BI-OBJECTIVE PSEUDOSPECTRAL OPTIMAL CONTROL TECHNIQUES FOR AIRCRAFT TRAJECTORY OPTIMISATION

Kenneth Chircop*, David Zammit-Mangion*,, Roberto Sabatini****
***University of Malta, ****Cranfield University**
kenneth.chircop@um.edu.mt

Keywords: *optimal control, optimisation, Pareto frontier, pseudospectral methods*

Abstract

This paper presents the adoption of a number of optimisation techniques to solve aircraft trajectory optimisation problems formulated as optimal control problems. The adaptive bisection ε-constraint method is adapted to enable the solution of problems in which two performance indices are to be minimized simultaneously leading to the generation of Pareto frontiers.

The techniques are applied to an aircraft trajectory optimisation problem in which a generic model of an Airbus A320 model aircraft is used. The problem involves the generation of a Pareto set of solutions which find a compromise between flight time and fuel consumption for a climb from 35 ft to a cruising level of 35,000 ft in a range of 900 km. The results are then analysed in-depth and corroborated with flight performance theory.

1 Introduction

Optimal control theory today is used in a wide range of engineering problems. It deals with finding optimal control laws that minimize a performance criterion (cost functional) of a dynamic system through mathematical optimisation techniques. The optimisation process involves translating the system dynamics and its desired objectives into the abstract language of mathematics, which give rise to what is called a control problem, and then to find the solution to this problem. Such a solution is also called optimal control and the path it follows to reach the desired goal is called the optimal trajectory. The physical system which is represented by a mathematical model, consists of a set of relations between the system states and its control inputs. Physical restrictions on the control inputs lead to a finite set of admissible inputs or controls. The solution of a control problem is to determine the admissible inputs which generate the desired output and which, in doing so, minimize the cost functional [10].

Traditionally, optimal control problems were solved using indirect methods, applying the calculus of variations or Pontryagin's maximum principle to satisfy first-order necessary conditions for optimality [12]. These methods are characterized by explicitly solving the optimality conditions stated in terms of the adjoint differential equations, the maximum principle, and associated boundary conditions [3]. This is practical for classical problems and some special weakly non-linear low dimensional systems. However, to obtain a solution of dynamic systems described by strongly non-linear differential equations, it is necessary to use numerical methods [12]. Even so, these methods suffer from the fact that adding new constraints can require deriving new necessary conditions. Also, in many complex problems, getting the necessary conditions in a useful form can be a very difficult task [5].

As problems became more complex, indirect methods became increasingly harder to use, eventually being replaced by the more computationally intensive direct methods. Direct methods

transcribe the continuous optimal control problem into a parameter optimisation problem. Satisfaction of the system equations is accomplished by integrating them stepwise using either implicit or explicit rules; in either case, the effect is to generate non-linear constraint equations which must be satisfied by the parameters, which are the discrete representations of the state and control histories [7]. The problem is thus converted from the original infinite dimensional optimal control problem into a finite Non-Linear Programming (NLP) problem which can be solved using standard NLP solvers.

The work presented in this paper describes the adoption of a number of techniques to solve optimal control problems (OCPs) using direct methods to find optimal trajectories. In particular, the adaptive bisection ε-constraint scalarisation method is adapted to enable the generation of Pareto frontiers for bi-objective optimal control problems (BOOCPs). The underlying NLP solver which is used for the solution of the discretised BOOCP is the open-source large-scale non-linear solver IPOPT [14], which has been integrated with the MATLAB environment [1], where all the algorithms were developed.

The paper is structures as follows. Section 1 gave an introduction to the work presented herein. Section 2 formulates a generic single-objective continuous optimal control problem which is the discretised using pseudospectral techniques in Section 3. The discretised optimal control problem is then formulated into a parameter optimisation problem which can be fed to the non-linear programming solver IPOPT in Section 4. The optimal control problem is further extended to support the solution of bi-objective problems in Section 5, and the adaptation of the adaptive bi-section ε-constrained scalarisation method is described in Section 6 which enables the solution of bi-objective problems with standard solvers. An aircraft trajectory optimisation problem is then applied to the optimisation techniques, the results of which are analysed in-depth and corroborated with flight performance theory. Finally a few concluding remarks on the work presented are given.

