Efficient methods for flight envelope estimation through reachability analysis

Stapel, J; de Visser, CC; van Kampen, EJ; Chu, QP

DOI
10.2514/6.2016-0083

Publication date
2016

Document Version
Accepted author manuscript

Published in
Proceedings of the AIAA guidance, navigation, and control conference

Efficient Methods for Flight Envelope Estimation through reachability analysis

J.C.J. Stapel[*] C.C. de Visser[†] Q.P. Chu[‡] E. van Kampen[†]

This paper contains a study to find faster numerical methods for Hamilton-Jacobi Isaacs partial differential equations in application to model-based flight envelope estimation. The aim is to identify new methods capable of providing the basic information needed for flight envelope protection. Useful insights have been obtained through assessing the reachable set theory associated to the problem, which permits integration of depth-first and breadth first estimation methods and the estimation of the flight envelope as a minimal time problem. The applicability of a class of non-iterative schemes, known as the fast marching methods, has been evaluated. The behavior of the studied methods is demonstrated on different example problems, including a simplified aircraft model.

I. Introduction

SAFETY has always been of primary importance in aviation. Even today the flight control community continues to improve safety by extending their systems' capabilities beyond the occurrence of non-critical failures or unanticipated changes in the system dynamics. Good progress has been made through adaptive dynamic model estimation and stability argumentation. An open problem is however to maintain an accurate estimate of the flight envelope, which is an essential tool in the prevention of loss of control.

loss of control (LOC) is considered a primary cause of fatalities in aviation.[?] Between 2002 and 2012 almost 40% of all fatal accidents are related to LOC.[?] In these accidents the most important causal factors were identified to be related to flight crew handling or mishandling after a technical failure.

The current practice is to estimate accurate flight envelopes beforehand by extensive flight tests and calculations. Although this method suffices under nominal conditions, it fails when the system dynamics change in the event of a failure. The unavailability of this protection system in impaired situations can be considered a significant weakness in the robust and adaptive methodology.

One promising model-based approach to resolve this problem is advocated in Tang et al.[?] where the flight envelope is described as the intersection between two non-linear, non-convex reachability problems. These reachability problems can be described by the Hamilton Jacoby Bellman (HJB) partial differential equation (PDE) or by the Hamilton Jacoby Isaacs (HJI) equation when disturbances are to be incorporated in the analysis. These equations have been solved numerically for dynamic flight envelopes with the Level Set method.[?,?,?] Due to the high computation times, this method cannot yet be considered for realistic real-time applications. Several studies have been undertaken in a search for faster or better performing variants of this method.

Adalsteinsson and Sethian[?] localize the computational domain to a narrow region around the front of the reachable set which reduces the number of calculations per iteration. Van Oort et all.[?] uses a semi-Lagrangian Level Set method to avoid the Courant-Friedrichs-Lewy (CFL) stability criteria which is otherwise limiting the permissible time step. De Weerdt and van Oort[?] replace the grids with interval analysis. Li et al.[?] adapts the partial differential equation to avoid the need for re-initialization, which may be necessary in some occasions. Govindarajan et al.[?] explore the opportunities of using simplex splines. Stipanovic

[*]MSc student, Department of Control & Simulation, Delft University of Technology, Kluyverweg 1 2629 HS Delft, student

[†]Assistant professor,Department of Control & Simulation, Delft University of Technology, Kluyverweg 1 2629 HS Delft, Professional Member.

[‡]Associate professor, Department of Control & Simulation, Delft University of Technology, Kluyverweg 1 2629 HS Delft, Professional Member.

and Tomlin[?] make over-approximations of the reachable sets with a polytopic approximation through feedback linearization. Helsen[?] constructs the reachable set with a depth-first approach where distance fields are computed over the grid with globally optimal controlled trajectories. Good introductory reads on the propagating front theory are chapters 10-13 of van Oort[?] and the books by Osher and Fedkiw[?] and Sethian.[?]

This study is an attempt to explore new ways to reduce the computational complexity of the reachable set approach. It will be assumed that the system is non-linear, but time-invariant before and after the failure event. It will be further assumed that the system model is correctly identified after the occurrence of the failure. It should be noted that this assumption is not entirely realistic since the current adaptive system estimation techniques only provide locally correct models, and may require some iterations to converge.

This paper considers stationary schemes for solving the HJI PDE and proves applicability to the search for flight envelopes. From a quality perspective the use of this approach is discouraged by Mitchell[?] as the method is limited to first order sub-grid accuracy, can form discontinuities and lacks additional information outside the reachable set. Mitchell also cautions for disappointing computational advantages. The occurrence of discontinuities is however intentional as the use of Huygens' principle of first arrival avoids the augmented viscosity needed in the Level Set method to find the correct weak solution. Although additional information outside the reachable set can be useful, it is not essential for determining the flight envelope itself and may therefore be sacrificed for an increase in computational performance. Additional advantage of the boundary value formulation is that it is not sensitive to the CFL stability criteria, which restricts the permissible step size in the Level Set method. The absence of global integration steps also makes the method more suitable for integration with depth-first methods or for localized set propagation, which may proof beneficial in real-time applications.

This paper will start with a brief overview of the reachable set theory. A formal proof will be given for the applicability of the boundary value formulation, along with some further insights on localized set propagation. The behavior of some of the solving schemes for the boundary value formulation will be investigated on a set of test problems. Finally a recommendation for further investigation will be formulated.

II. Description of the reachable set theory

The reachable sets can be defined as follows.[?] Consider a dynamic system Σ with dynamics $\dot{x} = f(x,u,d,t)$ where x is a state in state space χ, t is the time variable and $[u,d]$ are control and disturbance input signals from the permissible bounded sets $[\mathcal{U}(x,t), \mathcal{D}(x,t)]$. A trajectory is denoted as $\xi_{x_0,t_0,u(\cdot),d(\cdot)}(t) : t \to x \in \chi$, where x_0 is the initial state at $t = t_0$. The initial set $\mathcal{I}$ and target set $\mathcal{T}$ are defined such that $\mathcal{I}, \mathcal{T} \in \chi$.

A backwards reachable set is the collection of states from which the target set can be reached at time t_f when starting at time t while a forwards reachable set can be reached at time t when starting in the initial set at time t_0:

$$\begin{aligned} \text{Backwards reachable set: } \mathcal{R}^B_{\mathcal{T},t_f}(t) &:= \{x \in \chi : \forall d \in \mathcal{D}, \exists u \in \mathcal{U} | \xi_{x,t,u,d}(t_f) \in \mathcal{T}\} \\ \text{Forwards reachable set: } \mathcal{R}^F_{\mathcal{I},t_0}(t) &:= \{x \in \chi : \forall d \in \mathcal{D}, \exists [x_0,u] \in [\mathcal{I},\mathcal{U}] | \xi_{x_0,t_0,a,b}(t) = x\} \end{aligned} \tag{1}$$

The dynamic flight envelope for system Σ and $\mathcal{I} = \mathcal{T} := \mathcal{K}$ can now be represented as the intersection between the forward and backward reachable set. Apart from the reachable set, one may also observe the reachability tube, which is the union of all reachable sets over the observed time horizon. The reachability tube will be denoted as $\mathcal{R}_{\mathcal{K}}(\leqslant t) = \cup \mathcal{R}_{\mathcal{K}}(t)$. This study also uses the concept of a maintainable set which is the union of all maintainable or trimmable states $\{x | \exists [u,d] \in [\mathcal{U},\mathcal{D}] : \dot{x} = 0\}$. Further set types and their properties have been identified by Mitchell[?] and Kaynama et al.[?]

A. Level set method

To find the reachable sets efficiently the optimally controlled trajectories have to be identified. In the most general case the problem involves both a controller as a disturber. This optimality problem forms a differential game and has been used by Mitchell[?] and Lombaerts.[?] The differential game theory as well as the related differential equations are well described by for instance Evans[?] or Pierre.[?]

For the Level Set method the reachable set is described by a pursuit-evader zero-sum differential game with an objective function $J : u, d \rightarrow \mathbb{R}$ of the form $J(u,d) = g(x(t_f))$, where g is chosen such that it satisfies equation 2, where c is any constant; typically chosen to be zero. The reachable sets can then be found as stated in theorem 1, where Φ_t and $\bigtriangledown_x \Phi$ are the time derivative and spatial gradient of Φ.

$$g(x) = \begin{cases} < c & \forall x \in \mathcal{K} \\ = c & \forall x \in \delta\mathcal{K} \\ > c & \forall x \in \mathcal{K}^{-1} \end{cases} \tag{2}$$

Theorem 1. *Let Φ^+ be the viscosity solution of the (terminal) upper Isaac's equation and Φ^- the viscosity solution for the (initial) lower Isaac's equation:*

$$\mathcal{I}_B^+ \begin{cases} \Phi_t^+ + H_B^+(t, x, \bigtriangledown_x \Phi^+) = 0 \\ \Phi^+(t_f, x) = g(x) \end{cases} \qquad \mathcal{I}_F^- \begin{cases} \Phi_t^- + H_F^-(t, x, \bigtriangledown_x \Phi^-) = 0 \\ \Phi^-(t_0, x) = g(x) \end{cases} \tag{3}$$

where $H_B^+(t,x,p) = \min_u \max_d \{f(t,x,u,d) \cdot p\}$ and $H_f^-(t,x,p) = \max_u \min_d \{f(t,x,u,d) \cdot p\}$. Then $\mathcal{R}^B_{\mathcal{T},t_f}(t) = \{x : \Phi(x,t) \leqslant 0\}$ and $\mathcal{R}^F_{\mathcal{I},t_0}(t) = \{x : \Phi(x,t) \leqslant 0\}$.

