A Formal Model Approach for the Analysis and Validation of the Cooperative Path Planning of a UAV Team

Antonios Tsourdos
Brian White, Rafał Żbikowski, Peter Silson
Suresh Jeyaraman and Madhavan Shanmugavel

Guidance & Control Group
Department of Aerospace, Power and Sensors

Challenges in multiple UAV Systems

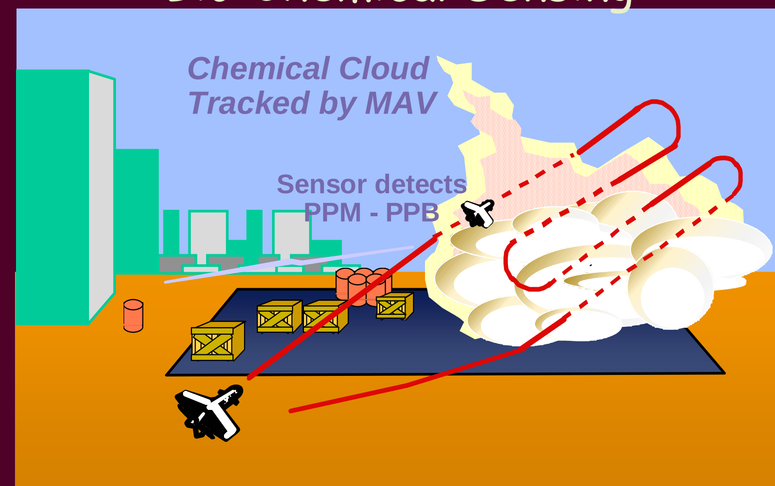
- Main driver is information
  - Timely
  - Accurate
  - Relevant
- Current focus on Autonomous Vehicles
  - Air vehicles
  - Ground vehicles
  - Underwater vehicles
- Homogeneous or Heterogeneous combinations

UAV Missions

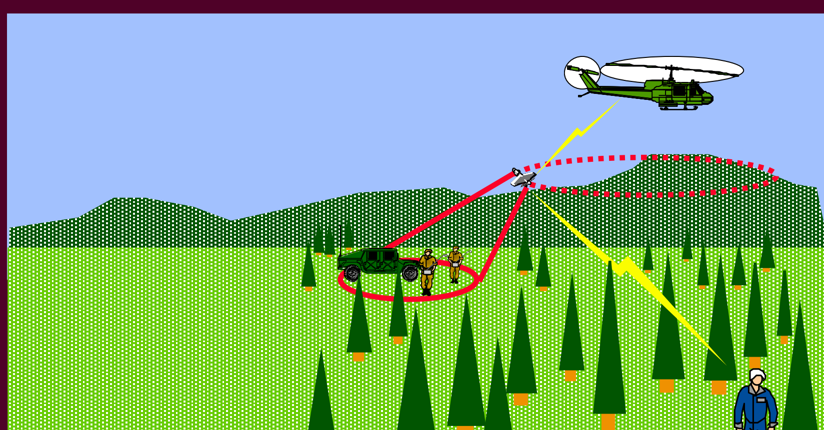
Cranfield UNIVERSITY

Cave Search

Bio-Chemical Sensing

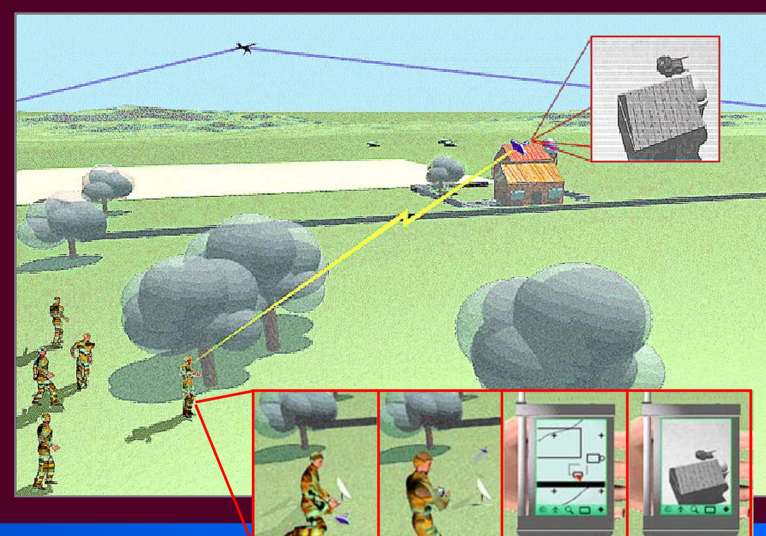


Rescue Missions



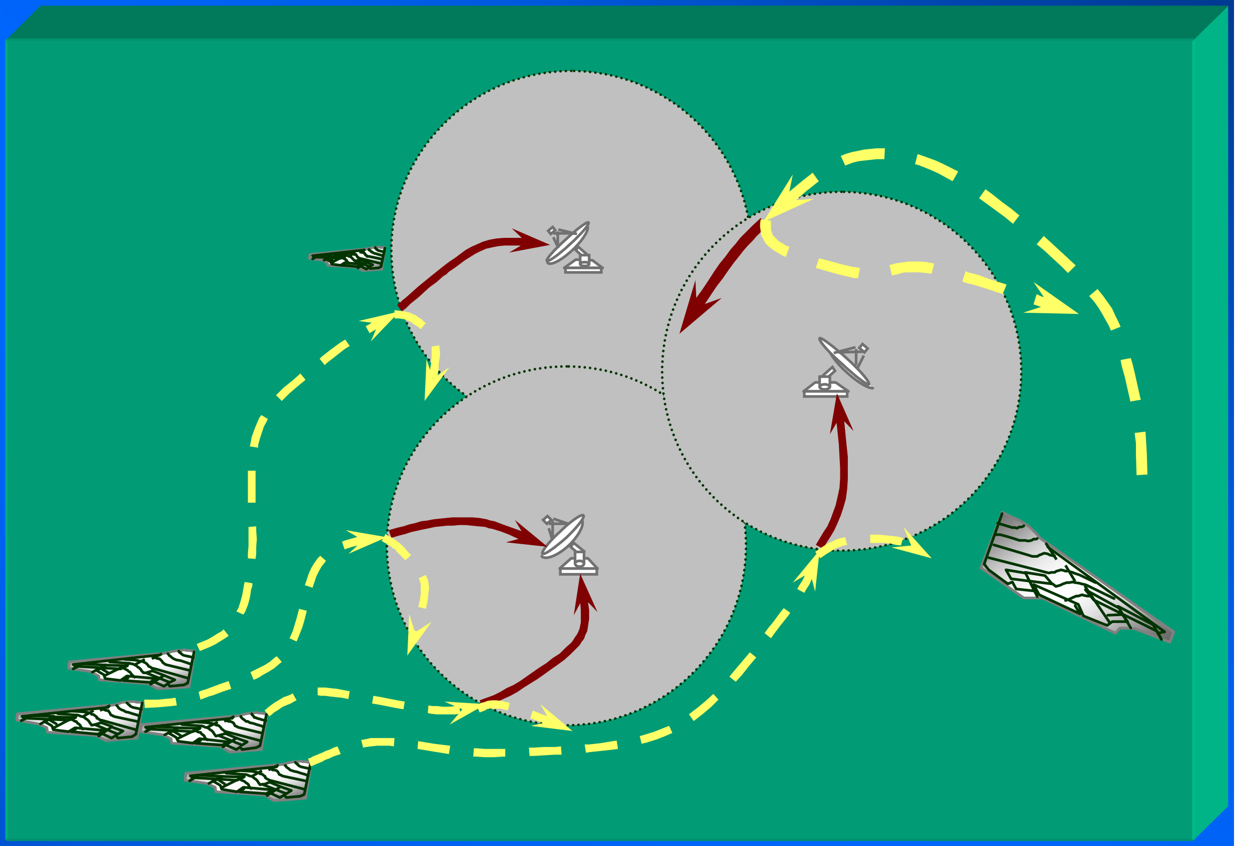
MAV provides situational awareness,
Provides beacon for rescue operations.

"Over-the-hill" Reconnaissance



UAV Cooperative Control Research

Objective
Develop new control theories to enable UAVs to cooperate autonomously



Technical Challenges

- Coupling
- Uncertainty
- Partial information

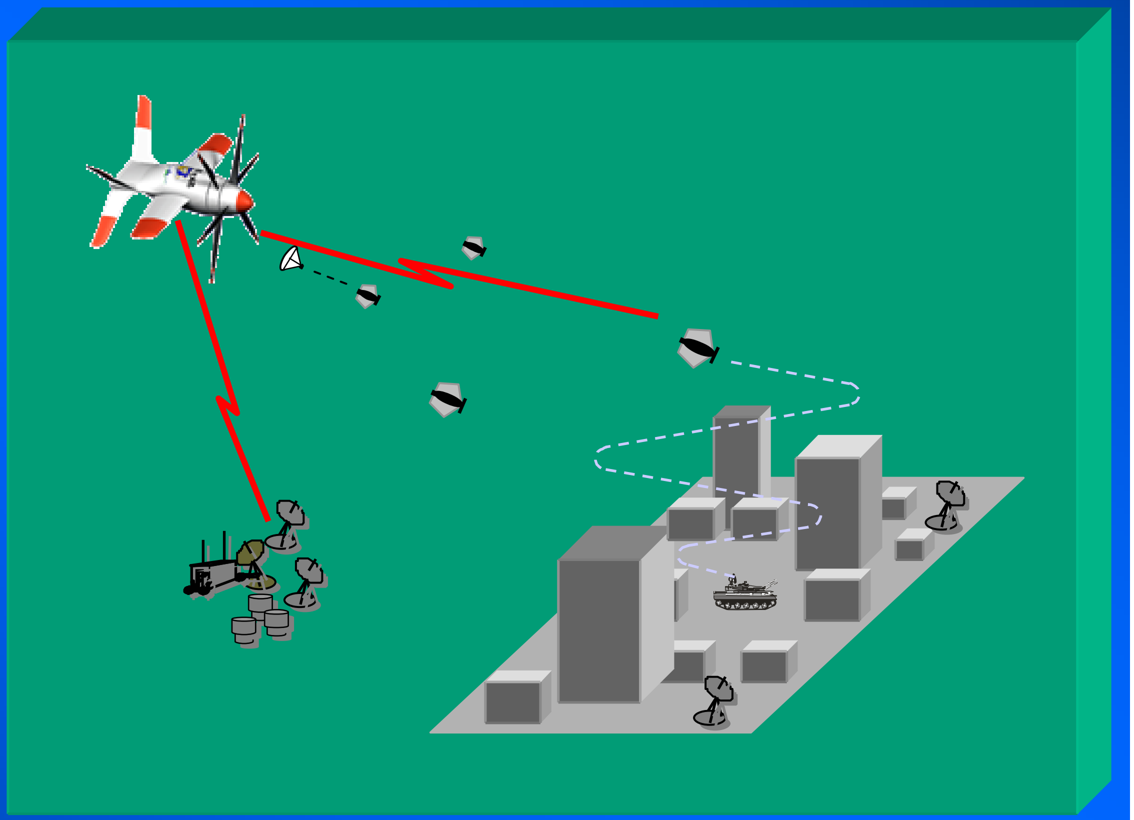
Approach

- Online re-planning and trajectory generation (Differential Geometry)
- Hierarchical multi-agent coordination architecture (Kripke Model)

Cooperative Operations in Urban Terrain

Goal

release micro vehicles from small surveillance UAV for positive target ID and tagging in urban terrain.