2 The Generic Optimal Control Problem

In general, a Single-Objective Optimal Control Problem (SOOCP) is solved by finding the state trajectories $x(t)$, the control trajectories $u(t)$, and times t_0 and t_f in the interval $t \in \left[t_0, t_f\right]$, that minimize the cost functional J. The problem is formulated as follows [4]:

$$J = \phi\left[x\left(t_f\right), t_f\right] + \int_{t_0}^{t_f} L\left[x(t), u(t), t\right] dt \tag{1}$$

where ϕ is the endpoint cost and L is the integrand cost, known as the Mayer and Lagrange cost respectively.

The SOOCP is subjected to the following constraints which must be satisfied by the solution:

$$\dot{x} = f\left[x(t), u(t), t\right], t \in \left[t_0, t_f\right] \tag{2}$$

$$h_l \leq h\left[x(t), u(t), t\right] \leq h_u, t \in \left[t_0, t_f\right] \tag{3}$$

$$e_l \leq e\left[x(t_0), x\left(t_f\right), u(t_0), u\left(t_f\right), t_0, t_f\right] \leq e_u \tag{4}$$

where $\dot{x}$ represents the system dynamics in the form of differential constraints, and h and e are the path and event constraints respectively.

The state, control and time variables are also bounded as follows:

$$u_l \leq u(t) \leq u_u, t \in \left[t_0, t_f\right] \tag{5}$$

$$x_l \leq x(t) \leq x_u, t \in \left[t_0, t_f\right] \tag{6}$$

$$t_{0l} \leq t_0 \leq t_{0u} \tag{7}$$

$$t_{fl} \leq t_f \leq t_{fu} \tag{8}$$

$$t_f - t_0 \geq 0. \tag{9}$$

3 Pseudospectral Discretisation

Solving the OCP analytically is generally a very difficult task. This problem can be eliminated if solutions are approximated in reasonable time with the utilization of efficient numerical techniques on digital computers. With this approach, the OCP is discretised such that the state and control trajectories are represented by vectors of points at nodes representing time.

The computation of a solution for the discretised optimal control problem involves a number

of mathematical computations, the most computationally intensive of which are those approximating the derivatives of the state trajectories at the discretisation nodes and integrating the cost functionals. Over the last few years, pseudospectral discretisation techniques have emerged as the most suitable computational methods for solving optimal control problems owing to their accuracy and speed, with an impressive convergence rate known as spectral accuracy [13]. In fact, for smooth problems, spectral accuracy implies an exponential convergence rate [11]. For such reasons, pseudospectral techniques have been adopted in this work.

The discretisation process commences with the introduction of the following transformation in the general optimal control problem [8]:

$$\tau \leftarrow \frac{2}{t_f - t_0} t - \frac{t_f + t_0}{t_f - t_0}, t \in \left[t_0, t_f\right] \tag{10}$$

This results in the mapping:

$$\tau \in [-1,1] \leftarrow t \in \left[t_0, t_f\right] \tag{11}$$

The OCP is now to find the state and control trajectories $x(\tau)$ and $u(\tau)$ respectively, in the interval $\tau \in [-1,1]$, and times t_0 and t_f, that minimize the performance index:

$$J = \phi\left[x(1), t_f\right] + \frac{t_f - t_0}{2} \int_{-1}^{1} L\left[x(\tau), u(\tau), \tau\right] d\tau \tag{12}$$

subject to the following constraints and bounds:

$$\dot{x}(\tau) = \frac{t_f - t_0}{2} f\left[x(\tau), u(\tau), \tau\right], \tau \in [-1,1] \tag{13}$$

$$h_l \leq h\left[x(\tau), u(\tau), \tau\right] \leq h_u, \tau \in [-1,1] \tag{14}$$

$$e_l \leq e\left[x(-1), x(1), u(-1), u(1), t_0, t_f\right] \leq e_u \tag{15}$$

$$u_l \leq u(\tau) \leq u_u, \tau \in [-1,1] \tag{16}$$

$$x_l \leq x(\tau) \leq x_u, \tau \in [-1,1] \tag{17}$$

$$t_{0l} \leq t_0 \leq t_{0u} \tag{18}$$

$$t_{fl} \leq t_f \leq t_{fu} \tag{19}$$

$$t_f - t_0 \geq 0. \tag{20}$$

In the Legendre pseudospectral approximation, the state and control trajectories $x(\tau)$ and $u(\tau)$ respectively, in the interval $\tau \in [-1,1]$, are approximated by N^{th} order Lagrange polynomials $x^N(\tau)$ and $u^N(\tau)$ based on interpolation at the Legendre-Gauss-Lobatto nodes [11]:

$$x(\tau) \approx x^N(\tau) = \sum_{k=0}^{N} x(\tau_k)\phi_k(\tau) \tag{21}$$

$$u(\tau) \approx u^N(\tau) = \sum_{k=0}^{N} x(\tau_k)\phi_k(\tau) \tag{22}$$

where $x^N(\tau)$ and $u^N(\tau)$ are the Lagrange interpolating polynomials, and $\phi_k(\tau)$ are known as Lagrange basis polynomials. The derivative of the state vector is approximated as follows:

$$\dot{x}(\tau_k) \approx \dot{x}^N(\tau_k) = \sum_{i=0}^{N} D_{ki} x(\tau_i), i = 0, 1, ..., N \tag{23}$$

where D is the differentiation matrix corresponding to the LGL nodes.

The differential constraints are evaluated at the LGL nodes such that:

$$f(\tau_k) = f_k\left(x^N(\tau_i), u^N(\tau_i), \tau_i\right), i = 0, 1, ..., N \tag{24}$$

The differential defects at the collocation points are calculated by subtracting the differential constraints from the derivative of the state vector:

$$\zeta(\tau_k) = \dot{x}(\tau_k) - f(\tau_k) \tag{25}$$

The path constraint functions are similarly calculated at the LGL nodes:

$$h(\tau_k) = h_k\left(x^N(\tau_i), u^N(\tau_i), \tau_i\right), i = 0, 1, ..., N \tag{26}$$

The objective function of the optimal control problem is approximated as follows:

$$J \approx \phi\left[x^N(1), t_f\right] + \frac{t_f - t_0}{2} \sum_{k=0}^{N} L\left[x^N(\tau_k), u^N(\tau_k), \tau_k\right] \omega_k \tag{27}$$

where the weights ω_k are defined at the LGL nodes.

4 NLP OCP Formulation

In Non-linear Programming (NLP), a system of equalities and/or inequalities is solved over a set of unknown variables, such that an objective function is minimized. In general, a NLP problem is defined as:

$$\min_{y} \{J\} \tag{28}$$

subject to:

$$g_j(y) \leq 0, (1 \leq j \leq r) \tag{29}$$

$$h_k(y) = 0, (1 \leq k \leq s) \tag{30}$$

$$y_{li} \leq y_i \leq y_{ui}, (1 \leq i \leq n_y) \tag{31}$$

where y is the vector of decision variables of size n_y to be optimised with lower and upper bounds y_{li} and y_{ui} respectively. $g_j(y)$ and $h_k(y)$ are the r inequality and s equality constraints respectively.

The OCP can be formulated into a NLP problem by following a simple procedure. The objective function can be calculated directly from equation (27), and the decision vector y is constructed from the state vector x^N, the control vector u^N and the initial and final times t_0 and t_f respectively. This results in vector y having dimension $n_y = n_x(N+1) + n_u(N+1) + 2$ and is constructed as follows:

$$y = [x^N \quad u^N \quad t_0 \quad t_f]^T \tag{32}$$

The decision vector is constrained by a lower bound vector y_{li} and an upper bound vector y_{ui}:

$$y_{li} = [x_l^N \quad u_l^N \quad t_{0l} \quad t_{fl}]^T; \tag{33}$$

$$y_{ui} = [x_u^N \quad u_u^N \quad t_{0u} \quad t_{fu}]^T; \tag{34}$$

The equality constraint vector $h_k(y)$ is constructed from the differential constraints such that:

$$h_k(y) = [\zeta^N]^T \tag{35}$$

with the dimension of the equality constraints vector r equal to $n_x(N+1)$.

The equality constraints vector incorporates the path, event and time constraints with their respective bounds:

$$g_k(y) = [h^N - h_l^N \quad h_u^N - h^N \quad e^N - e_l^N \quad e_u^N - e^N \quad t_f - t_0]^T \tag{36}$$

5 Generic BOOCP Problem Formulation

In flight trajectory optimisation, it is often the case that an engineer needs to find an optimal control law which minimizes the fuel consumption and the time of flight, or a compromise might be required between carbon dioxide and nitrous oxides emissions. These are inherently BOOCPs.