This theory states that the reachable set can be found by tracing the intersection between the defined value function and a horizontal plane at elevation c. The set is initialized such that this intersection represents the boundary of the initial set. This concept is illustrated in figure 1

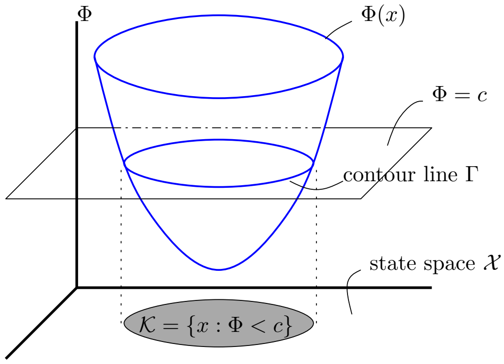


Figure 1: Illustration of the level set representation in its initial state

Together with its extended forms, the Level Set method is a commonly used reachability method as it covers a very general class of problems. It can be solved numerically with the schemes by Osher.[?] A Matlab toolbox implementing these schemes is developed by Mitchell.[?] The reachable sets of stationary problems can also be described by a boundary value formulation.

B. Minimal time problem

In a SIAM paper[?] Falcone and colleagues note that the burn equation, which is normally computed as a level set problem, can also be solved with the stationary boundary value formulation of the HJB PDE. The trick is that the reachability problem is not modeled as a pursuit-evasion differential game, but rather as a minimal time problem: $J(x,u,d) = \min(t : \xi_{x,u,d}(t) \in \mathcal{K})$. When a state is never reached, the assigned value defaults to $+\infty$ (or $-\infty$ in case of backward reachability). This formulation directly forms the HJI PDE given in theorem 2.[?,?,?]

Theorem 2. *Let T^+ and T^- be the viscosity solution of the upper and lower Isaac's equation:*

$$\mathcal{I}_B^+ \begin{cases} -H_B^+(x, \bigtriangledown_x T) = 1 \\ T(x \in \mathcal{K}) = t_f \end{cases} \qquad \mathcal{I}_F^- \begin{cases} H_F^+(x, \bigtriangledown_x T) = 1 \\ T(x \in \mathcal{K}) = t_0 \end{cases} \tag{4}$$

where $H_B^+(x,p) = \min_u \max_d \{f(x,u,d)\cdot p\}$ *and* $H_F^+(x,p) = \max_u \min_d \{f(x,u,d)\cdot p\}$. *Then* $\mathcal{R}^B_{\mathcal{K},t_f}(\geqslant t) = \{x : T(x) \geqslant t\}$ *and* $\mathcal{R}^F_{\mathcal{K},t_0}(\leqslant t) = \{x : T(x) \leqslant t\}$.

Note that the minus sign in the backward reachability problem disappears when a virtual time scale $\tau = -t$ is used. Since the method only tracks minimal arrival time, the boundary value formulation can only represent problems in which either $H < 0 \forall x, t$ or $H > 0 \forall x, t$ holds.

The minimal time problem comes with a different assortment of numerical schemes; each with its own advantages and limitations. A good review is given in.[?] The simplest scheme is referred to as the iterative method (ITM). It is generally applicable and the order of complexity is comparable to that of the Level Set method. Assuming N nodes in each dimension and a comparable number of iterations, the computational complexity is of the order $\mathcal{O}(N^{n+1})$. The fast sweeping method (FSM)[?] is a variant that updates the nodes in a directional manner. This modification improves the rate of convergence by exploiting the directionality of information flow in at least one of the 'sweeps' over the grid. Apart from this advantage, the methods have comparable complexity and applicability.

The Fast Marching method (FMM)[?] avoids the necessity for iteration completely by tracking the causality in which nodes will be reached. The complexity is $\mathcal{O}(N^n \log N^n)$. The original method is only suitable for Eikonal problems, but has been extended to the ordered upwind method (OUM)[?,?] which uses larger stencils to better cope with anisotropy, but also increases the overhead. Since the Fast Marching method calculates the nodes in a causal order, it is less suitable for parallelization. An alternative extension is the iterative Fast Marching method (IFM).[?,?] The order of complexity of the IFM lies between that of the ITM and the FMM. The safe Fast Marching (SFM)[?] only progresses the front if the used ordered upwind method is capable of finding a valid approximation.

C. Proofs for properties and usability

The following theorems prove that the flight envelope can be formulated as a stationary reachability problem. They also provide further insights that allow to progress the front locally, that permit to combine depth-first methods with the breadth-first front propagation and that show how to simplify the optimization problem.

Proposition 1. *Consider a reachability problem for a time-invariant dynamic system* $\dot{x} = f(x,u,d)$. *Let* $\mathcal{K}$ *be a maintainable set, then* $\mathcal{R}_\mathcal{K}(t_1) = \mathcal{R}_\mathcal{K}(\leqslant t_1)$. *(The set reachable at a specific time equals the reachable tube with the same horizon).*

Proof. Consider the case where a state is reached for the first time on $t = t_1$. It is then also possible to reach this state at an arbitrary moment $t_2 \geqslant t_1$ by staying in the initial maintainable state for a period of $\Delta t = t_2 - t_1$ after which the same trajectory is followed. This trajectory is permissible because of the shifting property. □

Corollary 1.1. *Consider the same reachability problem as in proposition 1. For* $\tau \geqslant 0$ *we can state that* $\mathcal{R}_\mathcal{K}(t_1) \subseteq \mathcal{R}_\mathcal{K}(t_1 + \tau)$ *(The reachable set can only grow monotonically).*

Proof. This property follows logically from Proposition 1, since every reached state can still be reached every moment thereafter. □

Proposition 2. *For a reachability problem where the front represents a cluster of permissible trajectories, consider two initial sets* $\mathcal{K}_1$ *and* $\mathcal{K}_2 \subset \mathcal{K}_1$. *Then their respective reachable sets satisfy* $\mathcal{R}_{\mathcal{K}_2}(t) \subseteq \mathcal{R}_{\mathcal{K}_1}(t)$

Proof. Since $\mathcal{R}(t) = \bigcup \xi_{x_0,t_0,u}(t)$, superposition holds and one can simply split the problem $\mathcal{K}_1 = \mathcal{K}_2 \bigcup (\mathcal{K}_1 \setminus \mathcal{K}_2)$, solve $\mathcal{R}_{\mathcal{K}_2}(t)$ and $\mathcal{R}_{\mathcal{K}_1 \setminus \mathcal{K}_2}(t)$. Now $\mathcal{R}_{\mathcal{K}_1}(t) = \mathcal{R}_{\mathcal{K}_2}(t) \bigcup \mathcal{R}_{\mathcal{K}_1 \setminus \mathcal{K}_2}(t) \supseteq R_{\mathcal{K}_2}(t)$ □

Corollary 2.1. *Consider a subset* $\mathcal{V}$ *of the reachable set* $\mathcal{R}_\mathcal{K}(t_1)$ *in a time-invariant problem. Then* $\mathcal{R}_\mathcal{V}(\tau) \subseteq \mathcal{R}_\mathcal{K}(t_1 + \tau)$

Proof. This is an extension of proposition 2, where a subset is considered of an intermediate reachability solution. The shifting property of time-invariant systems shows that since $\mathcal{V} \subset \mathcal{R}_\mathcal{K}(t_1)$, $\exists \xi_{x_0 \in \mathcal{K}}(t_1) = x \in \mathcal{V} \ \forall x \in \mathcal{V}$. The principle of dynamic programming dictates that extending this trajectory subset over a period of τ gives $\mathcal{R}_\mathcal{V}(\tau)$. □

Lemma 1. *Consider a stationary reachability problem with a maintainable initial set. Then on a bounded domain of finite dimensions there exists a reachable set $\mathcal{R}_{\mathcal{K}}(t_f)$ such that $\mathcal{R}_{\mathcal{K}}(t_f) = \mathcal{R}_{\mathcal{K}}(t_2)$ for all $t_2 > t_f$. This set is referred to as the steady set and t_f is defined as the infinite horizon.*

Proof. Since the set can only grow (corollary 1.1), the trivial case of reaching the full domain proofs existence of this steady set □

Remark. t_f does not have to be finite, as the steady set can be approached asymptotically. In practice t_f is approached sufficiently once a threshold for globally slow set growth is reached.

Proposition 3 (inclusion theorem)**.** *Consider a solution set $\mathcal{R}_{\mathcal{K}_1}(t_1)$ found by solving a time-invariant system initialized from a maintainable initial set $\mathcal{K}_1$ over a time horizon t_1. Consider an arbitrary valid trajectory (or set of trajectories) ξ such that $\xi(t=0) \in \mathcal{K}_1$ and $\xi \subset \mathcal{R}_{\mathcal{K}_1}(t_1)$. Then there exists a $t_2 \leqslant t_1$ and $t_3 \geqslant t_1$ such that $\mathcal{R}_{\mathcal{K}_1}(t_1) \subseteq \mathcal{R}_{(\mathcal{K}_2=\mathcal{K}_1\cup\xi)}(t_2) \subseteq \mathcal{R}_{\mathcal{K}_1}(t_3)$*

Proof. The proof consists of two parts. First it can be shown that $\mathcal{R}_{\mathcal{K}_1}(t_1) \subseteq \mathcal{R}_{\mathcal{K}_2}(t_2)$. This property follows from lemma 2 since $\mathcal{K}_1 \subseteq \mathcal{K}_2$.

To prove that $\mathcal{R}_{\mathcal{K}_2}(t_2) \subseteq \mathcal{R}_{\mathcal{K}_1}(t_3)$, consider reaching set $\mathcal{R}^*_{\mathcal{K}_1}(t)$ by first Reaching $\mathcal{K}_2$ from $\mathcal{K}_1$ over a time horizon Δt by restricting the set of permissible inputs such that only ξ can be made. Then the set is grown normally till time $T + \Delta t$. Since a smaller set of permissible inputs is used one can say that $\mathcal{R}^*_{\mathcal{K}_1}(T+\Delta t) \subseteq \mathcal{R}_{\mathcal{K}_1}(T+\Delta t)$.