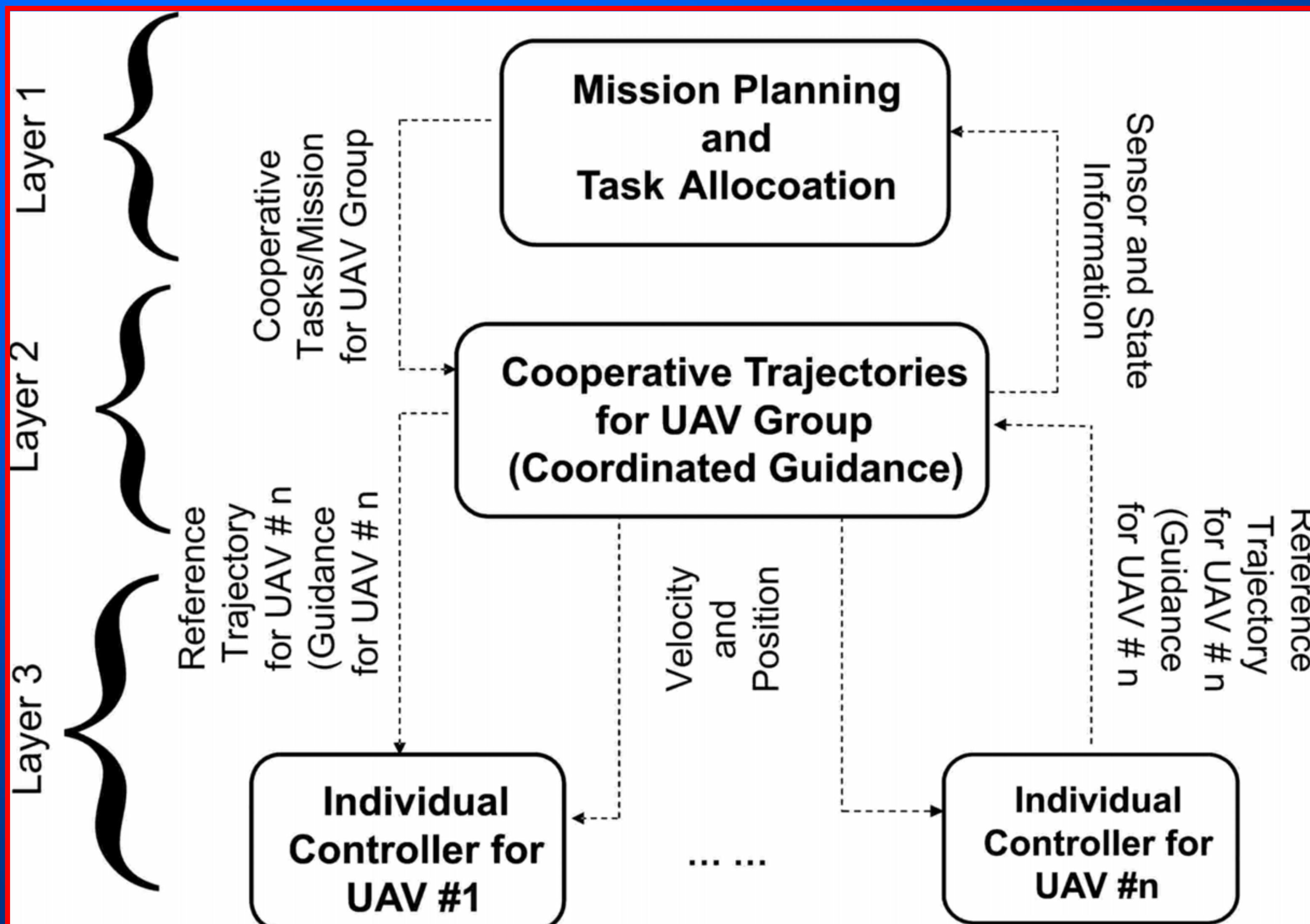


Issues:

- release micro vehicles
- cooperative search
- flight in congested environment
- no micro - micro comms
- limited information
- sensor integration by small vehicle
- presentation of information to operator

Cranfield UNIVERSITY

Hierarchy Levels of a UAV mission



Trajectory Shaping

and Cooperative Guidance

Cranfield UNIVERSITY

Cranfield UNIVERSITY

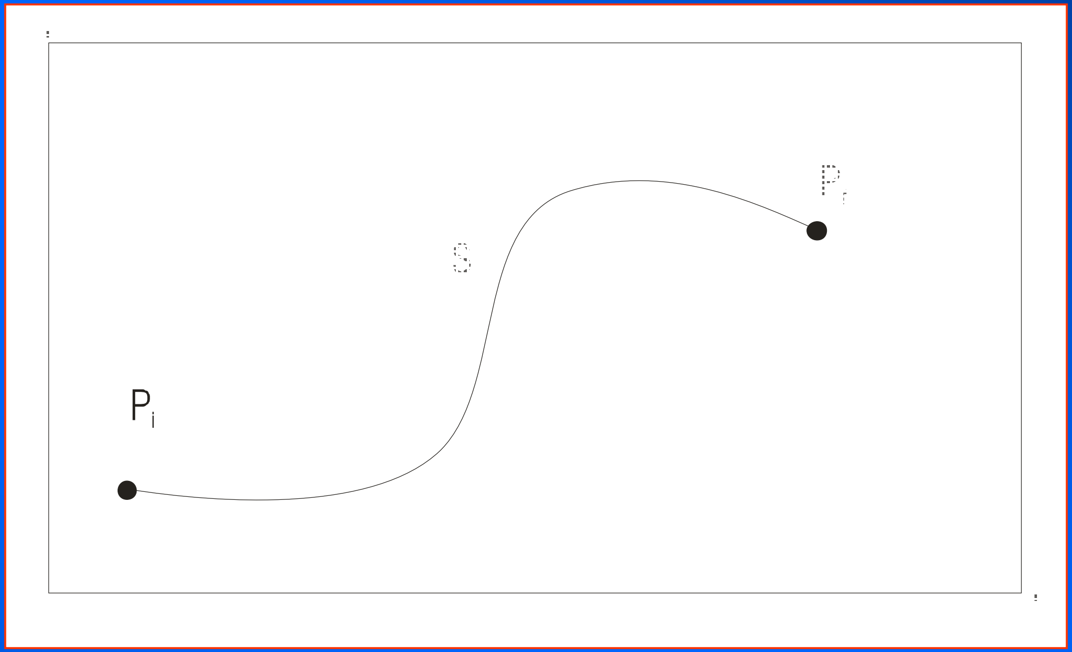
Trajectory Shaping

- Given initial Pose

$$P_i(x, y, z, q)$$

- Given final Pose

$$P_f(x, y, z, q)$$



- Find a smooth continuous path between them

Cranfield UNIVERSITY

Trajectory Shaping

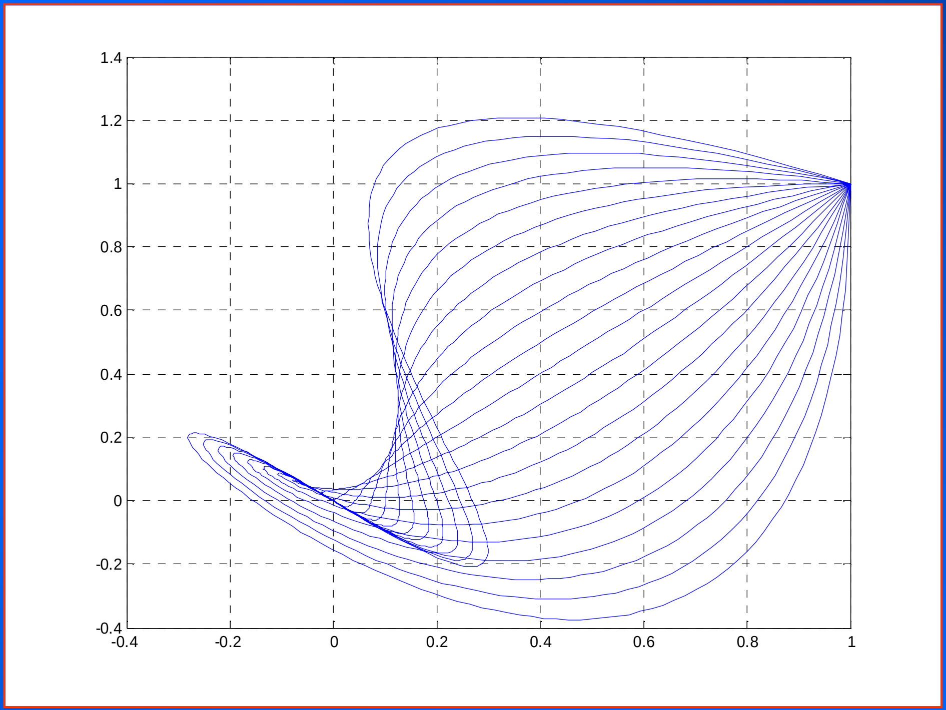
- Polynomial

$$P(s)=\sum_{i=1}^{n} a_i s^i$$

- Orthogonal Bases

$$P(s)=\sum_{i=1}^{3} \alpha_i b_i(s)$$

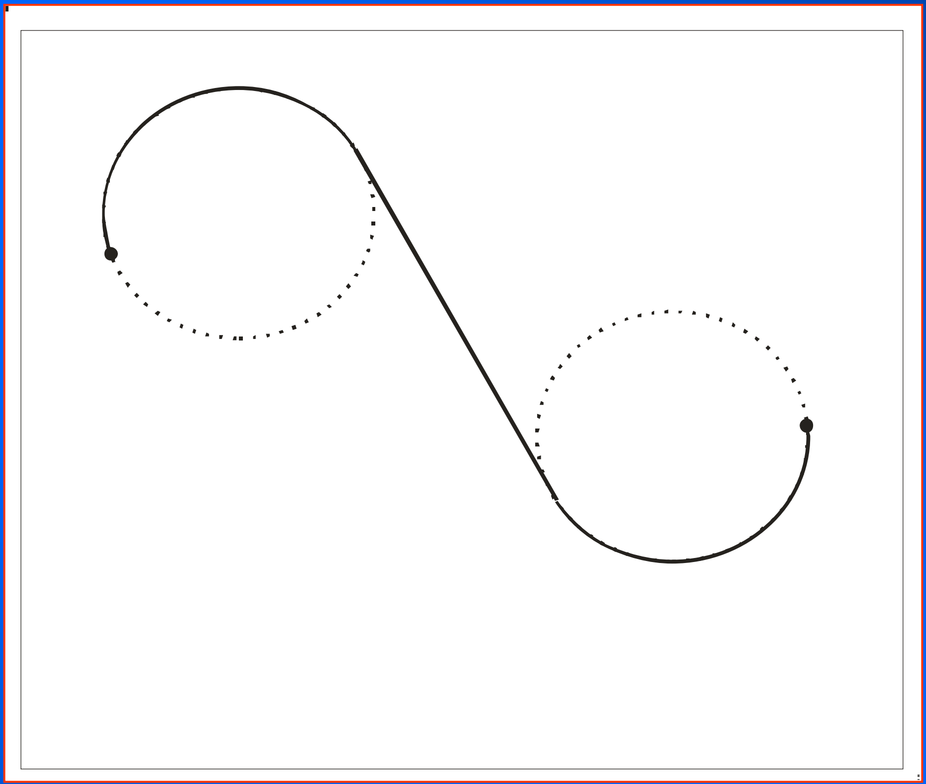
- Bezier Bases
- Hermite Bases



Cranfield UNIVERSITY

Trajectory Shaping

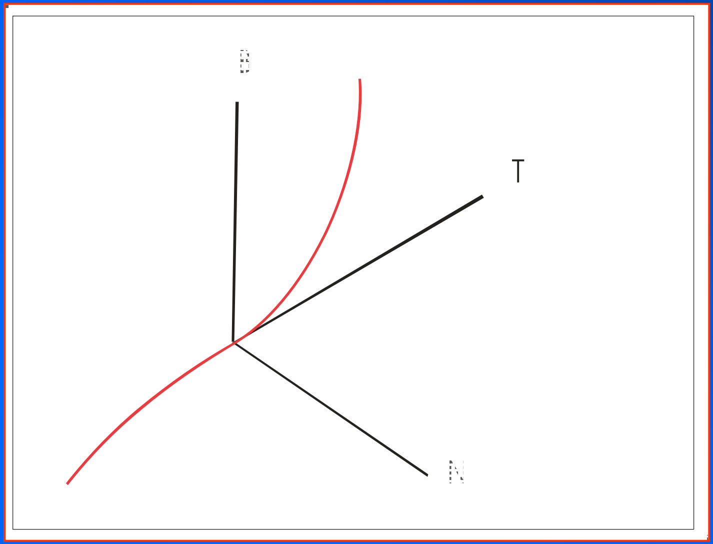
- Dubins Sets
  - Combines circles and lines
- Extend
  - Basic: 2 lines + circle
  - Module: 1 line + circle
- Control
  - Initial pose
  - Final pose
  - Path length
  - Path topology



Cranfield UNIVERSITY

Trajectory Shaping

- Differential Geometry
- Frenet Frame
  - Tangent vector T
  - Normal vector N
  - Binormal vector B
- Frenet Parameters
  - Curvature κ
  - Torsion τ

