4-D Trajectory Optimizers for Conflict Avoidance Using Speed Advisories

Arthur Richards and Oliver Turnbull

arthur.richards@bristol.ac.uk
oliver.turnbull@bristol.ac.uk

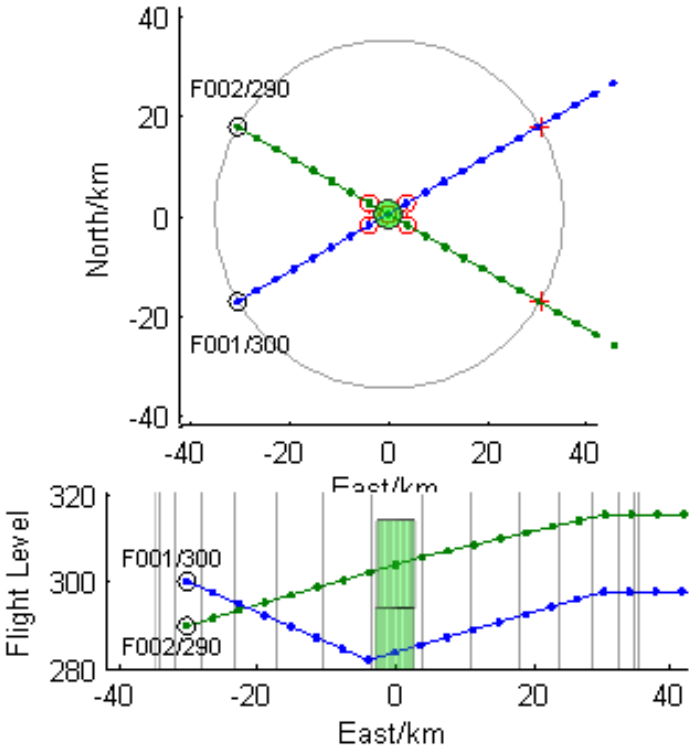
SUPEROPT

- Supervision of Route Optimizers
  - SESAR WP-E Research Project, 2011-13
  - Goal: provide ***intuitive*** interactions with trajectory optimizers
  - Converted common conflict resolution ***stratagems*** into mathematical ***constraints***
  - Optimized 4D trajectories then sent by datalink (*assumption*)

Speed Advisories

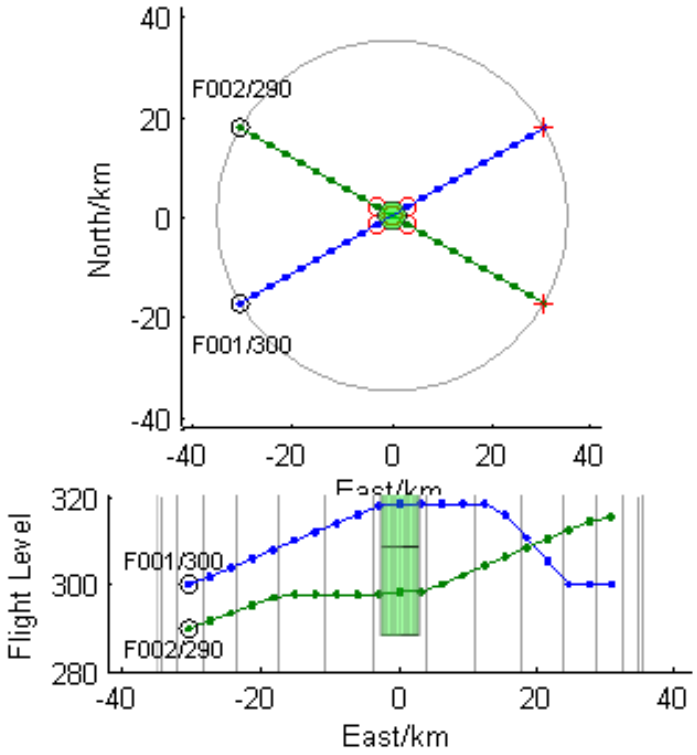
- ERASMUS found that speed changes alone could prevent many conflicts
- We want to add “resolve by speed change alone” to the SUPEROPT toolbox
  - Selectable by supervisor
  - Compatible with range of other constraints

SUPEROPT Toolbox 1



F002 over F001

(Unconstrained)

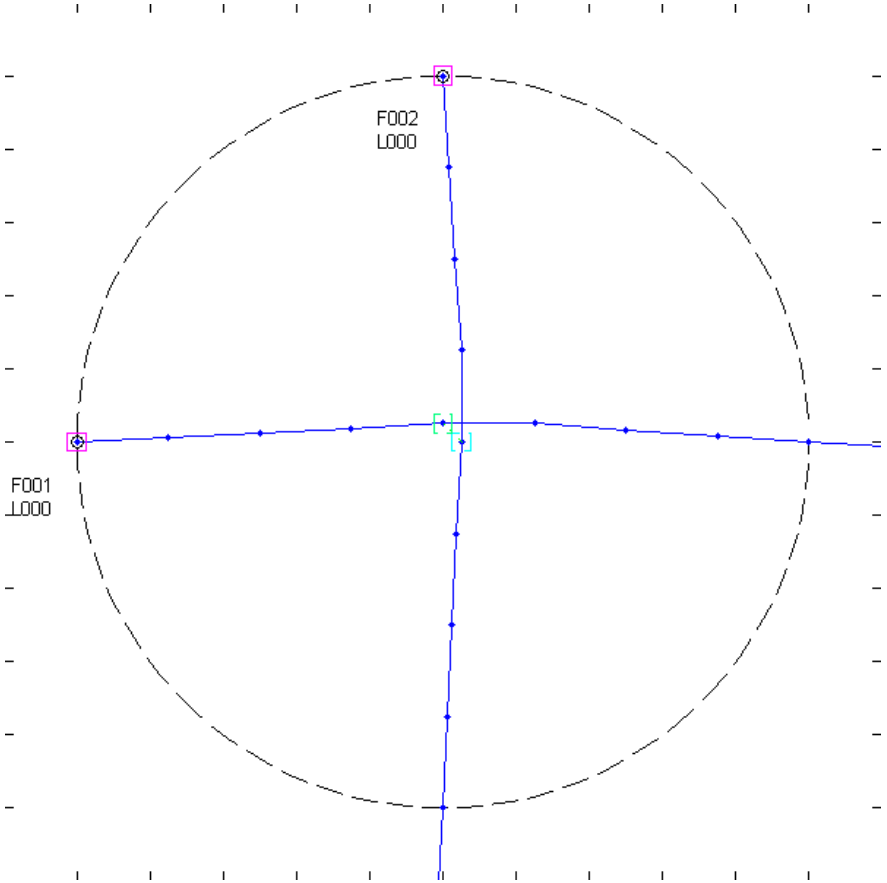


F001 over F002

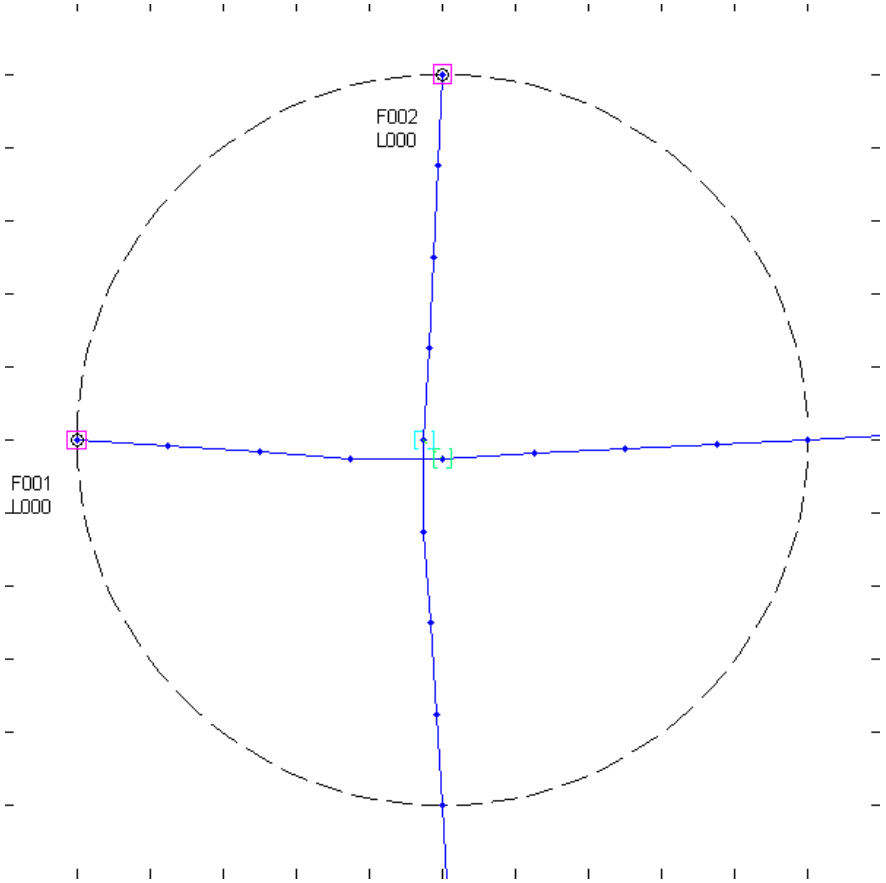
SUPEROPT Toolbox 2

F002 ahead of F001

(Unconstrained; 2D)