Since by definition, $\mathcal{R}^*_{\mathcal{K}_1}(\Delta t) = \mathcal{K}_2$ through the principle of dynamic programming it can also be stated that $\mathcal{R}^*_{\mathcal{K}_1}(T+\Delta t) = \mathcal{R}_{\mathcal{R}^*_{\mathcal{K}_1}(\Delta t)}(T) = \mathcal{R}_{\mathcal{K}_2}(T)$ and hence arrive at $\mathcal{R}_{\mathcal{K}_2}(T) \subseteq \mathcal{R}_{\mathcal{K}_1}(T+\Delta t)$ □

Corollary 3.1. *The two problems of proposition 3 with initial sets $\mathcal{K}_1$ and $\mathcal{K}_2$ have the same steady set, but may have different infinite horizons.*

Remark. These proofs only hold when the reachable sets represent clusters of trajectories. When the shape of the set has a physical meaning, like in the burn equations where temperature is depending on radius of curvature of the front, the superposition principle no longer holds in its used form.

D. Theorems' consequences

Prioritise regions of interest

In general a reachability calculation has to be initialized in the safe initial region and is propagated without any means to influence the progression of the front towards a region of interest. For real-time implementations, this requirement can entail considerable time before the reachability of a state far away from the initial set is verified. Proposition 2 and the inclusion theorem 3 provide the means to avoid this delay.

They show that it is possible to correctly identify reachable states from a larger initial set, provided that this set is contained in the reachable set of the original problem. The inclusion theorem also indicates that this set will find the reachable states sooner than the initial set alone. This also means that the steady set from lemma 1 can be found earlier. The only difference with the original solution is a change in the estimated arrival times.

It is therefore possible to combine depth-first trajectory optimization with the breadth-first dynamic programming principles of the moving front. A set of trajectories can be used to initialize the moving front in the vicinity of states of interest; for instance the regions near the aircraft's current state or close to where the limits are expected. The level set method (or any reachability scheme) can then start expanding the envelope directly in these regions. Even when there are no states of particular interest, a set of depth-first trajectories can be used to increase the area of the starting front, which also may lead to faster conversion of the sets. Although this trick may severely distort the original reachable set, it will correctly identify whether a state is reachable or not. This method is therefore recommended for further development.

Use of boundary value formulation

Corollary 1.1 proves that the monotonicity requirement can be satisfied by initializing the reachability problem with a trimmable initial set. The problem can therefore be fully described by the minimal time problem. The applicability of various solving schemes will be evaluated in section IV.

Localizing the problem

Corollary 2.1 proves that it is possible to find valid set regions when propagating only a subset of the set's front. This property gives potential application for localized front propagation. An attempt can be made to spend more computational resources to propagate the front in the vicinity of the current aircraft state and along the intended state transition. It should however be noticed that the flight envelope requires both a forward and backward reachable set. Since aircraft dynamics are highly asymmetric (i.e. a path traversed over the state-space can generally not be followed in both directions) it is unlikely that a both the forward and backward sets reach a state efficiently from the same initial state. Local extension may however work in locally controllable regions. Since aircraft are typically not locally controllable, this approach is not recommended for this application. It should further be noted that any form of local propagation will be very hard to accomplish with the level set method, as the algorithm has to iterate with globally equal time increments.

Reducing the optimization problem

The second proof of the inclusion theorem 3 shows that sub-optimal control can be used to find the reachable set faster. This insight shows that it is not required to reach all states in an optimal way. Instead of investing computational efforts in finding the optimal input, it suffices to identify any control that reaches the state. Once a state is reached, the principle of dynamic programming dictates that from there on the next state can be found irrespective of how the previous state was reached.

This approach reduces the optimal reachability problem to a sufficient reachability problem. The sub-optimality permits non-unique solutions on all time horizons. Still, since the actual arrival time is of a lesser interest than the quick availability of a correct flight envelope, this inconsistency may be acceptable in on-line envelope estimation.

It is expected that this observation can make an improvement in the selection criteria for the control inputs, and in case of the fast marching method may even circumvent the need to causally order the narrow band nodes, reducing the number of calculations by a factor $\log(N^n)$ and also improving the degree of parallelism.

III. Implementation of the Level Set schemes

For the purpose of verifying and demonstrating the behavior and properties of the Level Set method, a set of numerical schemes was implemented in MatLab based on Osher and Fedkew.[?] The implemented algorithms can solve two-dimensional problems with three different dissipation methods, second order Runge-Kutta integration and several options for handling differential games, augmenting maintainability and solving different reachability problems. Each implementation has been subjected to a variety of small tests to verify the operation and cooperation of the contributing functions. The algorithms were also compared to Mitchel's toolbox on a double integrator problem. Apart from a minor difference in the handling of the domain boundaries, the two implementations were found to produce the same results.

IV. Implementation of the minimal time schemes

For the minimal time problem, only the Eulerian Fast Marching and the (safe) semi-Lagrangian fast marching methods have been implemented. No satisfying external toolbox was found to be available. Two sets of algorithms have been developed. The first is capable of solving the classical Eulerian Fast Marching method for n-dimensional problems and has an integrated solving scheme for multiple input- multiple output (MIMO), input affine differential games. The second set is limited to two-dimensional problems but uses the semi-Lagrangian approach.

A. Eulerian method

The Eulerian scheme has been combined with an optimization scheme for non-linear, input affine MIMO problems. The scheme will be using first order upwind finite differencing. The assumed structure of the HJI

PDE is as follows:

$$\min_u \max_d \left(\left(\left(\begin{bmatrix} f_1 \\ f_2 \\ \vdots \end{bmatrix} + \begin{bmatrix} g_{1u_1} & g_{1u_2} & \cdots \\ g_{2u_1} & g_{2u_2} & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \end{bmatrix} + \begin{bmatrix} g_{1d_1} & g_{1d_2} & \cdots \\ g_{2d_1} & g_{2d_2} & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \end{bmatrix} \right) \right) \cdot \begin{bmatrix} \triangledown^{\pm}_{x_1} T_{ij\cdots} \\ \triangledown^{\pm}_{x_2} T_{ij\cdots} \\ \vdots \end{bmatrix} \right) = 1 \tag{5}$$

Where $\triangledown^{\pm}_{x_1} T_{ij\cdots}$ is either the lower ($\triangledown^{-}_{x_1} = \frac{(T_{i,j} - T_{i-1,j}}{\Delta x_1})$ or upper ($\triangledown^{+}_{x_1} = -\frac{(T_{i,j} - T_{i+1,j}}{\Delta x_1})$ spatial difference. In this implicit function the value $T_{i,j,\ldots}$ will be the unknown and either the left or right neighboring value will be available. Assuming that the optimal inputs u^*, d^* have been found, the PDE reduces to $f_{i,j}(u^*, d^*) \cdot \triangledown_x T_{i,j} = 1$ and it becomes straightforward to express $T_{i,j}$ explicitly:

$$T_{i,j} = \frac{1 + \sum_{i^-} f_i(u^*, d^*) \frac{T_{i-1}}{\Delta x_i} - \sum_{i^+} f_i(u^*, d^*) \frac{T_{i+1}}{\Delta x_i}}{\sum_{i^-} \frac{f_i(u^*, d^*)}{\Delta x_i} - \sum_{i^+} \frac{f_i(u^*, d^*)}{\Delta x_i}} \tag{6}$$

Where i^- are the dimensions in which $\triangledown^{-}_{x_1}$ is used, i^+ the dimensions with $\triangledown^{+}_{x_1}$ and in which T_{i-1} and T_{i+1} are the left and right neighbor in dimensions i respectively.

B. Semi-Lagrangian method

The semi-Lagrangian method is illustrated in figure 2. The theory below has been put together by using mainly the works of Cacace et al.[?,?] The idea is to select a grid point (for instance T_{ij}). From this starting position a trajectory can be calculated backwards in time until a certain distance criteria is met. On this end position, which is denoted as $\tilde{x}_{-\tau}$, the value $T_{\tilde{x}}$ is estimated by interpolation between the neighboring nodes (for instance $T_{i+1,j}$ and $T_{i,j+1}$). The new value at the starting node equals the interpolated value, plus the time τ spend for the trajectory to reach x_{ij} from $\tilde{x}_{-\tau}$.