In the bi-objective case, the optimal control problem is defined similar to the definition of the SOOCP in Section 2 with the exception for equation (1) which is now modified to cater for two objective functions:

$$J_i = \phi_i \left[x\left(t_f\right), t_f\right] + \int_{t_0}^{t_f} L_i \left[x(t), u(t), t\right] dt \tag{37}$$

where i = 1,2.

Following the pseudospectral discretisation and approximation process described in Section 3, the objective functions are calculated as follows:

$$J_i \approx \phi_i \left[x^N(1), t_f\right] + \frac{t_f - t_0}{2} \sum_{k=0}^{N} L_i \left[x^N(\tau_k), u^N(\tau_k), \tau_k\right] \omega_k \tag{38}$$

where $i = 1, 2$. NLP formulation of the BOOCP will lead to the following problem:

$$\min_{y} \{J_1, J_2\} \tag{39}$$

subject to:

$$g_j(y) \leq 0, (1 \leq j \leq r) \tag{40}$$

$$h_k(y) = 0, (1 \leq k \leq s) \tag{41}$$

$$y_{li} \leq y_i \leq y_{ui}, (1 \leq i \leq n_y) \tag{42}$$

Solving the BOOCP problem results in a Pareto set of optimal solutions. A standard NLP solver is limited to solving only single objective optimisation problems. Therefore, a scalarisation technique needs to be used to reduce the BOOCP to a series of SOOCPs to populate the Pareto frontier. The three most applied scalarisation methods in the literature are the linear weighted sum (LWS), the normal boundary intersection (NBI) and the normal constraint (NC) methods. In previous work, the authors of this paper have

proposed a new method, that of the adaptive bisection ε-constraint method which has been shown to perform better than the reviewed methods. Hence, the adaptive bisection ε-constraint method was adopted to generate Pareto frontiers for the BOOCPs in this work.

6 The adaptive bisection ε-constraint method

The adaptive bisection ε-constraint method [6], is a scalarisation method which is used for the generation of optimal Pareto frontiers in bi-objective optimisation problems, in the context of this work, BOOCPs. Similar to other scalarisation methods, the Pareto frontier is generated by solving a sequence of single-objective optimisation problems in a systematic manner. In this method, one of the objective functions is selected to be optimised while the other is converted into an additional constraint, leading to a solution that can be proven to always be weakly Pareto optimal. Systematic modification of the value of the objective function forming the additional constraint leads to the generation of an evenly distributed Pareto frontier.

The method transforms the BOOCP into n_p SOOCPs, where n_p is user specified and it determines the target number of points on the Pareto frontier. The algorithm starts by obtaining the anchor points J_1^* and J_2^* of the BOOCP, corresponding to the minimum values of each of the performance indices through solving the two SOOCPs formulated in equation (39). The anchor points define the extremal points on the Pareto frontier, ensuring the ensuing scalarisation method does not fail to consider any part of the Pareto frontier in the optimisation process. The intersection of the lines $J_1 = J_1^*$ and $J_2 = J_2^*$ defines the utopian point J_u, which, albeit being an ideal solution, does not lie in the feasible region of the optimisation problem (Fig. 1).

The remaining $(n_p - 2)$ problems to be solved

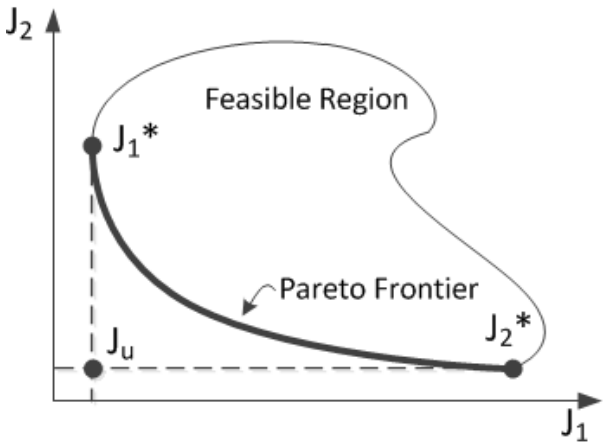

Fig. 1 Graphical representation of the design metric space of a BOOCP.

