet", i.e., the sonic boom's ground footprint that the aircraft produces. And that, in turn, requires predicting the exact flight trajectory as well as the boom's atmospheric propagation. Given the problem's high complexity, a sophisticated and automated flight trajectory generator would be of use both for scientific studies and for planning future operations.

The challenges are manifold. Flight paths have to be drafted that respect airplane performance characteristics, air traffic domains, and coastal buffer zones. Trajectories have to be simulated that consider airplane performance, atmospheric conditions, and ground topography all along the flight path. Those trajectories would form the basis for computing the sonic boom carpet, which requires considering the atmosphere and ideally, the aircraft's surrounding pressure field. And lastly, flight paths have to be adapted to simultaneously avoid the carpets' predicted coastal transgressions and unnecessarily large detours.

The present effort's goal is integrating as many of the named capabilities as possible in one modelling environment that enables computing realistic supersonic flight trajectories and missions. This study describes the current status, namely having closed the simulation loop, and presents according flight paths.

---

[1] Dr.-Ing.; Researcher; Institute of Air Transport, Blohmstrasse 20, 21079 D-Hamburg.
[2] Dr.-Ing.; Researcher; Institute of Air Transport, Blohmstrasse 20, 21079 D-Hamburg.
[3] M.Sc. Student; Institute of Air Transportation Systems, Blohmstrasse 20, 21079 D-Hamburg.

II. Implementation

The toolchain shares sets of basic input data:

- Airport locations and altitudes
- Aircraft performance specifications (masses, speeds, aerodynamic polars, engine maps)
- Global non-standard 3-D atmospheric data including winds[4]
- Global coastlines[5], optionally buffered using MATLAB as well as proprietary algorithms
- Global terrain topography[6]
- A proprietary collection of uninhabited islands supposedly ignorable for supersonic overflight

The chain consists of four modules.

A. Flight path drafting

Our flight path drafting code computes flight paths between pairs of airports with the objective of minimizing flight time. It uses a modified Θ* ("Theta-Star") algorithm to find paths on a grid in which the nodes are tuples of equidistance latitude and longitude coordinates, and each of them is connected to the closest nodes in north, south, east and west. As the Θ* algorithm dynamically adds new edges if needed, these 4 connections from each node are sufficient for computing relatively smooth routes. The cost associated with each edge is the time it takes to fly along it, following the great circle between the edge's origin to its destination. Said time is estimated based on two factors: Firstly, a maximum subsonic speed is assumed for both overland segments and segments within a configurable buffer zone close to the next coast line, whereas a configurable supersonic speed is assumed on overwater segments. This factor alone allows for computing relatively realistic routes. Secondly, acceleration is considered in order to model that aircraft cannot instantly switch between sub- and supersonic speed. This does not only increase result fidelity, but it also incentivizes routes with long continuous overwater segments, and it avoids multiple changes between subsonic and supersonic segments. Furthermore, the flight paths computed in this way do not seek to become seaborne as soon as possible, they rather consider that the aircraft will be subsonic for some distance after takeoff and before landing, respectively, and thus, over-land segments are acceptable in those cases. The maximum rate of acceleration, deceleration as well as the ratio between subsonic and supersonic speed have an impact on the optimal flight route.

Figure 1 shows a flight route between Dubai and Singapore on a setting of very low acceleration capabilities. Here, subsonic speed is shown in blue and supersonic speed in a range between white (Mach 1.0) and red (maximum speed, Mach 1.4 in this example). Starting in Dubai, the first part of the flight path is a large great circle segment, and due to the low acceleration assumed here (0.1 m/s^2) it takes a long time to reach maximum speed. Thus, the flight path does not cross India, as a detour is faster than flying a subsonic segment.

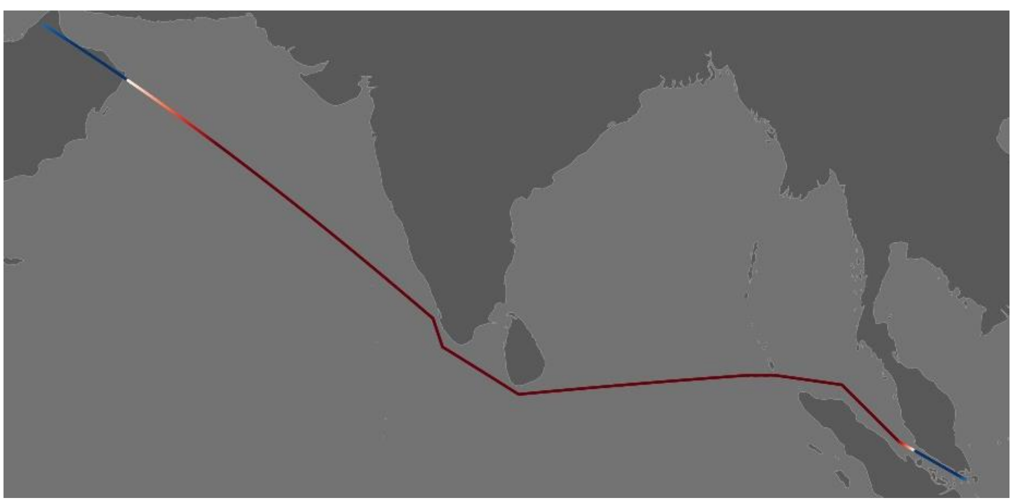

Figure 1. Flight path between Dubai and Singapore with slow acceleration

[4] ERA-Interim data, provided by the European Center for Medium-range Weather Forecast (ECMWF).
[5] World Vector Shorelines; provided by the University of Hawai'i and the U.S. National Ocean Service; available at https://www.soest.hawaii.edu/pwessel/gshhg/
[6] Global Multi-resolution Terrain Elevation Data set (GMTED2010), provided by the U.S. Geological Survey; available at www.usgs.gov/core-science-systems/eros/coastal-changes-and-impacts/gmted2010.