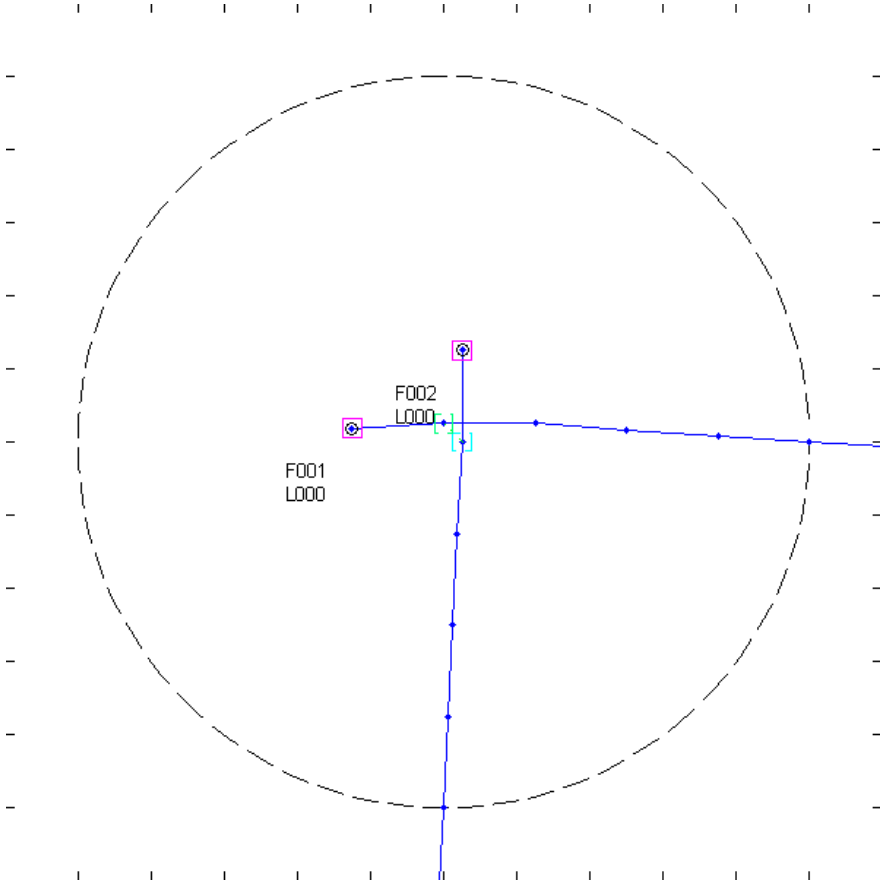
F002 behind F001



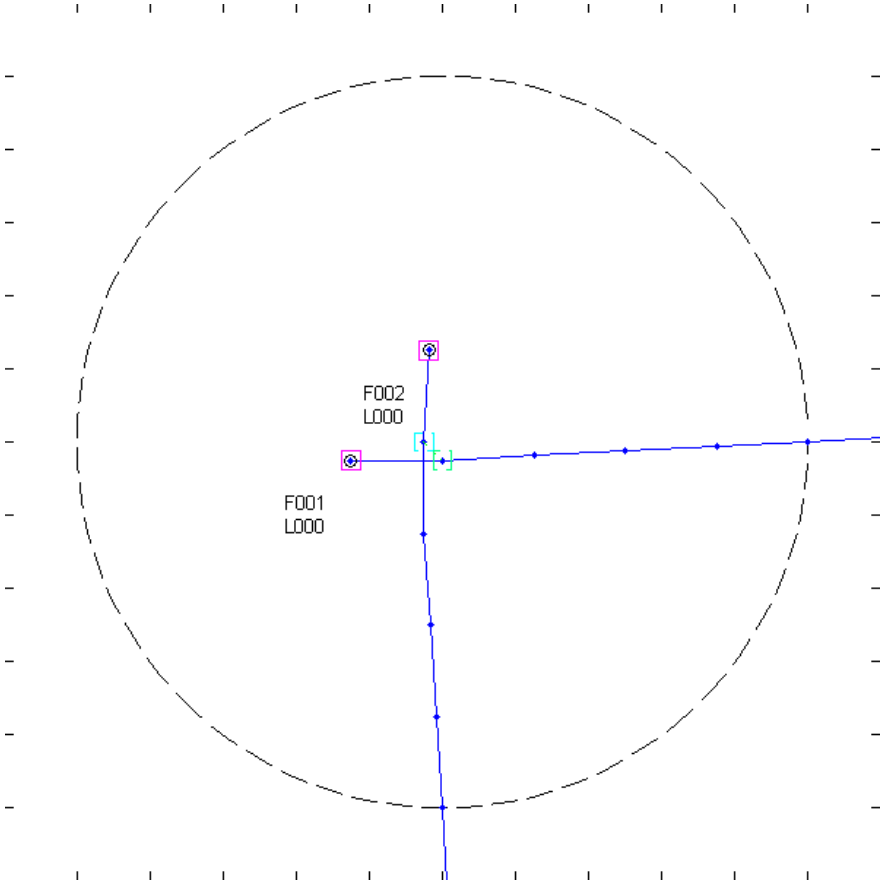
SUPEROPT Toolbox 2

F002 ahead of F001

(Unconstrained)



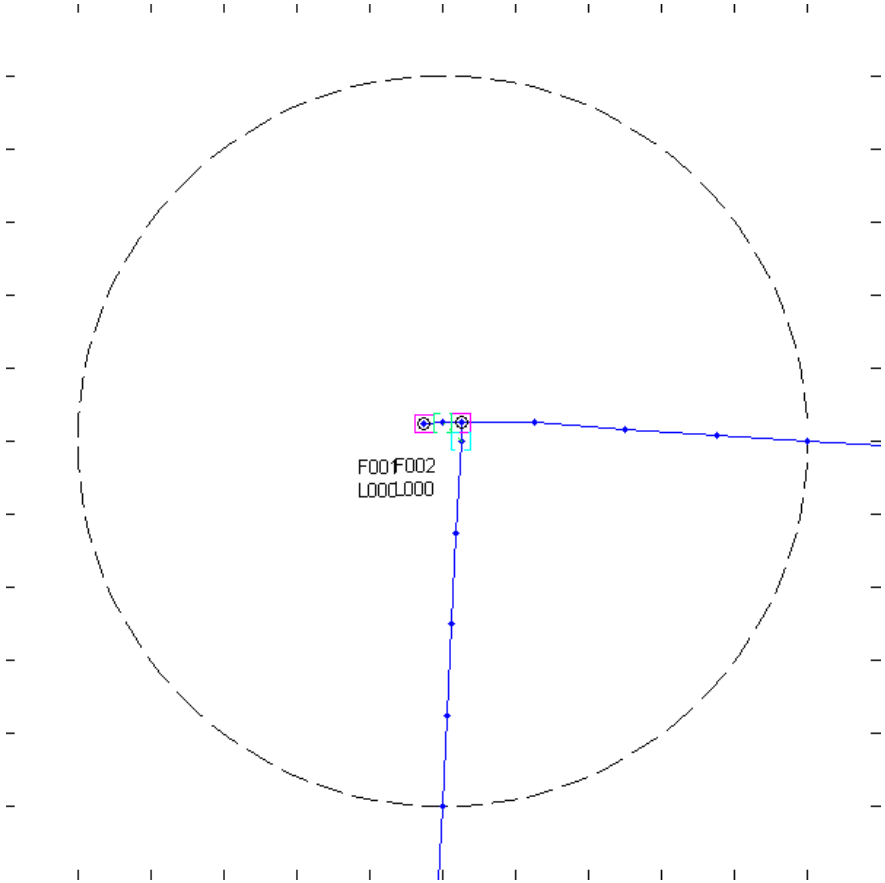
F002 behind F001



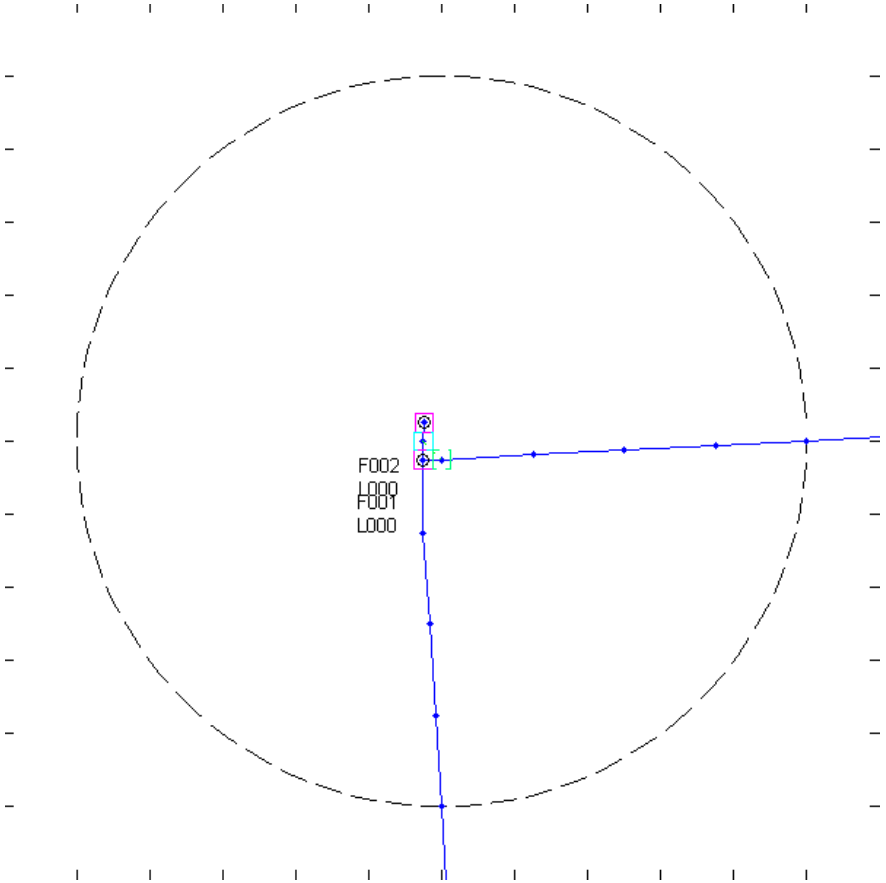
SUPEROPT Toolbox 2

F002 ahead of F001

(Unconstrained)