are formulated as follows:

$$\min_{y} \{J_1\} \tag{43}$$

subject to:

$$g_j(y) \leq 0, (1 \leq j \leq r) \tag{44}$$

$$h_k(y) = 0, (1 \leq k \leq s) \tag{45}$$

$$y_{li} \leq y_i \leq y_{ui}, (1 \leq i \leq n_y) \tag{46}$$

with the additional constraint:

$$J_2 = \varepsilon_c. \tag{47}$$

The value of ε_c for each of the remaining SOOCPs is calculated as follows. Once the the anchor points are determined, the utopian line, which is a straight line joining the anchor points, is bisected to obtain the first value of ε_c in equation (47). The solution of the resulting SOOCP will lead to an additional point on the Pareto frontier. Sometimes this will lead to an infeasible problem that leads to no solution. In this case, the line joining the anchor points is subdivided into four sections and the constraint ε_c is set to the value of J_2 at one-fourth the length of the line. If the problem is still infeasible, the value of J_2 at three-quarters of the line joining the anchor points is then tried. The line will continue being bisected until a solution is found or a constant K set by the user is reached.

Once an additional point J_3^* is found, the euclidean distance between the point and other points on the Pareto frontier is determined. The two points with minimal euclidean distance are then used to find an additional point on the Pareto frontier by using the method of line bisection as previously described. This process is repeated until the number of Pareto points n_p requested by the user is found. For each point on the frontier, the optimal state and control trajectories can be easily deduced from the optimisation variables.

7 Aircraft Trajectory Optimisation

An aircraft trajectory optimisation problem was considered in this paper to demonstrate the validity of the techniques presented. The trajectory optimisation problem involves generating a Pareto frontier of optimal climb trajectories for an aircraft flying from 35 ft above screen height to a cruising altitude of 35,000 feet while covering a range of 900 kilometres. Two cost functionals were considered in this case, the minimization of flight time and the minimization of fuel consumption.

7.1 Problem formulation

For such a problem, an aircraft performance model (APM) was developed to model the physical non-linear dynamic response of a generic A320 airliner. BADA coefficients [2] were used to develop the APM, following the methodology of Glover and Lygeros [9], described in the following state equations:

$$\dot{X} = V\cos\gamma \tag{48}$$

$$\dot{h} = V\sin\gamma \tag{49}$$

$$\dot{V} = -\frac{C_D S \rho V^2}{2m} - g\sin\gamma + \frac{T_{max} T_R}{m} \tag{50}$$

$$\dot{m} = -f \tag{51}$$

where X is the distance (range) covered by the aircraft on the ground, h is the height above ground level (AGL) (in this example it is considered as the altitude with the provision that the ground elevation is zero at the mean sea level (MSL)), V is the true airspeed (TAS) of the aircraft and m is the mass of the aircraft. X, h, V and m constitute the four states of the model which can be used only for vertical profile trajectories due to an omitted degree of freedom (DOF) in the lateral axis of the aircraft.

Event	Initial or Final Condition
1	X_i = 0 km
2	h_i = 35 ft
3	V_i = 165.2 kts
4	m_i = 68 tonnes
5	X_f = 900 km
6	h_f = FL 350

Table 1 BOOCP event constraints.

The control inputs to the aircraft are the flight path angle γ, and the thrust ratio T_R which is a fraction of the maximum thrust available from the engines at a particular altitude, T_{max}. The engine model integrated with the APM is a simplistic model of a turbofan engine as described in the BADA user manual [2].

C_D is the coefficient of drag which varies with the aircraft configuration i.e. take-off, initial climb or clean. S represents the total area of the lifting surfaces on the aircraft, ρ is the air density at a particular value of h assuming standard atmospheric conditions and f is the rate of fuel consumption. Finally, g is the gravitational acceleration assumed to be constant.

The initial and final conditions of the aircraft, formulated as event constraints in the BOOCP problem formulation are defined in Table 1. Furthermore, path constraints were applied to the trajectory optimisation problem which puts realistic limits on the performance of the aircraft and its propulsion system. The first path constraint is the stall speed of the aircraft which is a function of the aircraft configuration. The upper end of the speed scale is limited by V_{MO} and M_{MO} which are the maximum operating speed and the maximum operating Mach number respectively. Finally, the maximum thrust provided by the engines is also formulated as a path constraint.

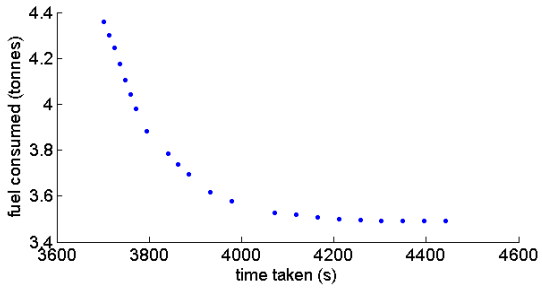

Fig. 2 Pareto frontier for minimum time and minimum fuel burn.