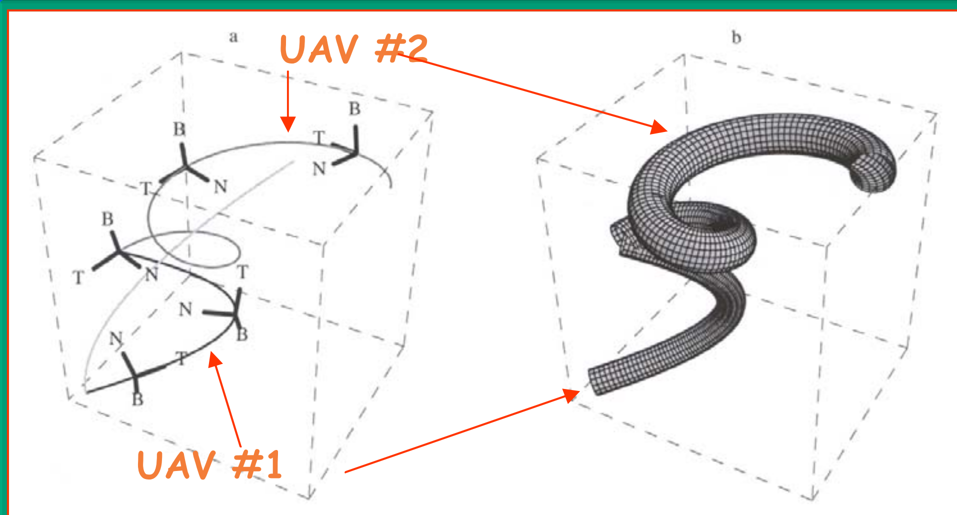


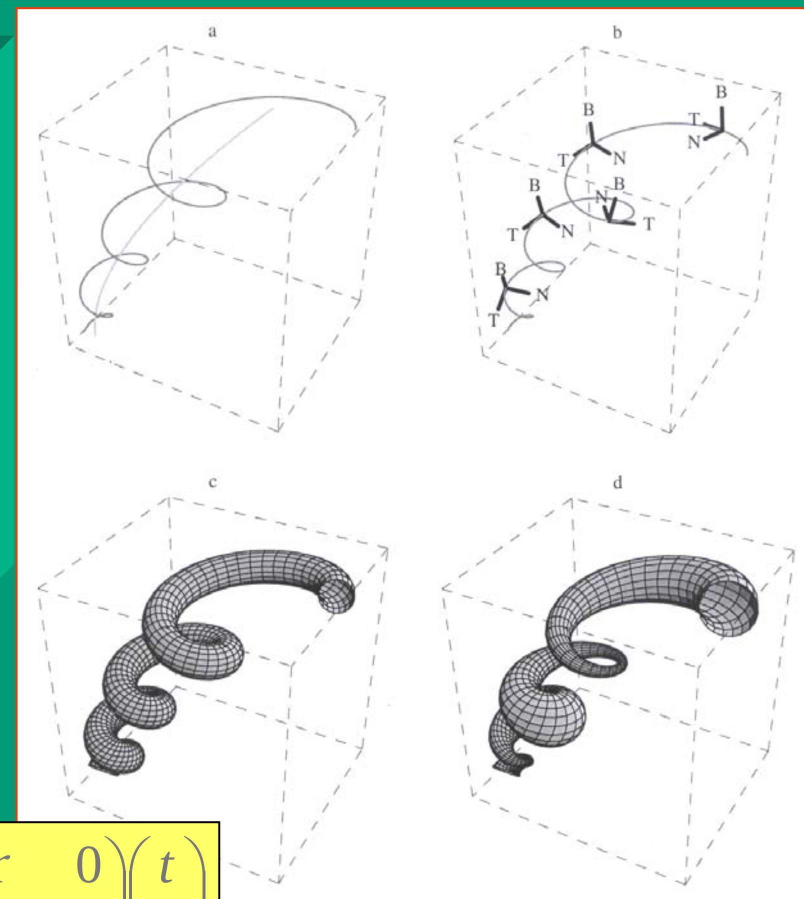
$$\kappa(s) = P_\kappa(s)$$
$$\tau(s) = P_\tau(s)$$

$$\begin{pmatrix} \dot{t} \\ \dot{n} \\ \dot{b} \end{pmatrix} = \begin{pmatrix} 0 & \kappa & 0 \\ -\kappa & 0 & \tau \\ 0 & -\tau & 0 \end{pmatrix} \begin{pmatrix} t \\ n \\ b \end{pmatrix}$$

Cranfield UNIVERSITY

Differential Geometric Guidance



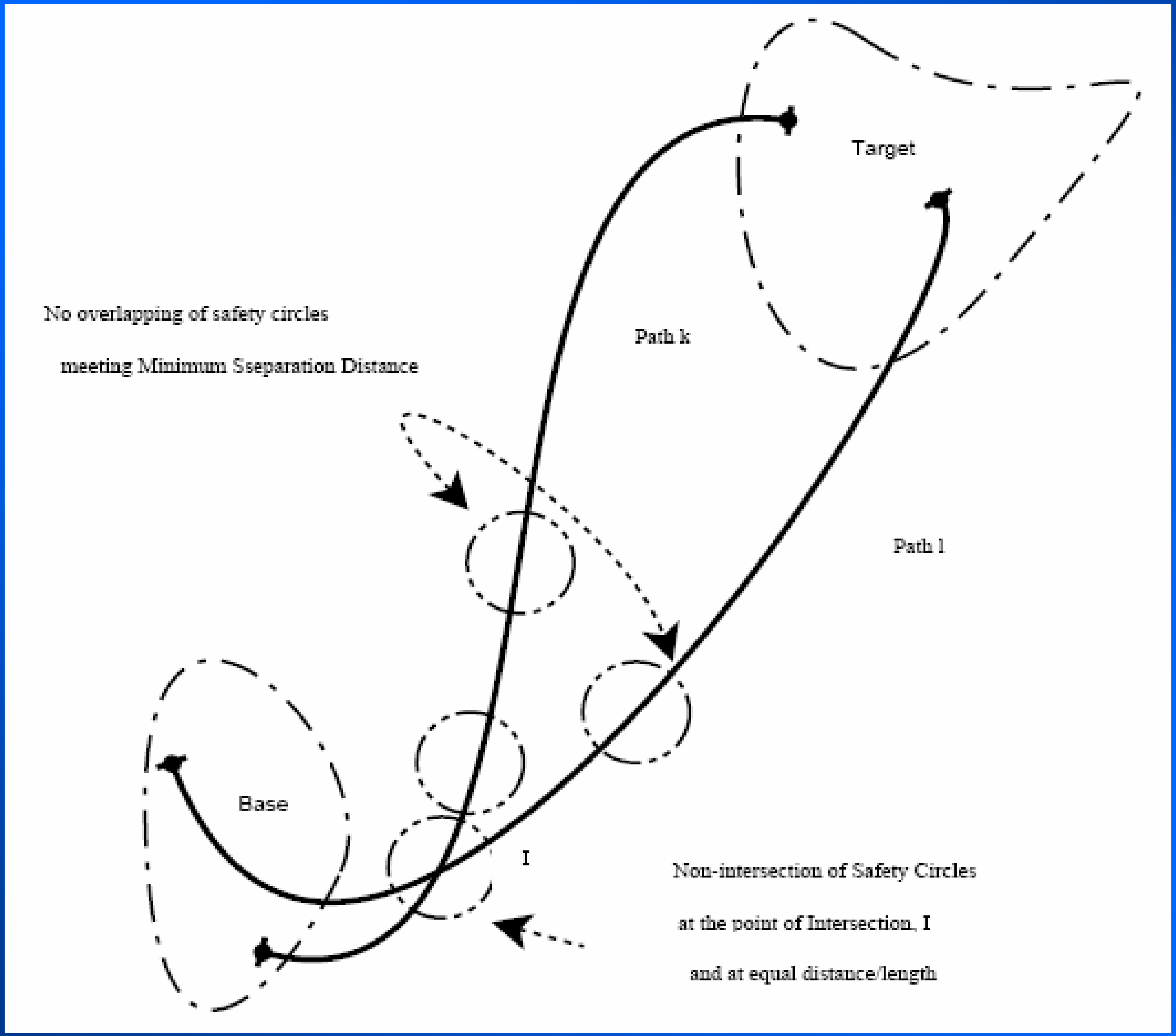


- Frenet Frame
  - Tangent vector T
  - Normal vector N
  - Binormal vector B

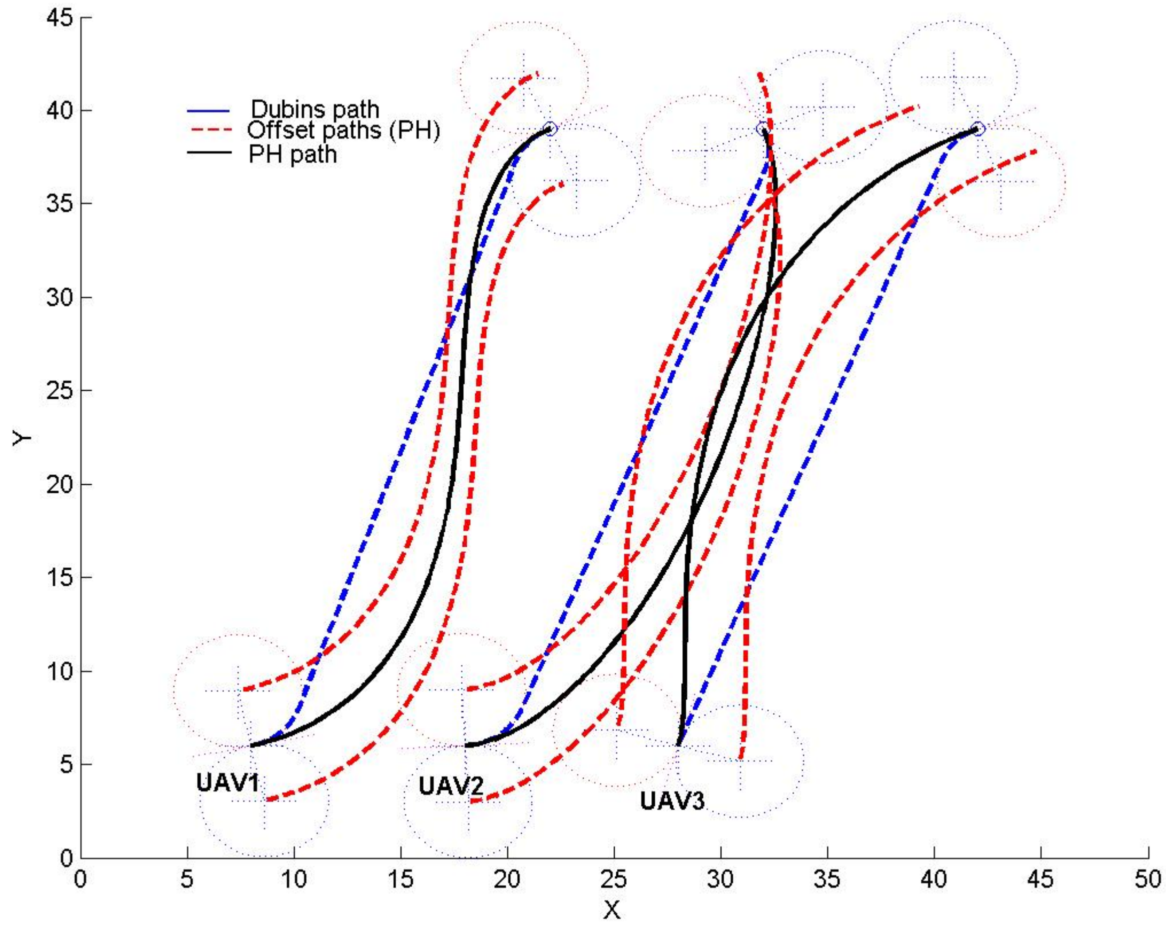
$$\begin{pmatrix} \dot{t} \\ \dot{n} \\ \dot{b} \end{pmatrix} = \begin{pmatrix} 0 & \kappa & 0 \\ -\kappa & 0 & \tau \\ 0 & -\tau & 0 \end{pmatrix} \begin{pmatrix} t \\ n \\ b \end{pmatrix}$$