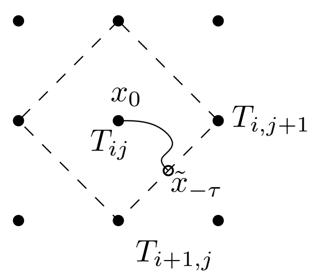


Figure 2: Semi-Lagrangian method

The semi-Lagrangian (SL) scheme can be seen as a discrete model. (See chapter 1.5 of[?]). For the minimal time problem the first order discretized PDE is given in equation 7, where h represents the time $-\tau$ necessary to reach the line or distance where the grid values are interpolated. Also the Kružkov transformed PDE is given with $\beta = e^{-h}$, which transforms $v = 1 - e^{-T}$ to bound the time domain to $t \in [0, 1]$:

$$\begin{aligned} T_h(x) &= \inf_a \sup_b \{T_h(x + hf(x, a, b)\} - h \\ v_h(x) &= \inf_a \sup_b \{\frac{1}{\beta} v_h(x + hf(x, a, b)\} + 1 - \frac{1}{\beta} \end{aligned} \tag{7}$$

These expressions can be implemented on a sampled grid in several ways. Here the two-point stencil from figure 2 will be used as well as a $3x3$ multi-stencil scheme. The two-point stencil is implemented for the normal and Kružkov transformed case. The multi-stencil has been built for the safe fast marching scheme. It adapts its neighbor usage depending on the availability of converged nodes and the direction of the velocity vector. In this case, the stencil is limited to the eight neighboring nodes. Without loss of generality, the node selection logic can be limited to the three neighboring nodes of one quadrant. Using the notation given in figure 3, five situations can be distinguished:

1. No neighbors are available
2. Neighbor n_1, n_2 or n_3 is available
3. Neighbors $\{n_1, n_2\}$ are available
4. Neighbors $\{n_1, n_3\}$ (or $\{n_2, n_3\}$) are available
5. Neighbors $\{n_1, n_2, n_3\}$ are available

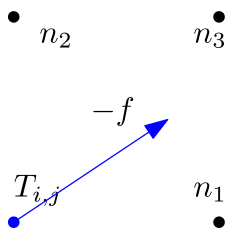


Figure 3: Neighbor notation in considered quadrant

In the first situation no neighbors exist with an accepted value. For the considered velocity it is not possible to calculate a candidate arrival time. Therefore the candidate value is set to $+\infty$. In the second situation only one of the neighbors is available. The safe fast marching scheme dictates that no interpolation is possible. The scheme can therefore only find a value if the negative velocity vector points exactly towards this neighbor. In all other cases, the candidate value is set to $+\infty$. The third situation has two neighbors and therefore allows to interpolate. This interpolation however is not generally applicable, since the third node is not accepted. An invalid situation can occur when a node is approached by two fronts, in which case the interpolation leads to unnecessary dissipation. Therefore a value may only be calculated if the vector points to one of the accepted nodes. In the fourth case it is possible to calculate a value if the negative velocity vector points somewhere between the two accepted nodes. In all other cases the value is set to $+\infty$. In the final situation all the required nodes are available to use the two- or three-point stencil. In this situation a finite value is guaranteed to exist.

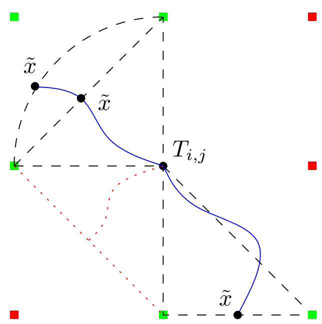


Figure 4: Geometry for semi-Lagrangian multi stencil

All cases but the first are illustrated in the quadrants of figure 4, where green nodes have converged values and red nodes are unknown.

C. applicability of the FM schemes

Both the Eulerian and the semi-Lagrangian Fast Marching method were found not to be applicable to anisotropic problems with uncontrollable regions. Even when an incorrect arrival time is acceptable, the Fast Marching method will not be able to separate reachable states from unreachable states. The update schemes assume that all upwind neighboring nodes have a converged value. In case of an anisotropic flow this assumption is not necessarily the case. For the classical Fast Marching methods there are two methods to resolve situations where the neighboring nodes are not available.

Sethian[?] suggests for the Eulerian method to ignore the dimensions in which no converged neighbour is available. The consequence is illustrated in figure 5. By reducing the dimensionality of the problem, it becomes possible to incorrectly conclude that a node is reachable. If the projected problem is extended back into the original size, the missing dimensions are extrapolated as flat, infinite extensions. It therefore ignores the finiticity of the set that would otherwise prevent a node from being reached.

For the semi-Lagrangian method Falcone[?] suggests to use the Kružkov transform. This transformation permits to interpolate between converged and unknown states. This interpolation will fail in unreachable regions as the interpolation will always result in a finite arrival time. The same argument holds when a finite

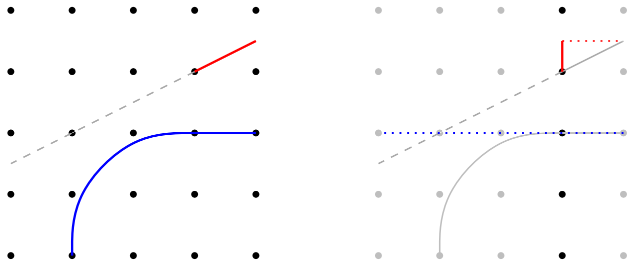


Figure 5: Illustration of cause of leakage. Left: node will not be reached from set. Right: Projected problem is reachable.

time ceiling is used instead of a Kružkov transform.

The SFM method handles the absence of converged nodes by not calculating a solution when the required information is not available. This abstention avoids the risk of calculating a finite value for an unreachable state but also limits the capabilities of identifying the reachable states. Consider the two situations visualized in figure 6. The evaluated node is not reached according to the safe semi-Lagrangian fast marching method, while in reality a trajectory can be found that connects the reachable set to the node. In the left case, the flow vector points directly to the front of the set, but does not reach it within the stencil space around the considered node. In the right case, the linearized trajectory misses the front altogether, while the continuous (and in this example, also the piecewise linearized) trajectory reaches the front. The ability to identify reachable nodes improves with the used limited by the stencil size.

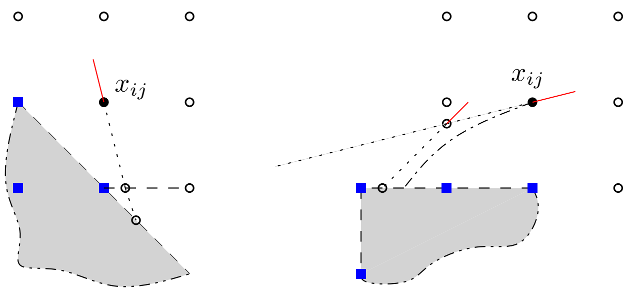


Figure 6: Two situations where the considered node is incorrectly ignored by the safe SL FM scheme

Despite these limitations, the safe semi-Lagrangian Fast Marching method is well capable of calculating accurate arrival times for all evaluated nodes, but may not be able to evaluate all reachable states. In practice this means that for some problems this single-pass method can be used to find part of the reachable set, after which the solution can be refined with a more robust but iterative method.

V. demonstrations and evaluations

The derived properties are demonstrated on three example cases. A simple Eikonal equation is solved with the Level Set method as well as the Fast Marching and safe Fast Marching method in comparison to the analytic solution. A mass spring damper system is then used to show the behavior of these methods on an anisotropic problem with uncontrollable regions. Finally the safe Fast Marching method is applied to a simplified aircraft model.

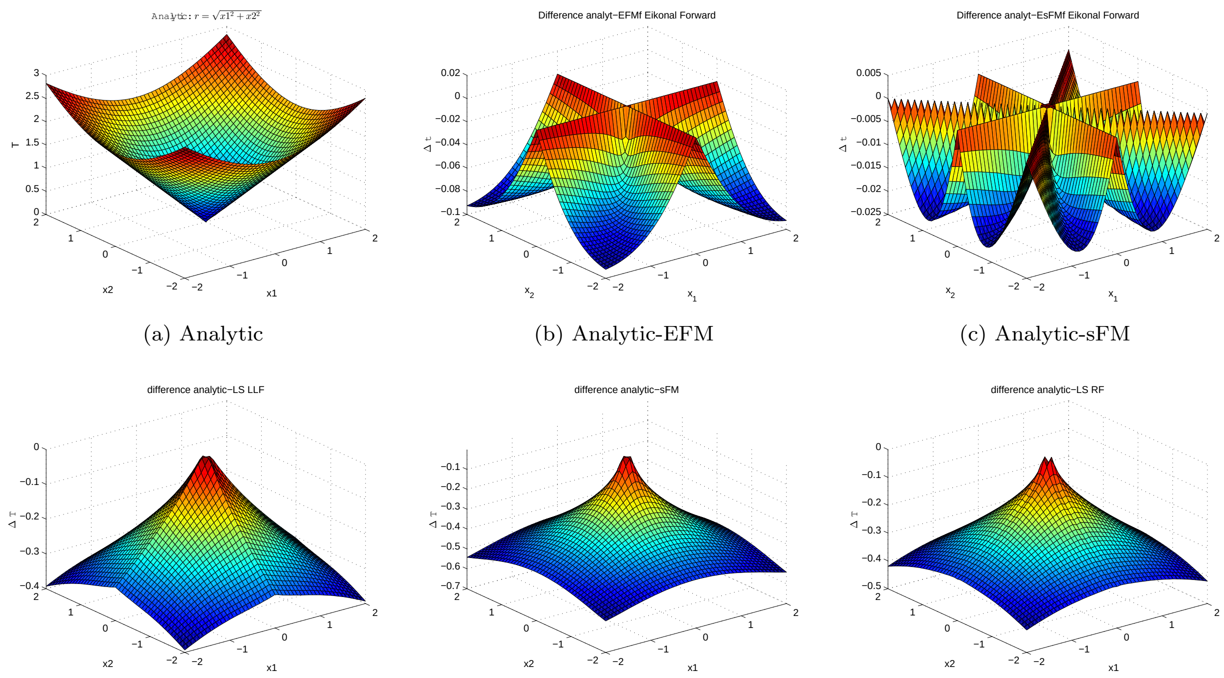
Uniformly expanding front

The uniformly expanding front problem is used to compare the performance of the Level Set and ordered upwind methods. It is an Eikonal problem and is therefore well suited for the Fast Marching schemes. The dynamics of this system are given below.

$$\dot{x} = f(x) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \frac{1}{\sqrt{x_1^2 + x_2^2}} = \frac{\nabla \Phi}{|\nabla \Phi|} \tag{8}$$

Assuming that the initial set is a point in the origin of the domain, the analytic solution to this problem is $T = \sqrt{x_1^2 + x_2^2}$. The dynamics in equation 8 are provided as a function of gradient as well as an external function of state. In this study the Fast Marching methods have not been implemented to accept gradient-dependent system dynamics, as this phenomenon is not occurring in reachability problems of trajectory based dynamic systems. The Level set method has been implemented to use either expression.