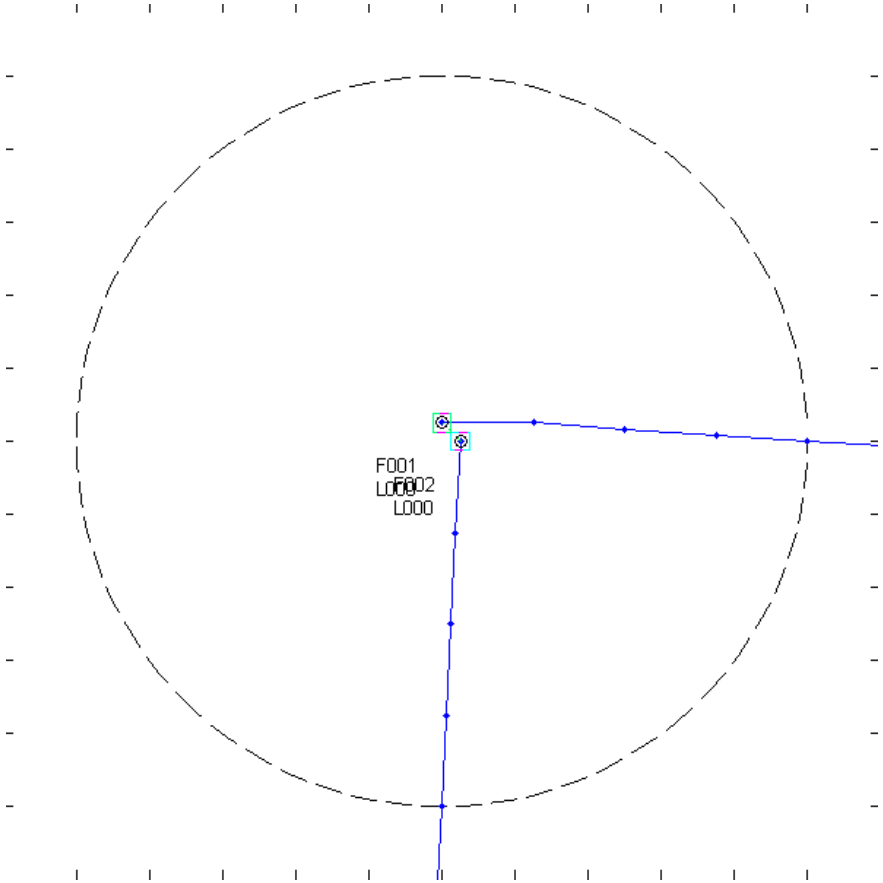
F002 behind F001



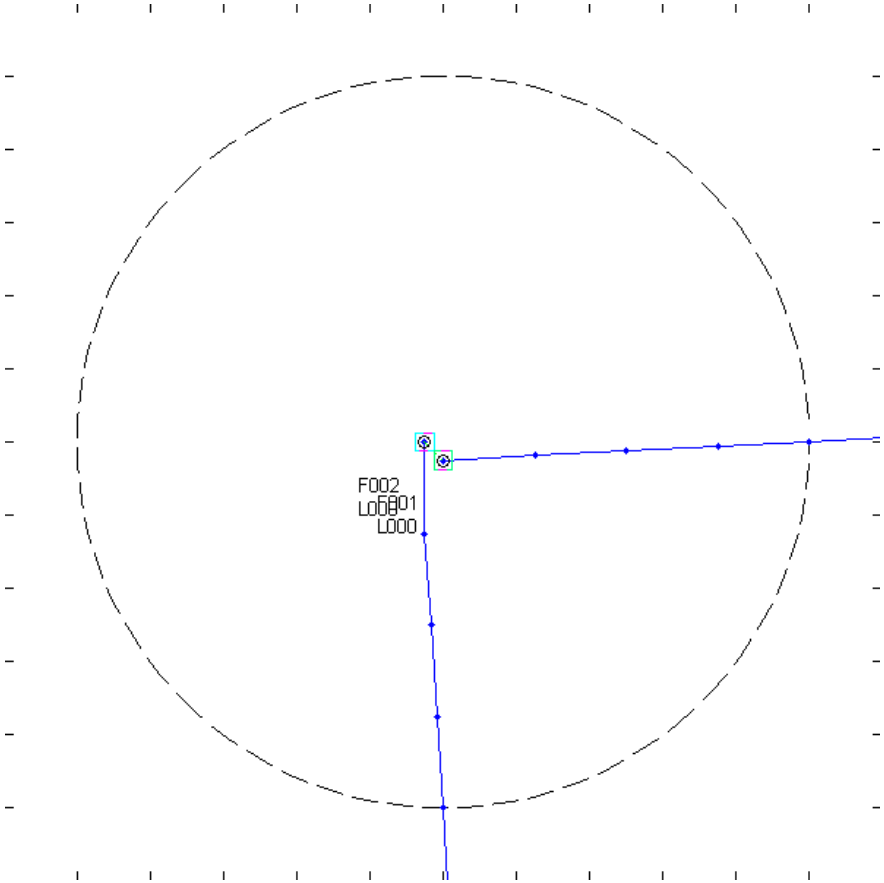
SUPEROPT Toolbox 2

F002 ahead of F001

(Unconstrained)



