NEXTOR Annual Research Symposium
November 14, 1997

Session III
Issues for the Future of ATM

Distributed ATM Concepts

Shankar Sastry, UC-Berkeley
and
Claire Tomlin, UC-Berkeley

Modeling, Specification and Safety Analysis of CTAS

George J. Pappas and Shankar Sastry
Department of Electrical Engineering
University of California at Berkeley
Berkeley, CA 94720

John Lygeros and Nancy Lynch
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139

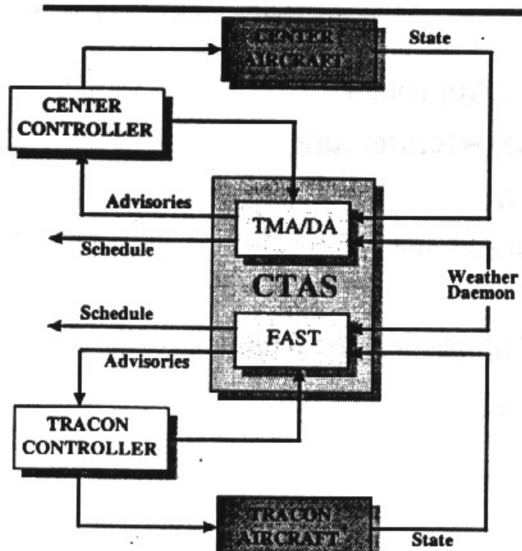
Talk Outline

- The Formal Systems Approach
- Description of CTAS Architecture
- Modeling Formalism
  - Hybrid Input-Output Automata
- Safety Notions
  - Nominal, Robust, Degraded and Structural
- Analysis Methodology
- Conclusions

The Formal Methods Approach

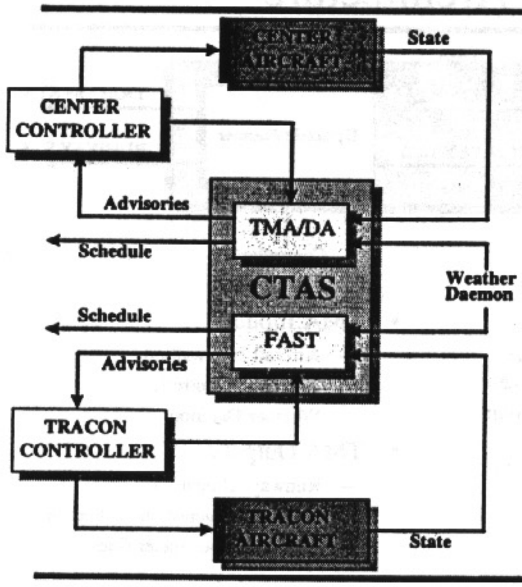
- Think of CTAS and NAS as a large scale system
  - Methodology should not depend on CTAS details
  - System view suggests correct safety notions
- *Formal Methods Approach: Given a model of the system and notions of safety, we utilize formal methods which provide conditions under which the system is guaranteed to be safe*
- Methodology
  - Formal Modeling of the System
  - Formal Specification
  - Formal Analysis of the System

CTAS Architecture



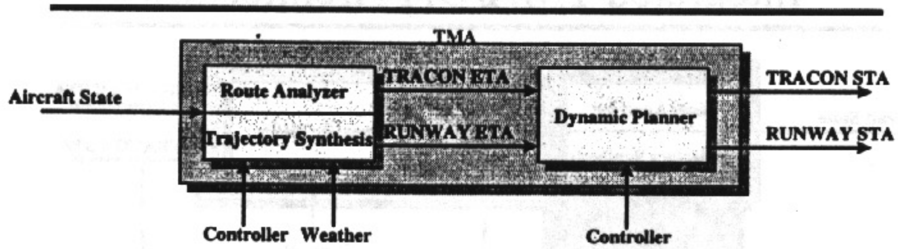
- CTAS consists of
  - Traffic Management Advisor
  - Descent Advisor
  - Final Approach Spacing Tool
- Future Additions
  - Expedited Departure Tool
  - User Preferred Routing
- TMA and DA exist in Center
- FAST exists in TRACON
- CTAS is
  - Human Centered
  - Reactive
  - Distributed and Hierarchical
  - Hybrid
  - SAFETY CRITICAL

CTAS Architecture



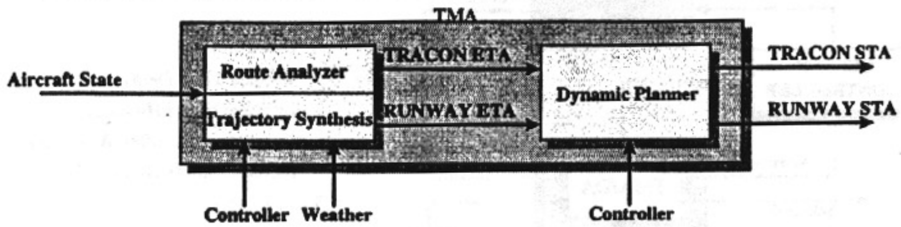
- CTAS Goals
  - Reduce Time Delays
  - Increase Fuel Efficiency
  - Decrease Controller Workload
  - Increase Safety Standards
- CTAS Inputs
  - Radar Daemon
  - Weather Daemon
  - Controllers and TMC
  - Various Databases
- CTAS Outputs
  - Runway Allocation
  - Landing Sequence/Schedule
  - Advisories to Controllers

TMA Architecture



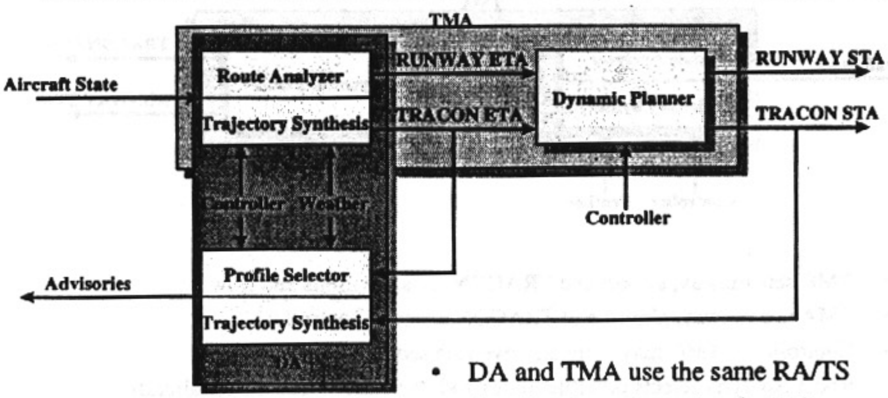
- TMC sets runway, airport and TRACON capacity limits and flow rates
- TMA is a runway, airport and TRACON capacity controller
- Controller or TMC may manually override sequence, schedule
- Route Analysis selects possible routes and degrees of freedom per aircraft
- Trajectory Synthesis puts 4D profiles on each possible route
- Possible ETAs per aircraft, route, degree of freedom are inputs to DP
- DP performs runway allocation, landing sequences and scheduling

TMA Architecture



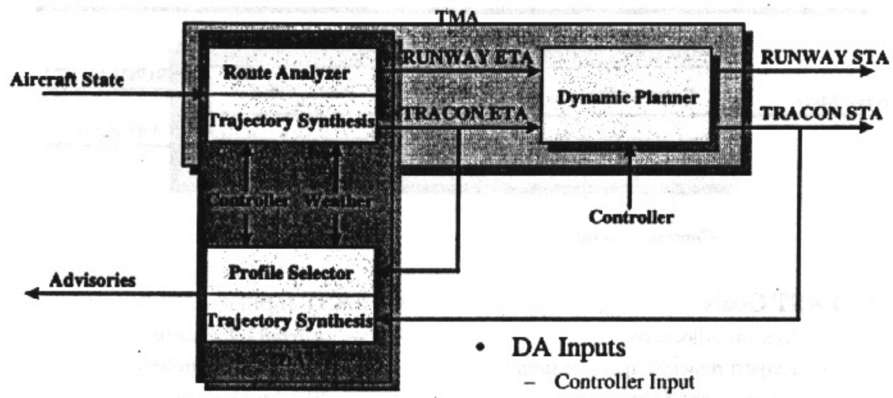
- TMA Goals
  - Meet separation requirements at runway threshold, TRACON Gates
  - Satisfy capacity and flow constraints at various fixes
  - Minimize time delay (STA-ETA)
- TMA Inputs
  - Aircraft Information
  - TMC and Controllers
  - Weather Daemon
- TMA Outputs
  - Runway Allocation
  - Landing sequence and schedule
  - STA at various meter fixes

Integrated TMA/DA Architecture



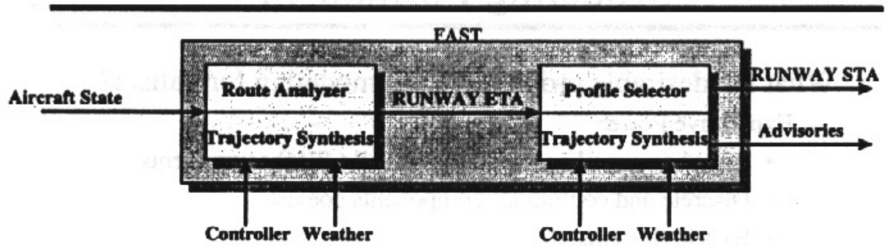
- DA and TMA use the same RA/TS
- Conflict detection exists in PFS
- Conflict resolution is manual
- TMA and DA are not integrated!