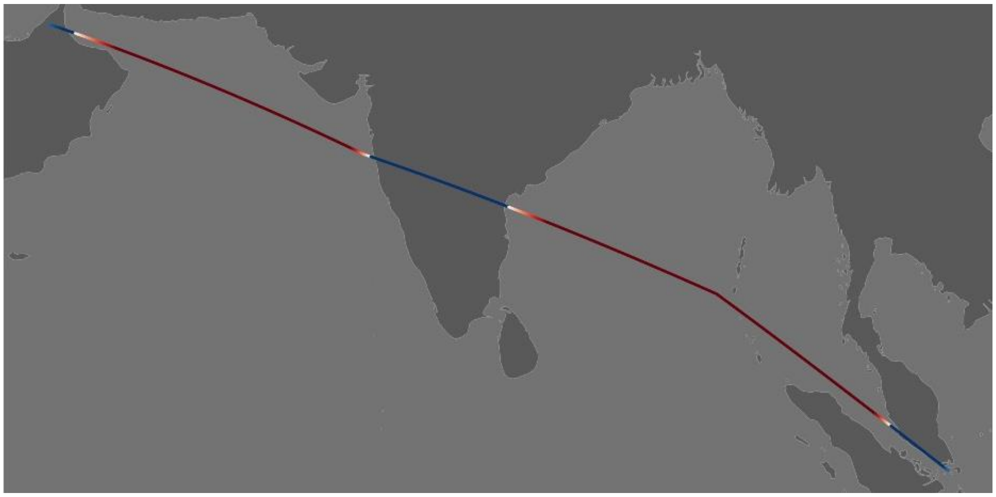

Figure 2. Flight path between Dubai and Singapore with fast acceleration

In a second example, a higher rate of acceleration (0.25 m/s^2) is assumed for the same route, c.f. Figure 2. The higher acceleration is visible by the shorter white and orange segments. Here, the solver avoids the detour south of India, as a subsonic segment decreases the route's length significantly. Using the higher acceleration, the maximum speed of the aircraft can be reached earlier. In both cases, a deceleration rate of 0.5 m/s^2 was assumed.

The speeds which can be reached on the route are computed based on the coast buffers. Over land or inside the buffer zones, only subsonic speed is allowed. Outside these zones, maximum speeds are propagated in forward and backward direction based on the given acceleration and deceleration rates. This allows for a precise calculation of the maximum speed reachable on each route segment.

B. Flight mission simulation

DLR's flight performance code SuperTraC (Supersonic Trajectory Calculator [2]) simulates missions in arbitrary subsonic-supersonic segmentation, airport to airport, by iteratively solving the equations of motion. It considers airport data, flight performance specifications, 3-D atmospheres, coastlines, and given flight paths. Using topography in addition, it can simulate so-called Mach-cutoff missions where the airplane flies at the highest possible (low-)supersonic speeds over land without the sonic boom reaching the ground [3]. The code's output is a fully discretized trajectory (a generic one is depicted in Figure 3).

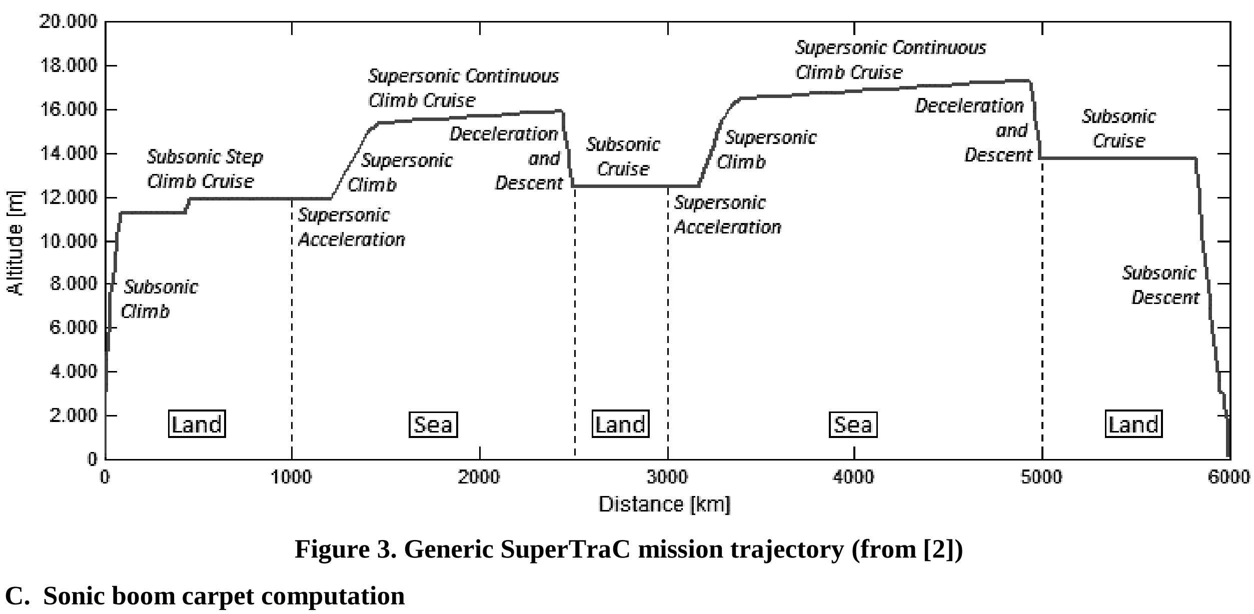

Figure 3. Generic SuperTraC mission trajectory (from [2])

C. Sonic boom carpet computation

In its core, the computation model employs the established method of atmospheric ray tracing: Points on the shock front are followed on their way through the atmosphere while changes in temperature and wind alter their direction of propagation (also see [4]). The model considers the flight trajectory, topography, the curvature of the earth, and 3-D atmospheres. To our knowledge, the latter is unique amongst the published sonic boom propagation models.