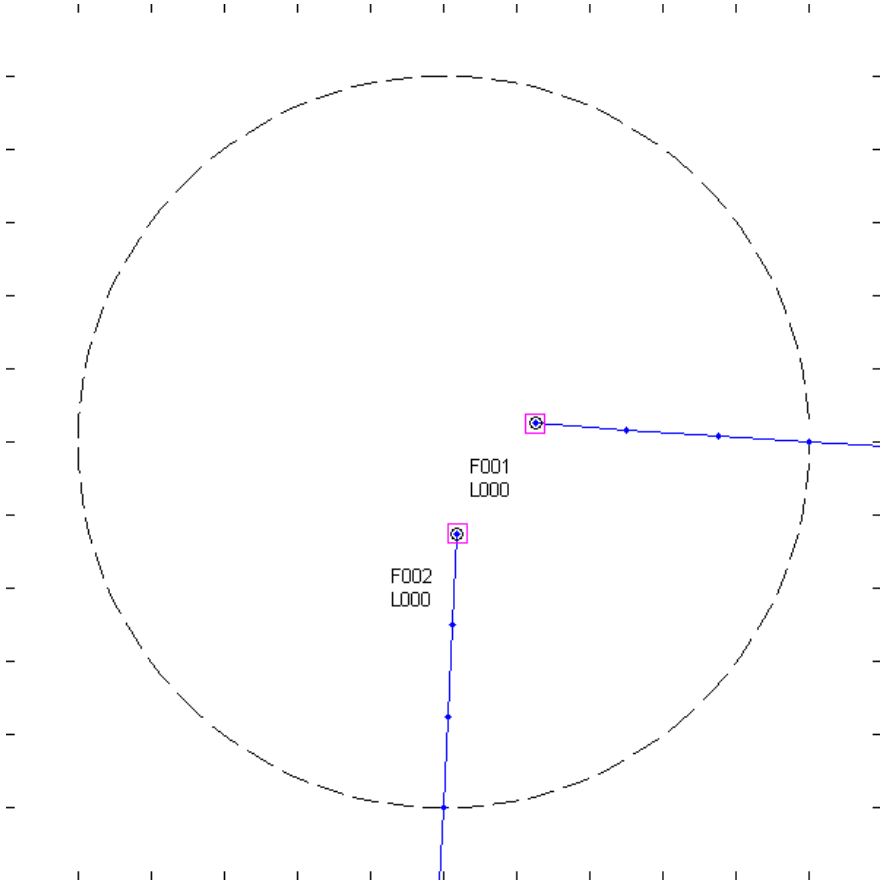
F002 behind F001



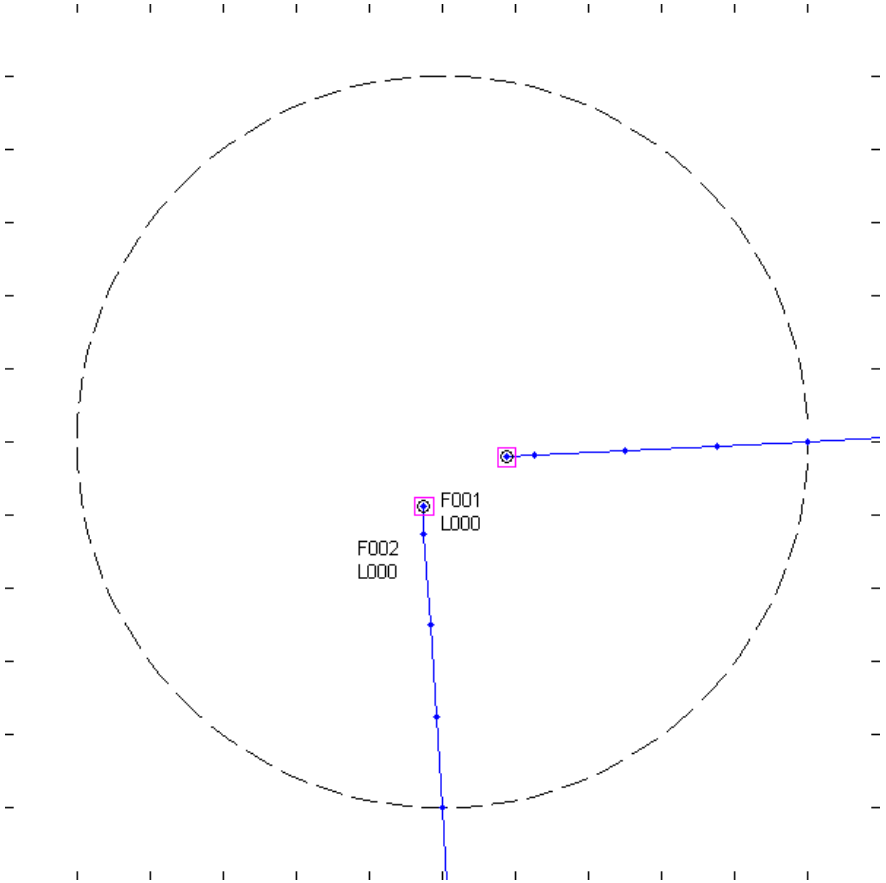
SUPEROPT Toolbox 2

F002 ahead of F001

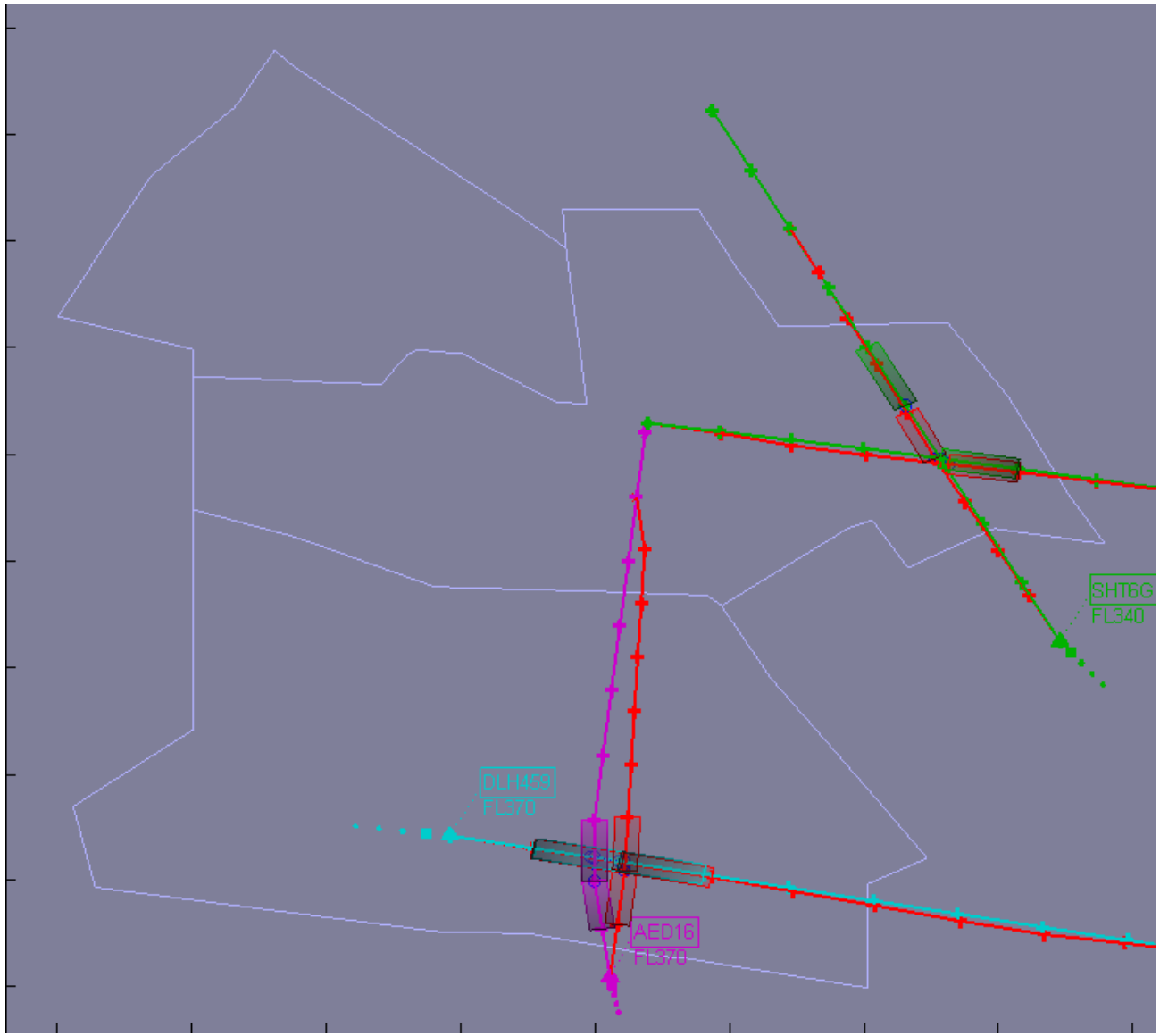
(Unconstrained)



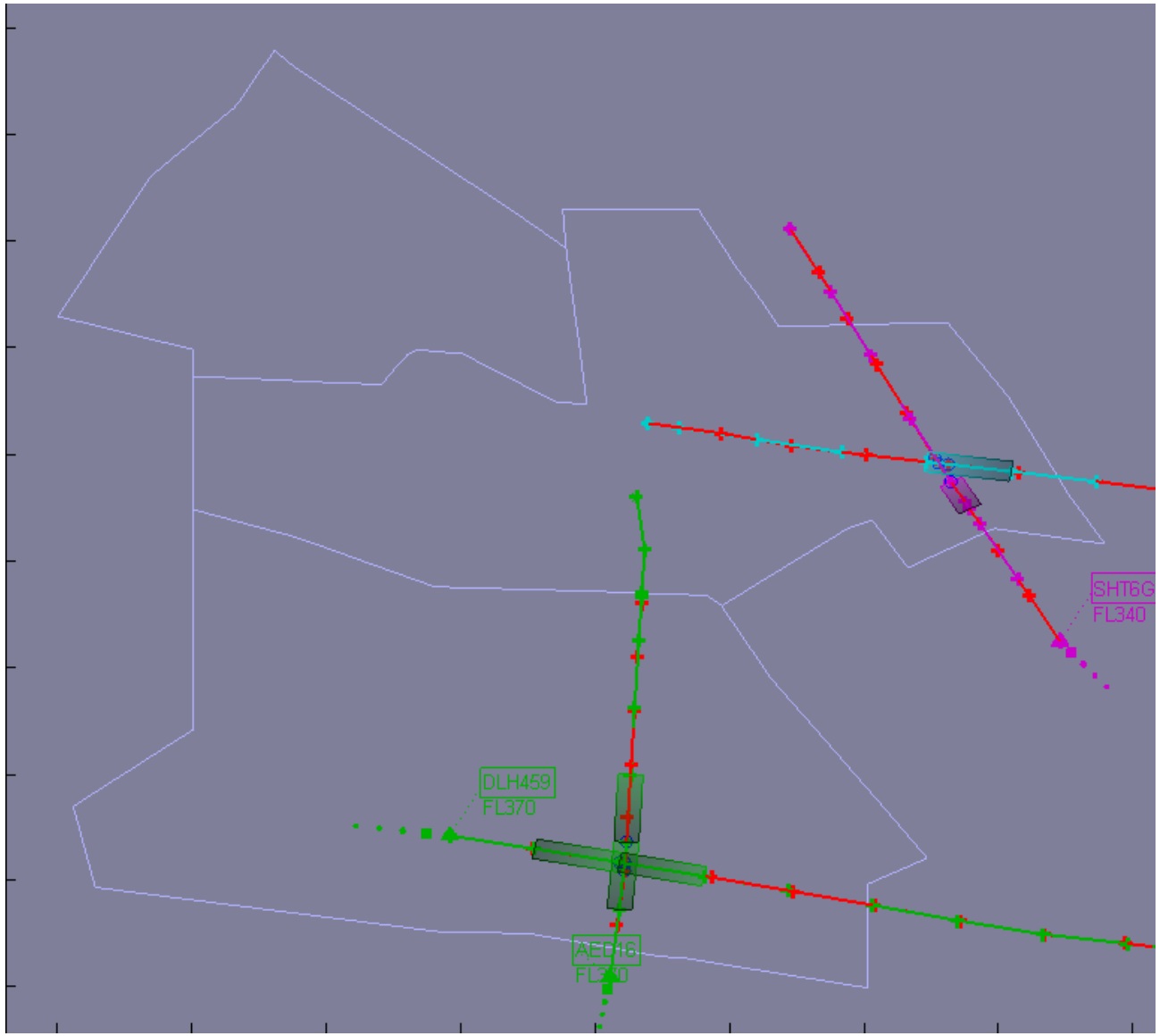
F002 behind F001



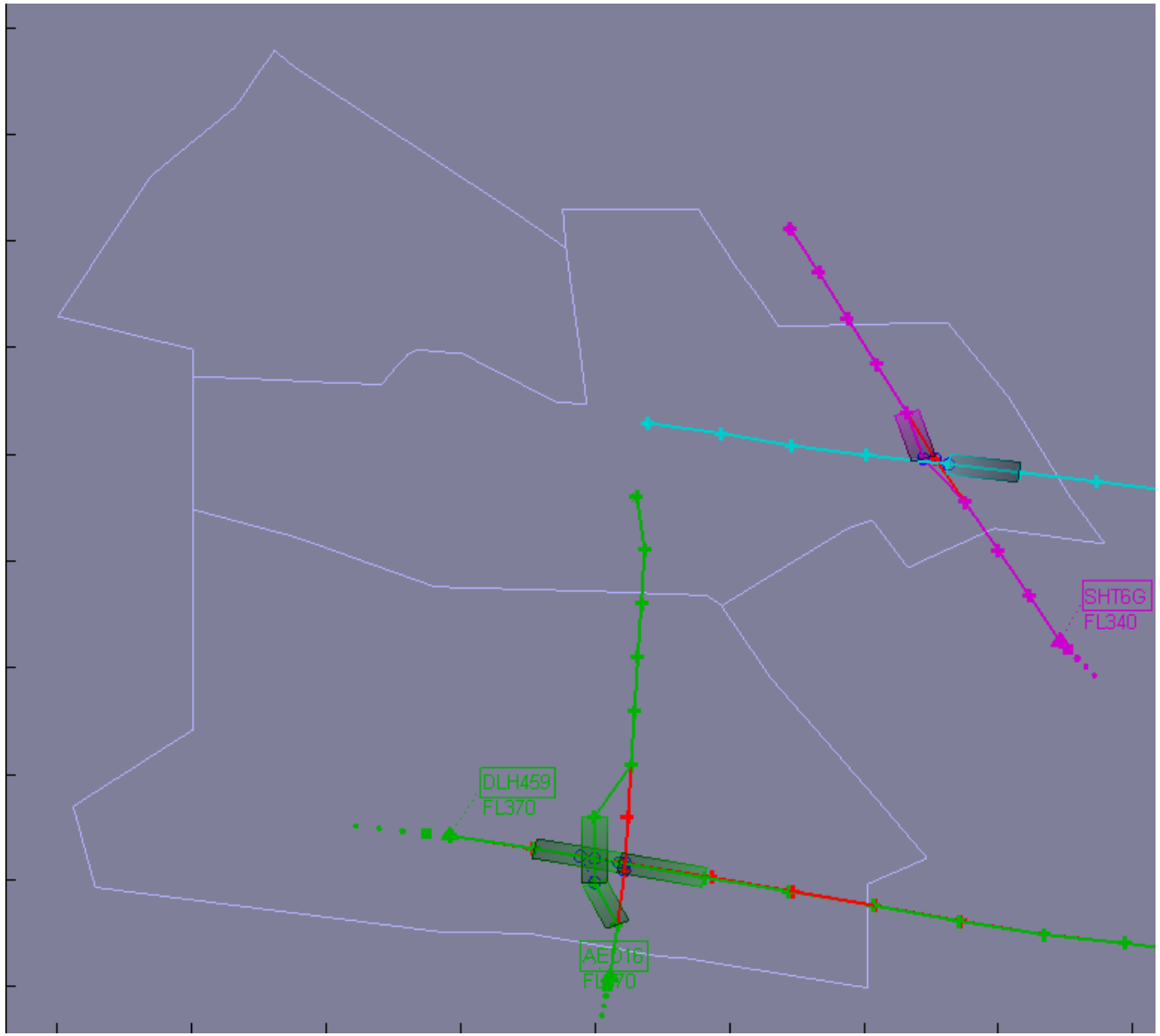
Costs: Min Time



Costs: Min Deviation



Costs: Max Separation



Optimizers

- Mixed-Integer Linear Programming
  - BADA-based performance modelling
  - Reliable commercial ***global*** solvers
  - Need to linearize everything
- Nonlinear with collocation
  - Less-developed solvers, for local only
  - Greater flexibility of cost etc

Problem Statement

- Given a 4D reference trajectory

  $T_R = \{ (x_R, y_R, z_R, t_R)_1 , (x_R, y_R, z_R, t_R)_2 , \ldots \}$

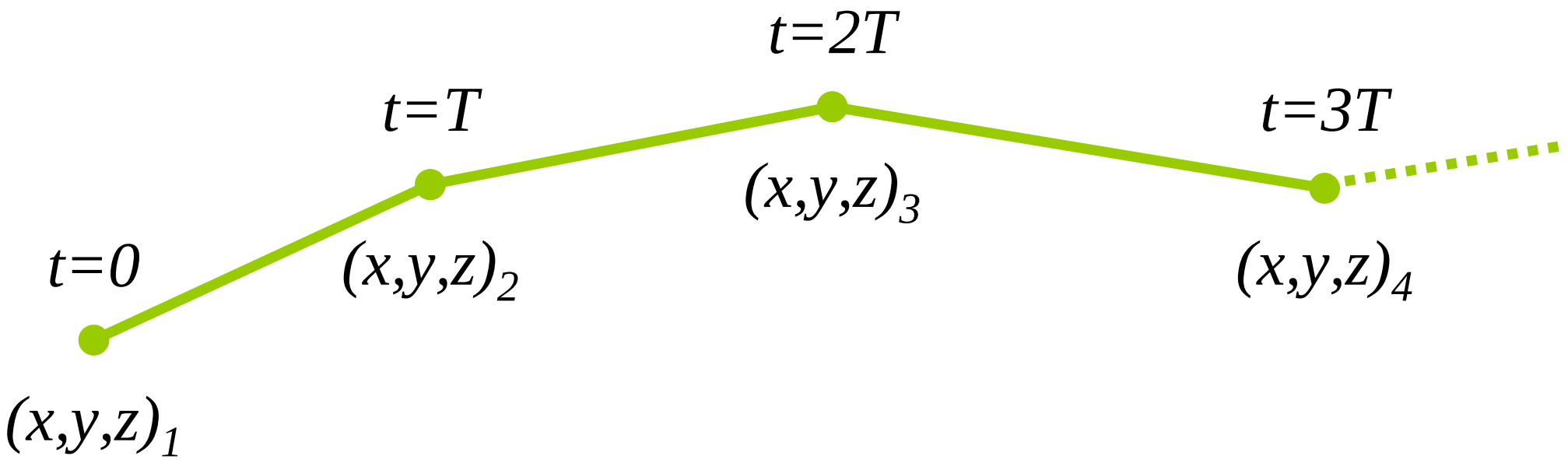
- Optimizing a new trajectory

  $T^* = \{ (x^*, y^*, z^*, t^*)_1 , (x^*, y^*, z^*, t^*)_2 , \ldots \}$

- **Constrain such that T^* differs from T_R only in speed changes**

Speed Advisories in MILP

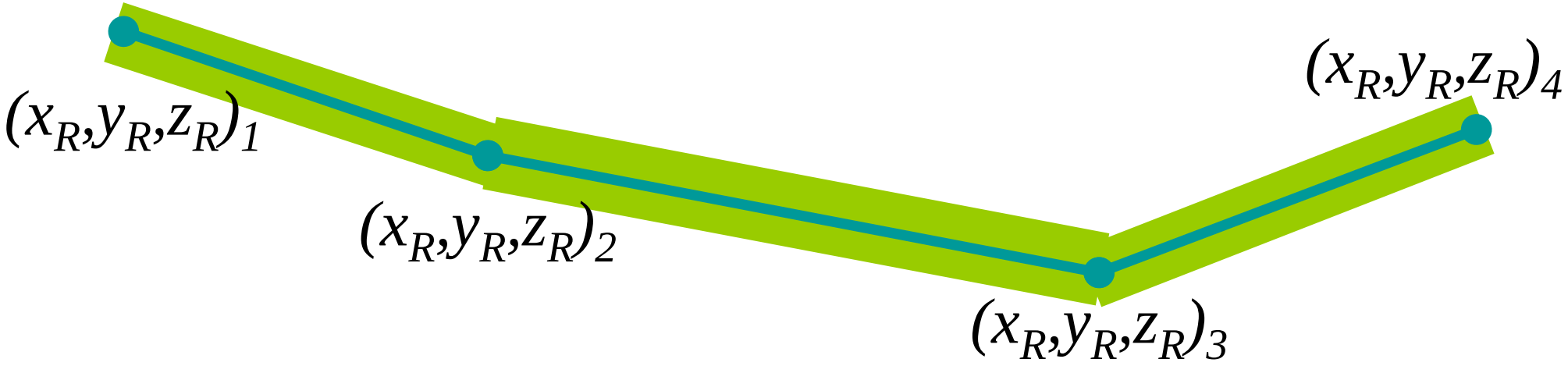
- MILP approach uses constant time steps



- Constraint: every (x*,y*,z*) must be on the reference trajectory

Speed Advisories in MILP

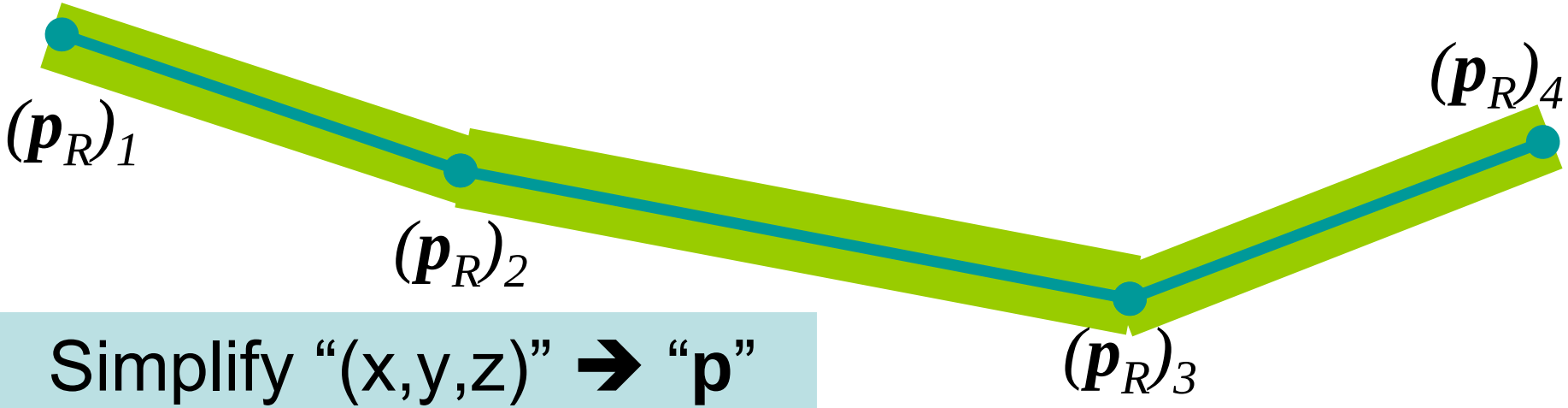
- Define the reference trajectory “legs” and corridors ***in 3D*** around them



  - Small margin for feasibility – may not be possible to follow reference exactly at a different speed

Speed Advisories in MILP

- Define the reference trajectory “legs” and corridors ***in 3D*** around them



Simplify “(x,y,z)” ➔ “**p**”

  - Small margin for feasibility – may not be possible to follow reference exactly at a different speed

Speed Advisories in MILP

$$\mathbf{p} = \alpha(\mathbf{p}_R)_1 + (1-\alpha)(\mathbf{p}_R)_2 + \Delta\mathbf{p} \qquad \textit{(on leg 1)}$$

OR

$$\mathbf{p} = \alpha(\mathbf{p}_R)_2 + (1-\alpha)(\mathbf{p}_R)_3 + \Delta\mathbf{p} \qquad \textit{(on leg 2)}$$

OR

$$\mathbf{p} = \alpha(\mathbf{p}_R)_3 + (1-\alpha)(\mathbf{p}_R)_4 + \Delta\mathbf{p} \qquad \textit{(on leg 3)}$$

OR …

with $0 \le \alpha \le 1$ and $\| \Delta\mathbf{p} \| \le$ (corridor width)

Speed Advisories in MILP

$$\mathbf{p} = \beta_1(\mathbf{p}_R)_1 + \beta_2(\mathbf{p}_R)_2 + \beta_3(\mathbf{p}_R)_3 + \ldots + \Delta\mathbf{p}$$

$$\beta_1 \leq b_1$$
$$\beta_2 \leq b_1 + b_2$$
$$\beta_3 \leq b_2 + b_3$$

$$b_1 + b_2 + b_3 + \ldots = 1$$

with $\beta_i \geq 0$ and still $\| \Delta\mathbf{p} \| \leq$ (corridor width)

Speed Advisories in MILP

$$\mathbf{p} = \beta_1(\mathbf{p}_R)_1 + \beta_2(\mathbf{p}_R)_2 + \beta_3(\mathbf{p}_R)_3 + \ldots + \Delta\mathbf{p}$$

$$\beta_1 \le b_1$$
$$\beta_2 \le b_1 + b_2$$
$$\beta_3 \le b_2 + b_3$$

$b_1 = 1$ if $\mathbf{p}$ on leg 1

$= 0$ otherwise

$$b_1 + b_2 + b_3 + \ldots = 1$$

with $\beta_i \ge 0$ and still $\| \Delta\mathbf{p} \| \le$ (corridor width)

Speed Advisories in MILP

$$\mathbf{p} = \beta_1(\mathbf{p}_R)_1 + \beta_2(\mathbf{p}_R)_2 + \beta_3(\mathbf{p}_R)_3 + \ldots + \Delta\mathbf{p}$$

$$\beta_1 \le b_1$$
$$\beta_2 \le b_1 + b_2$$
$$\beta_3 \le b_2 + b_3$$

b_1 = 1 if **p** on leg 1

= 0 otherwise

$$b_1 + b_2 + b_3 + \ldots = 1$$

with $\beta_i \ge 0$ and still $\| \Delta\mathbf{p} \| \le$ (corridor width)

Speed Advisories in MILP

$$\mathbf{p} = \beta_1(\mathbf{p}_R)_1 + \beta_2(\mathbf{p}_R)_2 + \beta_3(\mathbf{p}_R)_3 + \ldots + \Delta\mathbf{p}$$

$\beta_1 \le b_1$

$\beta_2 \le b_1 + b_2$

$\beta_3 \le b_2 + b_3$

$b_2 = 1$ if $\mathbf{p}$ on leg 2

$= 0$ otherwise

$b_1 + b_2 + b_3 + \ldots = 1$

with $\beta_i \ge 0$ and still $\| \Delta\mathbf{p} \| \le$ (corridor width)

Speed Advisories in MILP

$$\mathbf{p} = \beta_1(\mathbf{p}_R)_1 + \beta_2(\mathbf{p}_R)_2 + \beta_3(\mathbf{p}_R)_3 + \ldots + \Delta\mathbf{p}$$

$$\beta_1 \le b_1$$

$$\beta_2 \le b_1 + b_2$$

$$\beta_3 \le b_2 + b_3$$

$b_2 = 1$ if $\mathbf{p}$ on leg 2

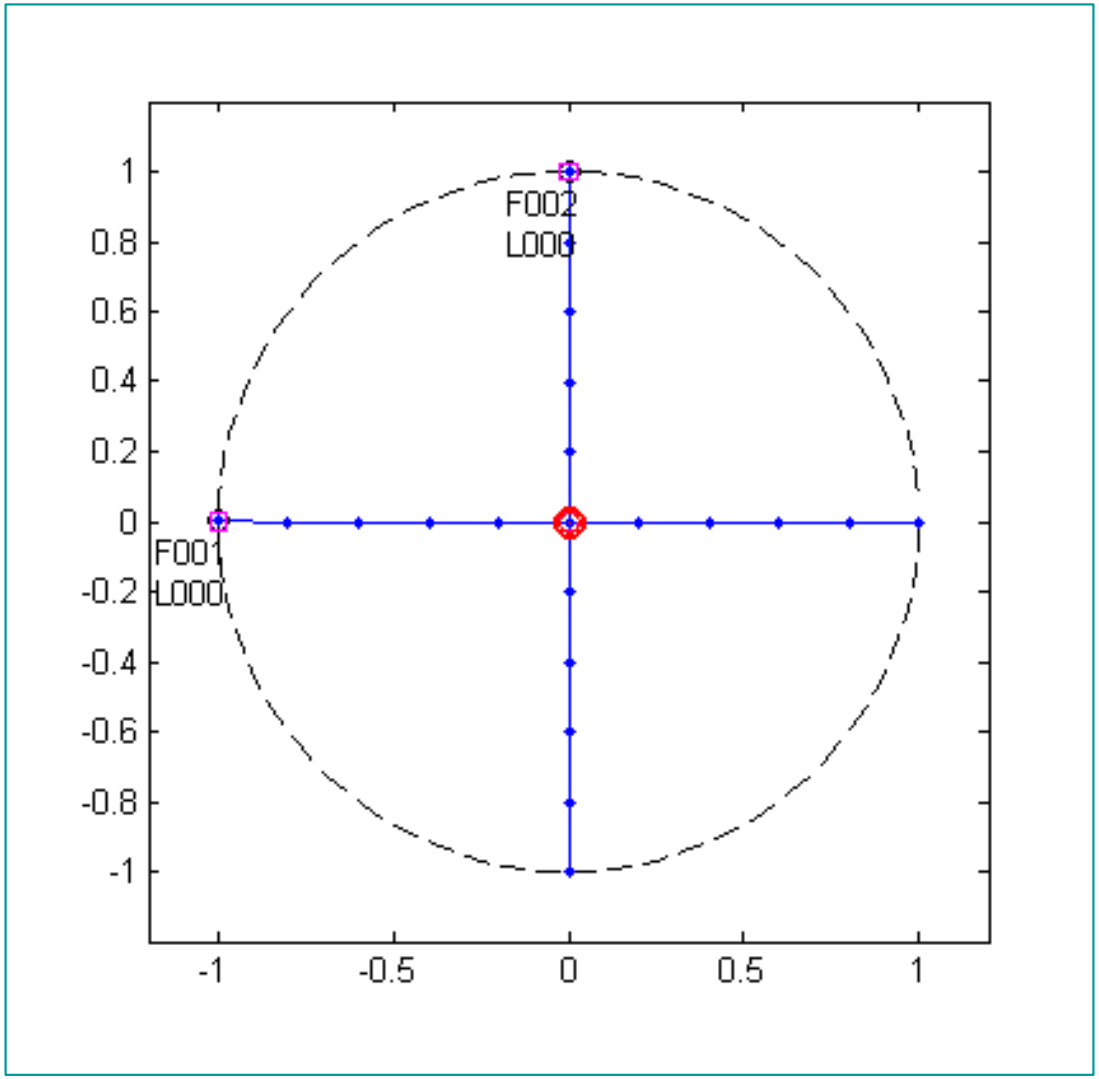
$= 0$ otherwise

$$b_1 + b_2 + b_3 + \ldots = 1$$

with $\beta_i \ge 0$ and still $\| \Delta\mathbf{p} \| \le$ (corridor width)

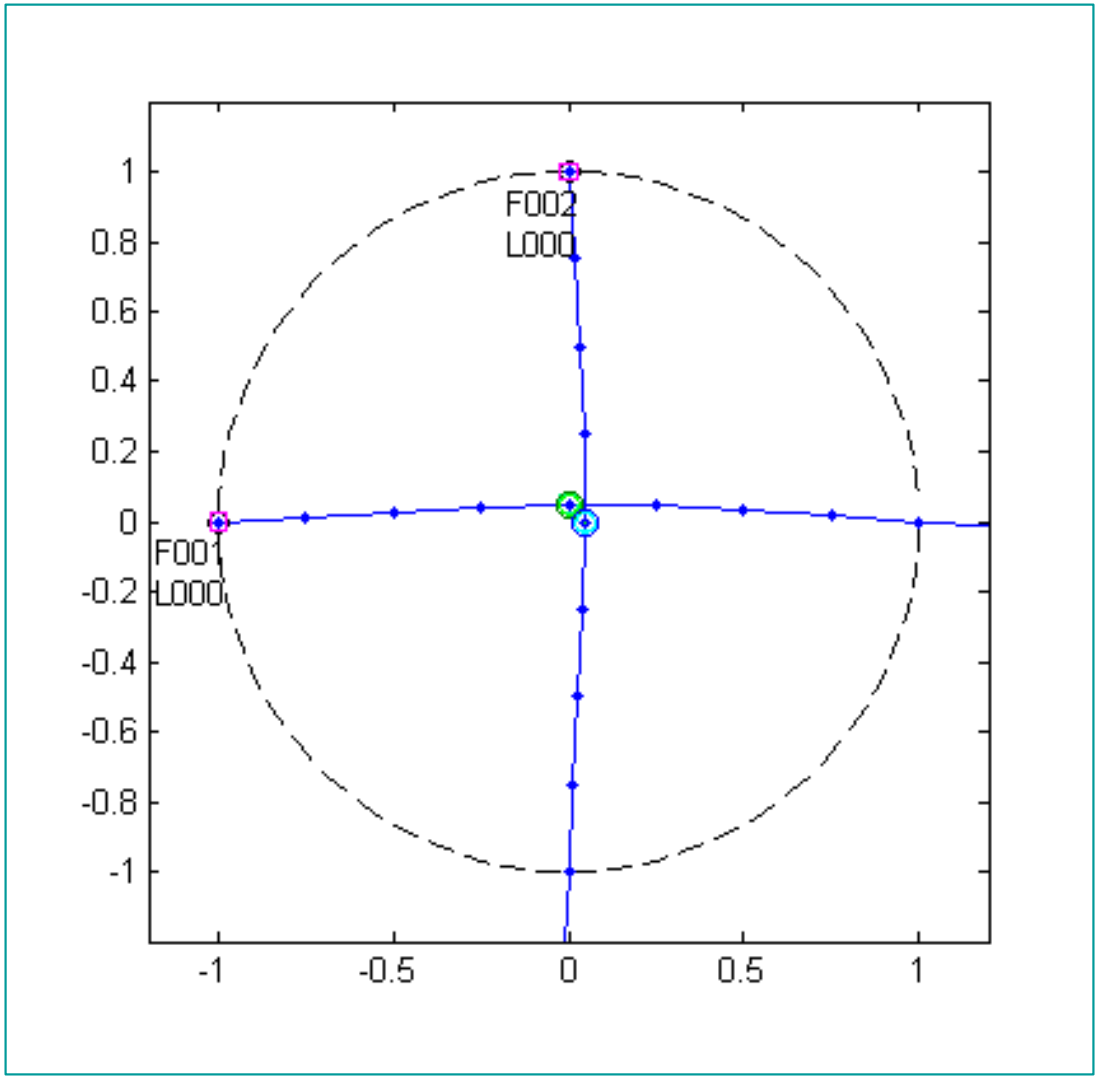
Example of MILP Result

- Reference trajectories
- Conflict halfway



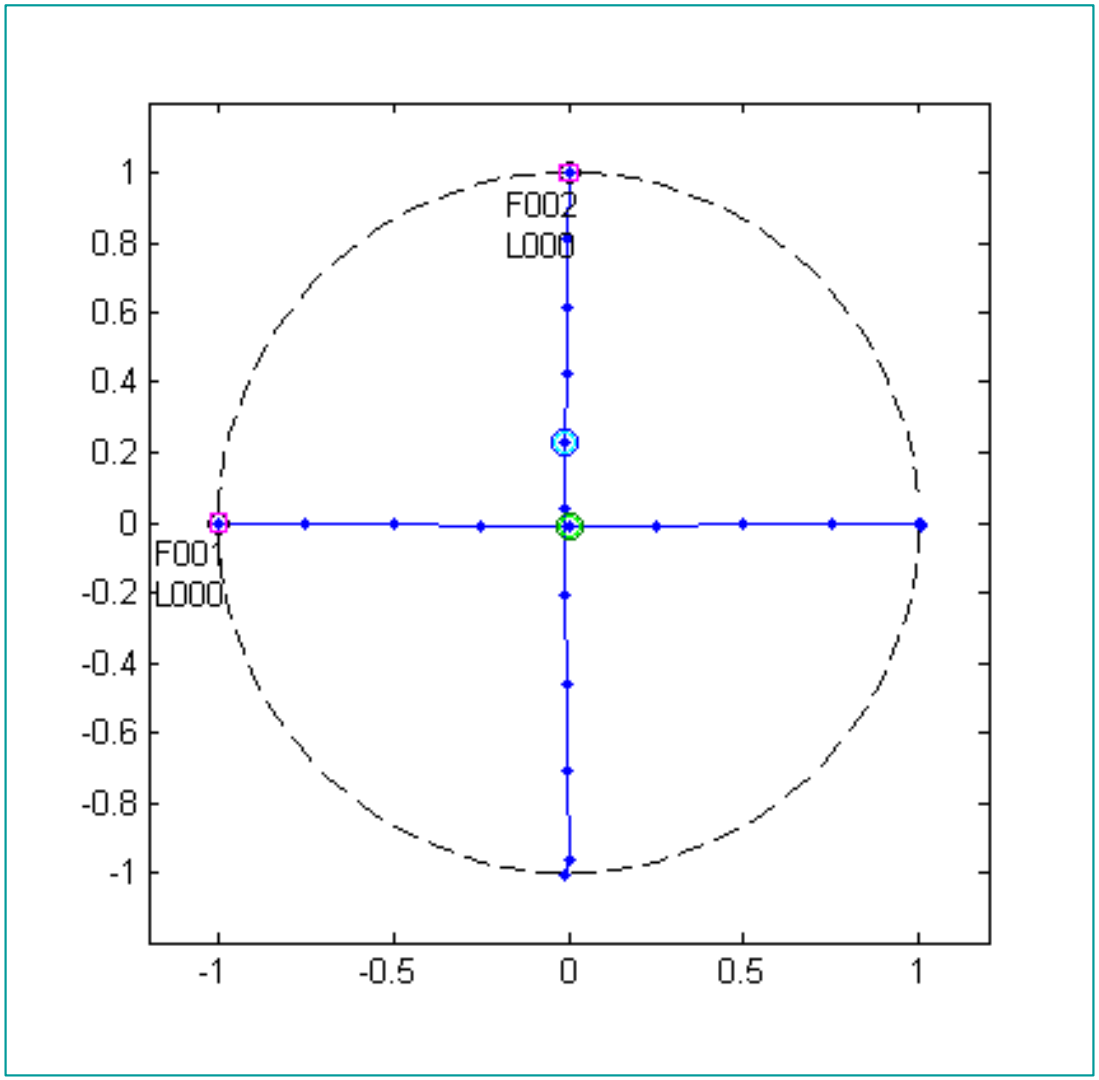
Example of MILP Result

- Resolve without SUPEROPT constraints
- Spatial resolution chosen



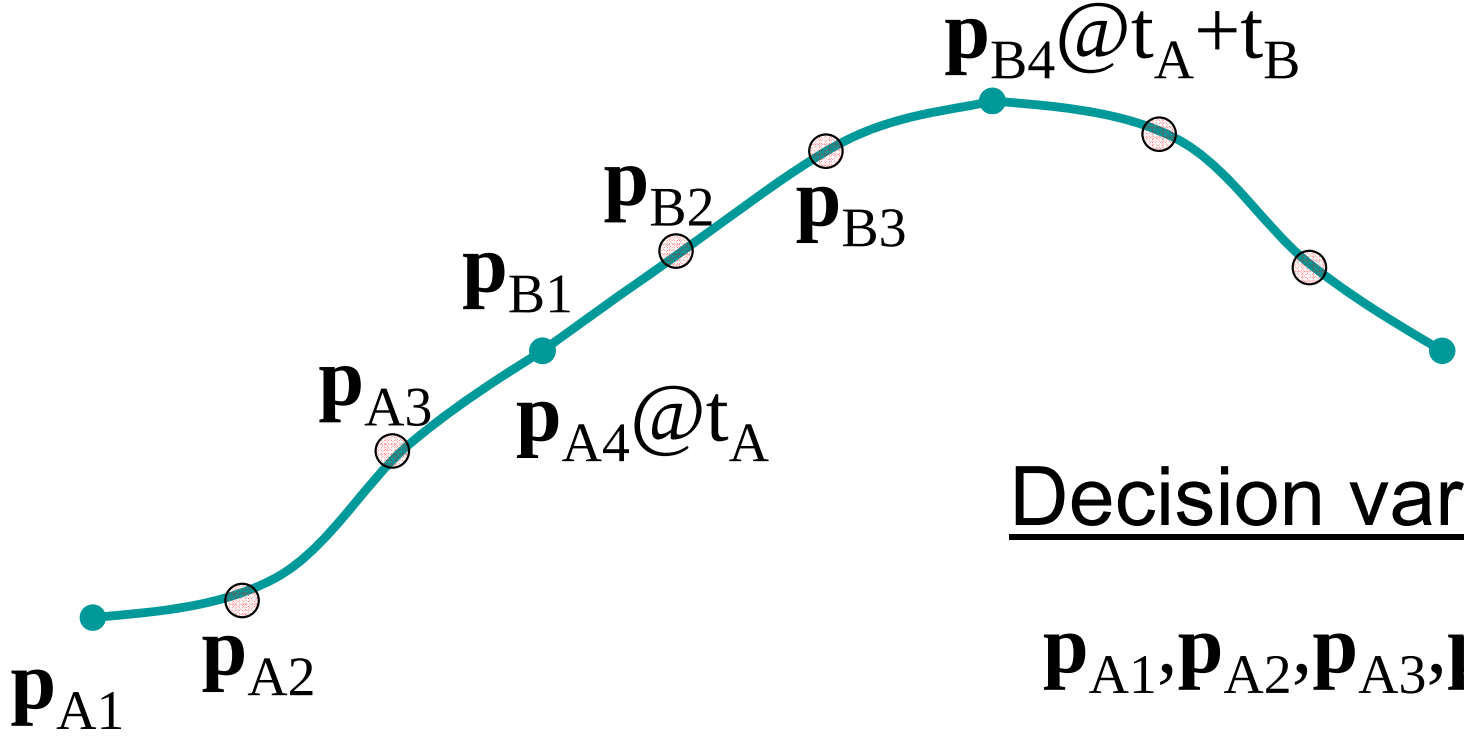
Example of MILP Result

- Resolve with speed change only
- F002 slows for F001 to pass in front
  - Slower computation



Nonlinear Collocation Optimizer

- Different formulation – now time is an explicit decision variable



Decision variables

$\mathbf{p}_{A1},\mathbf{p}_{A2},\mathbf{p}_{A3},\mathbf{p}_{A4},t_A$

$\mathbf{p}_{B1},\mathbf{p}_{B2},\mathbf{p}_{B3},\mathbf{p}_{B4},t_B$

Speed Advisories in Collocation

- Constrain positions to be close to reference positions:

$$\|\mathbf{p}_{A1}-(\mathbf{p}_{A1})_R\| < \text{(corridor width) and}$$
$$\|\mathbf{p}_{A2}-(\mathbf{p}_{A2})_R\| < \text{(corridor width) …}$$

- Leave the times free
  - General trend: because time is explicit variable in collocation, time/space tricks become straightforward

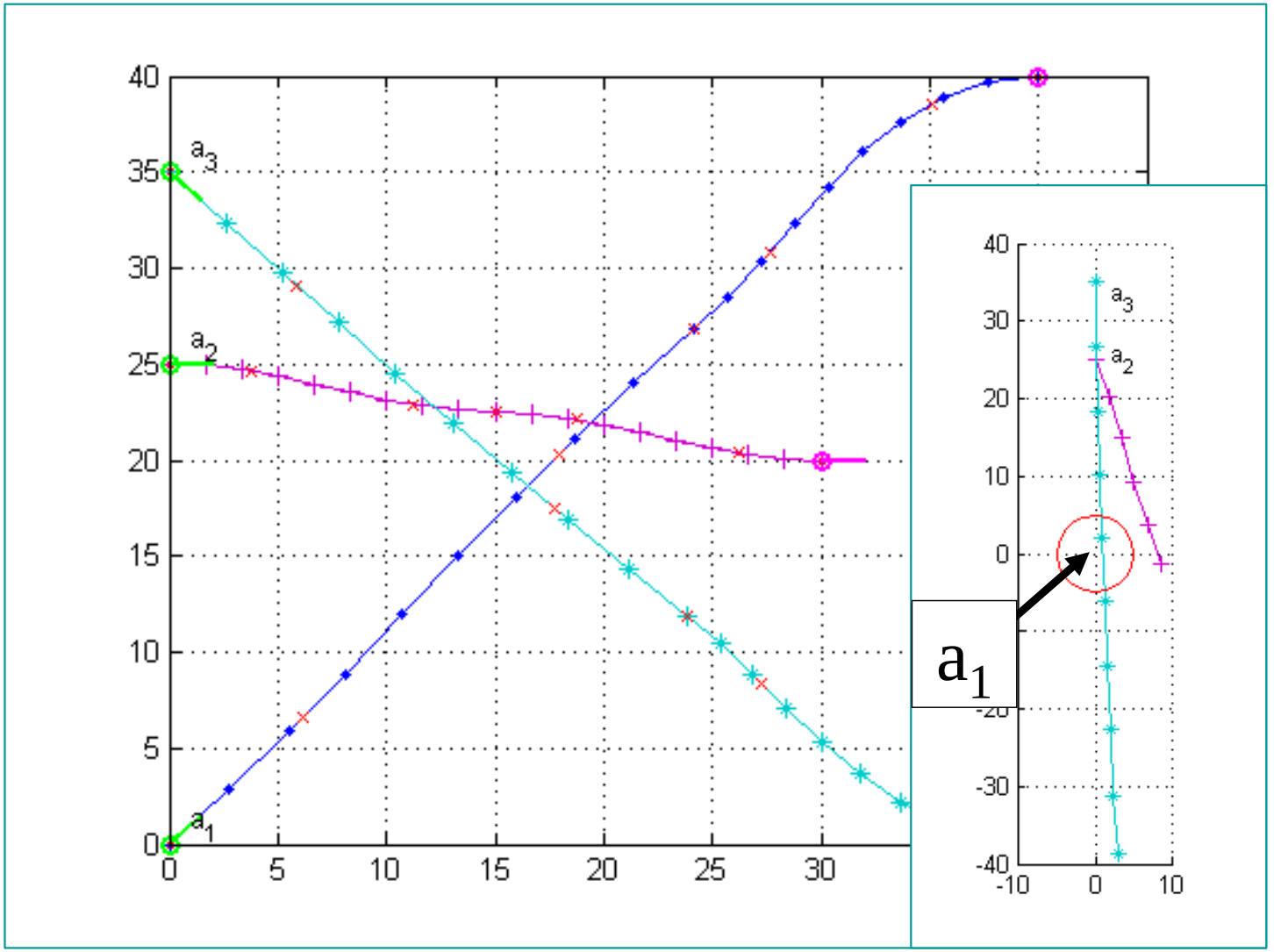
Time and Space Tricks

- Resolution strategies can usually be expressed in terms of time

- A ahead of B:
  $[\mathbf{p}_B(t) - \mathbf{p}_A(t+T)] \notin$ (conflict cylinder) $\forall T>0$
  - Complete spatial separation if true for all T
  - Also enables control of downstream sequencing