Integrated TMA/DA Architecture



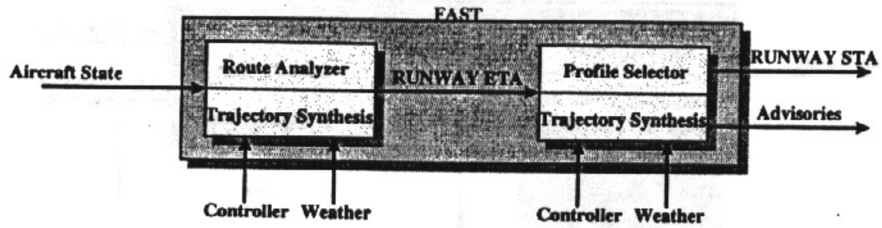
- DA Goals
  - Compute Fuel Efficient Descents
  - Meet TRACON STA of TMA
  - Perform Conflict Resolution
- DA Inputs
  - Controller Input
  - Weather and Radar Daemon
  - TRACON STA of TMA
- DA Outputs
  - Descent Advisories

FAST Architecture



- Passive FAST performs runway allocation, sequencing and scheduling
- Active FAST outputs heading, speed advisories to aircraft
- No integration between FAST and TMA or DA
- FAST schedule overrides initial schedule of TMA
- PFS resolves conflicts by iterating on routes/degrees of freedom
- Knowledge based system in PFS performs sequencing and scheduling

FAST Architecture

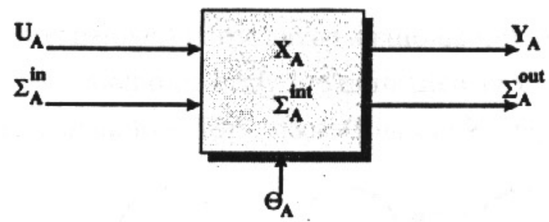


- FAST Goals
  - Runway Allocation
  - Respect runway/airport constraints
  - Sequence and Scheduling
  - Perform conflict detection/resolution
  - Suggest advisories to controllers
- FAST Inputs
  - Aircraft Information
  - TRACON Controllers
  - Weather Daemon
- FAST Outputs
  - Runway Allocation
  - Landing sequence and schedule
  - Advisories

Modeling Formalism

- What are desirable properties for a modeling formalism?
  - Expressive Power
    - Must be general enough to model all CTAS components
    - Discrete and continuous components coexist
    - Hybrid Systems
  - Compositionality
    - Allows modeling of large scale, distributed systems
    - Interconnection of various input-output components
    - Allows modular modeling and specification
  - Hierarchical Modeling
    - Allows modeling of a system at various levels of abstraction
    - Level of abstraction depends on task to be performed

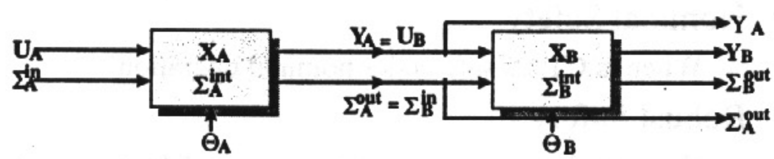
Hybrid Input/Output Automata



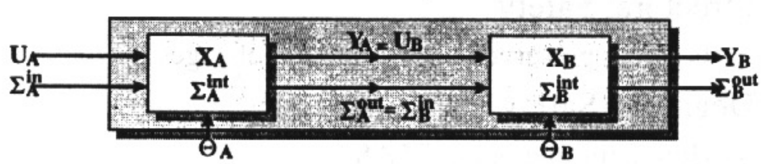
- A hybrid input/output automaton A is defined by
  - Input, output and internal typed variables
  - Input, output and internal actions
  - State space is set of all possible variable values
  - Initial conditions
  - A set W of trajectories of variables and D of discrete transitions
- Each action has an associated precondition and effect
- An execution of the automaton is $\alpha = w1\ a1\ w2\ a2\ w3\ a3.....$

Hybrid Input/Output Automata

- Compositions of compatible hybrid automata are hybrid automata



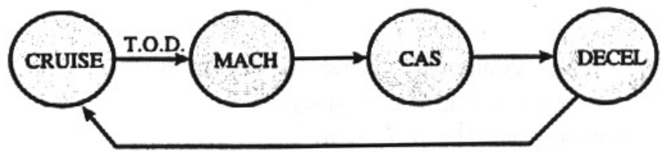
- Variable and action hiding allows building macrocomponents



- Composite system satisfies composite specification

Hybrid Input/Output Automata

- Hybrid I/O automata have formal linguistic descriptions
- Another visualization of hybrid automata
- Example: Nominal Descent Profile of an aircraft



- Trajectories are described by the differential equations
- Actions occur when capture conditions are satisfied
  - Top of Descent
  - Constant MACH or CAS captured

Safety Notions

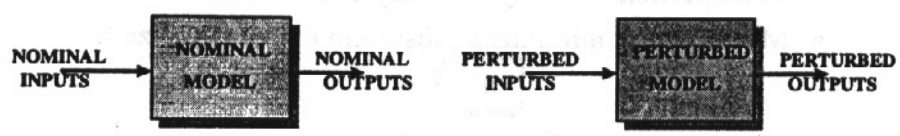
- Nominal Safety
  - When is CTAS safe under nominal operation?
- Robust Safety
  - How much error and uncertainty can CTAS tolerate?
- Structural Safety
  - Is safety preserved after structural changes?
- Degraded Safety
  - How fault tolerant is CTAS?

Nominal Safety

- Assume perfect system information and operation
  - No uncertainty in sensors, models, parameters or weather
  - No malfunctions or failures of components
  - Nice weather!
- Safety Specification at various levels of CTAS
  - Does TMA produced timeline respect separation requirements?
  - Does TMA scheduling respect airport and TRACON constraints?
  - Are conflicts always resolved in FAST?
- Additional nominal safety notions
  - Will CTAS issue an advisory? (Completeness)
  - Are the CTAS outputs stable? (Controller workload)
  - What are CTAS' operational limits? (Cost/Benefit Analysis)
  - Are CTAS advisories feasible by FMS? (Air/Ground Integration)

Robust Safety

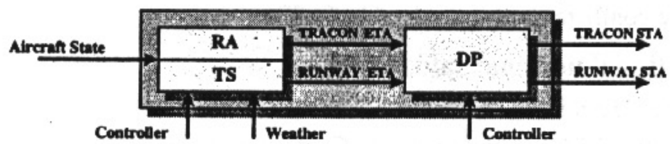
- What happens to nominal safety in the presence of errors?



- Error Sources
  - Modeling uncertainty (TS aircraft and aerodynamic models)
  - Sensor and parametric uncertainty (mass and inertia of aircraft)
  - Weather models (spatially and temporally coarse)
- Establish bounds between nominal and perturbed outputs
- Smallest deviation from nominal system that results in unsafe operation is a measure of robust safety
- Sensitivity analysis to various error sources

Structural Safety

- Will CTAS remain safe after structural changes?
  - Implementation Changes
  - Architectural Changes
- Implementation versus Specification
- Composite system remains safe if new component implementation satisfies old component specification
- Example: Incorporate improved Dynamic Planner

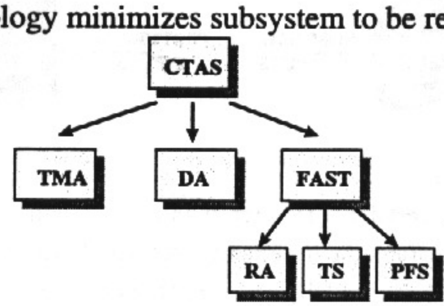


DP Spec: STAs meet certain flow rates
STAs are properly separated

September 22nd, 1997 CTAS Safety Analysis NASA Langley

Structural Safety

- If architectural or functional changes happen, reanalyzing some portion of the system may be necessary
- Methodology minimizes subsystem to be reanalyzed



- Moving PFS functionality to RA requires analysis only of FAST subsystems to verify that new FAST meets old spec
- Challenging Problem: Given components that satisfy their specs, is there an architecture to satisfy a system spec?