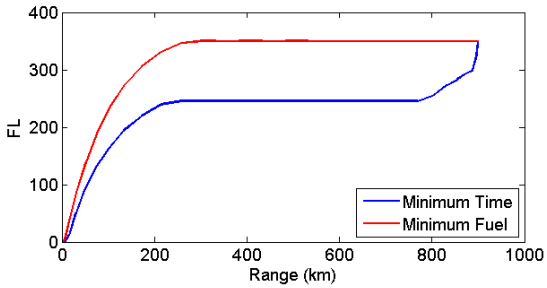

Fig. 4 Altitude profile for minimum time and minimum fuel burn.

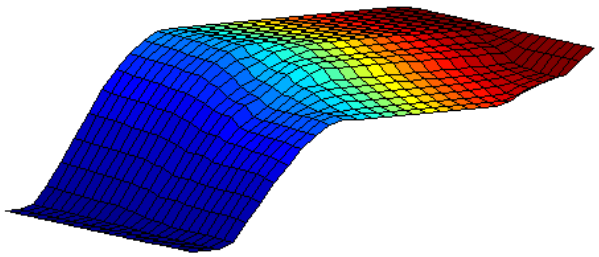

Fig. 3 Altitude profiles for Pareto set.

7.2 Results analysis

The Pareto frontier of the aircraft trajectory optimisation problem is presented in Figure 2. The frontier is evenly distributed and illustrates the utopian solutions at the extreme ends of the frontier, and intermediate solutions that provide a compromise between fuel consumed and flight time for a particular trajectory. The altitude-range profiles of the complete Pareto set are illustrated in Figure 3 as a three-dimensional surface plot. The trajectory at the front of the surface is the minimum time trajectory, whereas the profile at the far back represents the trajectory which consumes the least fuel. The complete surface is filled with successive plotting of trajectories representing the rest of the points in the Pareto optimal set. The discussion henceforth will focus on the utopian solutions.

Figure 4 illustrates the altitude-range profile for the two extremal solutions. On one hand, the minimum time solution consumes 4.36 tonnes of fuel and completes the trajectory in 1 hr 2 mins. On the other hand, the minimum fuel solution consumes 3.49 tonnes of fuel in a flight time of 1 hr 14 mins. This means that for this specific problem, an increased flight time of 12 mins over a 900 km leg can be traded for approximately 20% of the fuel consumed.

The minimum fuel trajectory involves a steep climb to the cruising altitude at which flight level the aircraft flies the rest of the leg. This is expected since gas turbines are most efficient at high altitudes where the air density is lower, which results in less drag on the aircraft. However, it is worth noting that the upper bound on the altitude in the BOOCP formulation was that of 45,000 ft giving the aircraft the possibility to fly higher than 35,000 ft in the level flight phase followed by a descent to the requested altitude at the end of the leg. This did not materialise, however, since the cost of gaining additional gravitational potential energy was larger than the savings in fuel resulting in flying a few flight levels higher.

It is interesting to note that the minimum time trajectory follows a completely different strategy. The flight starts with a shallow climb up to around 24,500 ft, followed by a level flight and finally a shallower climb to the requested altitude of 35,000 ft. The altitude the aircraft adopts to fly level at is the cross-over altitude i.e. the altitude at which the maximum operating speed and the maximum operating Mach number coincide. Consequently, this happens to be the altitude at which the aircraft can fly at the maximum true airspeed which results in minimum flight time.

In both cases, the climb from 35 ft occurs at

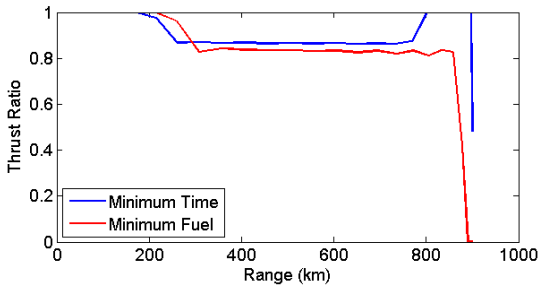

Fig. 5 Thrust ratio profile for minimum time and minimum fuel burn.