The boom carpet is the area where the sonic boom reaches the ground (see Figure 4). By using ray tracing only, the current model finds a *geometric* boom carpet where the boom *could potentially* cause a disturbance. Only an added sound propagation model can determine the border of the *acoustic* carpet where the boom dissipates into background noise. One is in development and should be integrated before long.

In any case, the geometric carpet will always exhibit a larger or equal extent compared to the acoustic one. This is because tailwinds in particular cause sonic rays to flatten out and "graze on" horizontally for long distances while acoustically, their sound would long have faded in background noise. Thus, by using the geometric carpet, the flight path optimization stays on the safe side whilst occasionally producing implausibly large carpet areas.

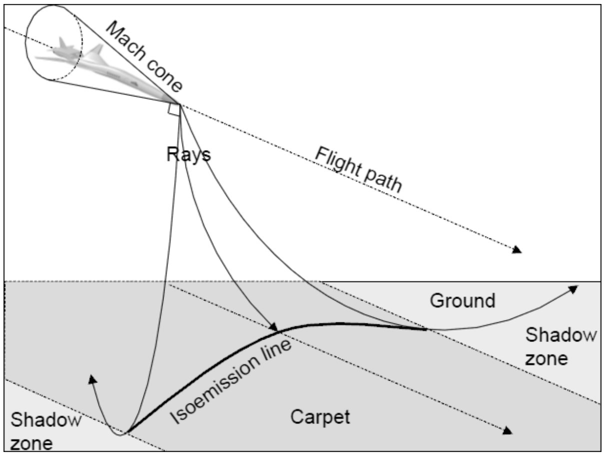

Figure 4. Sonic boom rays and carpet (modified from [5])

D. Coastline buffer modification

The model uses the flight trajectory and the boom carpet as inputs, plus two sets of buffered coastlines:

- a *coastal buffer*, using a smaller offset (e.g. 5 kilometers), that the final boom carpet shall not transgress;
- a *route buffer*, using a larger offset (e.g. 20 kilometers), that is used for flight path drafting.

The code determines the areas where the carpet violates the *coastal buffer*. In those areas, the *route buffer* is moved opposite to the direction of transgression, beyond the flight path, crossing it by the respective transgression distance.

Initially, the route buffer is chosen relatively small so that the flight route can benefit from the frequent occasions where the boom carpet is rather narrow (e.g. thanks to headwinds). Coincidentally, the buffer should be large enough to avoid all too narrow sea passages. By initially drafting routes close to coasts, it is sufficient for the algorithm to move the route buffer outward for the goal of optimization, not inward closer to the coast.

The modified route buffer constitutes a new coastline set for the next iteration of flight path drafting. Figure 5 displays an example for how the route buffer is altered due to a coastal buffer transgression of the boom carpet.

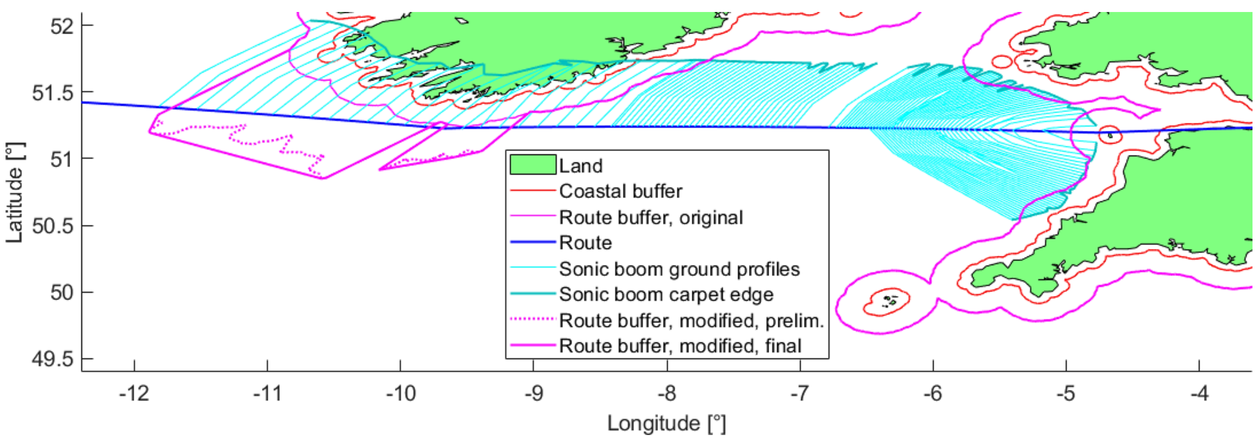

Figure 5. Example of route buffer modification: London-bound flight through the Celtic Sea

E. Simulation chain

A schema of the simulation chain is shown in Figure 6. The chain is to be set up in an appropriate simulation environment, e.g. DLR's RCE [6], or MATLAB, wherein the models B to D are programmed. For this work, the modified coastlines were forwarded to the next iteration of flight path drafting manually due to time constraints.