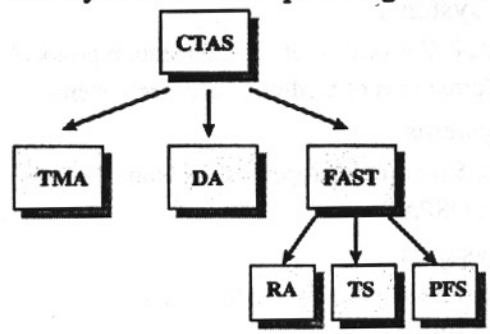
September 22nd, 1997 CTAS Safety Analysis NASA Langley

Degraded Safety

- **Possible System Failures**
  - Inclement weather and drastic weather changes
  - Faults in engines, sensors, power or communication devices
- **Safe but graceful performance degradation**
  - Small component failures do not affect the whole system
- **What combination of faults leads to unsafe operation?**
- **How can we quantify effect of failures to safety?**
- **Probabilistic Reasoning**
  - Given the probability distribution of malfunctions, compute the probability of the system becoming unsafe
- **Probability of unsafe operation measures degraded safety**

Safety Analysis

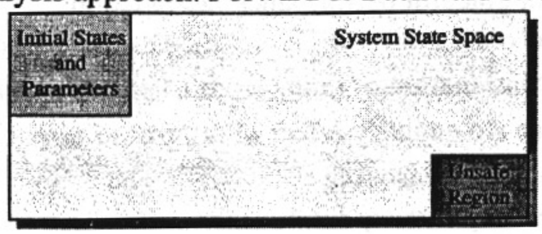
- **How can one analyze such a complex large scale system?**



- Step 1 : Top down specification refinement
- Step 2 : Verify that low level systems meet specification
- Step 3 : Abstract behavior of composite system

Safety Analysis

- **Safety specs can be expressed as undesirable state regions**
  - **Will aircraft lose separation? Is TRACON capacity exceeded?**
- **Specs can also be formulated using performance monitors**
- The analysis approach: Forward & Backward Reachability

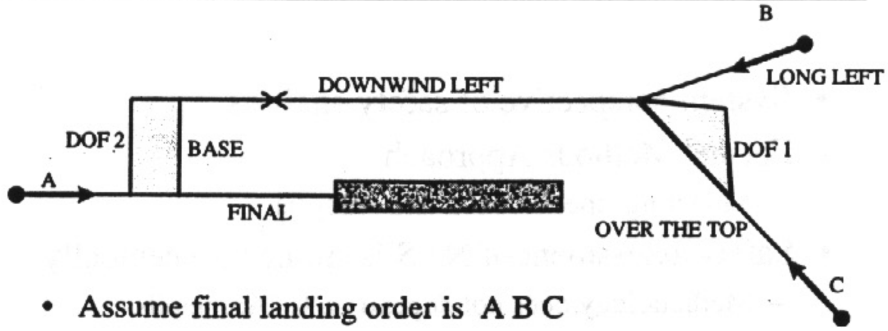


- Forward : *Verify safety given parameters and initial states or generate trajectory leading to unsafe operation*
- Backward : *Determine which initial states and parameters are reachable from the unsafe region*

Safety Tools

- Discrete Systems
  - COSPAN (Correctness of communication protocols)
  - VIS (Correctness of hardware/software systems)
- Timed Systems
  - KRONOS (real-time properties of communication networks)
  - Timed COSPAN
- Hybrid Systems
  - HyTech (Rectangular Hybrid Systems)
- Various Mathematical Tools from
  - Systems Theory
  - Probability Theory
  - Computer Science and Logic

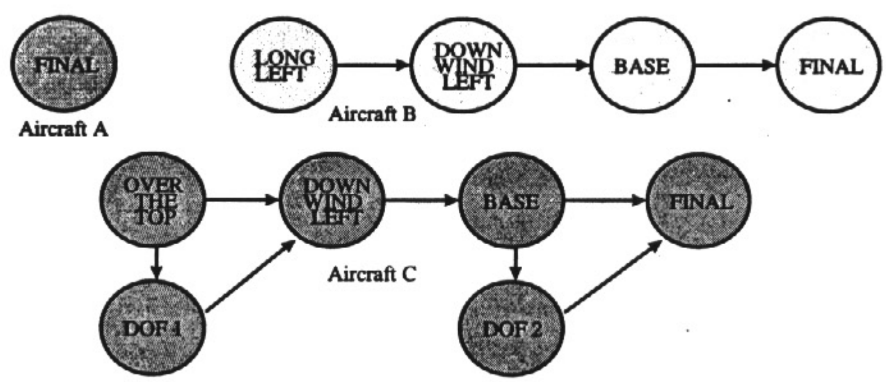
Conflict Resolution in FAST



- Assume final landing order is **A B C**
- Potential conflict between B and C on downwind left
- Aircraft C must be delayed using 2 degrees of freedom
- Speed and altitude profiles dictated by TRACON procedures
- Question: For what initial configurations (horizontal and vertical coordinates) of Aircraft C is conflict avoided?

Conflict Resolution in FAST

- Model Aircraft A, B and C as hybrid automata



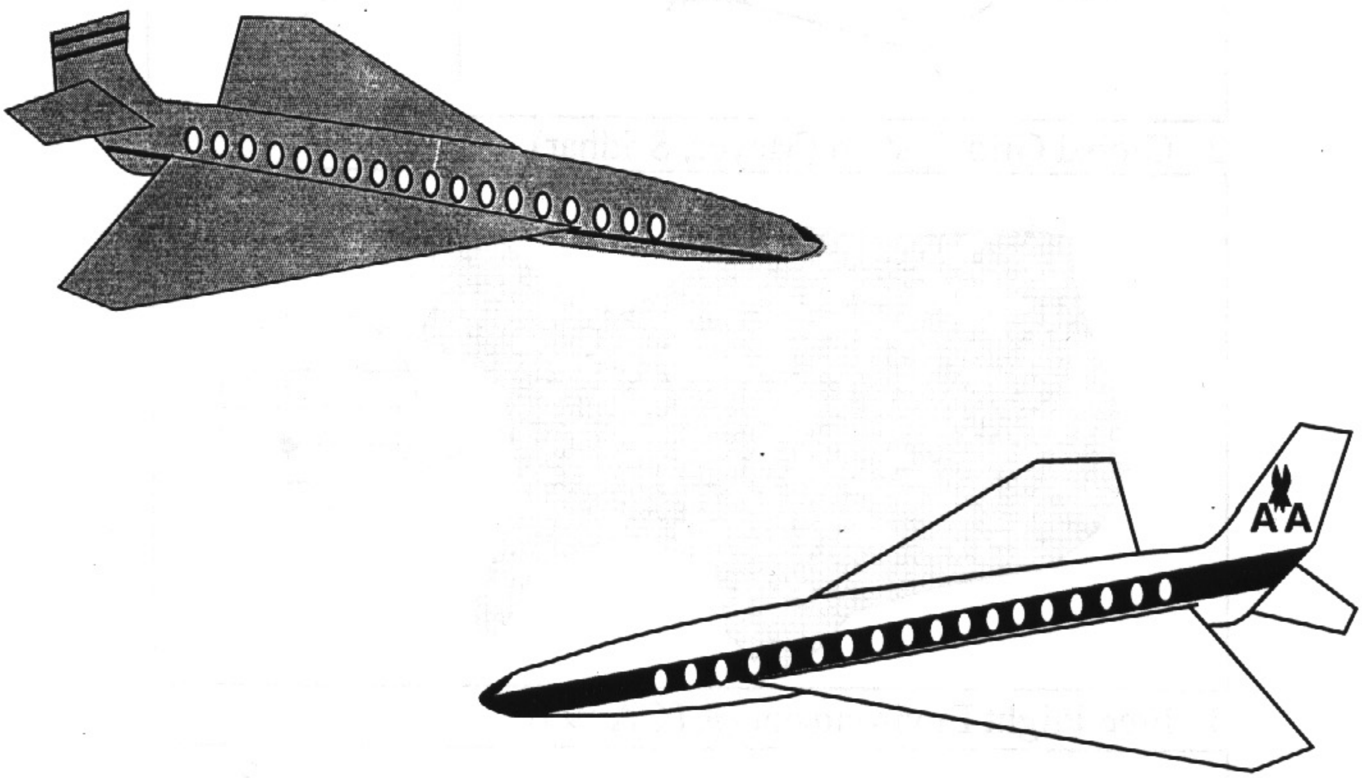
- System model : Aircraft A || Aircraft B || Aircraft C
- Specification : Aircraft A, B and C do not lose separation

Conclusions

- System perspective of safety analysis
- Formal Methods Approach
  - Modeling, specification and analysis
- Safety assessment of NAS is similar conceptually
  - Methodology does not depend on CTAS details
- Questions are challenging but also the right ones!

Algorithms for Distributed Air Traffic Management

Claire Tomlin, George Pappas, Jana Košecká,
John Lygeros, and Shankar Sastry



Intelligent Machines and Robotics Laboratory
Electrical Engineering and Computer Sciences
University of California at Berkeley