The Level Set method is solved using a Local Lax-Friedrichs scheme as well as a Roe-Fix scheme for dissipation and are initialized with $\Phi = \sqrt{x_1^2 + x_2^2} - 0.08$. The results are shown in figure 7. The calculations have been performed on a $N = [49, 49]$ grid and a domain of $x_1, x_2 = [-2, 2]$.



(a) Analytic (b) Analytic-EFM (c) Analytic-sFM

(d) Analytic-eLS LLF (e) Analytic-iLS LLF (f) Analytic-iLS RF

Figure 7: Results for the Eikonal problem

Figure 7a shows the analytic solution of the Eikonal equation. The remaining figures show the difference between the analytic and the numerical solutions. Figure 7b shows the difference pattern for the Eulerian Fast Marching method. No error is present on the principle axis, but the Fast Marching method estimates later arrival times elsewhere. Figure 7c shows the difference pattern for the safe Fast Marching method. In contrast to the Eulerian method, the error is about four times smaller in magnitude. Also the diagonals coincide with the exact solution. This improvement in accuracy is because the SFM method uses the larger multi-stencil which also incorporates the diagonal axis in the calculation.

Figure 7d shows the difference surface of the Level Set method with local Lax-Friedrichs scheme using the externally defined dynamics and is therefore comparable with the results of the Fast Marching methods. Although using the same stencil as the Eulerian Fast Marching method, the explicit Level Set method is less accurate in this Eikonal problem. Figure 7e shows the results for the same scheme using the implicitly defined dynamics. Comparing this surface to figure 7d gives an indication on the difference between evolving the set with pre-calculated and internal dynamics. The latter case uses different inputs in response to the approximation error. Figure 7f shows the difference for the Roe-Fix scheme. The results for the internally defined dynamics are a bit better when compared to the local Lax-Friedrichs scheme due to the reduced dissipation.

The Roe-Fix scheme was found to induce oscillations on the externally defined dynamics. The Roe-Fix scheme assumes that the derivatives of the Hamiltonian are a good indicator for discontinuities. Since the velocity vectors of the externally defined flow field are not affected by the gradient, this method fails as an indicator for induced oscillations. It is therefore suggested to add a second fix for problems that do not have the gradient in the derivative of the Hamiltonian. A good candidate may be to observe sign differences in the upper and lower derivatives themselves. The overall performance data is given in table 1. e and i stand for external and internal dynamics.

Table 1: Difference measures for the forward Eikonal problem; $N = [49, 49]$ and $dx = 1/12$

	L_∞	RMS
Analyt - E FSM	$9.24e^{-02}$	$5.35e^{-02}$
Analyt - s FSM	$2.23e^{-02}$	$1.22e^{-02}$
Analyt - e LS LLF	$3.93e^{-01}$	$2.53e^{-01}$
Analyt - i LS LLF	$5.42e^{-01}$	$3.82e^{-01}$
Analyt - e LS RF	$3.93e^{-01}$	$2.49e^{-01}$
Analyt - i LS RF	$4.20e^{-01}$	$2.86e^{-01}$

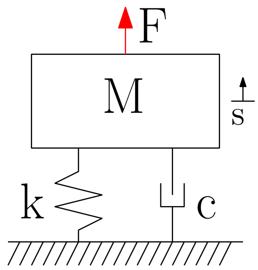


Figure 8: Schematic of the mass-spring-damper system with force excitation

Mass-spring-Damper system

The mass-spring-damper system forms a good test for the minimal time methods as the damping term provides unreachable regions inside the sate-space. The characteristics lie parallel to the reachable front on boundary of this region. The system is illustrated in figure 8 and the equation of motion is given below:

$$\begin{pmatrix} \dot{S} \\ \ddot{S} \end{pmatrix} = \begin{bmatrix} 0 & 1 \\ -k/M & -c/M \end{bmatrix} \begin{pmatrix} S \\ \dot{S} \end{pmatrix} + \begin{pmatrix} 0 \\ 1/M \end{pmatrix} F \tag{9}$$

The problem is set up with a specific spring and damping constant of 0.2 and a permissible specific forcing function of $[-0.15, 0.15]$. The calculations are performed on a 121x121 grid on domain $x_1, x_2 = [-3, 3]$. The initial level set is described by $g_0(S, \dot{S}) = |S| + |\dot{S}| - 0.2$.

The results for the minimal time problem are shown in figure 9. Two optimally exited trajectories are included to indicate the boundary of the reachability tube. Figure 9a shows the arrival time as calculated by the Eulerian Fast Marching method. The dimensions without converged nodes are ignored in the update scheme, which causes the solution to under-estimate arrival times and to "leak" into the unreachable region.

Figure 9b shows the semi-Lagrangian fast marching method where extrapolation between converged and not converged values uses the Kružkov transform. For $T = 35.5$ the increments are no longer visible to the floating-point resolution after the Kružkov transformation and the remaining state-space is reached with that arrival time. The solution produces conservative values until $T \approx 18$, after which the arrival times are under-estimated. The over-approximation is expected to be caused by the interpolation between not-converged nodes.

Figure 9c shows the arrival time surface when the Kružkov transform is replaced by a finite time ceiling. The nodes in the 'far' region are initialized with a value of $T_{\max} = 60$. Since the values are not transformed the interpolation is linear. The resulting solution is very conservative in its arrival times. The maximum arrival time also provides a stopping criteria as the arrival time cannot increase.

Figure 9d shows the result for an iterative scheme, where the semi-Lagrangian method is repeated 50 times. For each iteration, the nodes in the far region are given the values that were found in the previous iteration and thereby reduce the interpolation penalty. The value function now follows the two trajectories more closely. The arrival times are conservative everywhere except when approaching $T_{\max}$. Here the interpolation still allows unreachable nodes to receive a value smaller than $T_{\max}$. One attempt to overcome this problem is to use a threshold value $T_t < T_{\max}$. At the end of each iteration, all values larger than this threshold are set to the maximum value: $T(x) = T_{\max},\ \forall x | T(x) \geqslant T_t$. Provided that the difference between T_t and $T_{\max}$ cannot be overcome in one iteration for unreachable nodes, this threshold prevents over-estimation of the reachable set. When T_t is too small it can also prevent reachable states from being assigned a value. The main challenge is therefore to find an appropriate threshold. Figure 9e shows the result for $T_t = 0.95T_{\max}$. The reachable set is now conservative everywhere while still approximating the two trajectories. It is suspected that these extrapolation issues can also arise in similar iterative schemes like the fast sweeping method.

Figure 9f shows the results from the semi-Lagrangian safe Fast Marching method with a multi-stencil. Since the method is only allowed to use converged nodes, it stops when there are no candidate nodes inside the observed stencil. For the $3x3$ stencil used here, the calculated set is relatively small. The nodes that are found however closely follow the trajectories.

Figure 9g demonstrates the idea of combining the depth-first and breadth-first methods by including a trajectory in the initial set of the safe Fast Marching scheme. During implementation it was found that the initial values of the nodes visited by the trajectory can be set to the arrival times of this trajectory. This conservation of arrival time knowledge is not possible for the Level Set method where the front of the set must have the same value. One trajectory is used over a horizon of $t = [0, 40]$ and projected on the grid. The added trajectory greatly improves the region on which the method can solve the problem.

VI. Simplified aircraft model

To demonstrate the performance of the schemes on a more aerospace related system, the aircraft model used in[?,?,?] has been implemented. It consists of a simplified model of the slower symmetric motions. The state space is described by the velocity V and the flight path angle γ. The relevant axis and vectors are defined in figure 10. In addition, the influence of the side-slip angle β and roll angle ϕ is also included in this model. The considered equations of motion are given here:

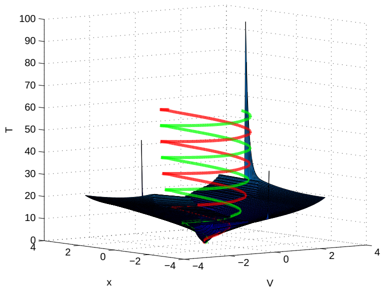
$$\begin{bmatrix} \dot{V} \\ \dot{\gamma} \end{bmatrix} = \begin{bmatrix} -\frac{\rho S}{2m} V^2 C_{D_0} - g\sin\gamma + \cos\alpha\cos\beta\frac{T}{m} - \frac{\rho S}{2m} V^2 \left(C_{D_\alpha}\alpha + C_{D_{\alpha^2}}\alpha^2 \right) \\ -\frac{g}{V}\cos\gamma + \left(\cos\phi\sin\alpha\cos\beta - \sin\phi\cos\beta\right)\frac{T}{Vm} + \frac{\rho S}{2m} V \left(C_{L_0} + C_{L_\alpha}\alpha \right)\cos\phi - \frac{\rho S}{2m} V C_{Y_\beta}\beta\sin\phi \end{bmatrix} \tag{10}$$

This model can be simplified by assuming small angles:

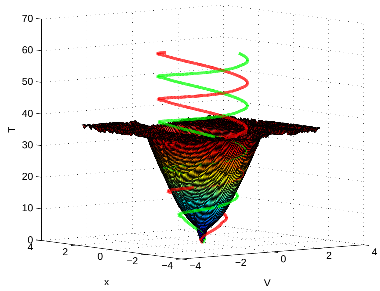
$$\begin{bmatrix} \dot{V} \\ \dot{\gamma} \end{bmatrix} = \begin{bmatrix} -\frac{\rho S}{2m} V^2 C_{D_0} - g\sin\gamma + \frac{T}{m} - \frac{\rho S}{2m} V^2 \left(C_{D_\alpha}\alpha + C_{D_{\alpha^2}}\alpha^2 \right) \\ -\frac{g}{V}\cos\gamma + \frac{\rho S}{2m} V \left(C_{L_0} + C_{L_\alpha}\alpha \right)\cos\phi - \frac{\rho S}{2m} V C_{Y_\beta}\beta\sin\phi \end{bmatrix} \tag{11}$$

The control parameters are $T \in [20546N, 410920N]$, and $\alpha \in [0°, 14.5°]$. The following aerodynamic derivatives will be used: $C_{D_0} = 0.1599$, $C_{D_\alpha} = 0.5035$, $C_{D_{\alpha^2}} = 2.1175$, $C_{L_0} = 1.0656$, $C_{L_\alpha} = 6.0723$ and $C_{Y_\beta} = -1.6$. The remaining terms are: $m = 120 \cdot 10^3\ kg$, $\rho = 1.225\ kg/m^3$, $S = 260\ m^2$ and $g = 9.81\ m/s^2$.