- Tubes
- Canal surfaces

Safe Flight Path

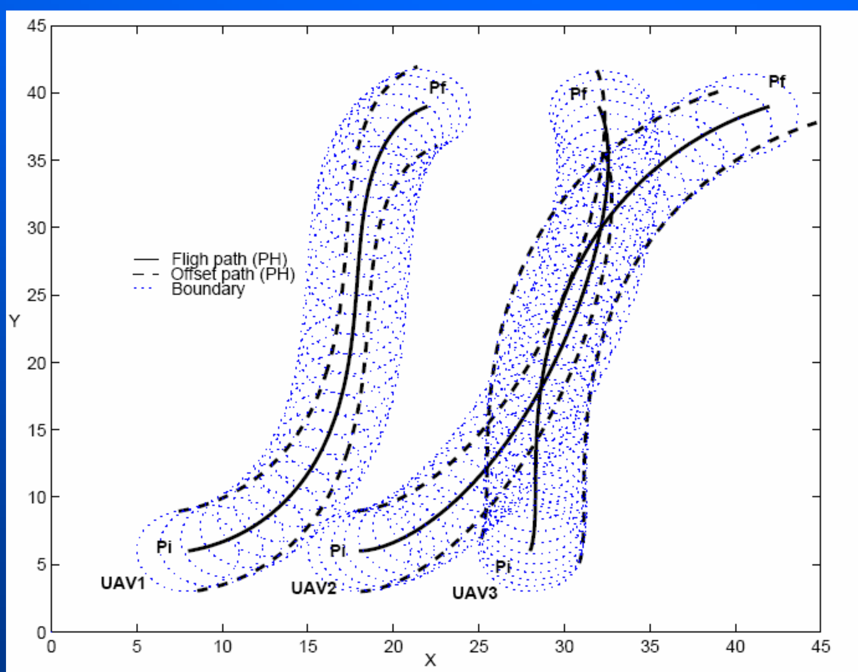


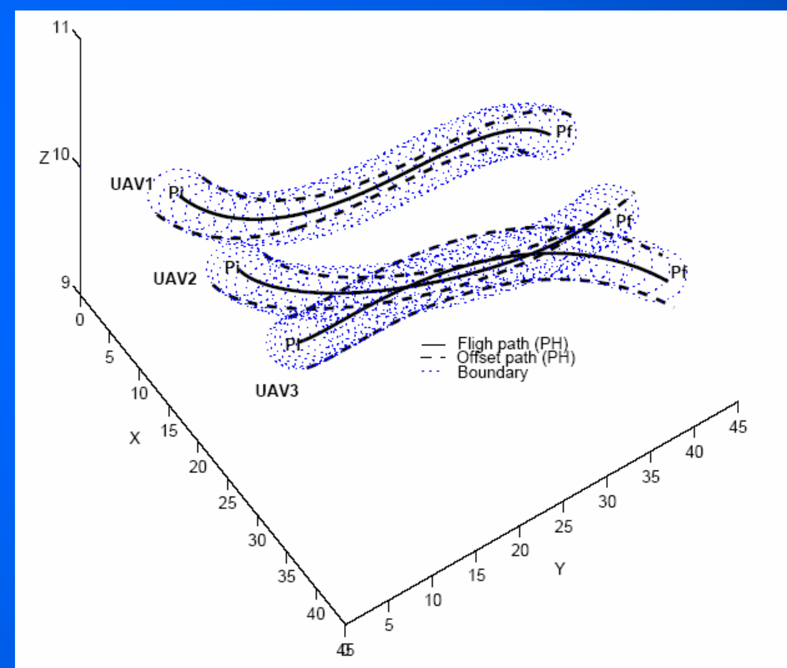
Approximate Dubins Paths



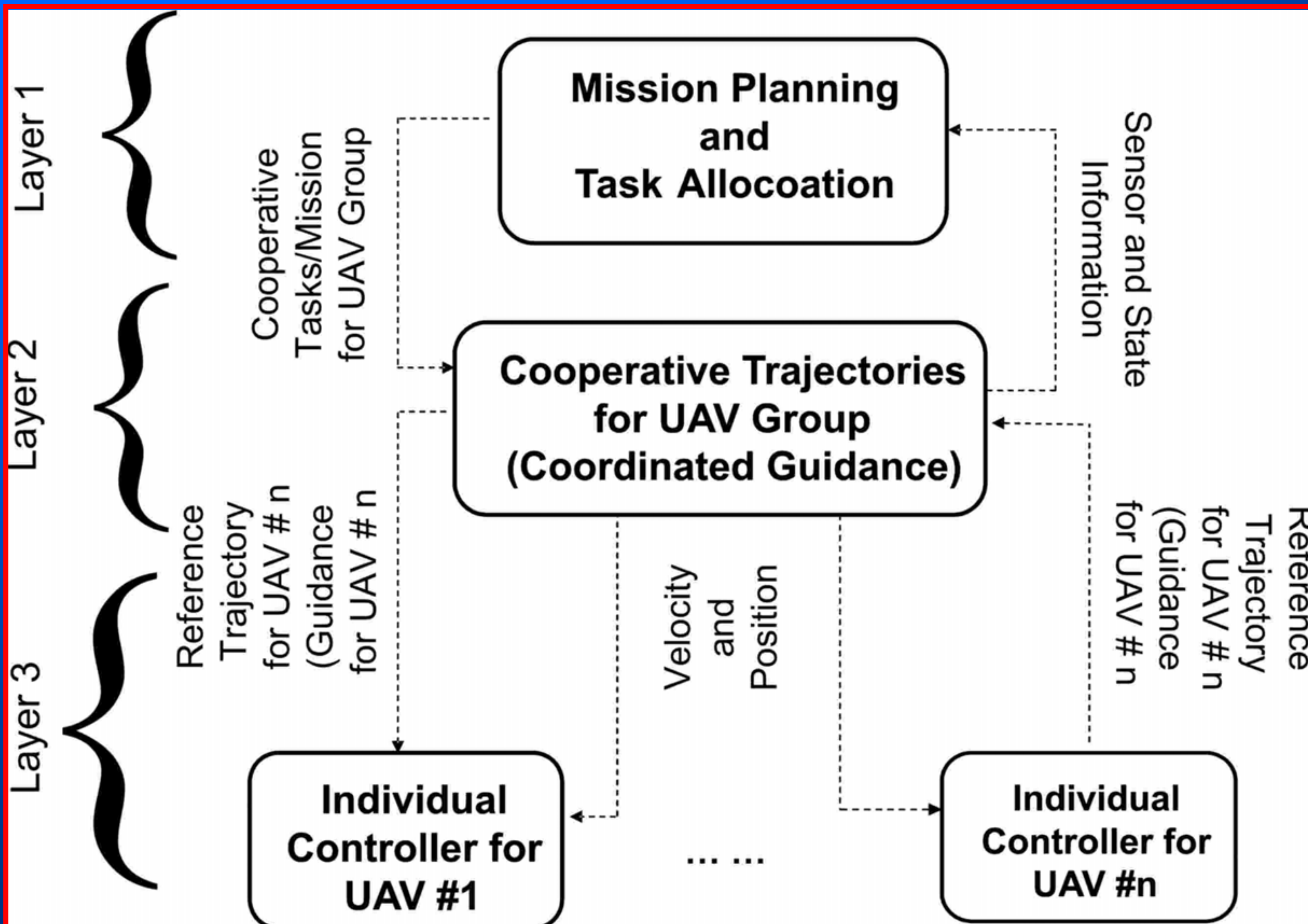
Cranfield UNIVERSITY

Approximate Dubin's Paths with Uncertainty





Hierarchy Levels of a UAV mission



Cranfield UNIVERSITY

Strategy

for

Mission Planning and Task Allocation

What is a swarm?

- Swarm of UAVs
  - a group (more than two)
  - flying together (not necessarily in formation)
  - heterogenous (same airframe, different sensors/paylods)
- Platform chracteristics
  - low cost
  - GPS-capable
  - air-breathing

What is swarm intelligence?

Swarm intelligence

is

limited

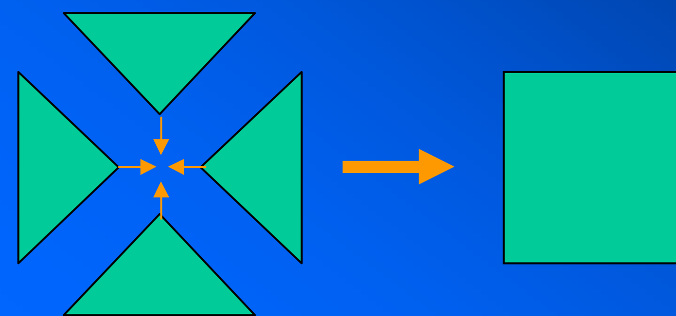
sensing, communication, decision and action

autonomy of a group of UAVs.

Cranfield UNIVERSITY

What is emergent property?

- Emergent property
  - group has it
  - group members have it not

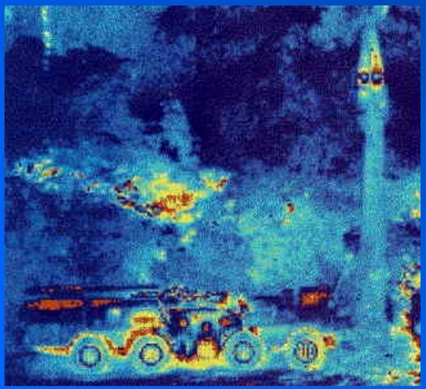


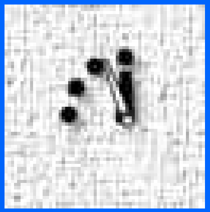
- Data fusion and decision capability
  - multi-spectral multi-sensor: combined seekers
  - distributed computing: networked on-board computers

Cranfield UNIVERSITY

Intelligence for UAV swarms

- Requirements:
  - real-time safety-critical operation
  - autonomous/remote operator override
  - flight dynamics
  - finite computational/storage resources
  - finite bandwidth communications
  - limited capability sensors
- Mathematical problems:
  - continuous dynamics
  - logic
  - discrete events





Tractability

Cranfield UNIVERSITY

Temporal logic: linear time

FUTURE

$\Box\phi$ means: ϕ will always be true

$\Diamond\phi$ means: ϕ will eventually be true

$\bigcirc\phi$ means: ϕ will be true at the next step

$\phi \mathrm{U} \psi$ means: ϕ will be true until ψ

j	0	1	2	3	4	5	...
x	4	5	3	7	8	9	...
$x > 3$	T	T	F	T	T	T	...
$(x > 3)$	F	F	F	T	T	T	...

PAST

$\blacksquare\phi$ means: ϕ has always been true

$\blacklozenge\phi$ means: ϕ was once true

$\bullet\phi$ means: ϕ was true at the previous step

$\phi \mathrm{S} \psi$ means: ϕ has been true since ψ

j	0	1	2	3	4	5	...
x	1	2	3	4	5	6	...
$x \leqslant 5$	T	T	T	T	T	F	...
$x = 3$	F	F	T	F	F	F	...
$(x \leqslant 5)\mathrm{S}(x = 3)$	F	F	T	T	T	F	...

Cranfield UNIVERSITY

Modal logic: syntax and semantics

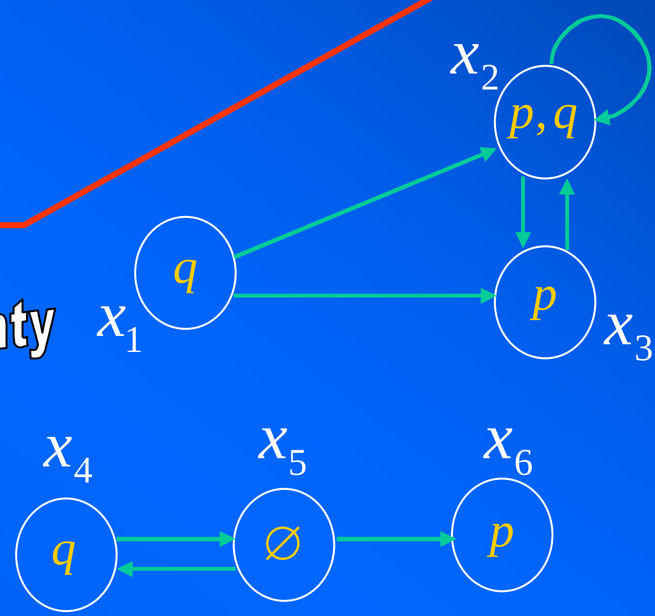
$$\phi ::= \bot \mid \top \mid p \mid \neg\phi \mid (\phi\wedge\phi) \mid (\phi\vee\phi) \mid (\phi\rightarrow\phi) \mid (\phi\leftrightarrow\phi) \mid \Box\phi \mid \Diamond\phi$$

Syntax of modal logic formulae (Backus Naur form)

- p - atomic formula
- ϕ - formula
- $\Box\phi$ - it is necessary that ϕ
- $\Diamond\phi$ - it is possible that ϕ

Semantics of modal logic formulae (Kripke models)

Reasoning about uncertainty



Kripke model (W, R, L) of basic modal logic:

1) Universe W of possible worlds $W = \{x_1, \ldots, x_6\}$
2) Accessibility relation R between worlds
3) Worlds' labelling function L

$R(x_1,x_2)$, $R(x_1,x_3)$, $R(x_2,x_2)$, $R(x_2,x_3)$
$R(x_3,x_2)$, $R(x_4,x_5)$, $R(x_5,x_4)$, $R(x_5,x_6)$

x_1	x_2	x_3	x_4	x_5	x_6
$\{q\}$	$\{p,q\}$	$\{p\}$	$\{q\}$	$\varnothing$	$\{p\}$

Cranfield UNIVERSITY

Research Method

Aims

- Formalised model of
  - the UAV group
  - system behaviour
- Model checking
- Simulation

Means

- Kripke Model of "possible worlds"
- Temporal logic
- SPIN model checker
- ANSI-C module

Result

Model checking results will proof-check system's behaviour as well as failings

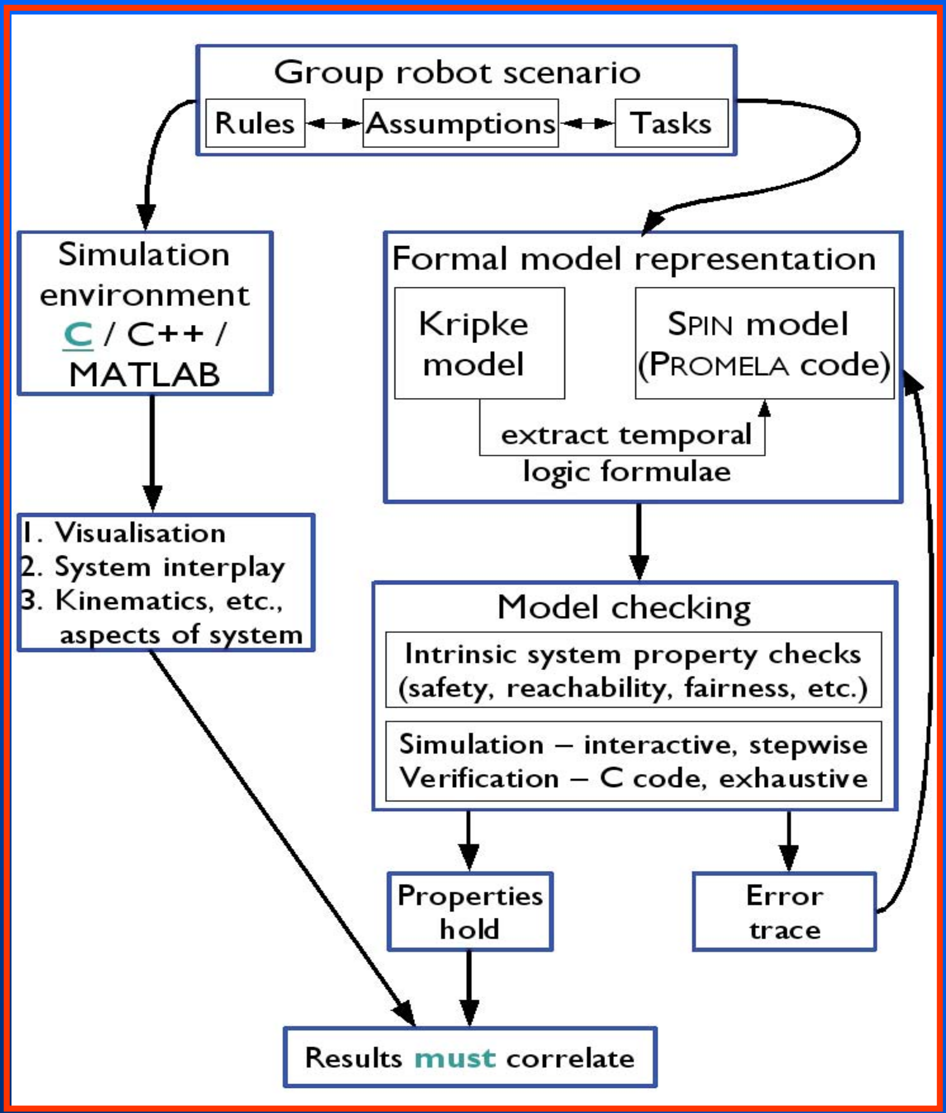
Model Checking

Cranfield UNIVERSITY

Property Definition	Specification Formula: LTL
reachability property — some particular property *can be reached*	Not Suitable: Expresses reachability negatively – nested reachability impossible
safety property — under certain conditions, something *never occurs*	$\Box\neg(\phi_1 \wedge \phi_2)$
liveness property — under certain conditions, something *will ultimately occur*	$\Box(\phi_1 \rightarrow \Diamond\phi_2)$
fairness property — under certain conditions, something will (or will not) occur *infinitely often*	possible using ω-automata

- Model checking – automated, exhaustive procedure, and always gives yes/no answers to system behaviour queries
- Common system critical properties are categorised as reachability, safety, liveness and fairness.
- The formal model must be an accurate replica of the actual scenario, as verification formulae are extracted from the model as shown

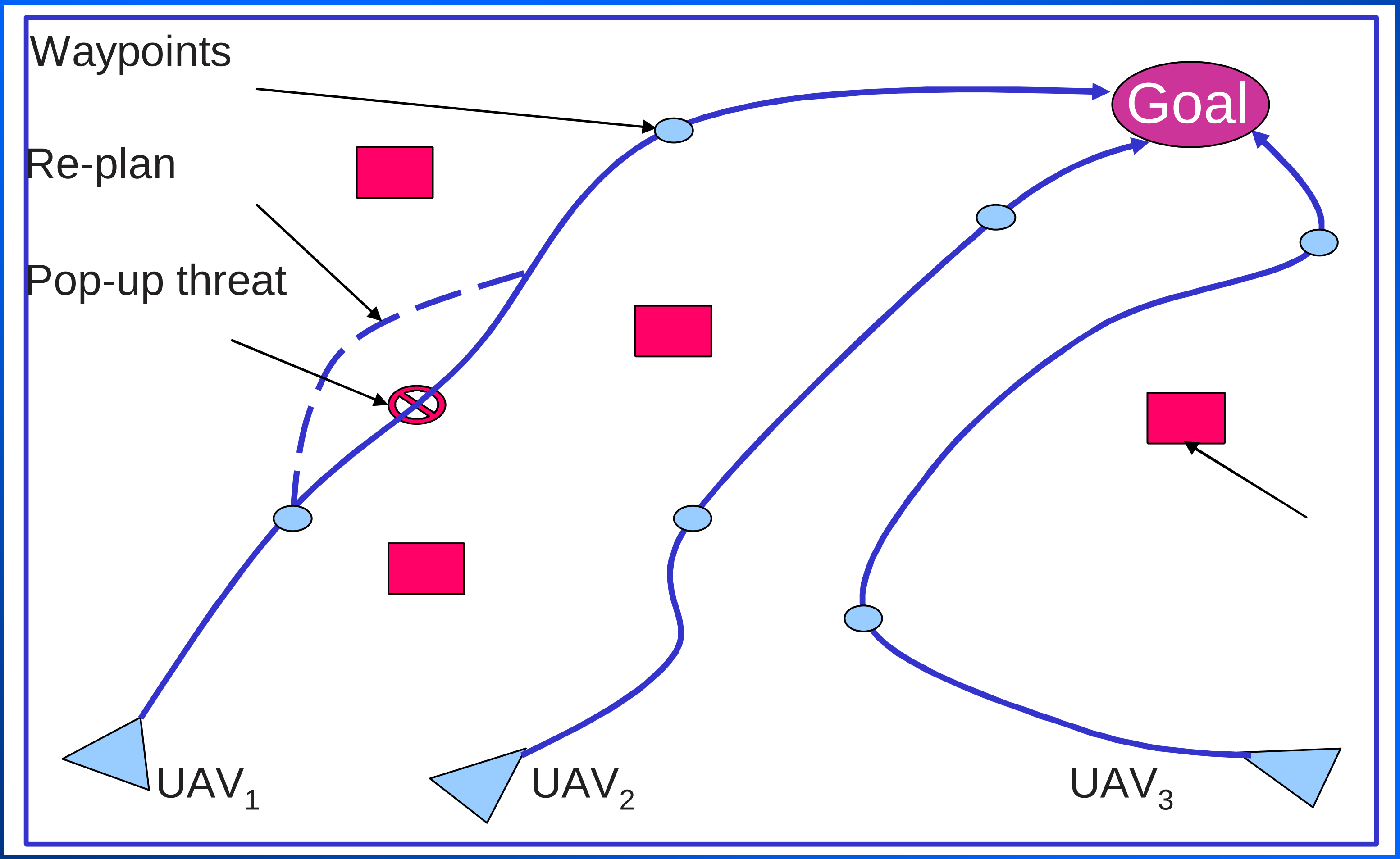
Model Checking



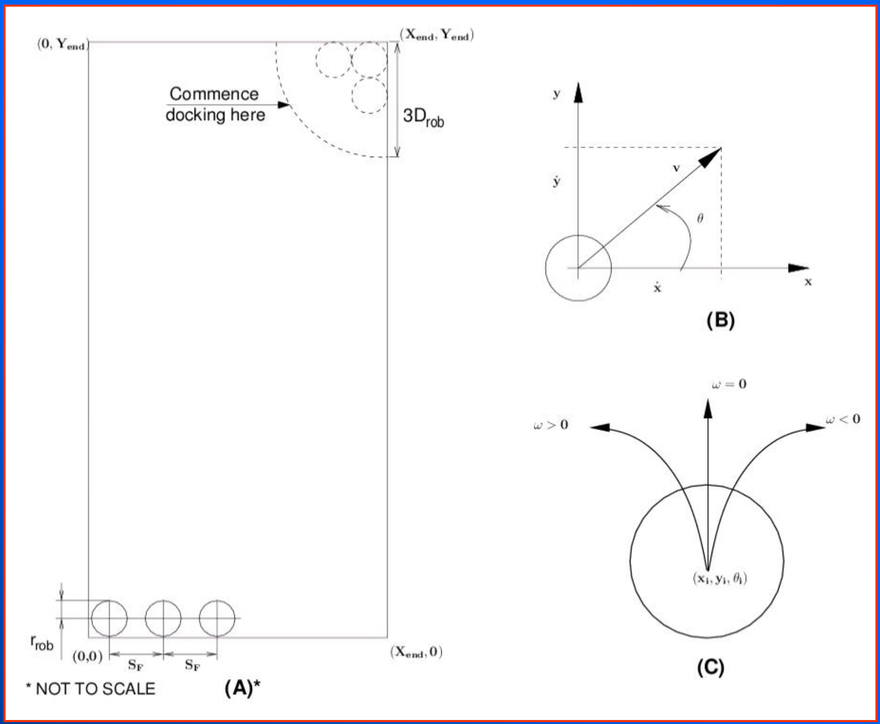
- Uses PROMELA for specifying verification model
- SPIN can be used in
  - Simulation runs
  - Verification runs
- Model specific verifier in ANSI-C – fast & fine tuneable execution
- Model generation is now automatic

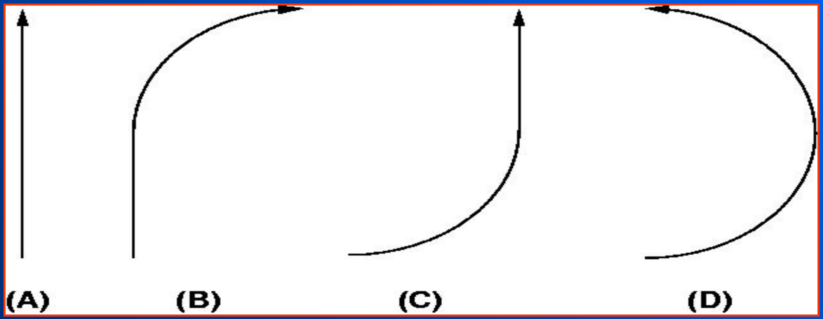
Cranfield UNIVERSITY

General Scenario



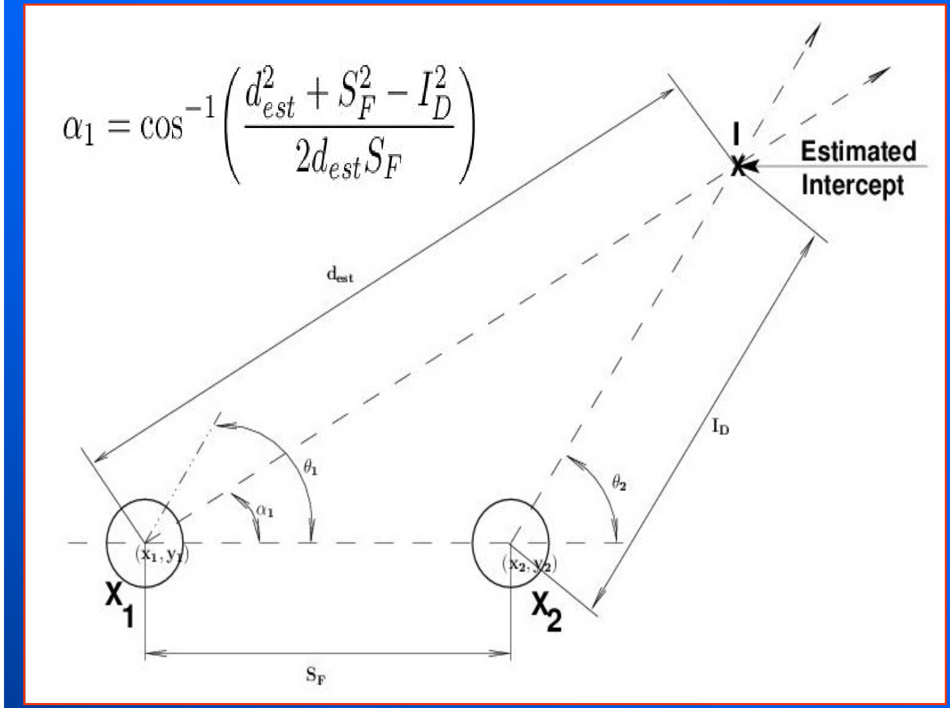
Scenario – Framework & Assumptions

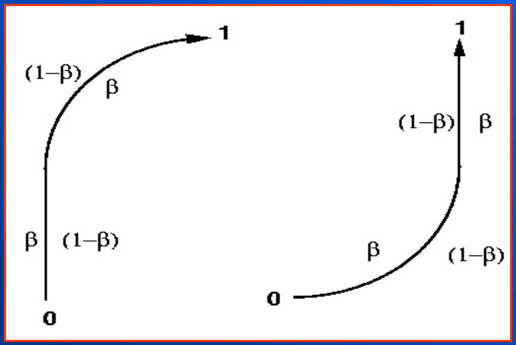




- Three UAVs – fixed turning radius for all UAVs
- Kinematics for UAV model, geometry controls UAV motion
  - Only Line, Arc or Combination manoeuvre possible
- Decision making rules
  - Minimum separation – TRUE
  - Optimum separation – TRUE
  - Collision avoidance – ALWAYS
  - Co-ordinated TOT – WHENEVER
  - No communication – TRUE

Interception without communication

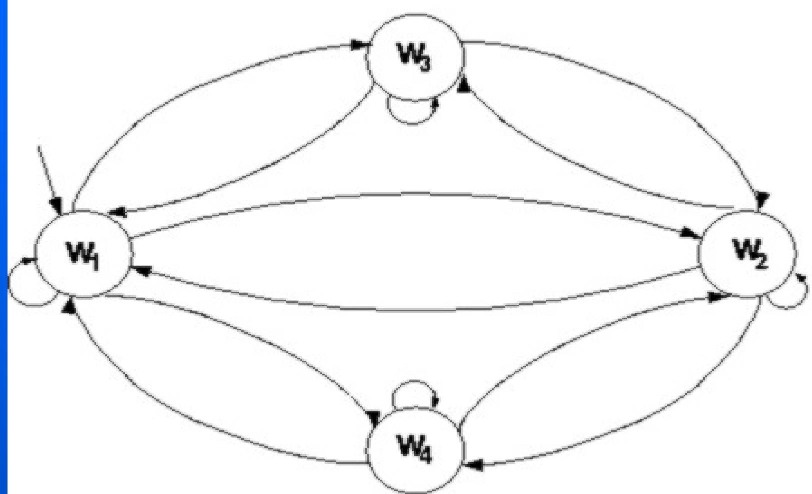




- No a-priori information – except starting points
- Ad-hoc sensing by UAVs
- Combination manoeuvre for attempting interception
- Interception triangle periodically redrawn
- Optimum separation kicks in, if sensors detect UAV
- Interception abandoned if no success

Cranfield UNIVERSITY

Scenario I – Move, Intercept & Separate

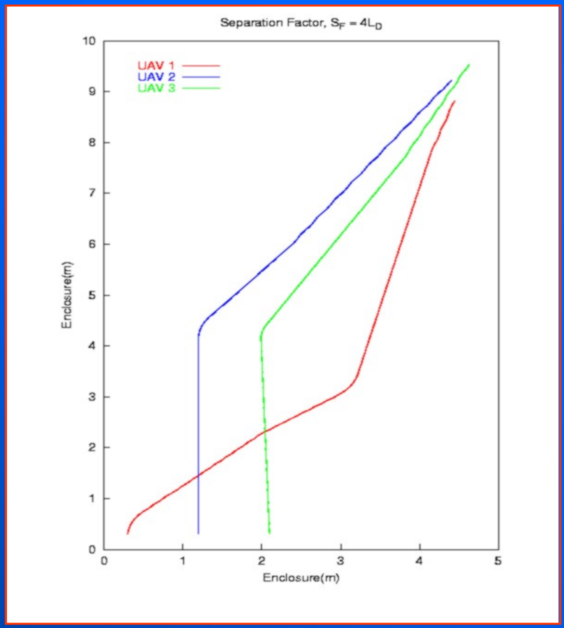


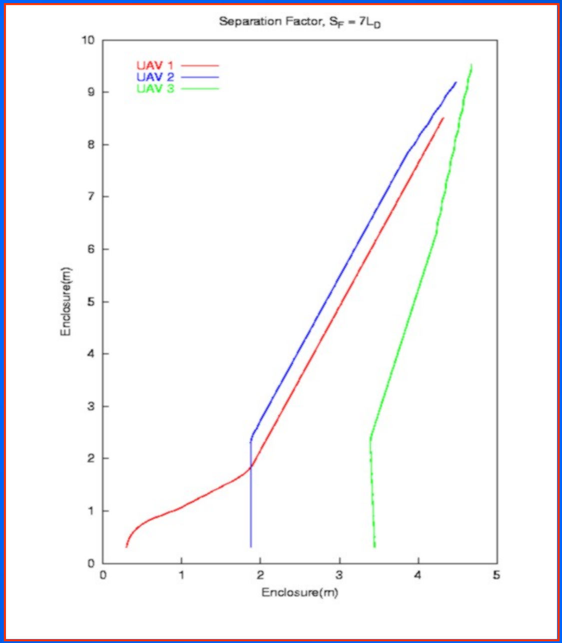
W1	move towards target
W2	interception behaviour
W3	opt. separation violation
W4	min. separation violation

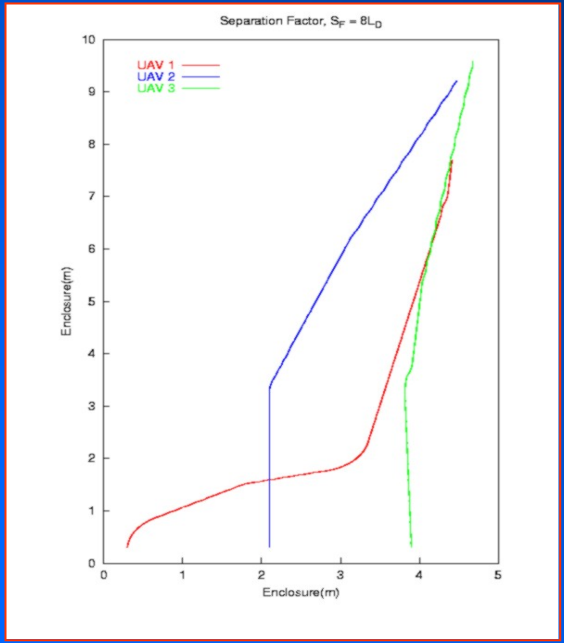
Term	reads as
$\underset{a}{\overset{l}{X}}_i$	Combination manoeuvre arc first, st. line next
$\underset{l}{\overset{a}{X}}_i$	The opposite of above
$\underset{a}{X}_i$	Arc only
$\underset{l}{X}_i$	St. Line only

Worlds	Labelling Function
W1	$\bigwedge_{i=1}^{N} \left[\underset{l}{\overset{a}{X}}_i \vee \underset{a}{\overset{l}{X}}_i \right]$
W2	$\bigvee_{i=1}^{N} \left[\underset{a}{\overset{l}{X}}_1 \vee \ldots \vee \neg \underset{a}{\overset{l}{X}}_N \right]$
W3	$\bigvee_{i=1}^{N} \left[\underset{l}{\overset{a}{X}}_1 \vee \ldots \vee \neg \underset{l}{\overset{a}{X}}_N \right]$
W4	$\bigvee_{i=1}^{N} \left[(\underset{a}{X}_1 \vee \underset{a}{\overset{l}{X}}_1) \ldots \vee \ldots \vee (\neg \underset{a}{X}_N \vee \neg \underset{a}{\overset{l}{X}}_N) \right]$

Simulation results







- Always, reaching the target is preferred over interception, in a UAV
- Sensors manage to detect kin in shorter separation cases
- Increased separation forces UAV3 to switch to task completion
- UAV1 performs interception manoeuvre each time – its direction of travel ties in with its interception orientation
- In Figs 1 & 2, UAVs 2 and 3 maintain a “loose” formation throughout

Extracting properties as LTL formulae

Reachability analysis, can be written in LTL as follows:

$$\Box\left[\bigwedge_{i=1}^{N} \overset{a}{\underset{l}{X}}_i\, \mathcal{U}\, (x_i, y_i) \in \left([x_{goal}, x_{end}], [y_{goal}, y_{end}]\right)\right]$$

The formula can be read as:

"*all the robots continue moving until they reach the area designated as the goal area*."

Cranfield UNIVERSITY

Extracting properties as LTL formulae

Safety properties are represented in LTL as follows:

$$\Box \neg \left[\bigwedge_{\substack{i,j=1 \\ i \neq j}}^{N} \sqrt{(x_{i2} - x_{j2})^2 + (y_{i2} - y_{j2})^2} < L_D \right]$$

The formula can be read as:

"no two robots can ever come closer than a pre-specified separation boundary."

Cranfield UNIVERSITY

Extracting properties as LTL formulae

By taking into account the lack of communication between the robots, interception is more weakly specified using the eventually and the disjunction operator as follows:

$$\Diamond\left[\bigvee_{\substack{i,j=1 \\ goal}}^{N} \overset{l}{\underset{a}{X_i}} \rightarrow L_D < \sqrt{(x_{i2} - x_{j2})^2 + (y_{i2} - y_{j2})^2} \leq 1.5 L_D\right]$$

The formula can be read as:

"in the course of goal seeking, two robots may intercept each other."

Cranfield UNIVERSITY

The critical areas of the code identified for verification are described below

Goal Completion. All robots are provided with a goal/task that needs completion. A critical section of the program executes the robot processes until the individual robots flag goal completion. We need to verify whether all robots do indeed complete their goal and whether the code does perform this check before termination.

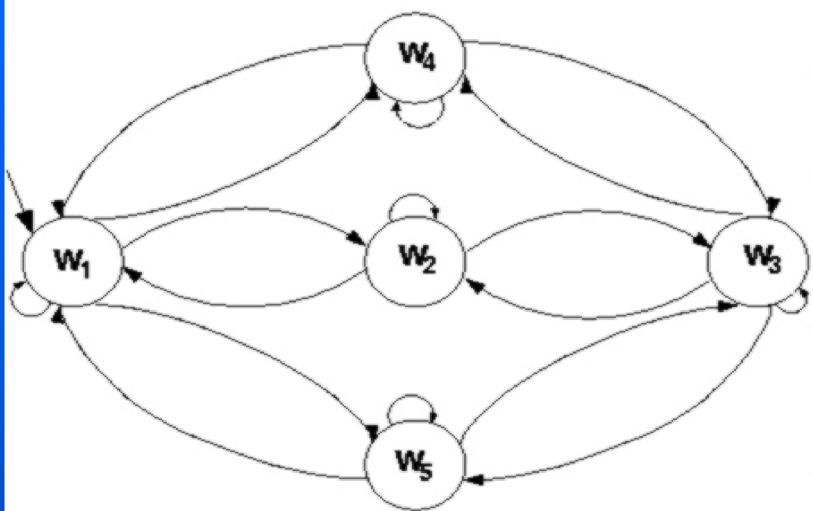
Interception. One contribution of this research work is demonstration of the ability of the robots to attempt interception of their immediate neighbour, without communication, but with their neighbours' initial co-ordinates known. We wish to verify this behaviour using the model checker.

Verification results for critical aspects of the system

Verifications	Task completion of robots	Interception
Livelocks	No livelocks	No livelocks
Deadlocks	No deadlocks	No deadlocks
Assertion Violation	No `assert` property violated	`assert(`Rob_A `-` Rob_B `between` $(L_D, 1.5L_D)$`)` violated
Completion	Yes	Yes

Cranfield UNIVERSITY

Scenario II - Scenario I & Obstacles

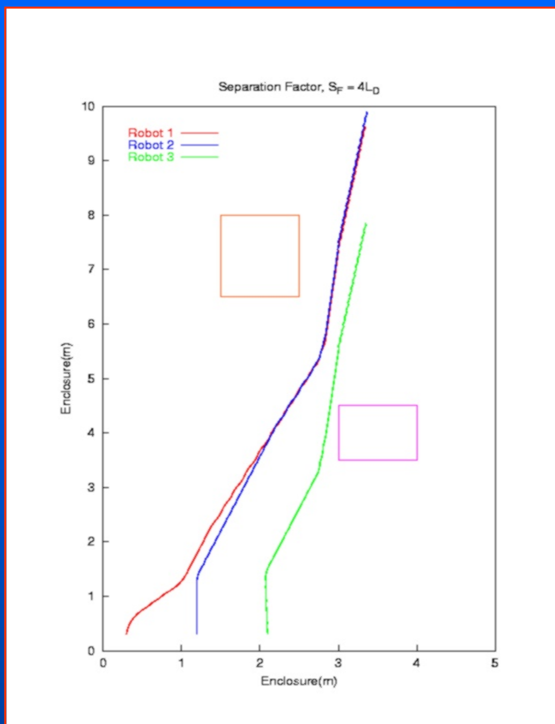


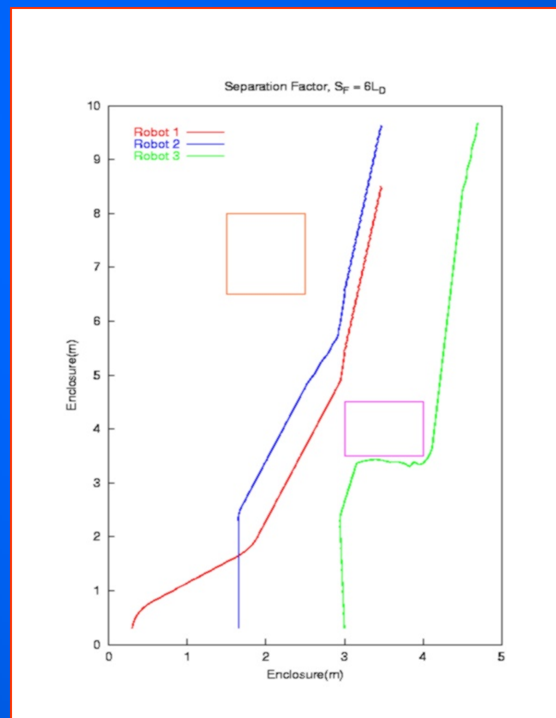
Worlds	Labelling Function
W1	$\bigwedge_{i=1}^{N} \left[\overset{a}{\underset{l}{X}}_i \vee \overset{l}{\underset{a}{X}}_i \right]$
W2	$\bigvee_{i=1}^{N} \left[\overset{l}{\underset{a}{X}}_1 \vee \ldots \vee \neg \overset{l}{\underset{a}{X}}_N \right]$
W3	$\bigvee_{i=1}^{N} \left[\underset{a}{X}_i \vee \underset{l}{X}_i \vee \overset{l}{\underset{a}{X}}_i \vee \overset{a}{\underset{l}{X}}_i \right]$
W4	$\bigvee_{i=1}^{N} \left[(\underset{a}{X}_1 \vee \overset{l}{\underset{a}{X}}_1) \ldots \vee \ldots \vee (\neg \underset{a}{X}_N \vee \neg \overset{l}{\underset{a}{X}}_N) \right]$
W5	$\bigvee_{i=1}^{N} \left[\overset{a}{\underset{l}{X}}_1 \vee \ldots \vee \neg \overset{a}{\underset{l}{X}}_N \right]$

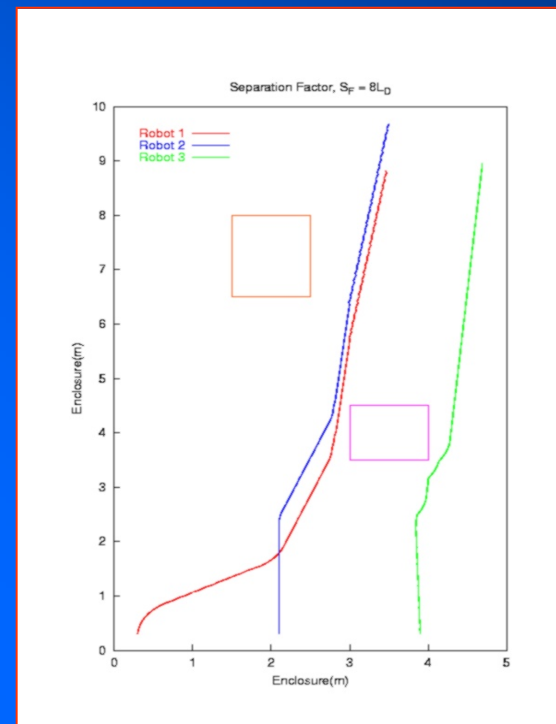
W1	move towards target
W2	interception behaviour
W3	avoid obstacles
W4	min. separation violation
W5	opt. separation violation

Term	reads as
$\overset{l}{\underset{a}{X}}_i$	Combination manoeuvre arc first, st. line next
$\overset{a}{\underset{l}{X}}_i$	The opposite of above
$\underset{a}{X}_i$	Arc only
$\underset{l}{X}_i$	St. Line only

Scenario II: Obstacle Avoidance



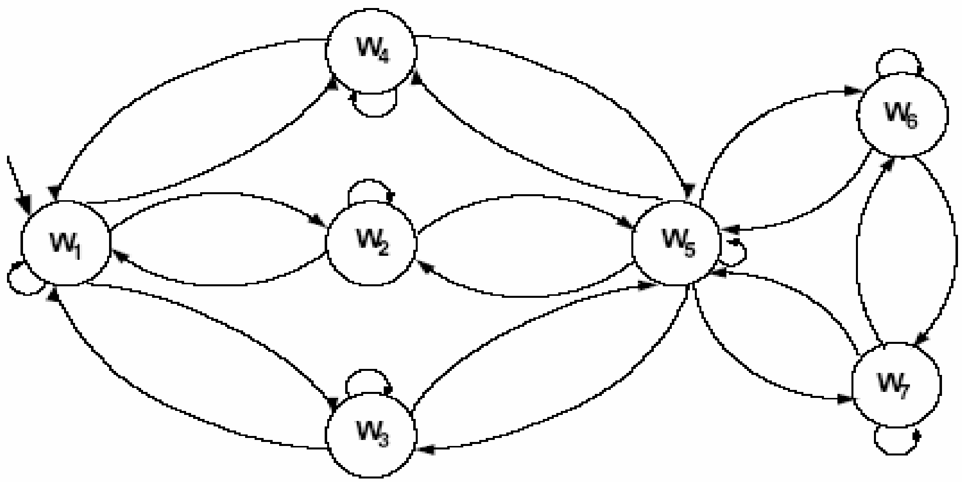




- Obstacle avoidance is successful in each separation scenario
- No communication between robots, hence interception is not achieved by all three robots before goal completion

Cranfield UNIVERSITY

Kripke Model for navigation based on Dubins Curves

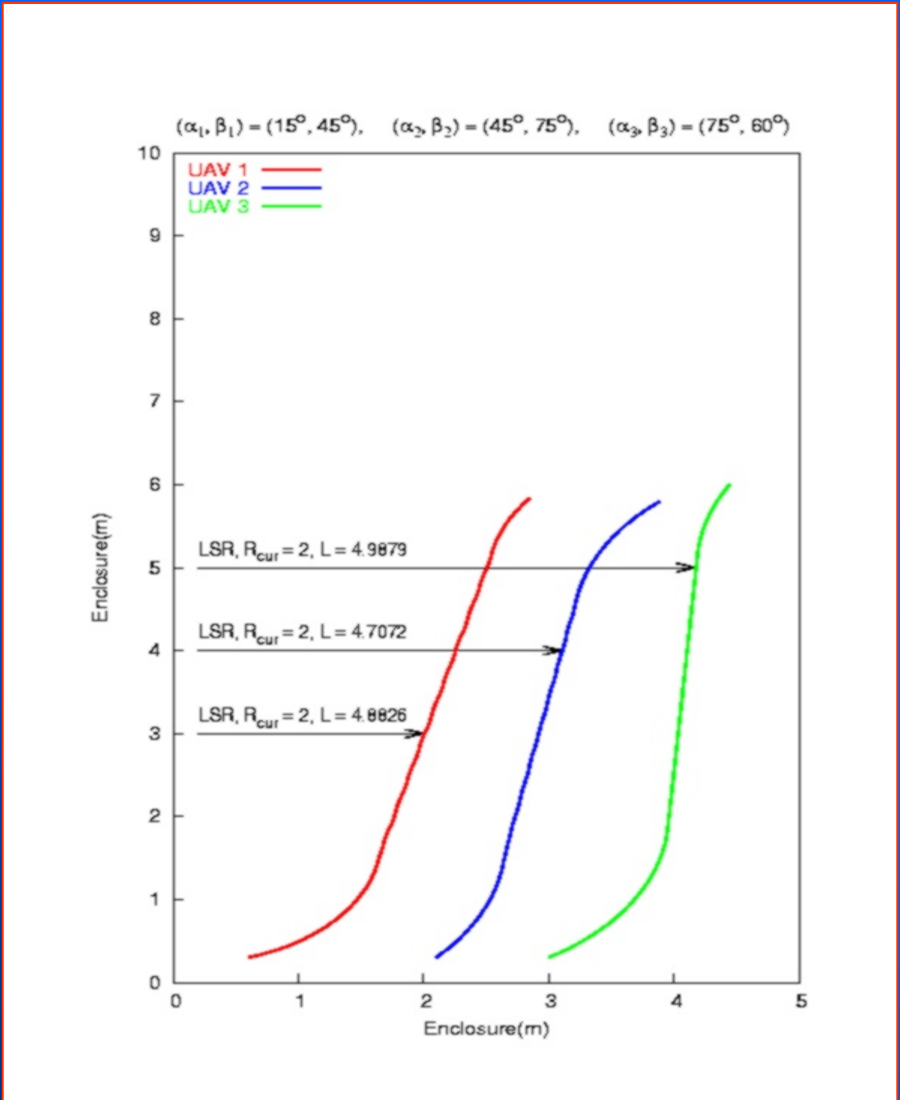


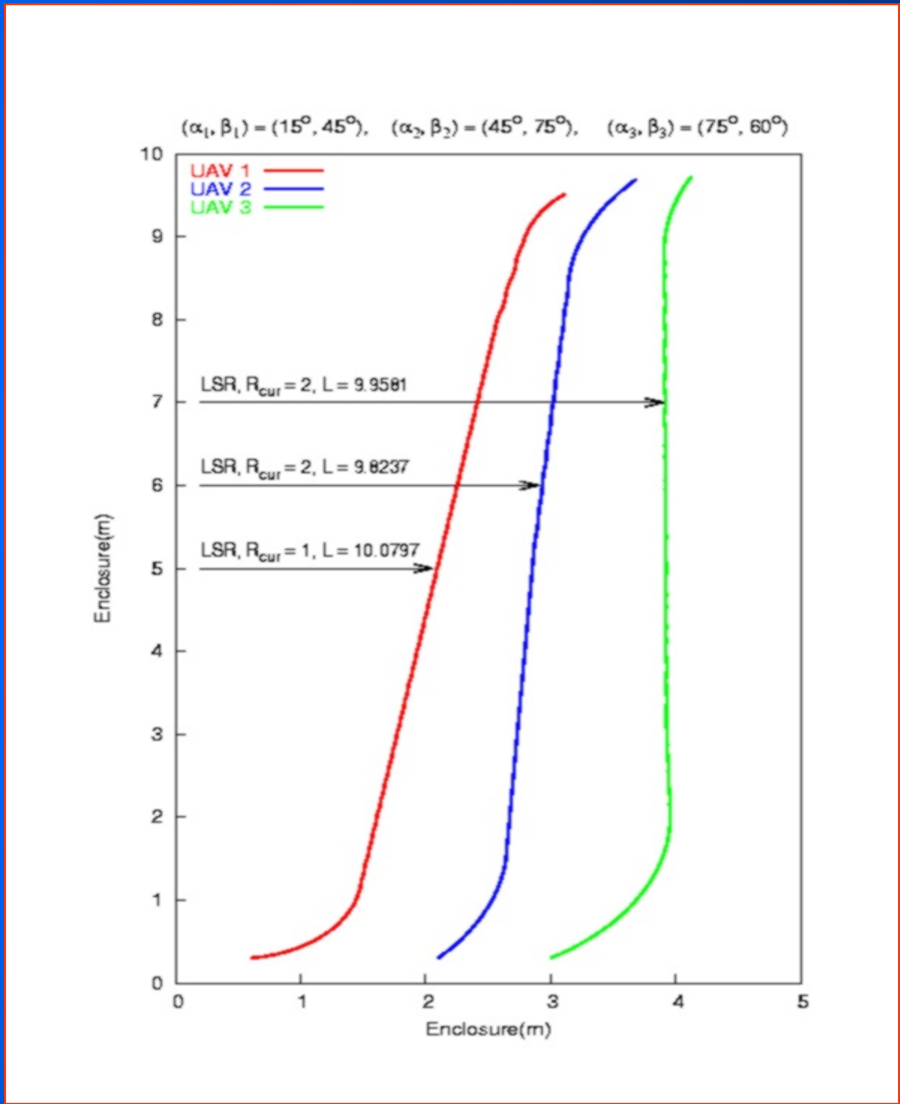
Worlds	Labelling Function
W1	$\bigwedge_{i=1}^{N}\left[\bigcap_{i=1}^{N} X_i\right]$
W2	$\bigwedge_{i=1}^{N}\left[\{X_i\}_{Dub} - \left[\bigwedge_{j=1}^{k,k<N}\{X_j\}_{Dub}\right]\right] = \min$
W3	$\bigvee_{i=1}^{N}\left[{X_i}_a \vee {X_i}_l\right]$
W4	$\sum_{i=1}^{N}\left[\{X_i\}\bigcap_{j}^{k,k<N}\{X_j\}\right]$
W5	$\bigwedge_{i=1}^{N}\left[{X_i}_{Dub}\right]$
W6	${X_i}_l$
W7	${X_i}_a$

Worlds	denote
W1	start
W2	path planner
W3	separation violation
W4	communication
W5	move
W6	straight line motion
W7	arc motion

Term	reads as
$\{X_i\}_{Dub}$	Set of Dubins paths for a family of curves
$\bigcap_{i=1}^{N} X_i$	Communication, as an inter-section operation on robots
${X_i}_a$	arc only
${X_i}_l$	straight line only

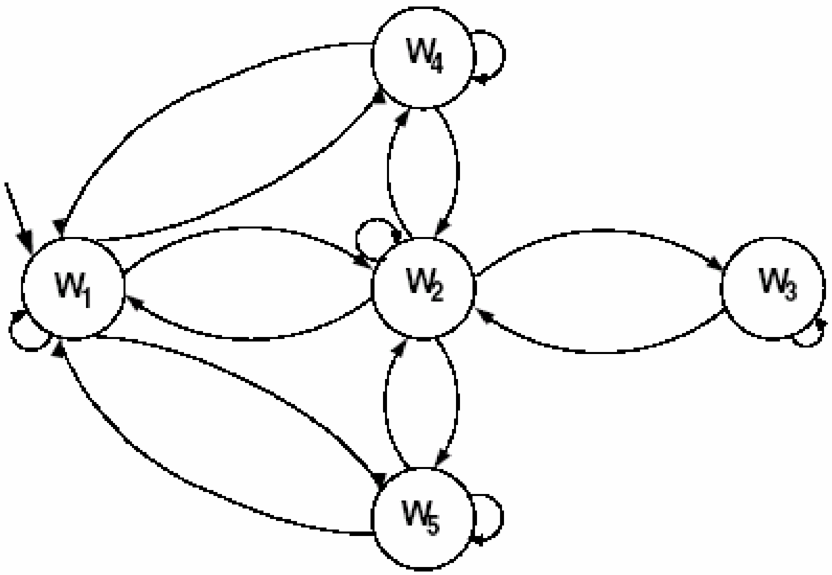
Dubins implementation





Cranfield UNIVERSITY

Effect of communication on co-ordinated TOT

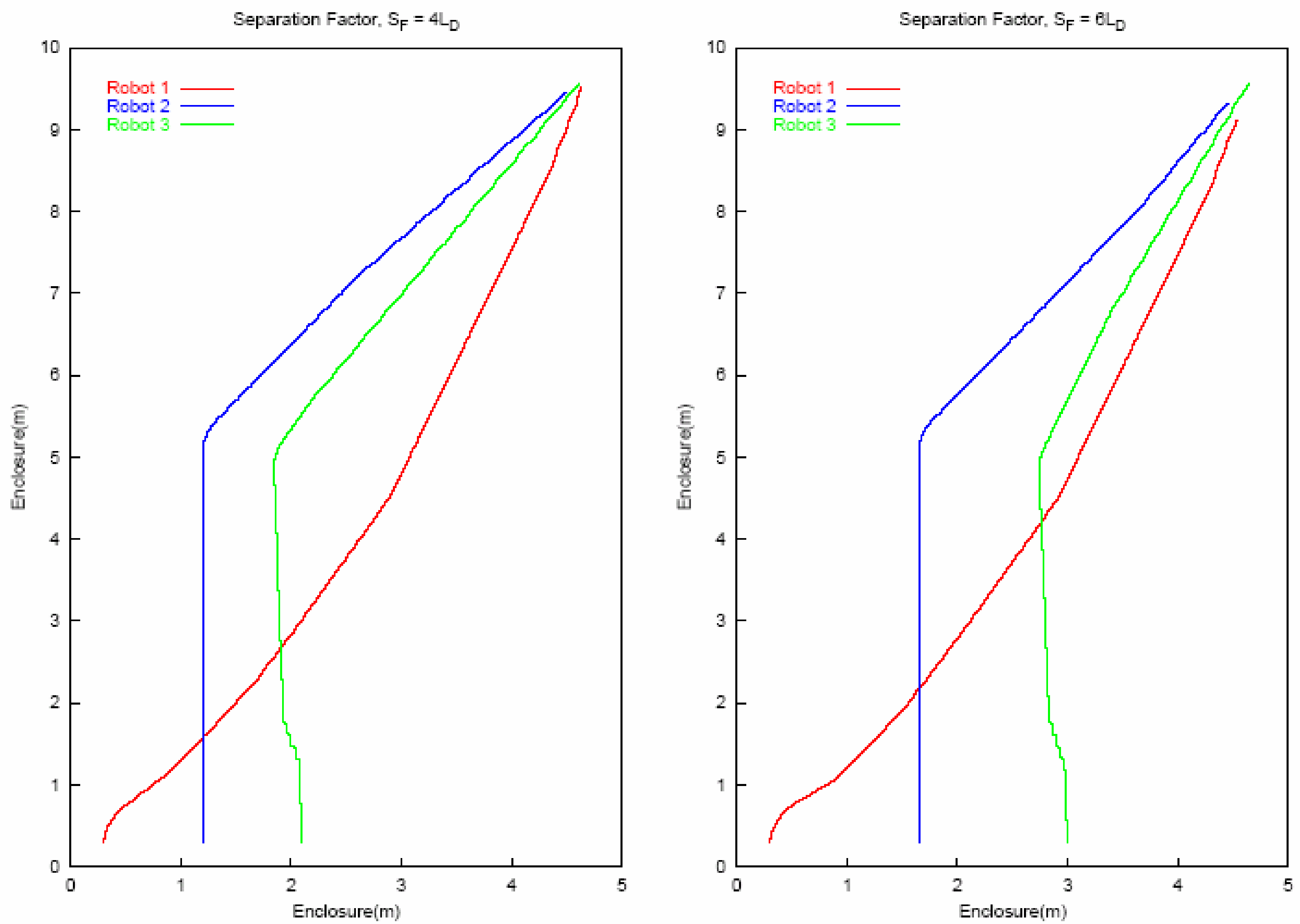


Worlds	Labelling Function
W_1	$\bigwedge_{i=1}^{N} \left[\overset{a}{\underset{l}{X_i}} \vee \overset{l}{\underset{a}{X_i}} \right]$
W_2	$\bigvee_{i=1}^{N} \left[\overset{l}{\underset{a}{X_1}} \vee \ldots \vee \neg \overset{l}{\underset{a}{X_N}} \right]$
W_3	$\sum_{i=1}^{N} \left[\{X_i\} \bigcap_{j}^{k,k<N} \{X_j\} \right]$
W_4	$\bigvee_{i=1}^{N} \left[(\underset{a}{X_1} \vee \overset{l}{\underset{a}{X_1}}) \ldots \vee \ldots \vee (\neg \underset{a}{X_N} \vee \neg \overset{l}{\underset{a}{X_N}}) \right]$
W_5	$\bigvee_{i=1}^{N} \left[\overset{a}{\underset{l}{X_1}} \vee \ldots \vee \neg \overset{a}{\underset{l}{X_N}} \right]$

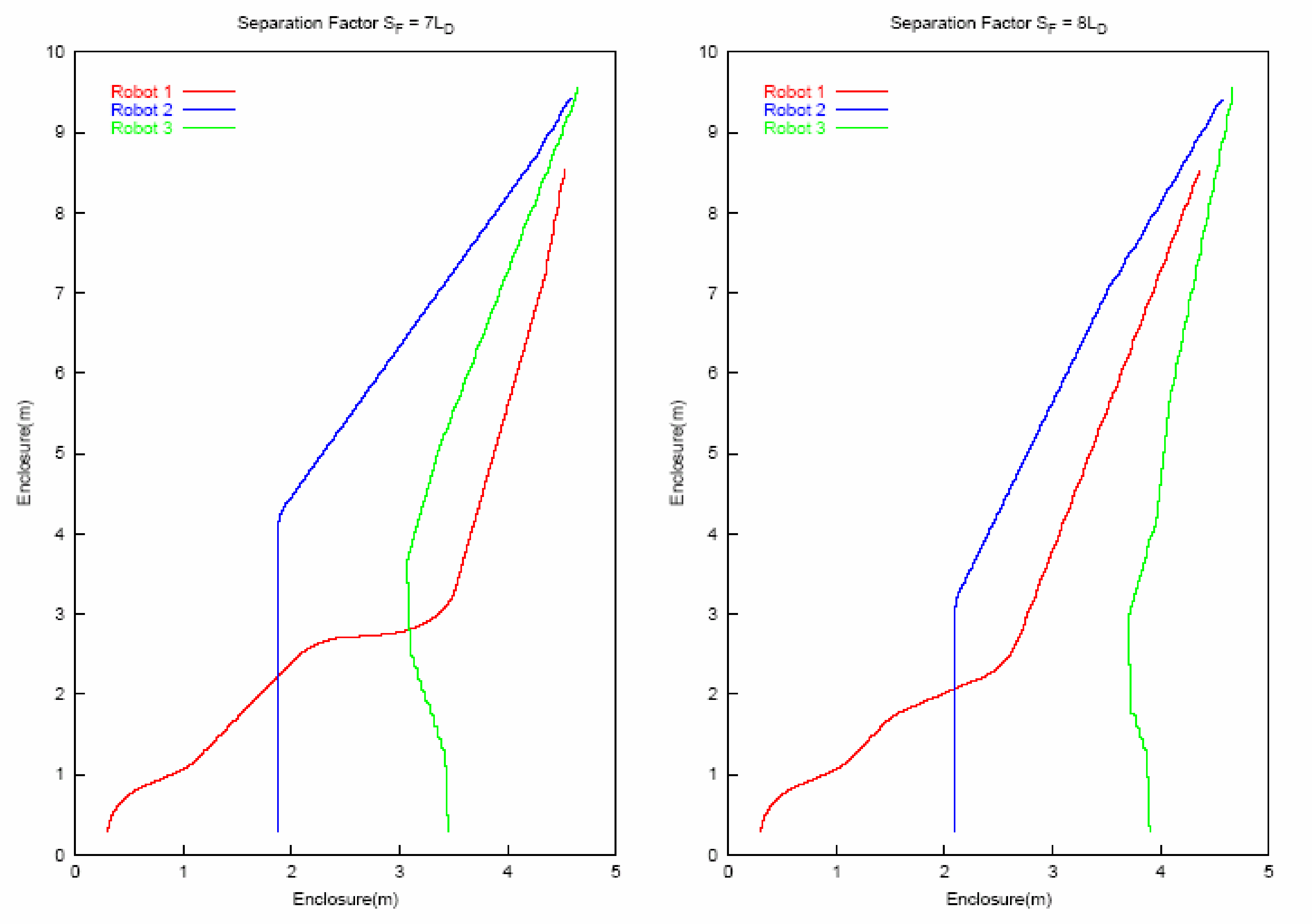
W_1	move towards target
W_2	interception behaviour
W_3	broadcast positions
W_4	min. separation violation
W_5	opt. separation violation

Term	reads as
$\overset{l}{\underset{a}{X_i}}$	Combination manoeuvre arc first, st. line next
$\overset{a}{\underset{l}{X_i}}$	The opposite of above
$\underset{a}{X_i}$	Arc only
$\underset{l}{X_i}$	St. Line only

Cranfield UNIVERSITY



Cranfield UNIVERSITY



Any questions?