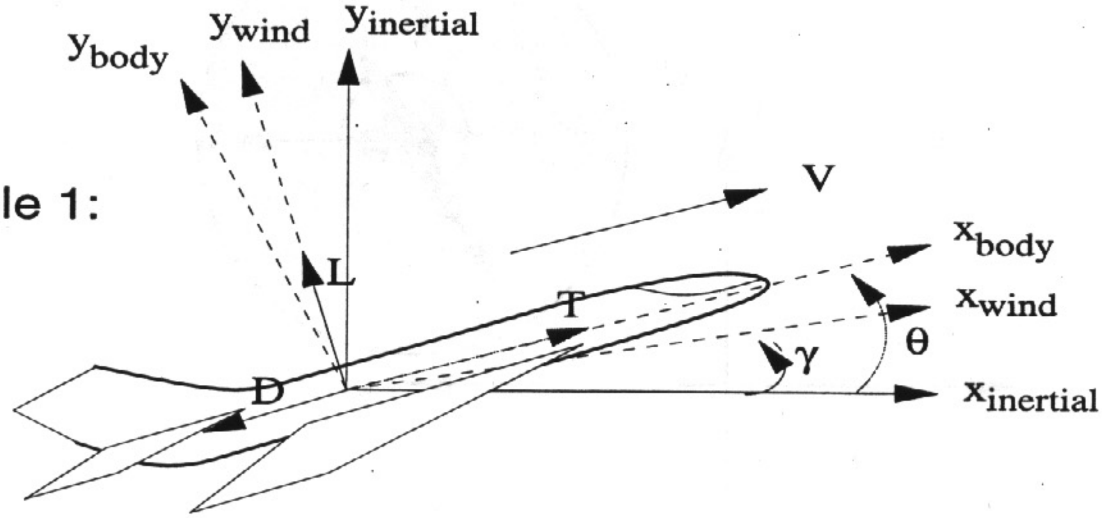
Model

Dynamic aircraft model: $\dot{x} = f(x, u, d)$

Alphabet of modes: Q

Action set: $\Sigma = \{ \sigma_1, \sigma_2, \sigma_3 \ldots \}$

example 1:



$$\dot{V} = -\frac{a_D V^2}{m} - g \sin\gamma + \frac{1}{m} T$$

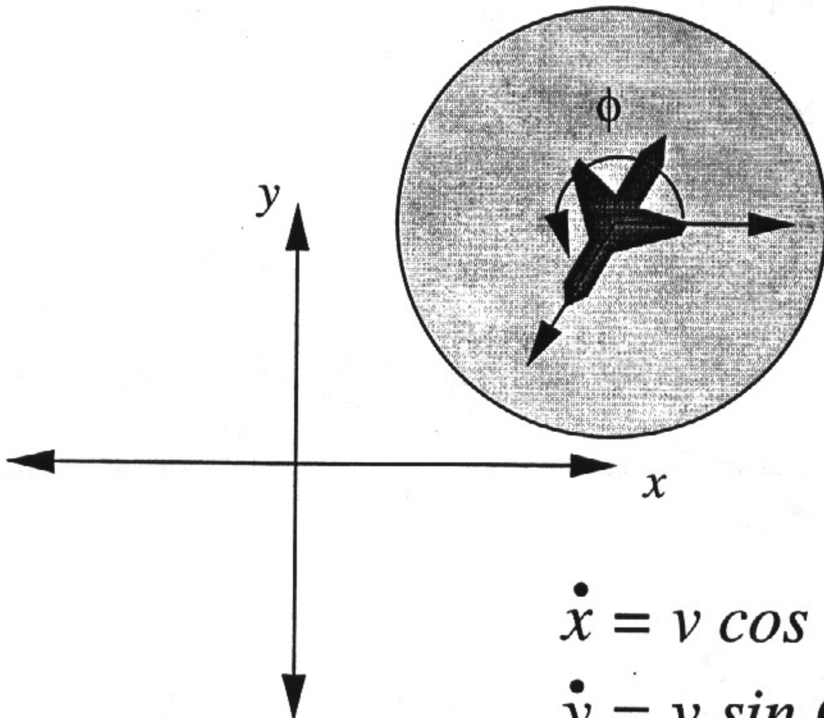
$$\dot{\gamma} = -\frac{a_L V(1-c\ \gamma)}{m} - \frac{g \cos\gamma}{V} + \frac{a_L V c}{m}\theta$$

Modes	Constant Inputs
Cruise	(vertical rate & speed)
Level decel/accel	(thrust & vertical rate)
Mach desc/asc	(Mach & thrust)
CAS desc/asc	(CAS & thrust)
desc/asc accel/decel	(thrust & vertical rate)

Actions: capture conditions on Mach, CAS, altitude

Model

example 2:

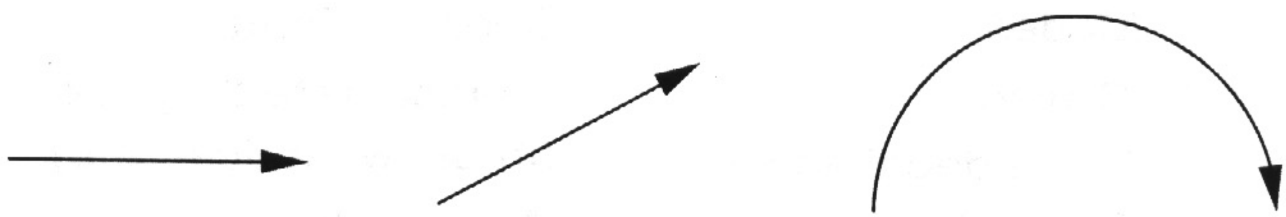


$$\dot{x} = v \cos \phi$$

$$\dot{y} = v \sin \phi$$

$$\dot{\phi} = \omega$$

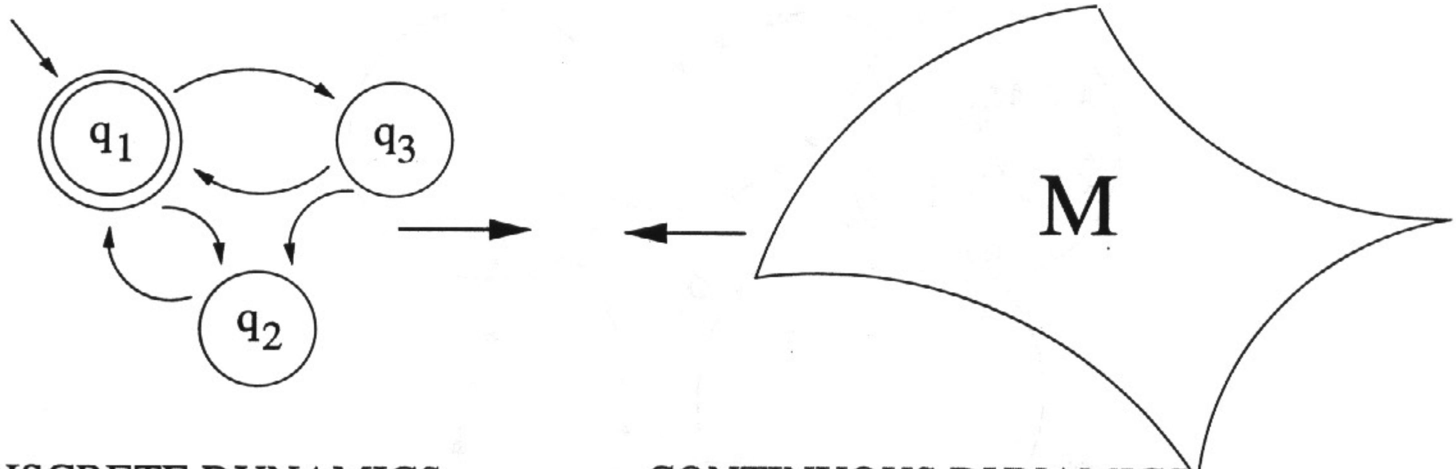
Modes:



Actions: length of maneuver

switching time between maneuvers

Hybrid Systems



DISCRETE DYNAMICS

ie. Finite State Machine

$Q = \{q_1, q_2, \ldots, q_m\}$

CONTINUOUS DYNAMICS

ie. Manifold M

f: $M \rightarrow TM$, $\dot{x} = f(x)$

Computer Scientists ⟶

- Model checking
- Automatic theorem proving

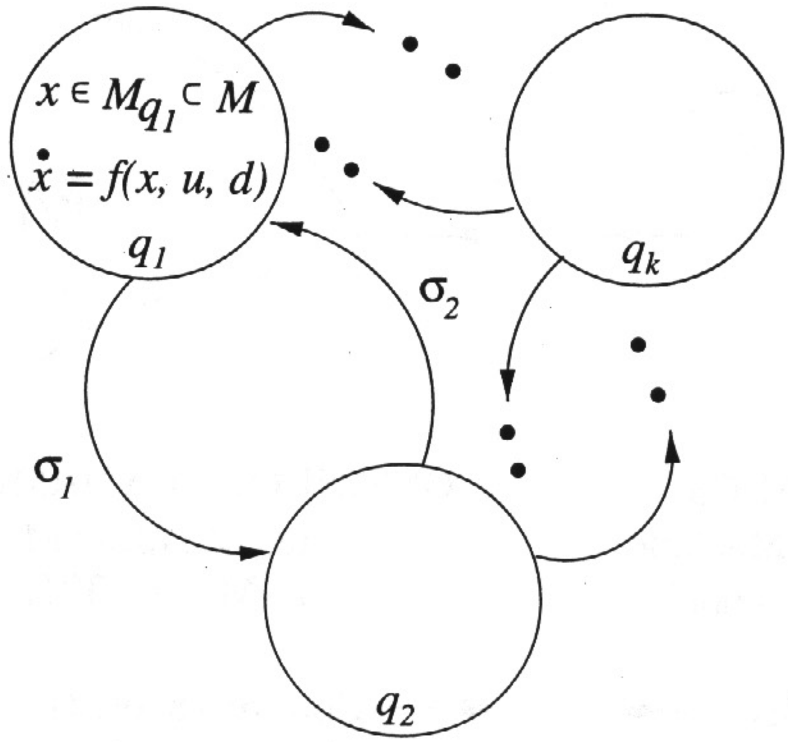
⟵ Control Theorists

- Lyapunov's Theorems
- Optimal Control
- Discrete Event Control

(Alur, Dill) Timed Automata
(Sifakis) Timed, Hybrid Automata
(Henzinger) Hybrid Automata
(Lynch) Theorem Proving
(Kurshan) Coordinating Automata
(Manna, Pnueli) Temporal Verification

(Brockett) Hybrid Models
(Nerode, Kohn) Multi-agent
(Branicky) Optimal Control
(Michel) Switched Systems
(Lygeros, Godbole, Sastry) Game Theoretic Approach
(Caines) Hierarchical Lattices

Hybrid System Model



M	n-Manifold
$Q \to 2^M$	Invariants
$f(q) : M \to TM$	Flows
$\Sigma \subset Q \times M \times Q \times M$	Jumps
$I \subset Q \times M$	Initial Conditions

Safety does there exist a sequence of jumps and flows from an initial state to an unsafe state?