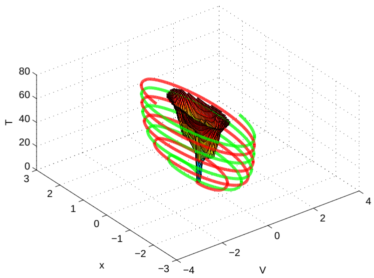
For the Level Set method it is possible to derive the optimal inputs directly from the objective function's gradient. The control logic including model uncertainty is worked out in[?] and is repeated here. Define the critical value $\hat{\alpha} = \frac{p_2 C_{L_\alpha}\cos\phi - p_1 V C_{D_\alpha}}{2 p_1 V C_{D_{\alpha^2}}}$ and mid-range angle $\bar{\alpha} = \frac{\alpha_{\min} + \alpha_{\max}}{2}$. Then to minimize the Hamiltonian, one has to choose the optimal inputs in accordance to the following rules:



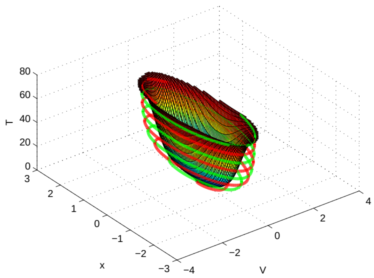
(a) Eulerian Fast Marching



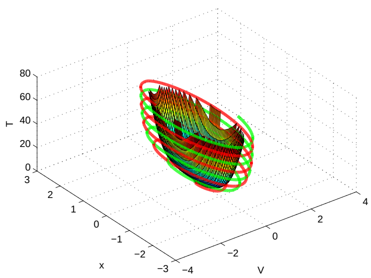
(b) Kružkov semi-Lagrangian Fast Marching



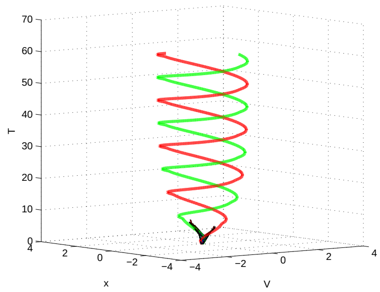
(c) semi-Lagrangian Fast Marching



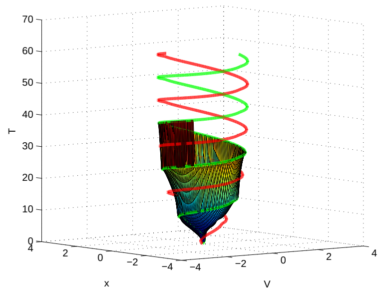
(d) Iterative semi-Lagrangian Fast Marching



(e) Iterative semi-Lagrangian Fast Marching with threshold



(f) safe Fast Marching



(g) safe Fast Marching with trajectory

Figure 9: Results for problem 2 as estimated using the Fast Marching methods

Control logic:

- If $p_1 > 0$ then $T^* = T_{\min}$ and
  - if $\hat{\alpha} \geqslant \bar{\alpha}$ then $\alpha^* = \alpha_{\min}$
  - if $\hat{\alpha} < \bar{\alpha}$ then $\alpha^* = \alpha_{\max}$
- If $p_1 = 0$ then $T^* = T_{\min}$ and
  - if $p_2 \geqslant 0$ then $\alpha^* = \alpha_{\min}$
  - if $p_2 < 0$ then $\alpha^* = \alpha_{\max}$
- If $p_1 < 0$ then $T^* = T_{\max}$ and
  - if $\hat{\alpha} \leqslant \alpha_{\min}$ then $\alpha^* = \alpha_{\min}$
  - if $\hat{\alpha} \leqslant \alpha_{\min}$ then $\alpha^* = \alpha_{\min}$
  - if $\alpha_{\min} \leqslant \hat{\alpha} \leqslant \alpha_{\max}$ then $\alpha^* = \hat{\alpha}$
- If $p_2 \sin\phi \geqslant 0$ then $\beta^* = \beta_m in$
- If $p_2 \sin\phi < 0$ then $\beta^* = \beta_m ax$

Disturbance logic:

- If $p_1 \geqslant 0$ then $C^*_{D_0} = C_{D_0\min}$, $C^*_{D_\alpha} = C_{D_\alpha\min}$, $C^*_{D_{\alpha^2}} = C_{D_{\alpha^2}\min}$
- If $p_1 \leqslant 0$ then $C^*_{D_0} = C_{D_0\max}$, $C^*_{D_\alpha} = C_{D_\alpha\max}$, $C^*_{D_{\alpha^2}} = C_{D_{\alpha^2}\max}$
- If $p_2 \geqslant 0$ then $C^*_{L_0} = C_{L_0\max}$, $C^*_{L_\alpha} = C_{L_\alpha\max}$
- If $p_2 \leqslant 0$ then $C^*_{L_0} = C_{L_0\min}$, $C^*_{L_\alpha} = C_{L_\alpha\min}$
- $C^*_{Y_\beta} = C_{Y_\beta\max}$

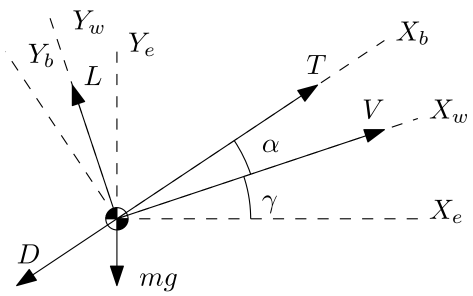


Figure 10: FBD for the aircraft model

Here p_1, p_2 are the components of the value-function's gradient. The simple condition for $C^*_{Y_\beta}$ follows from the fact that β will always be selected to satisfy $p_2 \sin\phi\beta < 0$.

Determining the optimal inputs for the ordered upwind methods is a bit more complicated as no gradient is readily available. For the purpose of demonstration, a brute-force optimization is implemented. The controller can select from ten samples for α and choose between the upper and lower limits for the remaining controls. This approach results in a total of 40 different combinations of control inputs. 160 combinations are needed when model-uncertainty is considered.

To make a better comparison, Mitchell's Level Set toolbox uses first order spatial differences and a first order runge-Kutta integration scheme. The implementation has been verified by re-producing some of the results from[?] and[?] with the Level Set scheme. All calculations have been made on a uniform $150x150$ grid with the dimensions as shown in the figures.

Figure 12 shows the forward and backward reachable sets for zero roll on ten arrival times to a horizon of five seconds. The difference between the safe Fast Marching method and the Level Set method is shown in figures 12c and 12f.

The safe Fast Marching method stops prematurely in regions where the gradient of the set and the characteristic curves are not sufficiently aligned, leaving distinctive edges. In contrast to the Eikonal equations the safe Fast Marching method arrives a little later in most regions, except on the paths visited by the corners of the initial set and on the upper right edge of the forward reachable set, where it arrives earlier. It is suspected that the augmented dissipation of the Level Set method has a significant impact on how these regions are extended.

The backward reachable set has two regions where the safe fast marching method arrives considerably later than the Level Set method. There are also two small regions where the Fast Marching method is a little faster. Especially the right region experiences a relative increase in arrival times. There are a couple of possible explanations for this increase. First of all the anisotropy prevents the safe fast marching method from progressing the front in the same manner as the Level Set method. A given node may only be solved once it is approached from a different direction at a later iteration or is only reachable form the current front when using a control input that is sub-optimal compared to what the Level Set method can use. The safe Fast marching method also has to bridge the space between grid nodes in a single iteration while the Level Set method may use several steps. A final reason for different arrival times in general is the fact that the

Level Set method uses an initial value function whose overall shape influences the progression of the front through stability criteria, dissipation and numerical derivatives.

The data in table 2 has been collected to compare the computational performance. Even with more efficient optimization the Level Set method requires 226 times as many node evaluations for the forward reachable set and 158 times as many for the backward reachable set. The difference for the number of model evaluations is less impressive in this example due to the brute-force optimization method used in the Fast Marching method, where 40 input combinations are tested for each calculation. The Level set method requires only nine model evaluations to figure out the correct upwind and dissipation terms. An additional advantage of the safe fast marching method is that no computations are performed on nodes outside the reachable set.

Apart from the computational performance, one can also observe the quality of the result. Table 2 also shows the maximum and average difference between the two methods. The percentage of missed nodes of either method is measured against the union set of reached nodes. The surplus of nodes reached in the Level Set method is caused by the inability of the safe Fast Marching method to capture the regions with high anisotropy. The few nodes that are uniquely identified by the safe Fast Marching method are most likely not found by the Level Set method due to the shrinking of the set caused by dissipation.

Table 2: Operational performance of the Level Set method and safe Fast Marching method on the aircraft model

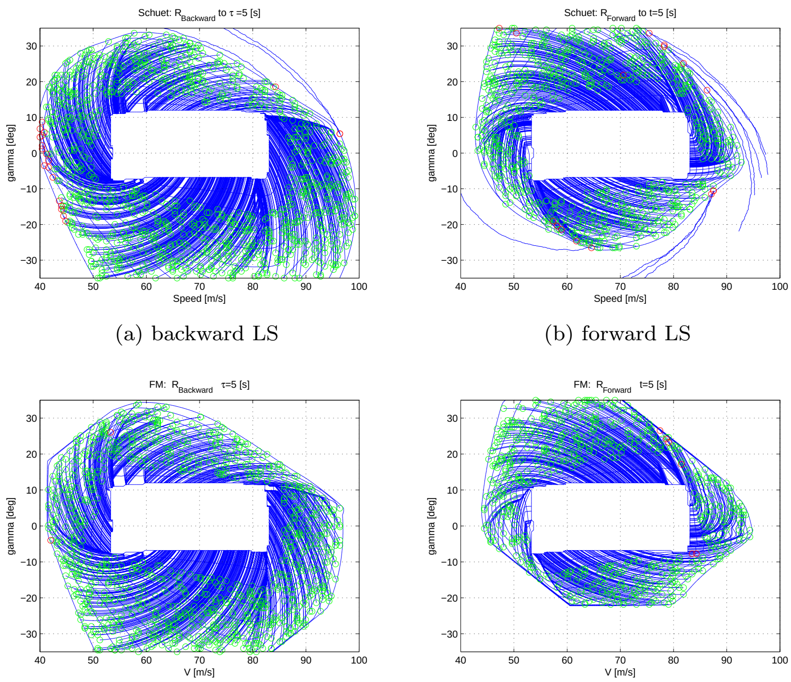
Parameter	Forward LS	Backward LS	Forward sFM	Backward sFM
# iterations	364	400	3.84	3.89
# model evaluations	3276	3600	1443640	2269000
# model evaluations/node update	9	9	40	40
# nodes/evaluation	22500	22500	1	1
tot # node evaluations	8190000	9000000	36091	56725
tot # model evaluations	73710000	81000000	1443640	2269000
nodes evaluated	22500(100%)	22500(100%)	8940(39.7%)	14582(64.81%)
nodes missed by sFM	10.7%	7.89%		
Nodes missed by LS	1.21%	0.89%		
RMS	0.158	0.207		
L_∞	1.511	1.487		

As an additional verification step, a Monte Carlo analysis has been made on the reachable sets. 2000 random sample states are generated. Only the states for which an estimated arrival time larger than one second exists are considered. For each, a single, stochastically optimal trajectory is generated. Instead of producing a large number of global trajectories, the principle of dynamic programming is applied by first generating a set of candidate trajectory increments end then selecting the one that results in the best performance. On each time increment of 0.02 seconds, the dynamic system is solved for a set of 51 random permissible inputs. The candidate end state of this increment with the smallest value on the arrival time surface as estimated by the Level Set or Fast Marching method is accepted. The corresponding set of inputs is included in the set of trial inputs for the next iteration.

Figure 11 shows the resulting trajectories and initial states. The states for which the trajectory successfully reaches the initial set within 110% of the estimated arrival time are colored green while the remaining initial states are red. A small number of trajectories escape from the region in which the estimated arrival time surface is properly defined and can no longer use it to optimize the control inputs. These trajectories should closely resemble the time-optimal paths. Figure 11a has a set of unverified nodes on the left edge of

Computed on an EliteBook 8570w Mobile Workstation running Windows®7 Home Premium and MatLab R2013a. Logged with MatLab's Profile toolbox the reachable set. This concentration of unverified nodes could indicate an overestimate. It is expected that the the Level Set method has incorrectly extended the lower-left corner of the trimmable set due to dissipation. Alternatively, the considered states can be particularly unforgiving for using sub-optimal inputs. The trajectories found from the unverified states pass close to the bottom-left corner. A truly optimal trajectory might have been able to reach this corner.

Table 3 gives an overview of these trajectories. The reach-rate indicates the fraction of initial states for which the trim set was reached successfully. The Success rate gives the fraction of these trajectories that reach the trim set within the time estimated by the arrival time surface. The remaining values give information on the arrival times obtained by the trajectories in comparison to the estimated arrival time. Interesting to note is that on average the successful trajectories of the Monte Carlo simulation predict approximately 60% faster arrival times than the Level Set and safe Fast Marching method. Although only first-order approximations were used in the estimation process and although no design effort was put in optimizing the estimation accuracy, these results give a strong indication that the estimation techniques tend to be very conservative. It is therefore recommended to obtain further insight in techniques to improve accuracy of the arrival time estimations.



(a) backward LS

(b) forward LS

(c) backward sFM

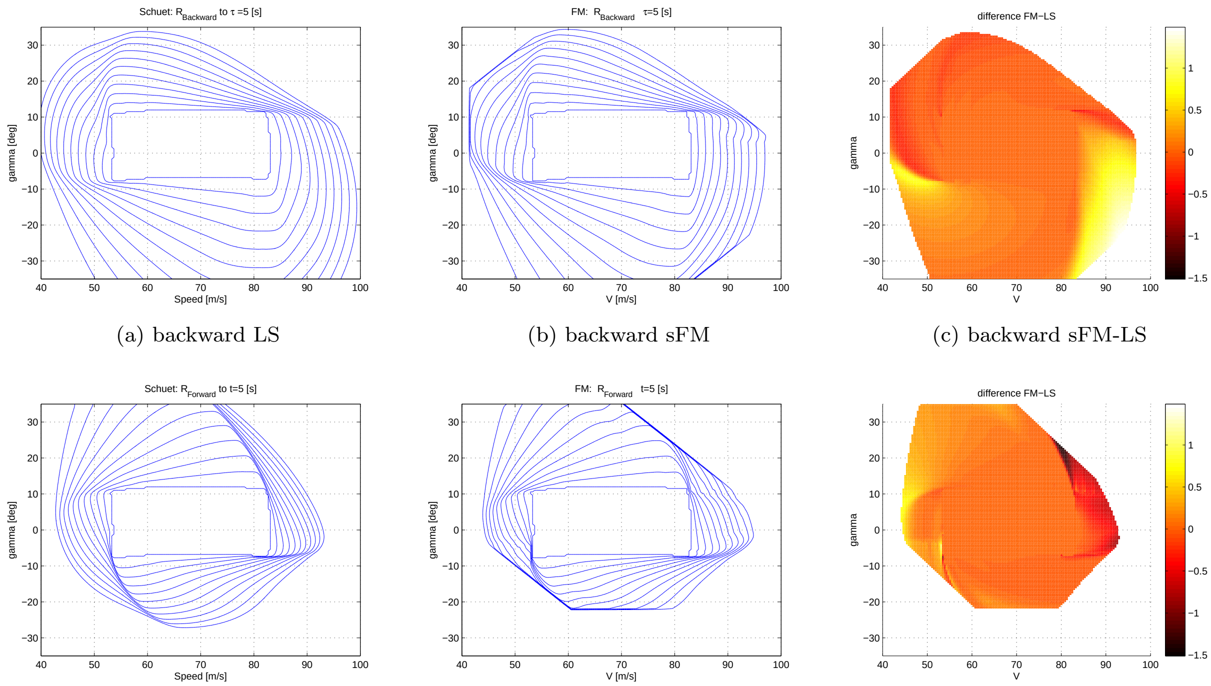
(d) forward sFM

Figure 11: Monte Carlo simulation with optimal trajectories as generated from random control inputs evaluated on the estimated arrival time surface.

Table 3: Statistical results from the Monte Carlo simulation with randomly generated optimal inputs.

problem	Reach rate	Success rate	Minimum T	Maximum T	mean T	std
LS forward	79.86%	96.93%	20.40%	108.00%	60.13%	21.69%
LS backward	98.18%	93.98%	19.20%	109.60%	60.68%	23.01%
sFM forward	99.05%	98.57%	17.60%	105.60%	56.64%	21.89%
sFM backward	99.80%	96.30%	14.80%	108.00%	57.51%	22.71%

Figure 13 shows the full reachable sets and flight envelopes for a perfect model as well as a model with 20%



(a) backward LS (b) backward sFM (c) backward sFM-LS

(d) forward LS (e) forward sFM (f) forward sFM-LS

Figure 12: Arrival time iso-lines and difference surface for the forwards and backward reachable sets

uncertainty. The sets are calculated with both the Level Set method and the Fast Marching method. The green surfaces in the center of each set represents the trimmable states for zero side-slip. The trimmed set requires higher velocities for larger roll angles, as can be expected. In the uncertain case the trim set as well as the reachable set become considerably smaller. The safe Fast Marching method is mostly conservative, also for the larger values of Φ.

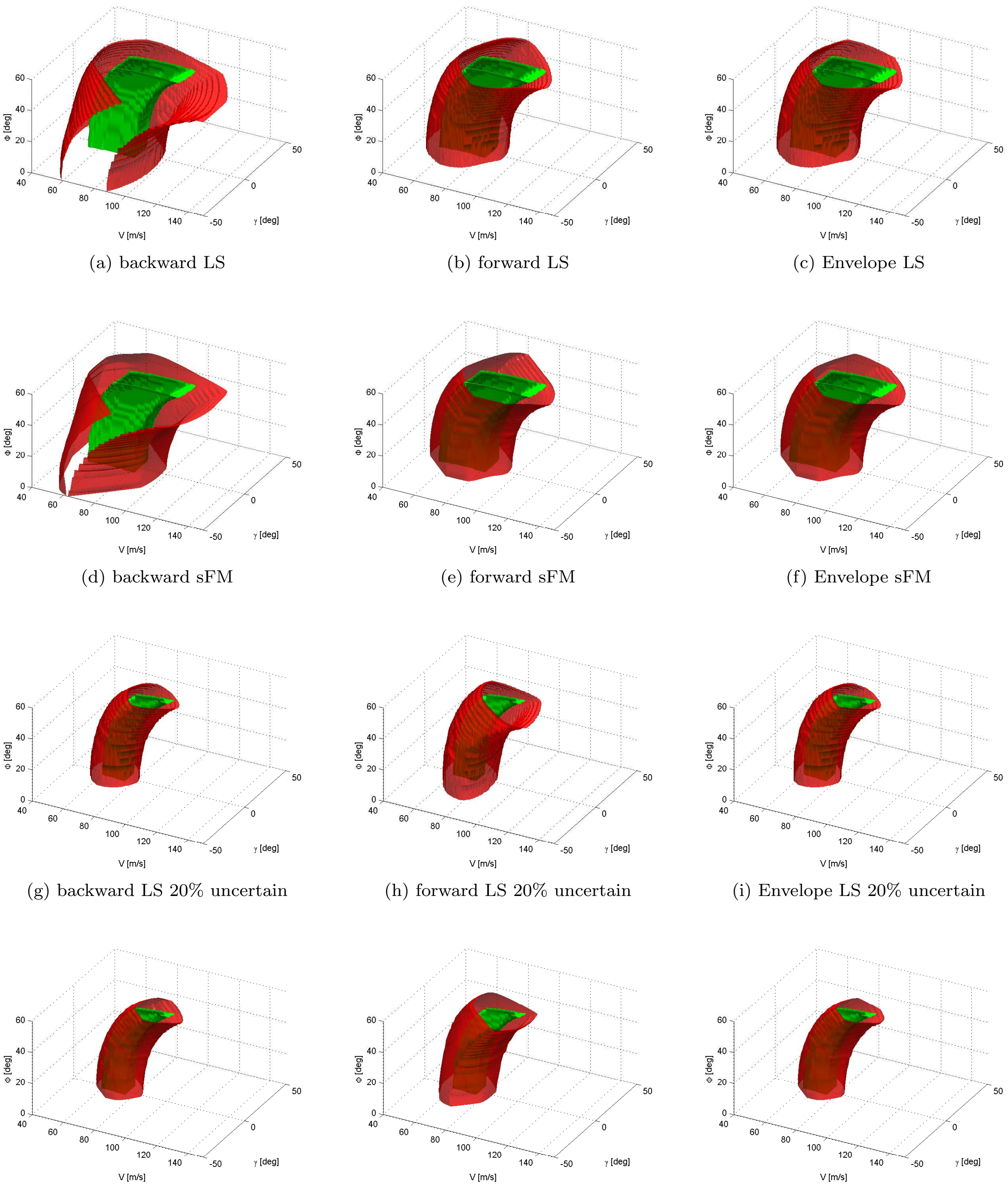
VII. Conclusion & recommendation

A simple thought experiment has shown that it is not possible to guarantee convergence of an arbitrary initial set. Similarly, it has been shown that a set found by recursively propagating a reachable set over a time-variant event cannot be used for the backward reachable set due to causality and does not produce a reachable set suitable for flight envelope estimation.

It was argued that permitting sub-optimal solutions by simplifying the optimization problem will result in more conservative minimal arrival times but should not affect the accuracy of the steady set. This simplification could save considerable computation time in the optimization aspect of the problem solving, which is of particular interest when considering more complex control problems. The simplification may be extended to the update schemes as well. Since no application of this idea has been tested, it is still open for further investigation.

It was further proven that under the assumption of a time-invariant system and a maintainable initial set the reachable set can be found from the minimal arrival time. This time can be found by the stationary boundary formulation. The schemes for this method are not subjected to the CFL stability criteria and do not require augmented dissipation in order to find a weak solution.

A variety of Fast Marching methods have been tested for applicability on a set of simple test problems. Although it was hypothesized that the classical Fast Marching method may be able to correctly identify the steady reachable set it was found that this is not the case. A physical explanation for this inability



(a) backward LS (b) forward LS (c) Envelope LS

(d) backward sFM (e) forward sFM (f) Envelope sFM

(g) backward LS 20% uncertain (h) forward LS 20% uncertain (i) Envelope LS 20% uncertain

(j) backward sFM 20% uncertain (k) forward sFM 20% uncertain (l) Envelope sFM 20% uncertain

Figure 13: Full nominal and uncertain reachable sets and envelope for arrival time of 5 s (red) and trimmed initial sets (green)

was found for both the Eulerian and the semi-Lagrangian approach. The safe Fast marching method has been demonstrated to find correct but incomplete reachable sets. An iterative upwind method was shown to require special modifications to prevent over-estimated reachable sets. The performance of these methods is compared to the Level Set method on a variety of test systems and a simplified aircraft model. The behavior of the safe Fast Marching method was demonstrated on a simplified aircraft model for the slow aircraft dynamics. The method required fewer calculations despite of an inefficient optimization scheme. The estimated arrival times differ from those calculated with the level set method. This difference is partially attributed to the low resolution of attempted control input samples. The results of the double integrator demonstrate that the SFM method can outperform the Level Set method in terms of accuracy on an Eikonal equation. The method is expected to perform better in anisotropic problems when a larger stencil size is used, as suggested for the ordered upwind method.[?] Since the stencil size can give considerable overhead in larger dimensions it is not recommended to use this method to find solutions in highly anisotropic regions. Combined with the depth-first trajectories it may however serve as a method to quickly identify a first accurate region of the flight envelope that can be extended afterwards with another scheme. An overview on the applicability of the solving schemes is given in table 5.

In a search for more localized solving schemes, it was proven that a set of depth-first trajectories can be combined with the propagating front methods to quickly reach out to state-regions of immediate interest. This technique can be applied to both the Level Set method and the Minimal time method. Although this method may alter the computed arrival times when applied to the Level Set method it can be guaranteed to correctly identify the reachable states. This approach has been demonstrated with the safe Fast Marching scheme on a mass-spring-damper system where it was found to aid the solution finding of the safe OUM. During implementation it was realized that this technique is better suited for the minimal time method as the notion of arrival time can be maintained in the initialization. Iterative minimal arrival schemes should able to converge to the true minimal arrival time even when the selected trajectories are not optimal. The advantages of this technique are earlier verification of important regions, faster convergence and better handling of anisotropy for the safe OUMs.

Based on the experience gained in this study, the following recommendations can be made:

- Continue the exploration of Minimal arrival time calculation methods. In particular, investigate the IFM and FSM as they should result in competitive computational efficiency. The observations made with the mass-spring-damper problem suggest that a modification may be required to prevent the occurrence of over-estimation when approaching the steady reachable set. The FSM may be improved by extending it to a narrow band method.
- Of all the single-pass methods only the safe upwind method is believed suitable for producing correct reachable sets in anisotropic problems. The SFM method may have applications for problems where an incomplete but otherwise accurate reachable set has to be quickly available.
- It is expected that the Level Set method, but in particular the boundary value scheme can result in significant computational savings when combing the methods with a depth-first method. Particular applications and implementation methods could be explored further.
- The idea to accept sub-optimal solutions by simplifying the optimization problem at the loss of accurate arrival times has not been assessed beyond the theoretical concept and remains open for further investigation. No indications were found that would restrict the applicability.
- The narrow band Level Set method is still a good competitor to many of the alternative methods explored in this study and should not be forsaken in further research.
- The Roe-Fix scheme may be extended to also handle non-smooth value surfaces and ambiguities that are independent of the flow field.
- The performed demonstrations have shown indications of excessive under-estimation of arrival times for all approximation schemes. Some investigation may be in order to obtain further insight in techniques to improve accuracy of the arrival time estimations.

The recommendations are summarized in table 4.

Finally it is stressed that in order to use any form of model-based moving front theory it is required that the model is accurate in all regions where the front is propagated. Although a more localized front propagation technique may make this requirement easier to achieve, it is stressed that the current adaptive model estimation techniques are only accurate near the states where measurements have been made. In order to safely use any of the proposed techniques, the model estimation method has to be able to predict the occurrence of bifurcation in unexplored states. Unless a suitable approximation technique is found, the techniques discussed in this study will be of limited use on on-line applications.

Table 4: Recommendations on considered simplification methods.
- - = not possible, - = not recommended, + = acceptable with loss of information, ++ = recommended

Method	Moving front	Minimal time
Split to problems of lower dimension	-	-
Convergence of estimated set	- -	- -
Recursion on time-invariant system	- -	- -
Localized front propagation	- -	-
Collaboration with trajectories	+	++
Acceptance of sub-optimal inputs	+	+
General applicability	++	+
Use of Narrow band method	++	++

Table 5: Overview of solving schemes

	Complexity	CFL	Upwind	Iterative	Anisotropic	speed	Suitable
LSM	$\mathcal{O}(N^{n+1})$	Y	N	Y	Y		Yes
NB-LSM	$\mathcal{O}(kN^n)$	Y	N	Y	Y		Yes
ITM	$\mathcal{O}(N^{n+1})$	N	N	Y	Y		No
FSM	$\mathcal{O}(N^{n+1})$	N	Y	Y	Y	$< ITM$	Maybe
NB-FSM	$\mathcal{O}(kN^n)$	N	Y	Y	Y		Maybe
E FMM	$\mathcal{O}(N^n \ln N)$	N	Y	N	N		No
SL FMM	$\mathcal{O}(N^n \ln N)$	N	Y	N	N		No
E OUM	$\mathcal{O}(kN^n \ln N)$	N	Y	N	Y		No
SL OUM	$\mathcal{O}(kN^n \ln N)$	N	Y	N	Y		No
s OUM	$\mathcal{O}(kN^n \ln N)$	N	Y	N	Y		Yes
BFM	$\mathcal{O}(kN^n \ln N)$	N	Y	Y	Y	$> ITM$	No
PFM	$\mathcal{O}(kN^n \ln N)$	N	Y	Y	Y	$> ITM$	No
FIM	$\mathcal{O}(kN^n \ln N)$	N	Y	Y	Y	$\approx FSM$	Maybe