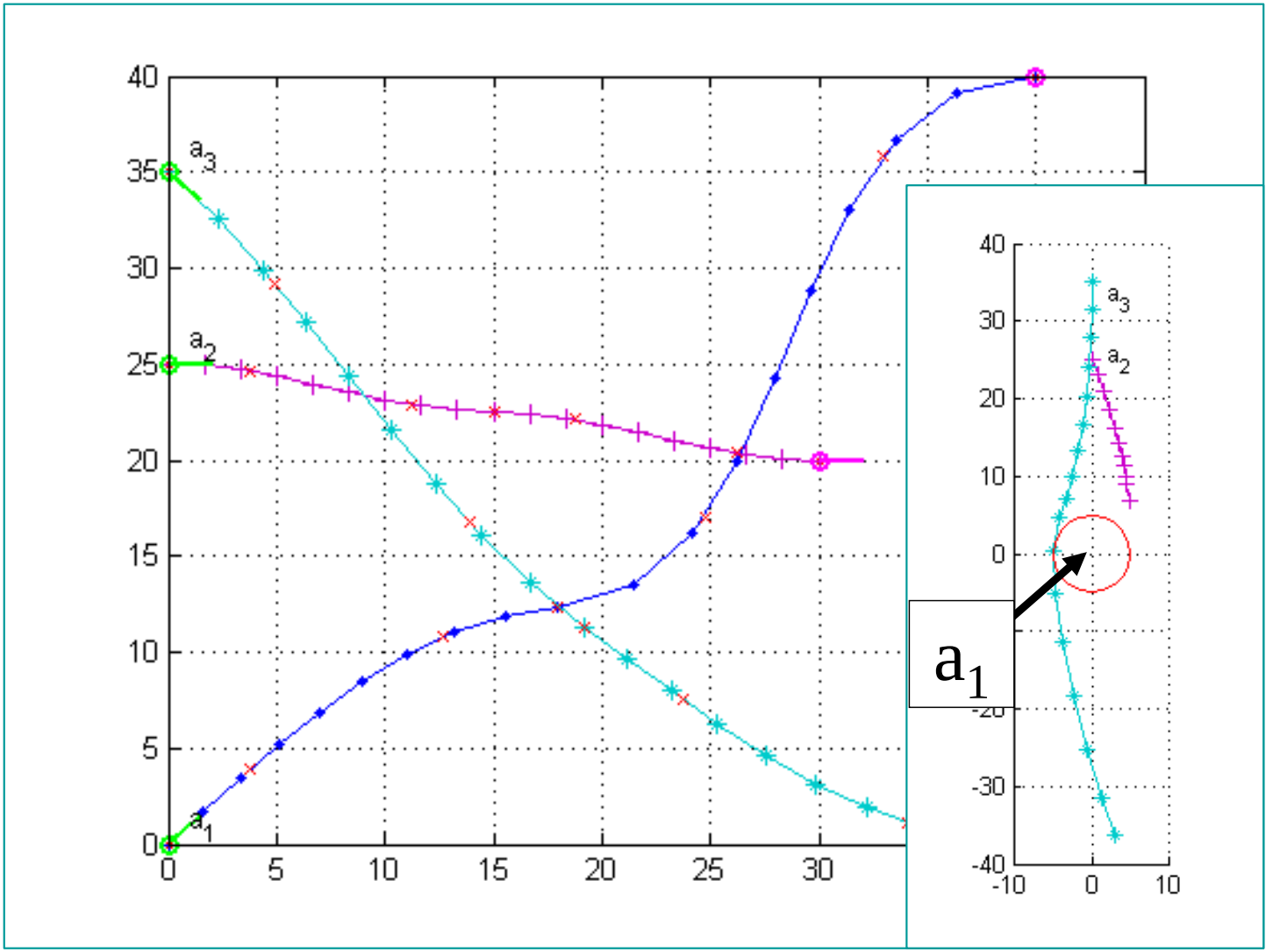
Collocation Example

- Reference
- a1 and a3 in conflict



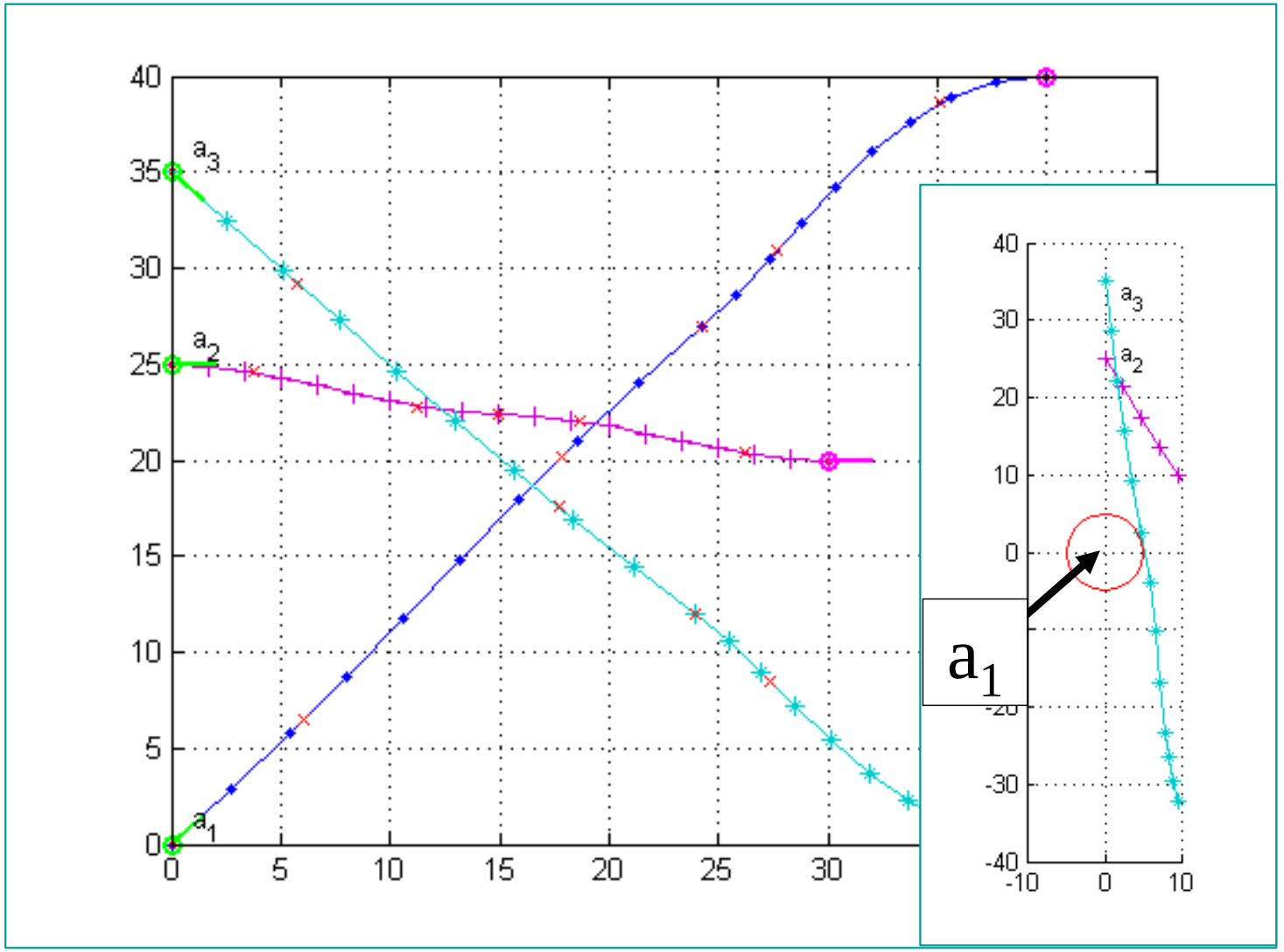
Collocation Example

- Free resolution
- a1 diverts



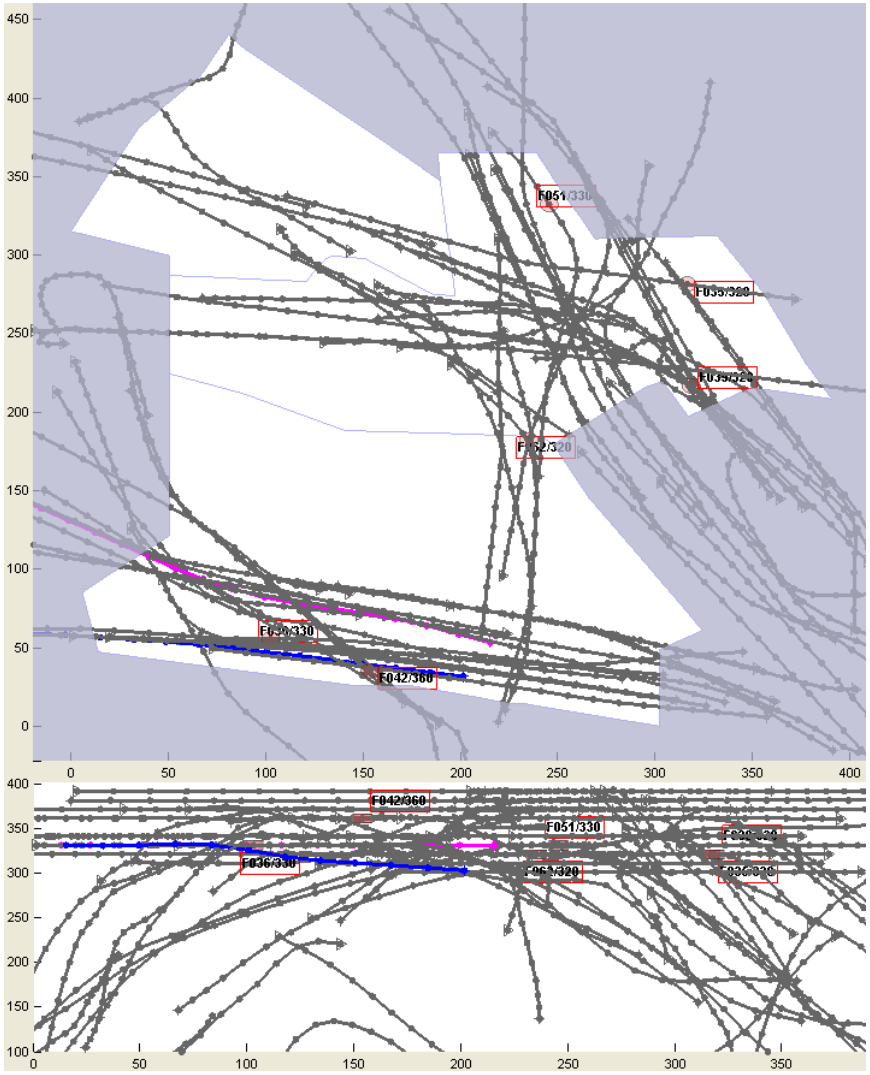
Collocation Example

- Speed resolution
- a1 slows



Computation

- Studied a “Multi Sector Area” over western UK
  - Spatial: two minutes
  - Temporal: 10+
    - Lifelines: iterative conflict handling; cut-down binaries



Conclusions

- Added constraints to trajectory optimizers to enforce “speed changes only”
  - MILP and nonlinear collocation
- Joins a family of other constraints to capture supervisor requirements without over-constraining
  - Fix lateral; Fix vertical;
  - A ahead of B; A over B