the maximum thrust provided by the engine as can be seen in Figure 5. In the minimum time trajectory it is clear that engine thrust is kept at the maximum level permissible by operational constraints to gain the maximum possible airspeed which will result in minimal flight time. When minimising fuel consumption, however, this is less obvious because fuel consumption is directly related to thrust levels. Reducing the thrust during the climb while keeping a constant flight path angle will result in a decrease in the rate of climb. As a result, it will take longer to reach the target altitude. Since the total fuel consumed during the climb is the integral of the rate of fuel consumption for the duration of the trajectory, the longer the flight time, the larger the fuel consumption. The optimisation results suggest that it is cheaper to climb at high thrust levels and high rates of fuel consumption at high rates of climb rather than by simply keeping low thrust levels for a prolonged time. This is also understandable in the context that, for a given calibrated airspeed (CAS), TAS increases disproportionately with altitude. In still air, such as the case considered, TAS is what defines the ground speed and hence the flight time. As a result, it is expected that it will be advantageous to expedite climb beyond that providing minimum fuel burn during climb in order to reach higher altitudes quicker.

The true airspeed and flight path angle (FPA) profiles illustrated in Figures 6 and 7 correlate perfectly with the observations previously made. The minimum time trajectory yields a speed profile which is larger in magnitude throughout the whole flight range over the minimum fuel counterpart. The former adopts a TAS of 495 kts, equivalent to 0.86 Mach, at the crossover altitude, while the latter flies at a nearly constant TAS varying from 419 kts to 410 kts at 35,000 ft. This corroborates with the fact that during cruise, the aircraft would be expected to climb or slow down gradually as the aircraft becomes lighter through progressive fuel burn. In this case, the latter strategy was adopted. The steep climb of the minimum fuel trajectory is reflected in Figure 7 in which a maximum FPA of 6.2^{o} is adopted as opposed to a maximum FPA of 4.4^{o} in the minimum time case. The minimum time trajectory exhibits a lower climb gradient than its counterpart in order to afford a quicker TAS during the climb, since in both cases the engine thrust is set to the maximum level. Furthermore, in the minimum time trajectory, the second climb at the end of the trajectory, which is shallower than the first with a maximum FPA of 1^{o}, is explained by the fact that at higher altitudes turbine engines generate less thrust due to the thinner atmosphere. Moreover, no trade-off is occurring between kinetic and potential energy to provide the additional rate of climb until the very end of the flight.

In Figures 4 to 7, it can be observed that the end of the trajectory is salient. In particular, in the minimum time profile the last 5,000 ft are climbed in just 12 km. Clearly, this is not practical for both operational and safety reasons. The speed profile during this time indicates a rapid decrease in TAS from 495 kts to 363 kts. The speed change occurs due to the trading of kinetic energy (speed) with potential energy (altitude) and also due to a rapid reduction in the thrust ratio. This solution falls within the feasible search space of the optimiser since no final condition is set on the speed in the problem formulation. A similar but less prominent strategy is adopted in the minimum fuel trajectory. In this case, the aircraft is already flying at the cruise altitude requiring only to cover further range to complete the mission. Since fuel consumption is of primary importance, in the final part of the trajectory the thrust is reduced to zero to conserve fuel leaving the aircraft to decelerate slowly to-

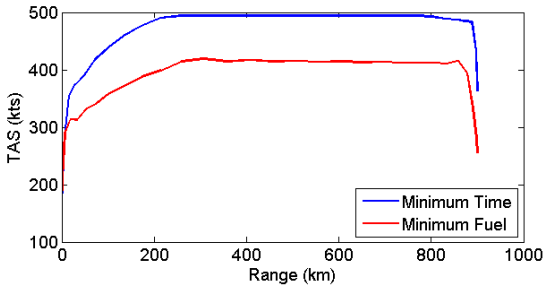

Fig. 6 True Airspeed profile for minimum time and minimum fuel burn.

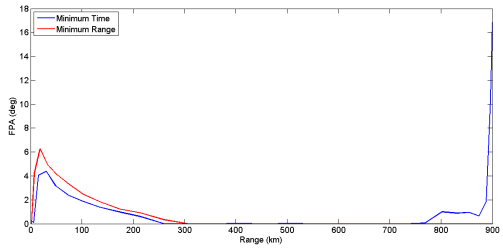

Fig. 7 Flight path angle profile for minimum time and minimum fuel burn.

wards the stall speed as it flies through the atmosphere which produces a retarding drag force on the airframe. Theoretically this is possible since no constraints in the problem formulation are compromised even though for practical reasons such a flight is not flown.