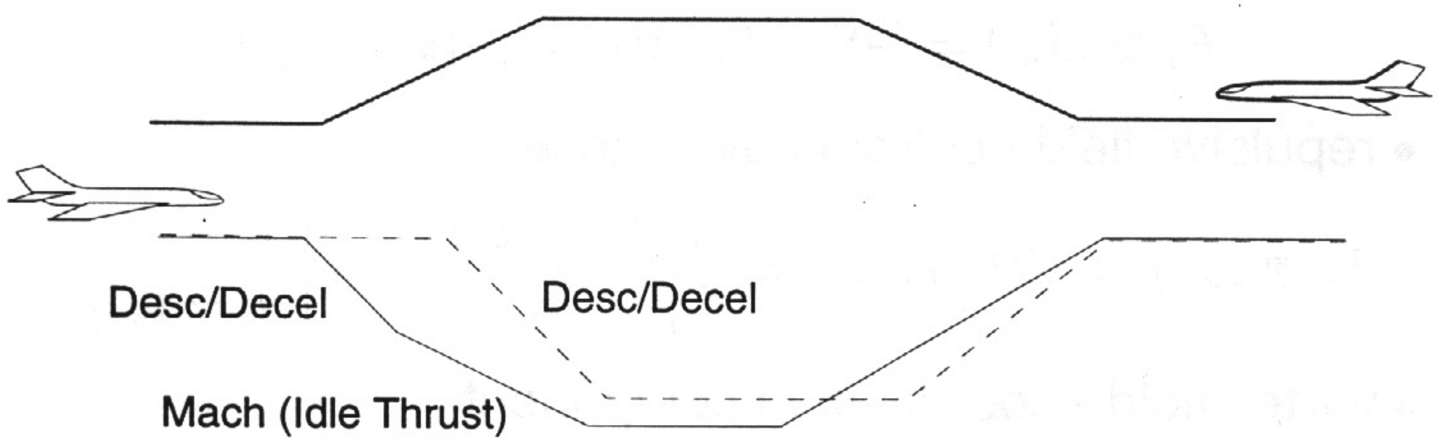
Pre(R) $R \subset Q \times M$: All initial states for which there are trajectories linking these states to some state in R

Method

$$\dot{x} = f(x, u, d), \quad Q, \quad \Sigma$$

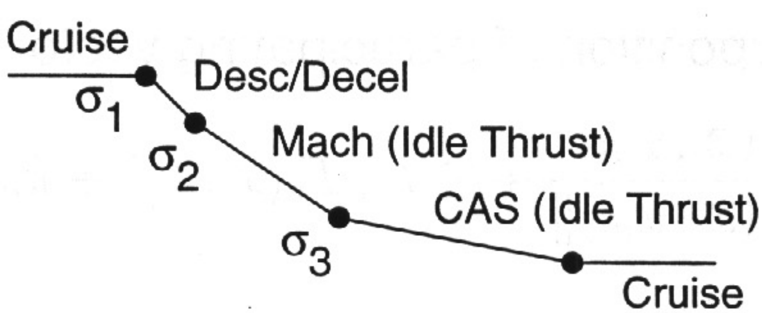
1. Avoidance Protocol: generates possible sequences of flight modes

example:



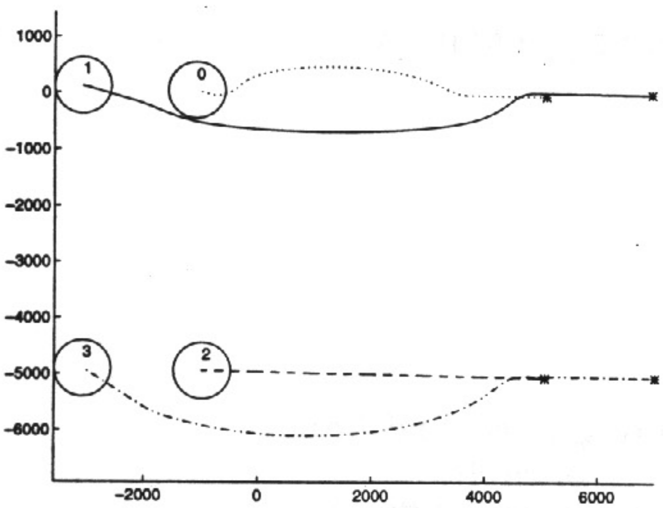
2. Controller Synthesis: generates control input u, discrete actions σ

example:

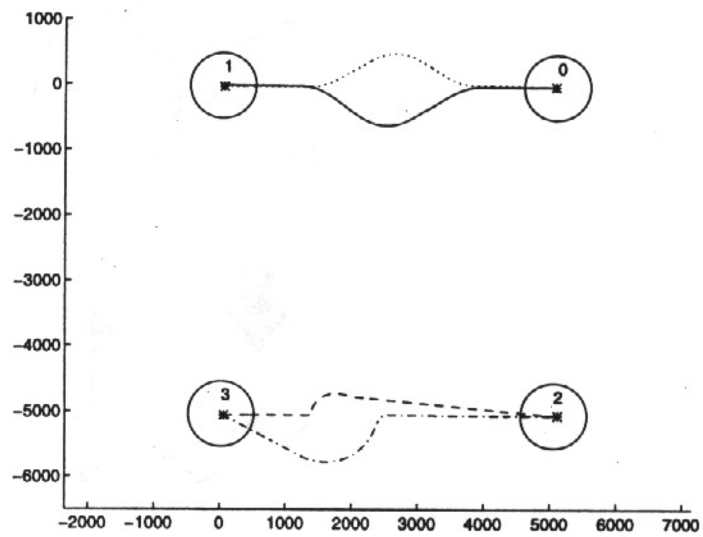


Examples

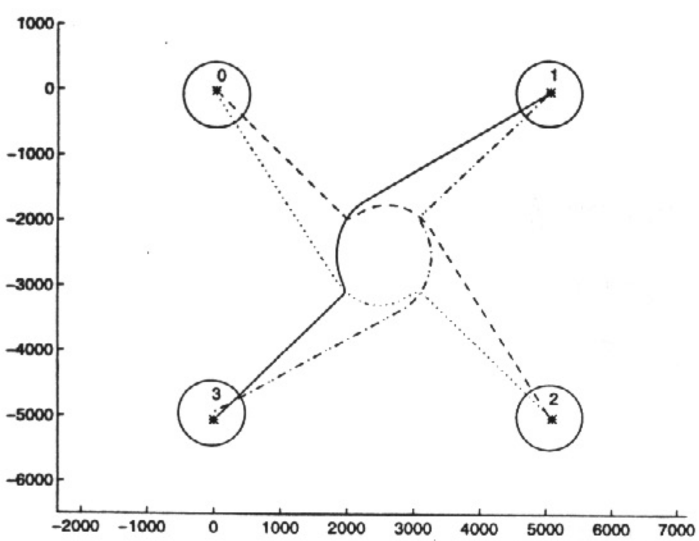
Overtake



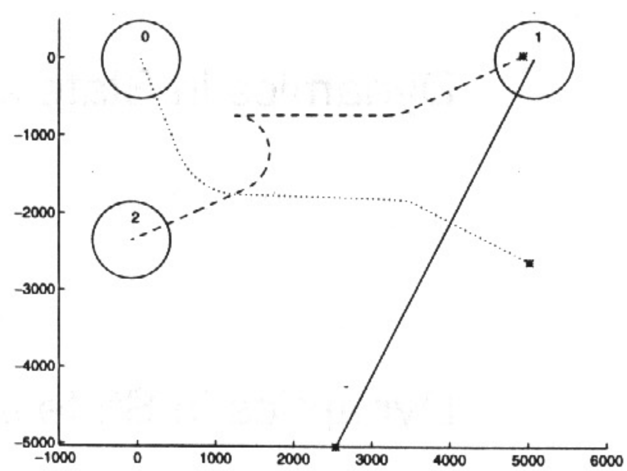
Headon



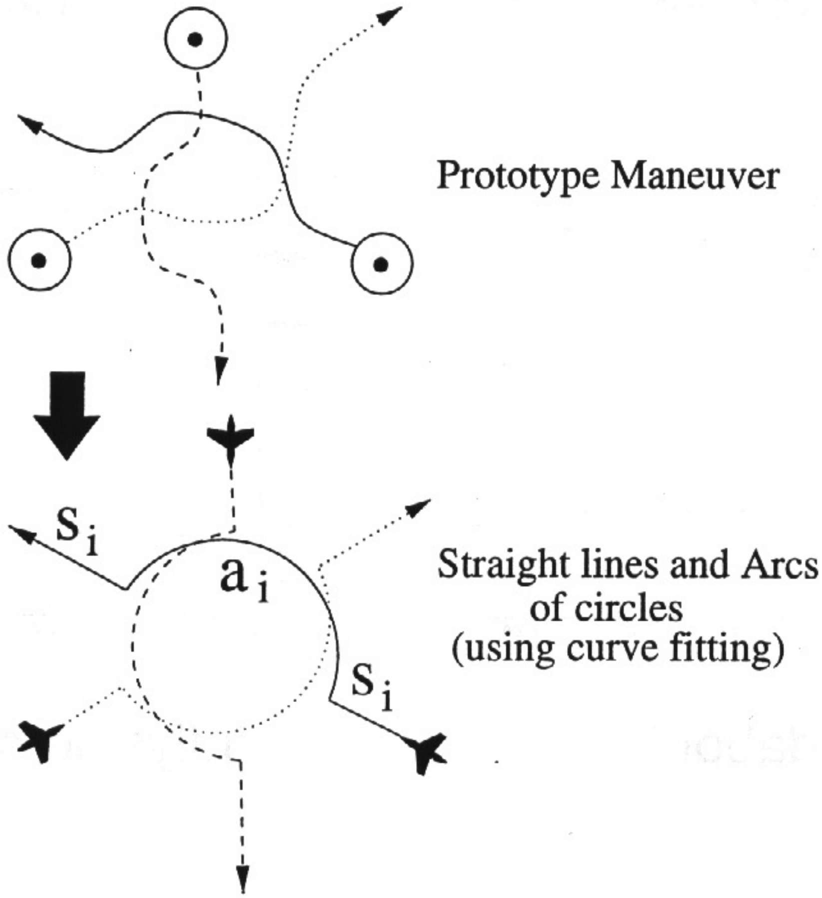
Roundabout



Unflyable maneuver



"Roundabout" Example



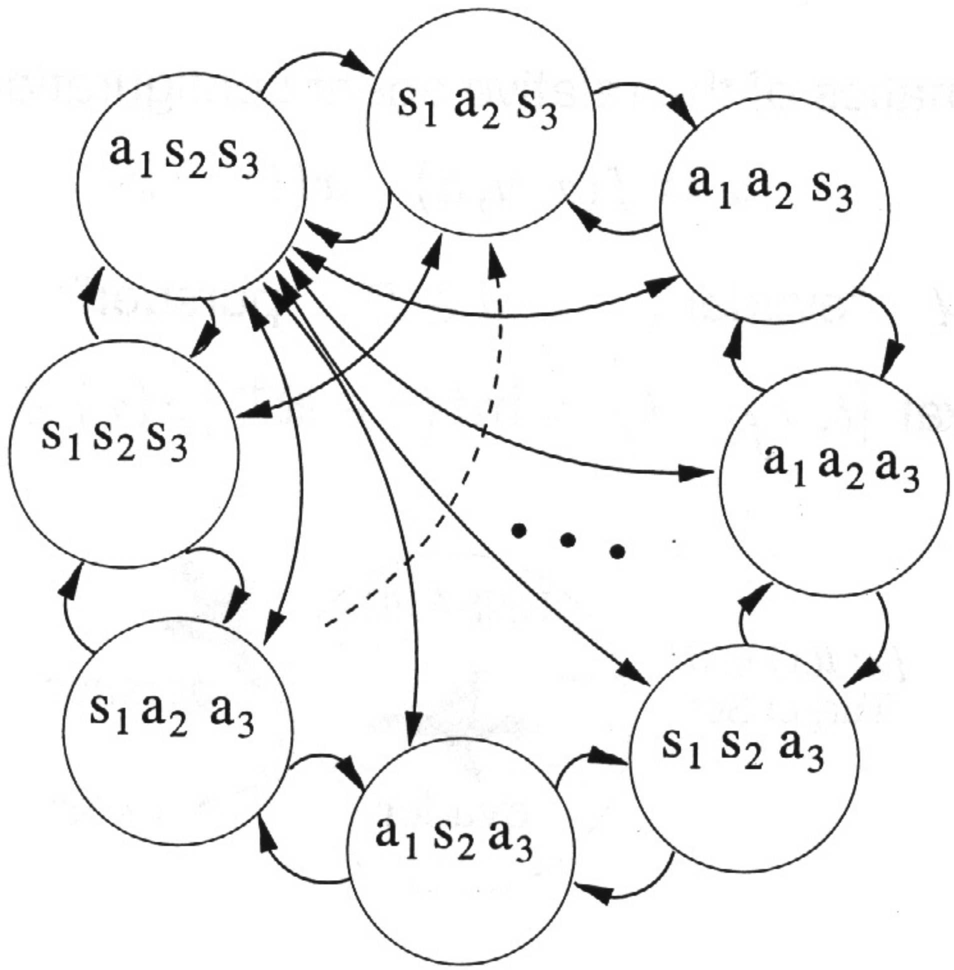
Dynamics in State s_i:

$$\begin{aligned}\dot{x}_i &= v_i \cos\phi_i \\ \dot{y}_i &= v_i \sin\phi_i \\ \dot{\phi}_i &= \omega_i \equiv 0\end{aligned}$$

Dynamics in State a_i:

$$\begin{aligned}\dot{x}_i &= v_i \cos\phi_i \\ \dot{y}_i &= v_i \sin\phi_i \\ \dot{\phi}_i &= \omega_i \neq 0\end{aligned}$$

Resulting Maneuver is a Hybrid System



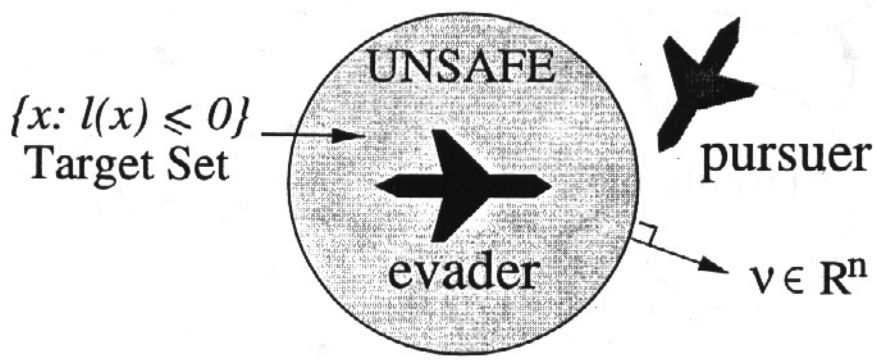
Controller Synthesis

Kinematics of the relative agent configuration:

$$\dot{x} = f(x, u, d) \quad x(t) = x$$

$u \in \mathcal{U}$ "evader"; $d \in \mathcal{D}$ "pursuer"

Interval $[t,\ t_f]$, $t_f = \inf\{\tau \in \Re^+ \mid x(\tau) \in T\}$



$$T = \{x \mid l(x) \leq 0\}$$
$$\partial T = \{x \mid l(x) = 0\}$$

$\nu = \frac{\partial l}{\partial x}(x(t_f))$ outward pointing normal

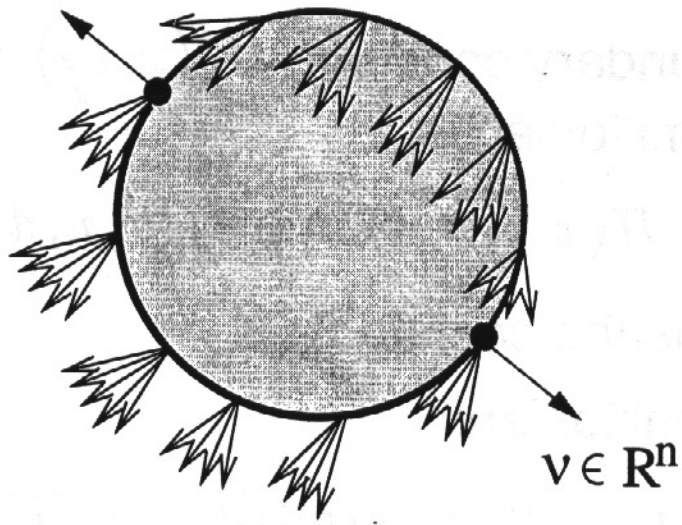
Variational problem without a running cost:

$$J_1(x, t, u, d) = l(x(t_f))$$

Calculating Pre(T)

$$\{x(t_f) : \exists u \forall d \quad \nu^T f(x(t_f), u, d) \geq 0\} \text{ Safe } \partial T$$

$$\{x(t_f) : \forall u \exists d \quad \nu^T f(x(t_f), u, d) < 0\} \text{ Unsafe } \partial T$$



Optimal control: $u^* = \arg\max_{u \in \mathcal{U}} J_1(x, t, u, d)$

Worst disturbance: $d^* = \arg\min_{d \in \mathcal{D}} J_1(x, t, u, d)$

Saddle Solution ...

$$\begin{aligned} J_1^*(x,t) &= \max_{u \in \mathcal{U}} \min_{d \in \mathcal{D}} J_1(x, t, u, d) \\ &= \min_{d \in \mathcal{D}} \max_{u \in \mathcal{U}} J_1(x, t, u, d) \end{aligned}$$

... is the optimal strategy for *each* player under the assumption that the other player plays its optimal strategy