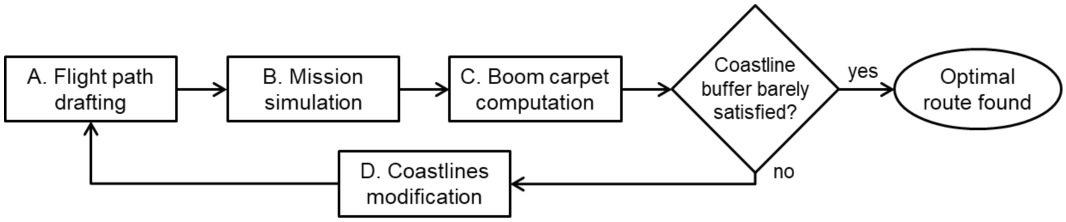

Figure 6. Supersonic flight path optimization toolchain

III. Results

The toolchain was applied to a small number of city pairs, employing one set of realistic 3-D atmospheric data and our Mach 1.6 supersonic business jet model [3] for flight performance. The buffers were set to 5 kilometers for the coast and 20 kilometers for the route, respectively.

In the following, examples are presented to showcase the toolchain's functionality.

A. New York City to London

In its first iteration, the flight path drafting algorithm proposes a route through the English Channel (Figure 7). Subsequently, the boom carpet calculation reports transgressions on Nantucket Island, Newfoundland, and multiple ones on both sides of the Channel. The route buffer is modified accordingly (Figure 8).

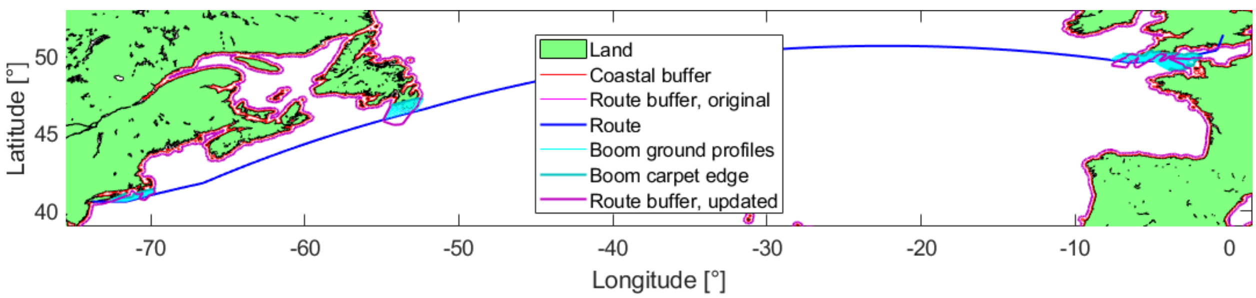

Figure 7. New York City to London, iteration 1

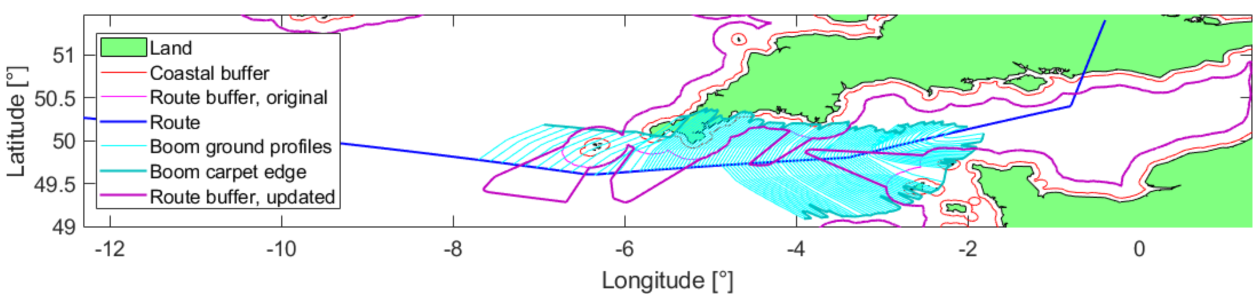

Figure 8. New York City to London, iteration 1, English Channel

Due to the new constraints, the route moves away from the North American coasts and up to the Celtic Sea and Bristol Channel, respectively (Figure 9). On the Eastern side, the boom carpet causes transgressions in southern Ireland and Cornwall. Eventually, the issues are resolved in iteration 3 (Figure 10).

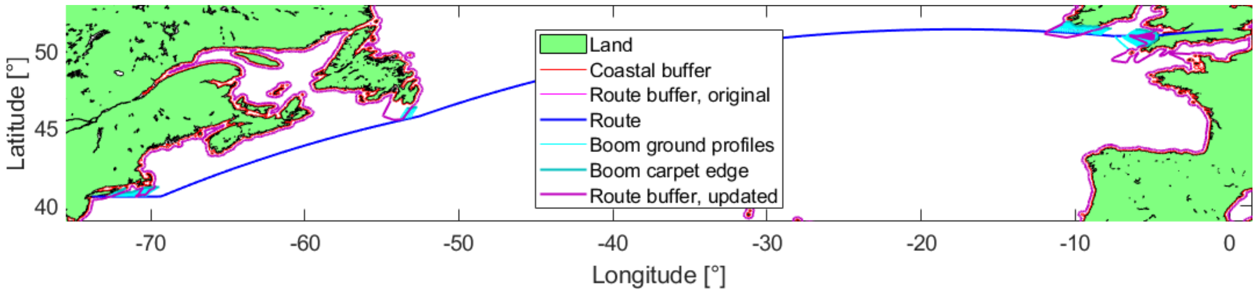

Figure 9. New York City to London, iteration 2

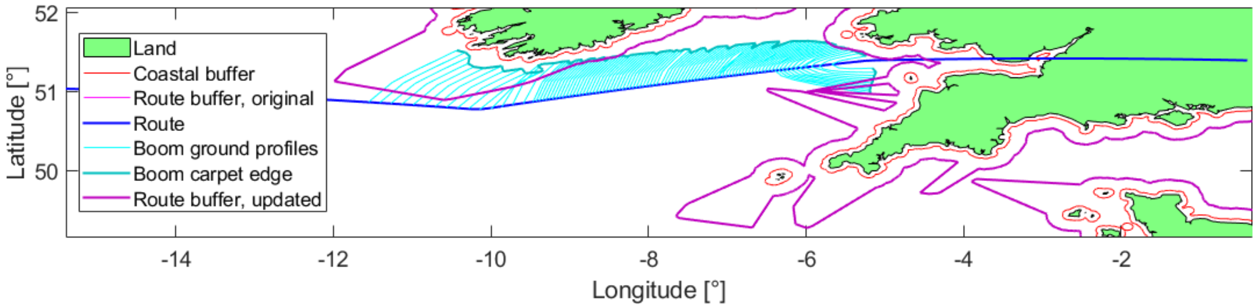

Figure 10. New York City to London, iteration 3, Bristol Channel

B. Singapore to Brisbane

On this particularly challenging route, numerous Indonesian islands and islets stand in the way. The algorithm attempts to find a way through, but eventually has to allow for two decelerated segments over southernmost Sulawesi and the Torres Strait of northernmost Australia (see Figure 11).

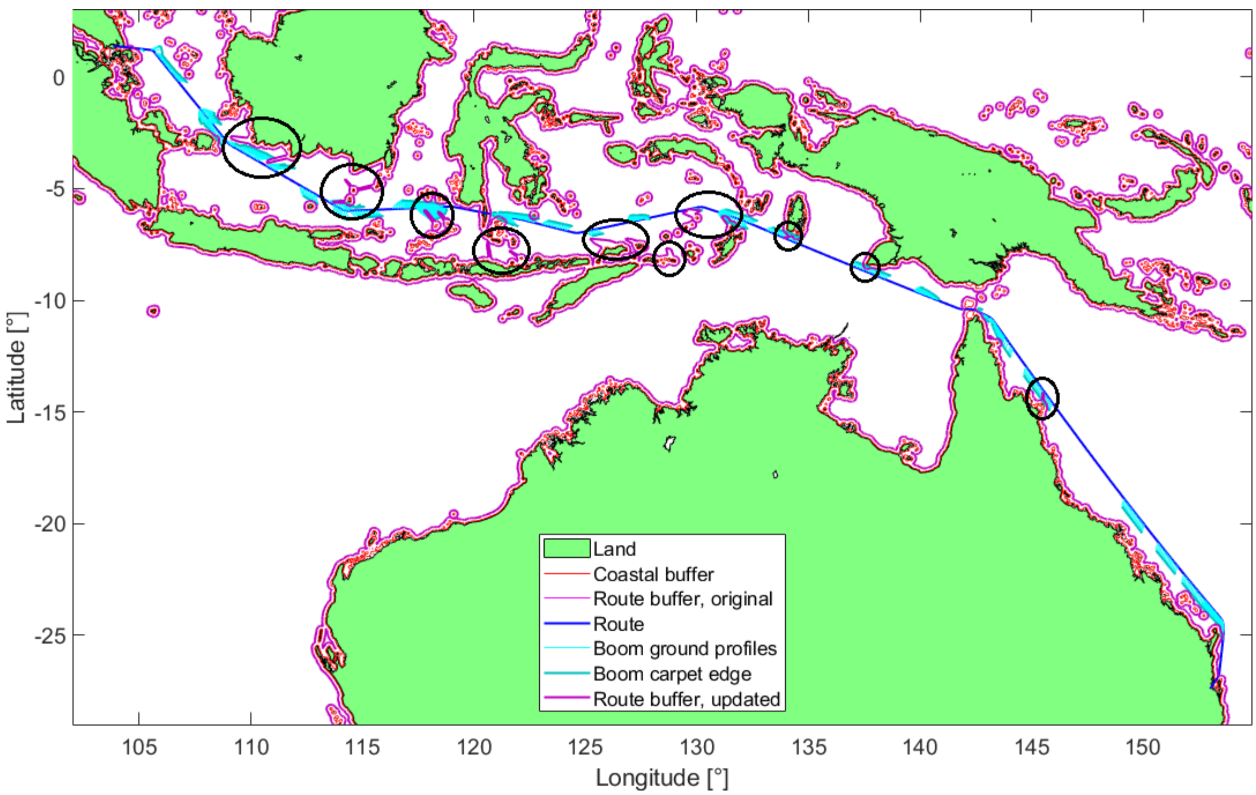

Figure 11. Singapore to Brisbane, iteration 4; modified route buffers ringed

C. Auckland to Perth

The algorithm has the alternatives of routing through the Bass Strait or around Tasmania. It decides for the former (Figure 12) and only has to make small corrections in the second iteration (Figure 13) because the boom carpet happens to be narrow enough. The Kent Group islands were willfully ignored because they appear to be uninhabited. In reality though, inhabitation alone might not suffice as a criterion for supersonic overflight permission, as governments might rule to prohibit disturbing, for instance, wildlife reserves.

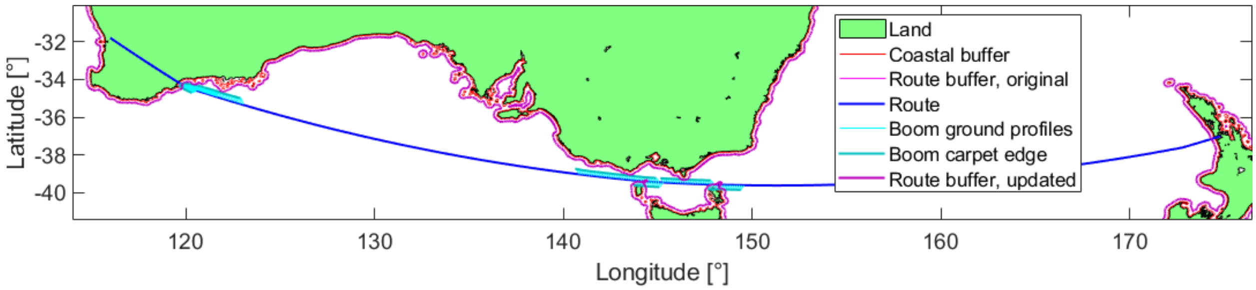

Figure 12. Auckland to Perth, iteration 1

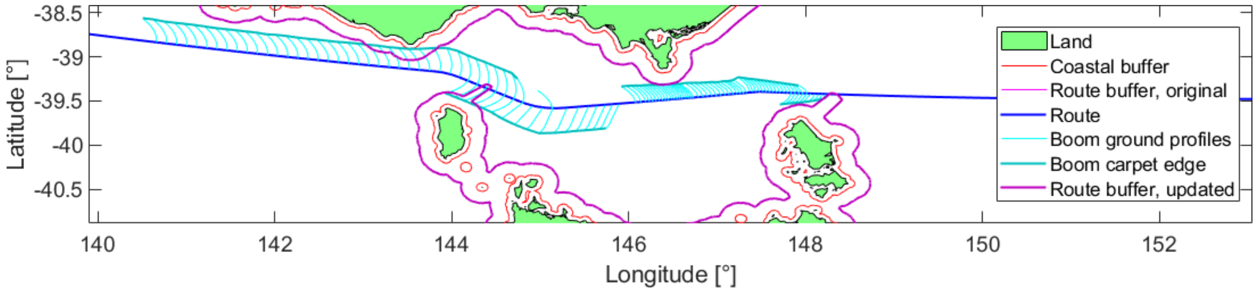

Figure 13. Auckland to Perth, iteration 2, Bass Strait

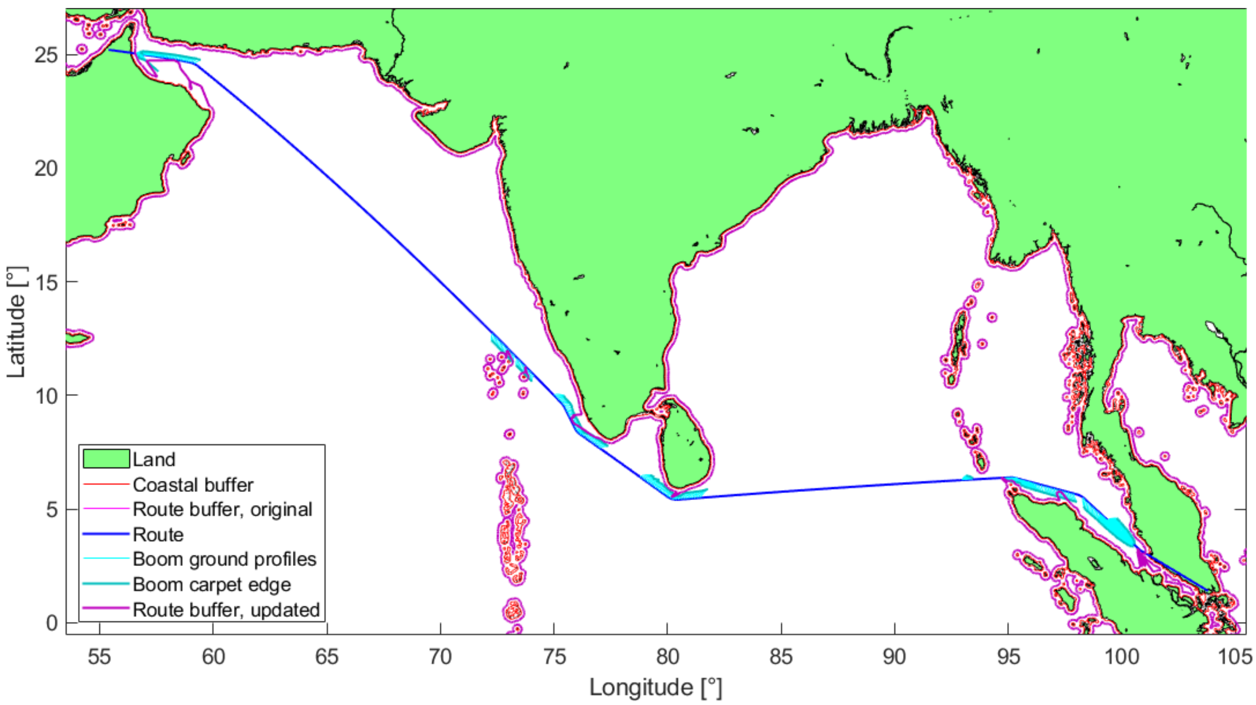

Figure 14. Dubai to Singapore, iteration 3

D. Dubai to Singapore

The algorithm chooses to circumvent India and Sri Lanka due to the aircraft's superior overwater speed (Figure 14). It keeps the obvious flight path and adapts it locally, correcting all too close approaches in the initial two iterations.

E. New York City to Lima

Initially, the algorithm tries to cross the Florida Strait, but the passage between Florida and the Bahamas turns out to be too narrow. Next, it tries to circumvent the Turks and Caicos Islands to stay supersonic, but the extended detour due to a carpet transgression makes it choose a straight, subsonic leg over the Greater Antilles. The fourth iteration only adapts for a small transgression close to Jamaica. See Figure 15.

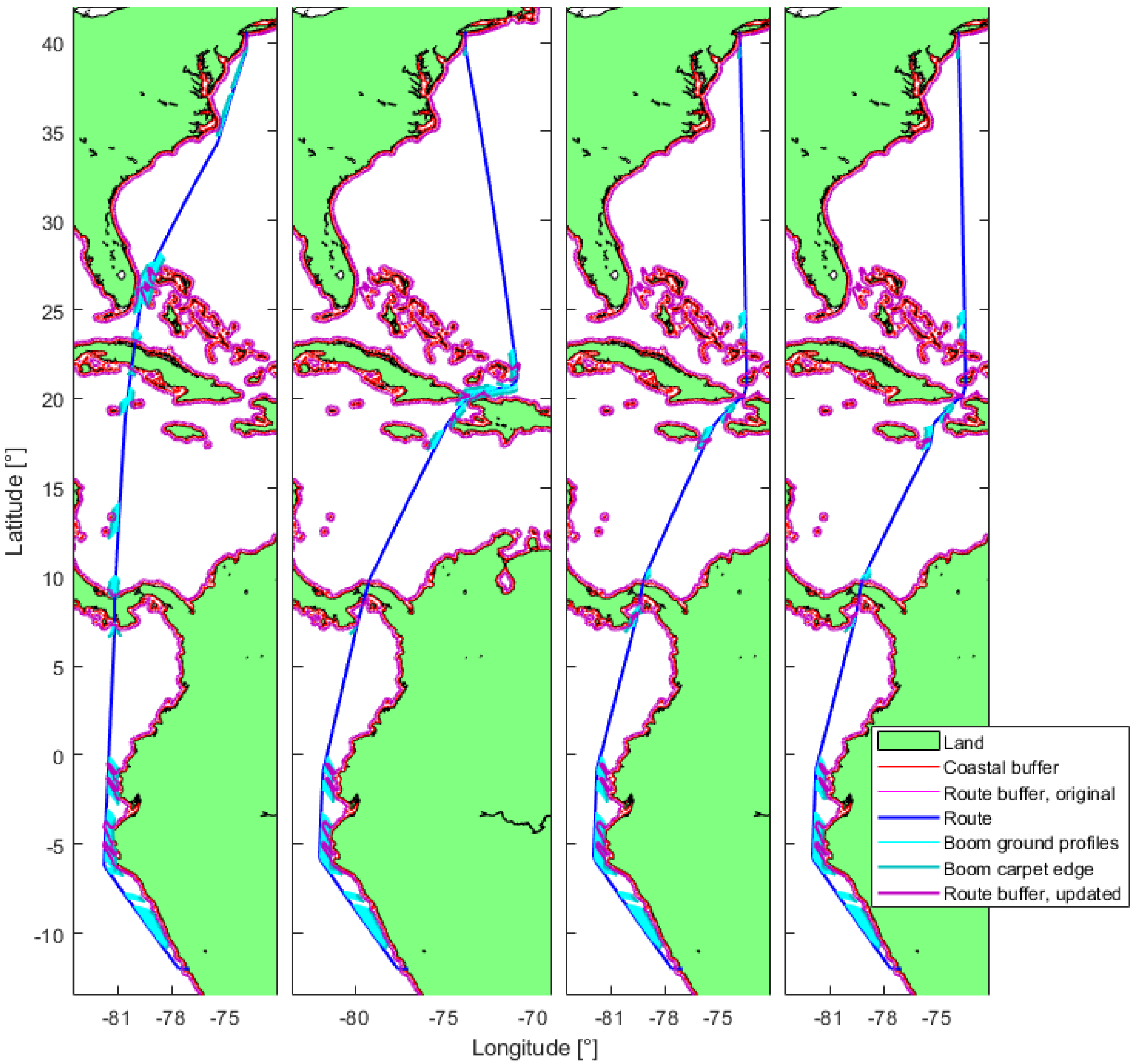

Figure 15. New York City to Lima, iterations 1-4

IV. Discussion

A. Comparable methodologies from other sources

Other institutions have already implemented fully automatic routines for overwater flight route generation. EUROCONTROL (the European Organisation for the Safety of Air Navigation) and the MIT (Massachusetts Institute of Technology) both issued presentations on their tools in 2020 and 2022, respectively, whereas no official publications are available and to the authors' knowledge, nothing new has emerged from them since.

Georgia Tech have issued several publications involving their tool since 2020 [7–12], which happens to be the only one with comparable complexity to the present one. It equally employs a Theta* graph search algorithm for route generation that notably includes a parameter for gauging time-optimality and fuel-optimality. In contrast to using unique atmospheric conditions, a probabilistic approach to boom carpet estimation is implemented in a surrogate model. Flight performance is considered by employing path segments with different speed attributes.

Due to its more generic approach, Georgia Tech's toolchain presumably produces results much faster whereas it might lack some of the present chain's accuracy and flexibility. It might be better suited for scientific big-data studies whereas the one presented here could help assess issues of real-world operations.

B. Secondary sonic boom

The current analysis only considers the primary sonic boom, i.e. the one running directly downward from the airplane. Other kinds of sonic booms, generally called secondary booms, can appear beyond the primary boom carpet. They could potentially be ruled harmful and could hence be prohibited from reaching land. This would result in much wider coastal buffers and much fewer possible supersonic flight routes, potentially eliminating, for instance, the Mediterranean Sea, the Persian Gulf, or the Red Sea.

The present ray-tracing algorithm only requires simple modifications to consider over-the-top and reflected secondary booms as well.

C. Entry and exit points

Supersonic airplanes will have to mix with subsonic air traffic. On overland segments, flying at subsonic speed will probably force them on shared flight levels, whereas over the sea, supersonic airplanes will probably cruise above all subsonic traffic. Consequently, they could receive their own skyways over sea (or the freedom to not use flight routes at all), but they would still have to enter and exit regional airspace systems.

For this purpose, dedicated waypoints could be implemented for supersonic flights, as has been the case for Concorde. Such waypoints would also reduce the complexity for flight route generation (stopping short of prescribing dedicated supersonic routes altogether, which would stifle the potential for optimization in favorable weather).

D. Way forward

As mentioned, a methodology for acoustic sonic boom propagation shall be implemented to estimate the acoustic boom carpet instead of the geometric one. This is expected to cut carpet widths particularly on the downwind side.

Further, all implemented modules feature numerous parameters and consequently, high complexity. Robustness can be improved particularly for the coastline modification module, next to its computation speed, which requires additional effort and perhaps smarter heuristics.

The path planning algorithm can be improved by better emulating actual flight performance instead of using averaged rates of acceleration.

Eventually, the toolchain can be used to generate routes for diverse atmospheric conditions and to test route robustness and flight times, respectively, for operational use.

V. Conclusion

A methodology was presented to generate and optimize flight routes for civilian supersonic airplanes on long-haul missions. It employs all-proprietary models, namely a graph search algorithm for flight route generation, the SuperTraC model for flight physics-based mission trajectory simulation, an atmospheric propagation model for sonic boom carpet computation, and a novel model for coastline modification that feeds back into flight route generation. The methodology aims for relatively high accuracy by using realistic flight performance and weather data as well as high-fidelity computation models. Several examples of automatically generated flight routes were showcased, demonstrating supposedly plausible results. Next, it is planned to implement an acoustic propagation algorithm to compute more realistic boom carpets and to study routings in diverse atmospheric conditions.

References

[1] Coen, P., Loubeau, A., Rathsam, J., and Shah, G., *Achieving Global Consensus on Acceptable Sound Levels for Overland Supersonic Flight*, Institute of Noise Control Engineering, 2023.

[2] Liebhardt, B., Linke, F., and Dahlmann, K., "Supersonic Deviations: Assessment of Sonic-Boom-Restricted Flight Routing," *Journal of Aircraft*; Vol. 51, No. 6, 2014, pp. 1987–1996. doi: 10.2514/1.C032591.

[3] Liebhardt, B., Linke, F., and Knaack, M., "Supersonic Overland Without a Sonic Boom: Quantitative Assessment of Mach-Cutoff Flight," *Journal of Aircraft*; Vol. 59, No. 5, 2022, pp. 1257–1266. doi: 10.2514/1.C036637.

[4] Liebhardt, Bernd, Lütjens, Klaus, Swaid, Majed, Müller, Matthes N., and Ladewich, Maik, "Sonic Boom Carpet Computation as a Basis for Supersonic Flight Routing," *AIAA Aviation Forum,* Dallas, TX, 2019.

[5] Coulouvrat, F., "The Challenges of Defining an Acceptable Sonic Boom Overland," *15th AIAA/CEAS Aeroacoustics Conference (30th AIAA Aeroacoustics Conference),* American Institute of Aeronautics and Astronautics, Reston, Virigina, 2009.

[6] Boden, B., Flink, J., Först, N., Mischke, R., Schaffert, K., Weinert, A., Wohlan, A., and Schreiber, A., "RCE: An Integration Environment for Engineering and Science," *SoftwareX*; Vol. 15, 2021, p. 100759. doi: 10.1016/j.softx.2021.100759.

[7] *A Methodology for Supersonic Commercial Market Estimation and Environmental Impact Evaluation (Part I),* 2020.

[8] Wen, J., Weit, C. J., Mayakonda, M., Anand, A., Zaidi, T., and Mavris, D., "A Methodology for Supersonic Commercial Market Estimation and Environmental Impact Evaluation (Part II)," *AIAA AVIATION 2020 FORUM,* American Institute of Aeronautics and Astronautics, Reston, Virginia, 2020.

[9] Weit, C. J., Wen, J., Zaidi, T. A., and Mavris, D. N., "Estimating supersonic commercial aircraft market and resulting CO$$_2$$ emissions using public movement data," *CEAS Aeronautical Journal* [online], Vol. 12, No. 1, 2021, pp. 191–203, https://link.springer.com/article/10.1007/s13272-020-00486-3.

[10] Pfaender, H., *Sonic Boom over land avoidance and the impact on economic feasibility*, Institute of Noise Control Engineering, 2023.

[11] Wen, J., Pfaender, J. H., and Mavris, D. N., "An Integrated Path Planning Tool for Civil Supersonic Flight Operations," *AIAA SCITECH 2024 Forum,* American Institute of Aeronautics and Astronautics, Reston, Virginia, 2024.

[12] Wen, J., Pfaender, J. H., and Mavris, D. N., "Commercial Supersonic Operations and Network Analyses," *AIAA SCITECH 2024 Forum,* American Institute of Aeronautics and Astronautics, Reston, Virginia, 2024.