8 Conclusion and Future Work

This paper has presented the adoption of a number of techniques to solve optimal control problems using direct methods to find optimal trajectories. In particular, the adaptive bisection ε-constraint scalarisation method was adapted to enable the generation of Pareto frontiers for BOOCPs. The techniques were then applied to a bi-objective aircraft flight trajectory optimisation problem. This included a climb from 35 ft to a cruise altitude of 35,000 covering a range of 900 km bound by operational constraints in terms of speed and thrust provided by the engines. The results were analysed in depth and have illustrated the strategies adopted by the optimisation process in the presence of such constraints. Moreover corroboration with flight performance theory was confirmed. The work has formed the basis of a trajectory optimisation tool which will be used to solve more complex and practical problems in the future.

The ultimate goal of the ongoing research is to develop innovative real-time trajectory optimisation methods which have the potential to be used in the next generations of ATM Planning/Negotiation Systems and Avionics Flight Management Systems. The recent advent of Performance-Based Navigation (PBN) concepts represents a shift from sensor-based to performance-based navigation. PBN specifies that aircraft navigation systems performance requirements be defined in terms of accuracy, integrity, availability and continuity required for the proposed operations in the context of a particular airspace, when supported by an appropriate Air Traffic Management (ATM) infrastructure. The extensive introduction of advanced Communication, Navigation and Surveillance (CNS) technologies in the Civil/Military ATM field is reshaping the global air traffic network. However, the international aviation community is now facing important integration and harmonization challenges for the US NextGen (Next Generation Air Transportation System) and the European SESAR (Single European Sky ATM Research) initiatives. Important research efforts are also ongoing to demonstrate the feasibility of Avionics/ATM technologies capable of contributing to the emissions reduction targets set by the European governments. Significant contributions are possible in the areas of trajectory optimisation, air traffic flow management and control, airport operations and capacity analysis, as well as aircraft systems integration and performance enhancement.

References

[1] *MATLAB version 7.11*. Natick, Massachusetts: The Mathworks Inc., 2010.

[2] *User Manual for the Base of Aircraft Data*

(BADA) Revision 3.7. Brétigny-sur-Orge, France: EUROCONTROL Experimental Centre, 2009.

[3] Betts, J. T., *Survey of numerical methods for trajectory optimisation*. Journal of Guidance, Control, and Dynamics, vol. 21, no. 2, pp. 193-207, March-April 1998.

[4] Betts, J. T., *Practical methods for optimal control using non-linear programming*. SIAM: Philadelphia, PA, 2001.

[5] Betts, J. T., Campbell, S. L. and Engelsone, A., *Direct transcription solution of inequality constrained optimal control problems*. Proceeding of the 2004 American Control Conference, Boston, Massachusetts, June 30 - July 2, 2004.

[6] Chircop, K. and Zammit-Mangion, D., *On ε-constraint based methods for the generation of Pareto frontiers*. Manuscript submitted for publication.

[7] Conway, B. A., *A Survey of Methods Available for the Numerical Optimisation of Continuous Dynamic Systems*. Journal of Optimisation Theory and Applications, vol. 152, no. 2, pp. 271-306, 2012.

[8] Gill, P. E., Murray, W. and Wright, M. H., *Practical Optimisation*. Academic Press Inc: San Diego, CA, 1987.

[9] Glover, W. and Lygeros, J., *A Multi-Aircraft Model for Conflict Detection and Resolution Algorithm Evaluation*. Technical Report WP1, Deliverable D1.3, Version 1.3, HYBRIDGE, February 18, 2004.

[10] Khan, S., *Flight Trajectories Optimisation*. 23rd Congress of the International Council of the Aeronautical Sciences (ICAS), Toronto, Canada, September, 2002.

[11] Ross, I. and Fahroo, F., *Legendre pseudospectral approximations of optimal control problems*. Nonlinear Dynamics and Control and their Applications, vol. 295, pp. 327-342, 2004.

[12] Stryk, O. and Bulirsch, R., *Direct and indirect methods for trajectory optimisation*. Annals of Operations Research, vol. 37, no. 1, pp. 357-373, 1992.

[13] Trefethen, L. N., *Spectral Methods in MATLAB*. SIAM, Philadelphia, PA, 2000.

[14] Wächter, A. and Biegler L. T., *On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming*. Journal of Mathematical Programming, vol. 106, no. 1, pp. 25-57, 2006.

Copyright Statement