Hamilton-Jacobi (Isaacs) Equation

If $J_1^*(x,t)$ is a smooth function of x and t:

$$\frac{\partial J_1^*(x,t)}{\partial t} = -H^*(x, \frac{\partial J_1^*(x,t)}{\partial x})$$

with the boundary condition $J_1^*(x,t_f) = l(x(t_f))$
and the Hamiltonian is:

$$H(x,p,u,d) = p^T f(x,u,d)$$

$p \in T^*\Re^n$ is the costate

Optimal Hamiltonian:

$$\begin{aligned} H^*(x,p) &= \max_{u \in U} \min_{d \in D} H(x,p,u,d) \\ &= H(x,p,u^*,d^*) \end{aligned}$$

Steady state solution: $J_1^*(x,-\infty)$

$$\begin{aligned} &\Rightarrow H^*(x, \frac{\partial J_1^*(x,-\infty)}{\partial x}) = 0 \\ &\Rightarrow \frac{\partial J_1^*(x,-\infty)}{\partial x} \perp f(x,u^*,d^*) \end{aligned}$$

Problem: Shocks, ie. discontinuities in J as a function of x

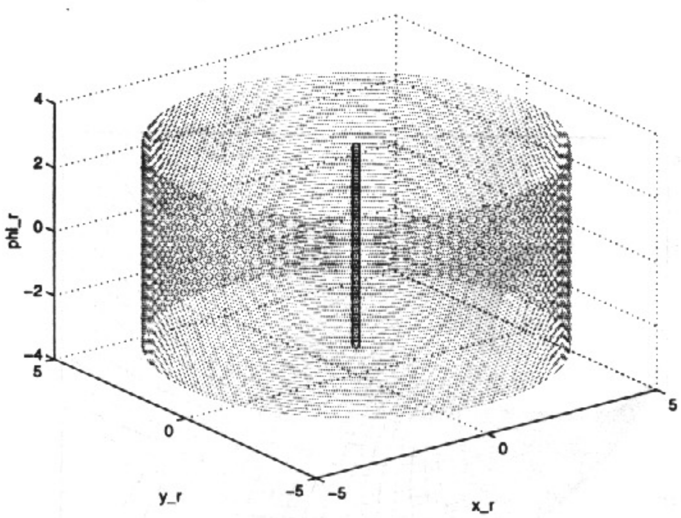
Example 1: Resolution by Angular Velocity

Model: $u = \omega_1$, $d = \omega_2$

$$\begin{aligned} \dot{x}_r &= -v_1 + v_2 \cos\phi_r + u y_r \\ \dot{y}_r &= v_2 \sin\phi_r - u x_r \\ \dot{\phi}_r &= d - u \end{aligned}$$

Target set:

$$T = \{(x_r, y_r) \in \Re^2, \phi_r \in [-\pi, \pi) \mid x_r^2 + y_r^2 \leq 5^2\}$$

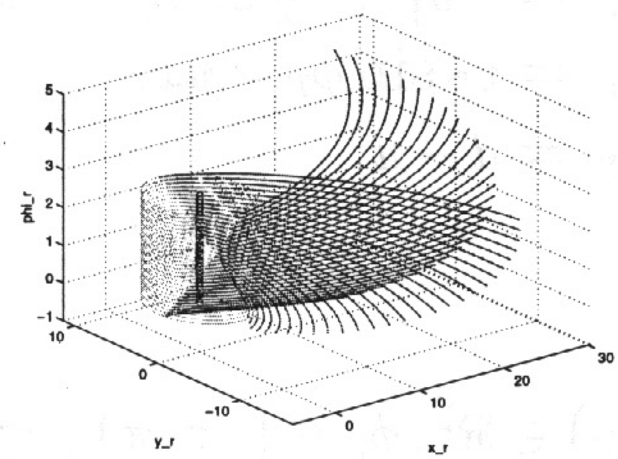


Cost:

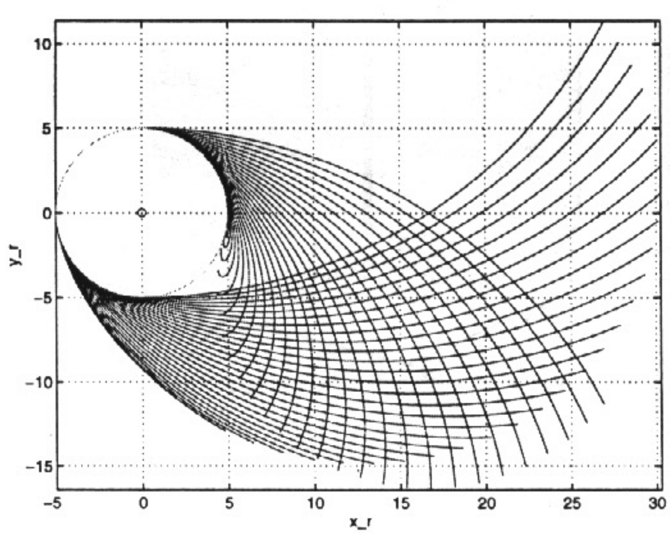
$$l(x) = x_r^2 + y_r^2 - 5^2$$

Determination of Pre(T)

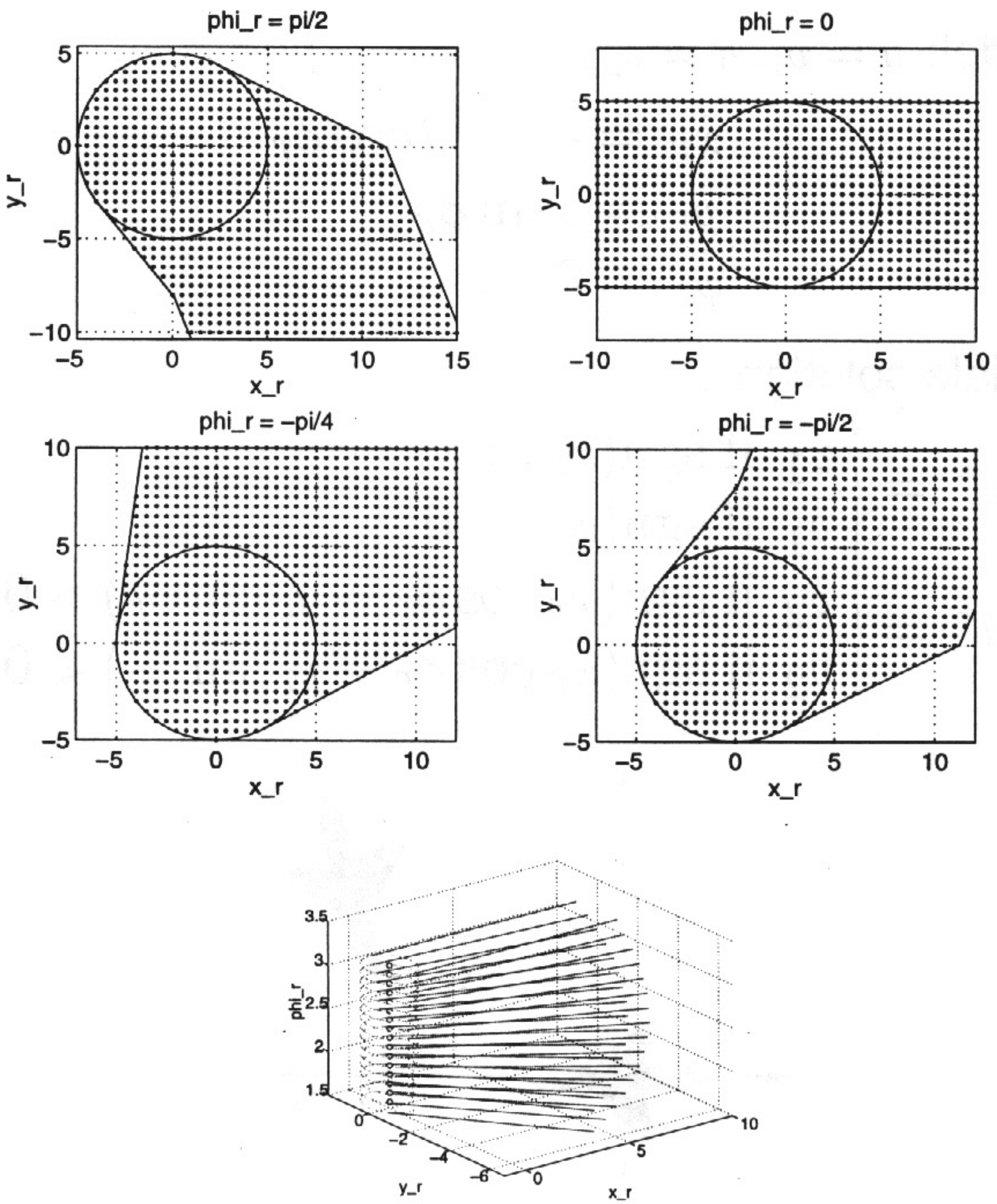
3D View:



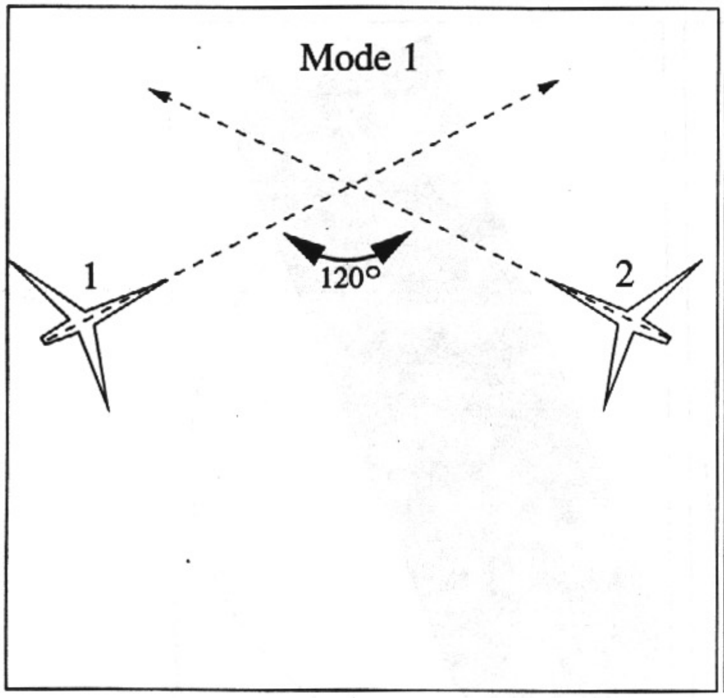
Top View:

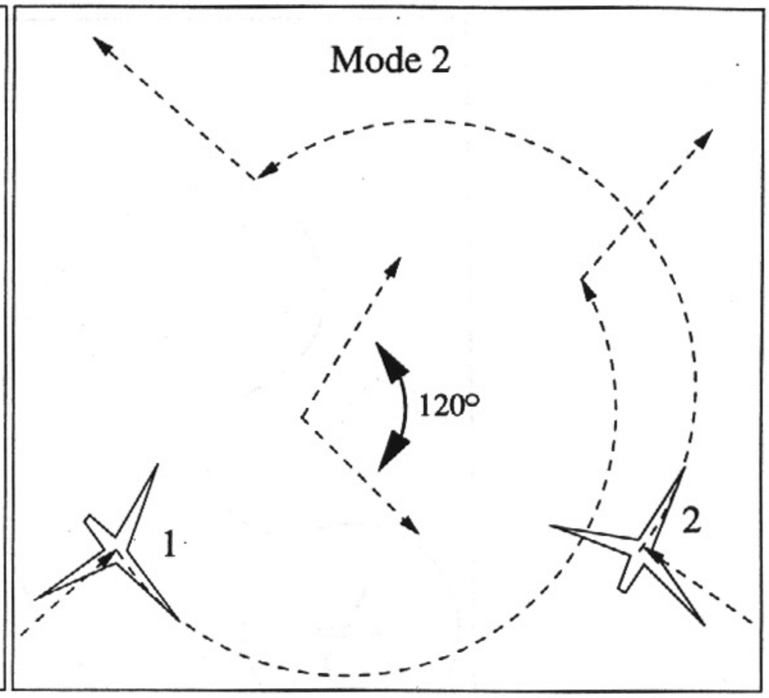


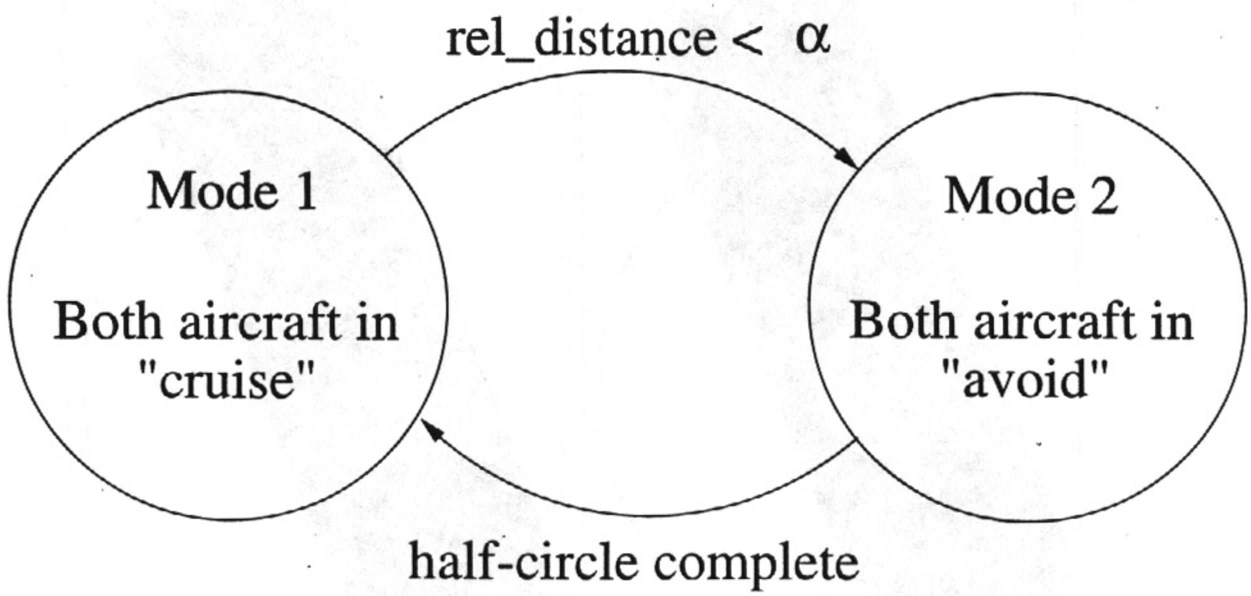
Determination of Pre(T)



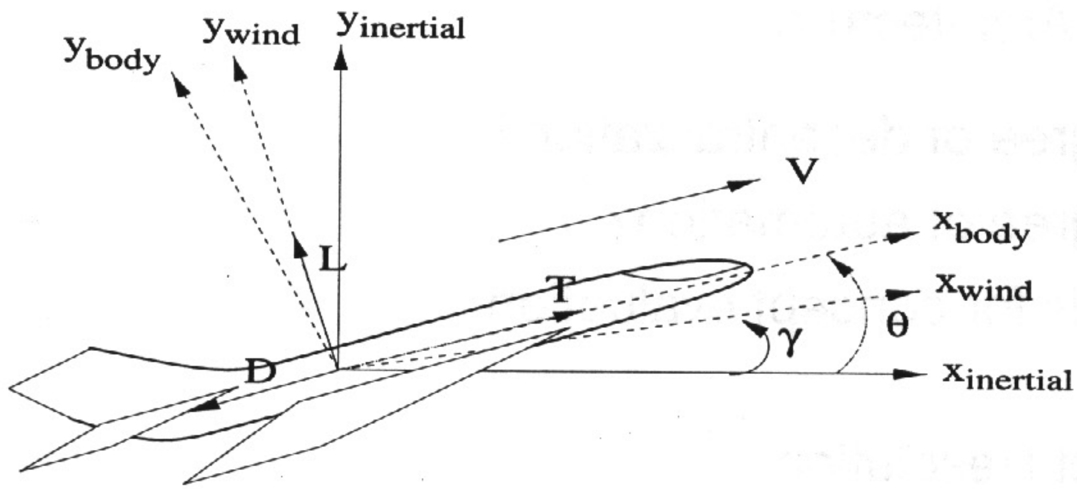
Synthesizing “Roundabout”







Flight Mode Switching



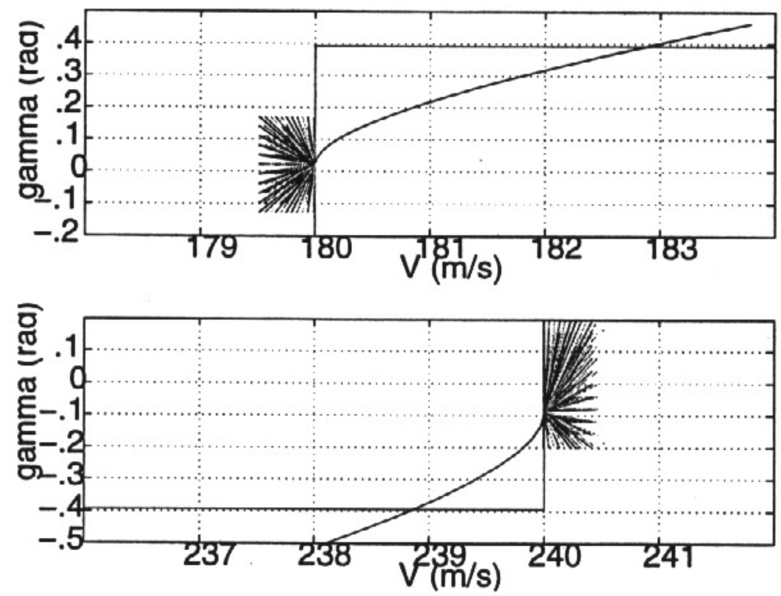
Flight modes: modes of aircraft operation

	Control Input	State	Output	
1.	(T, θ)	(V, γ)	(V, γ)	Airspeed Flight Path Angle
2.	θ $T = T_{min} \vee T_{max}$	(V, γ)	V	Airspeed
3.	θ $T = T_{min} \vee T_{max}$	(V, γ)	γ	Flight Path Angle

Design the least restrictive safe and efficient control scheme, such that the mode switching logic is implicitly defined.

Calculation of the Safe Sets of States

Cones of vector fields along the boundary:



The safe set of states: