Colloquium d'informatique de l'UPMC
22 October 2013

Gilles Dowek

- Slides 2–41: *Are formal methods the future of air traffic control? (Is there an autopilot on board?)*
- Slides 42–62: *Teaching informatics in high-school: what have we learnt?*

Are formal methods the future of air traffic control?

(Is there an autopilot on board?)

Gilles Dowek

Joint work with César Muñoz and many researchers of the NASA-Langley research center

I. From complex objects to formal methods

Formal methods

From 10^2 components (steam engine, radio, etc.) to 10^8 (program, computer, etc.)

Humans cannot avoid making mistakes

All the methods invented to avoid (reduce the number of) mistakes

Empirical (e.g. testing), a priori (e.g. model checking, proofs), etc.

Reformulation

To introduce some redundancy: two algorithms to solve the same problem, proved to be equal same idea as error correcting codes

To give less information: Gaussian elimination, $AB = I$

This talk

Formal methods

Examples: air traffic control

Bugs are not always as harmful: transportation, medicine, power plants, etc.

Future of transportation: human out of the loop (more efficient, paradox: safer / perceived as more dangerous)

Two examples

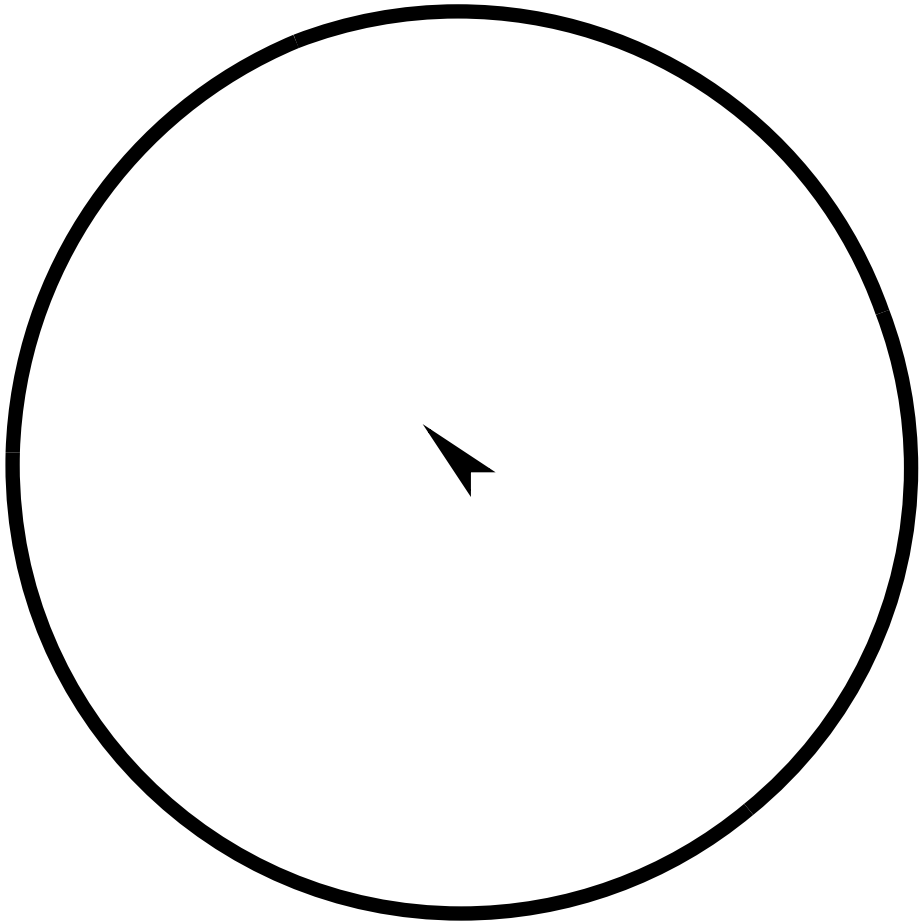
I. Prevention band algorithms

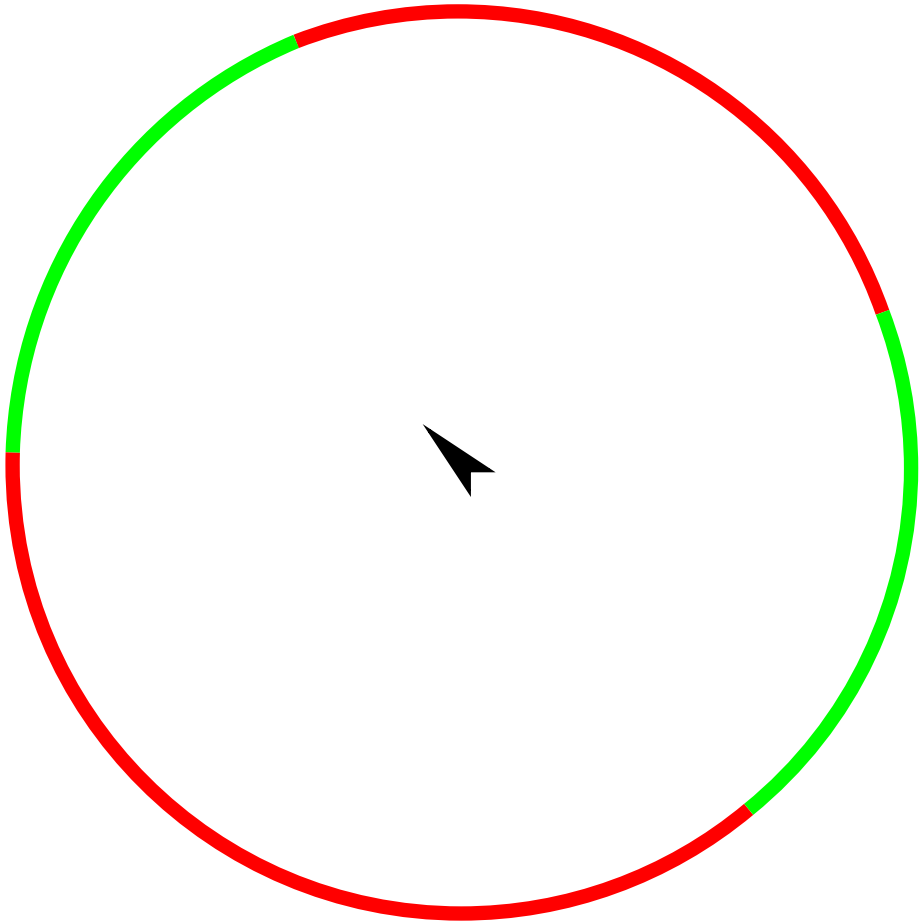
II. Small aircraft transportation systems

Both part of the ACCoRD library

But very different problems (s)

I. Prevention band algorithms





The goal

A prevention band algorithm

A formula expressing its correctness:

$$\text{green}(\mathbf{v}, \mathbf{w}) \Rightarrow \text{noconflict}(\mathbf{v}, \mathbf{w})$$

Prove this formula in the proof-checking system PVS (Coq, Isabelle, HOL, etc.)

Prevention bands = conflict detection?

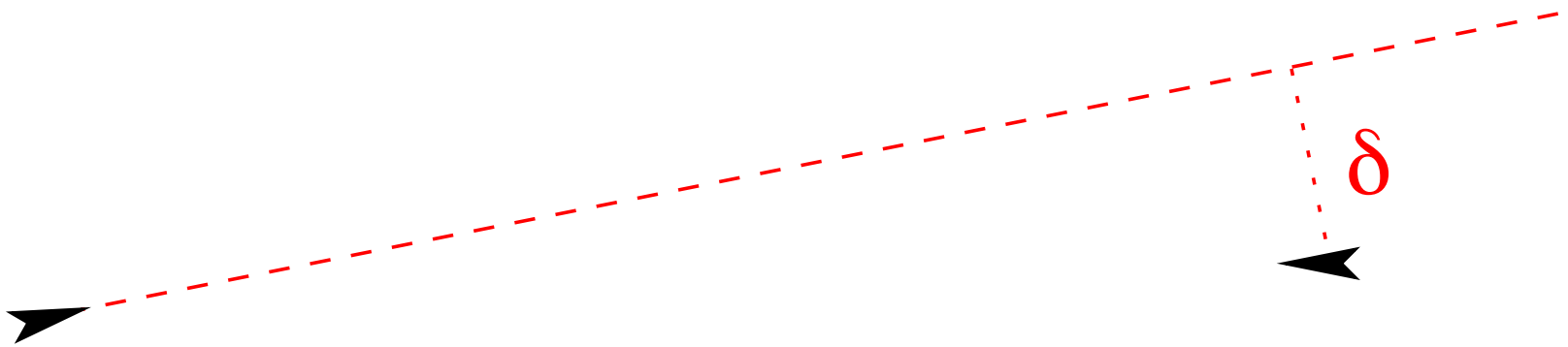
If Ownship takes heading 45°, will a conflict result?

Try all the possible headings: infinite number

Discretize: ok but discretization step arbitrary

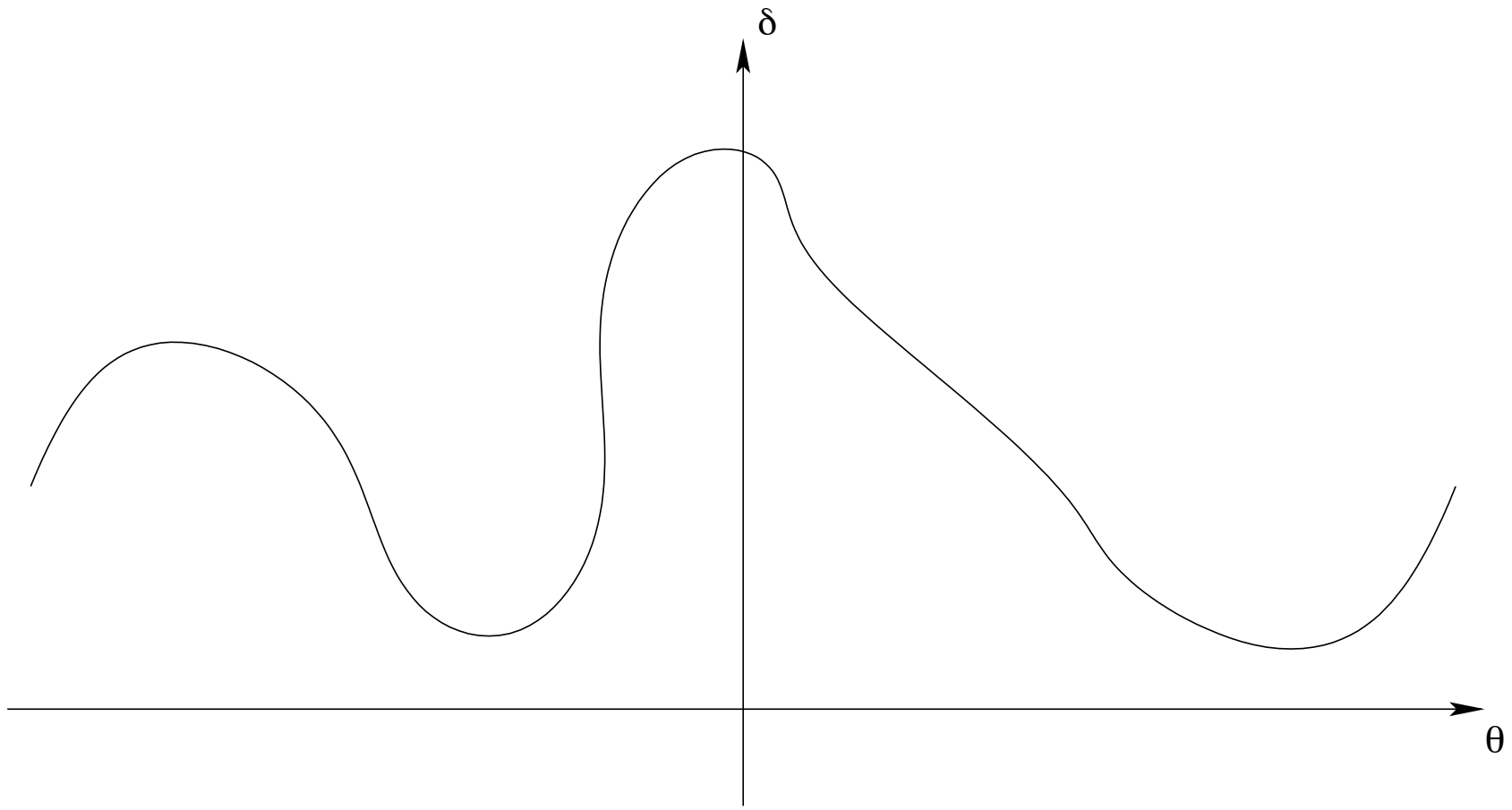
The essence of the problem is continuous, not discrete

From conflict to minimal separation

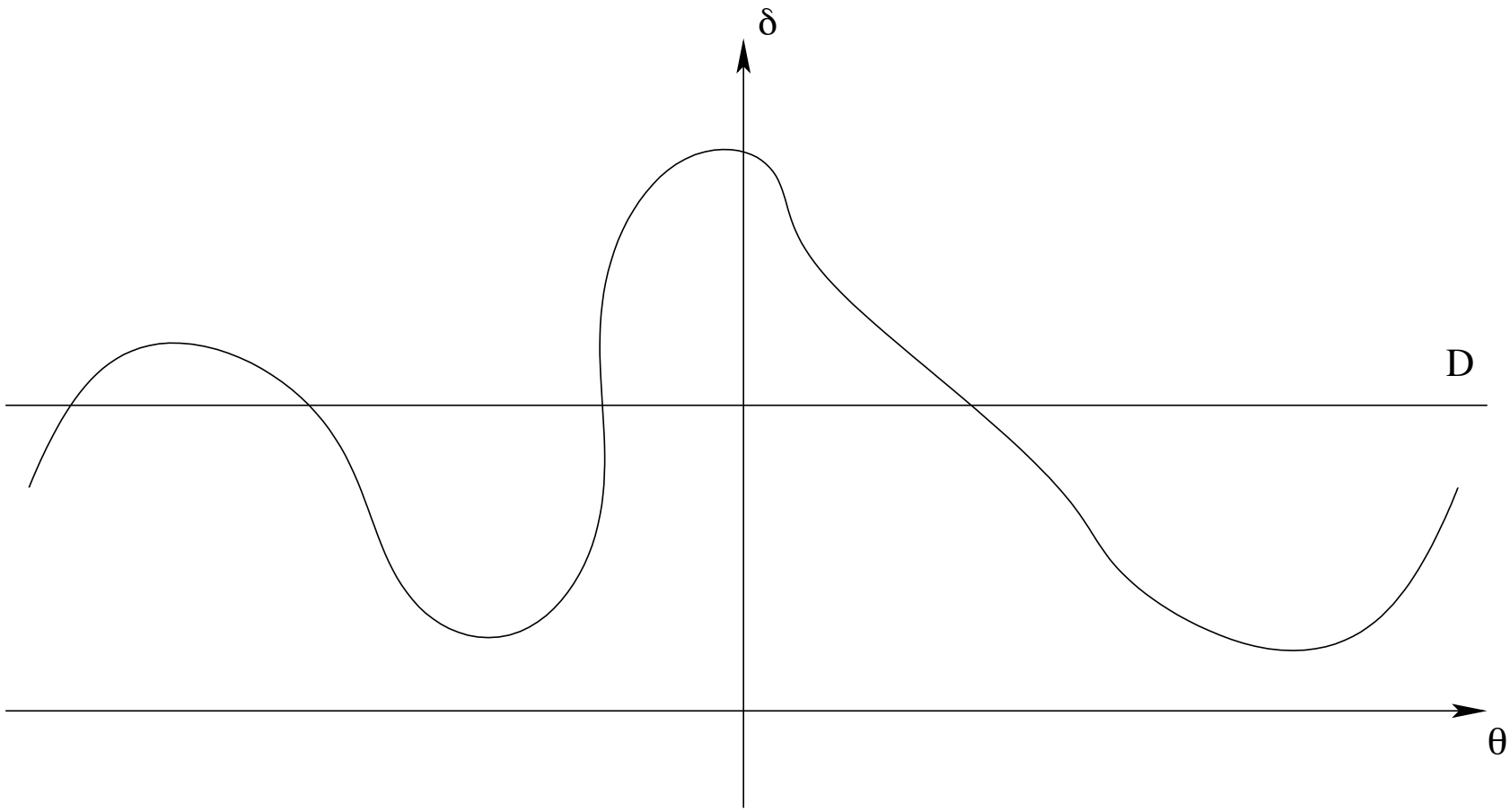


Already a choice here: the full line (including past conflicts), the half line (future conflicts), a segment (finite look-ahead time)

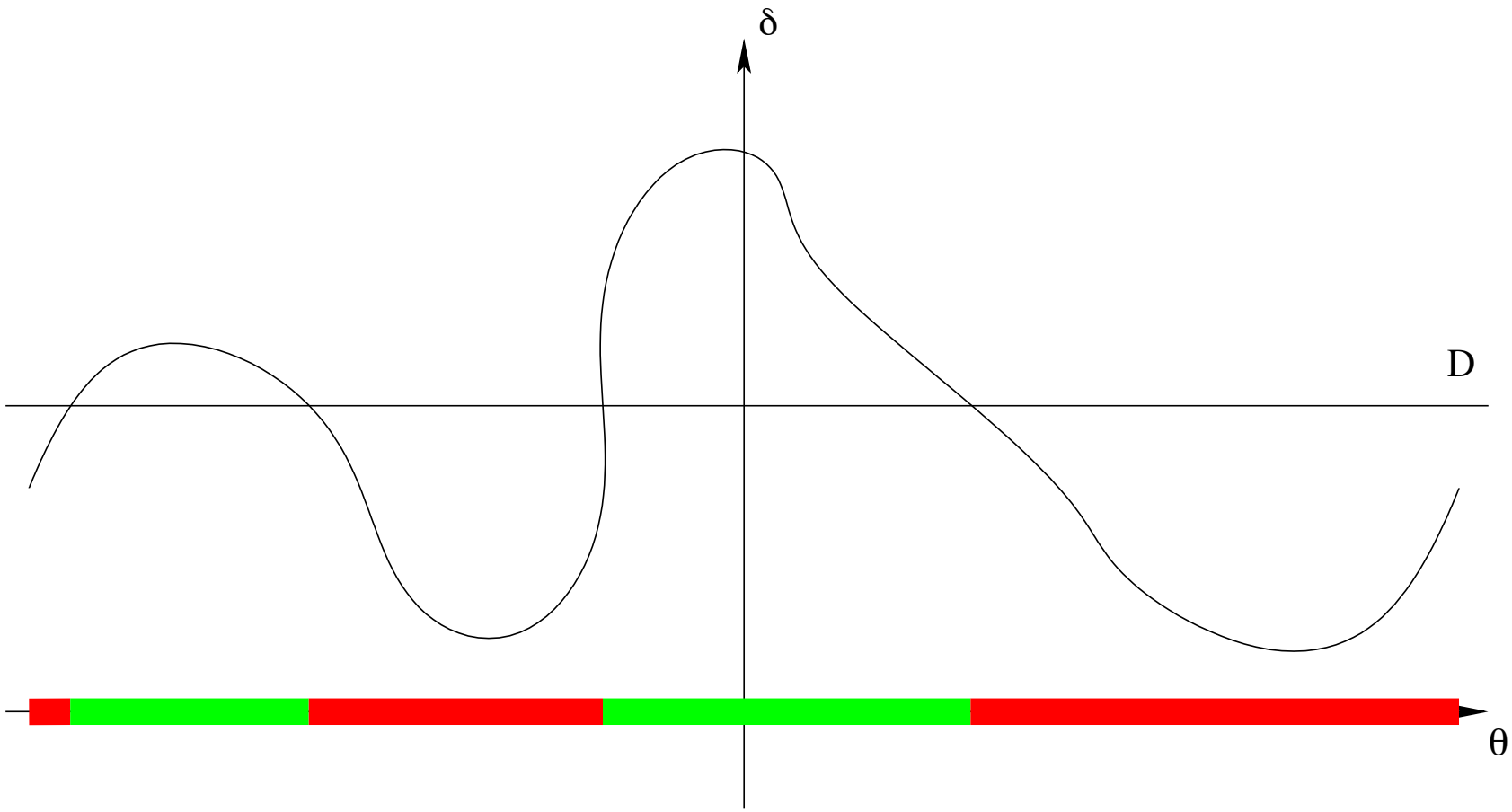
Minimal separation



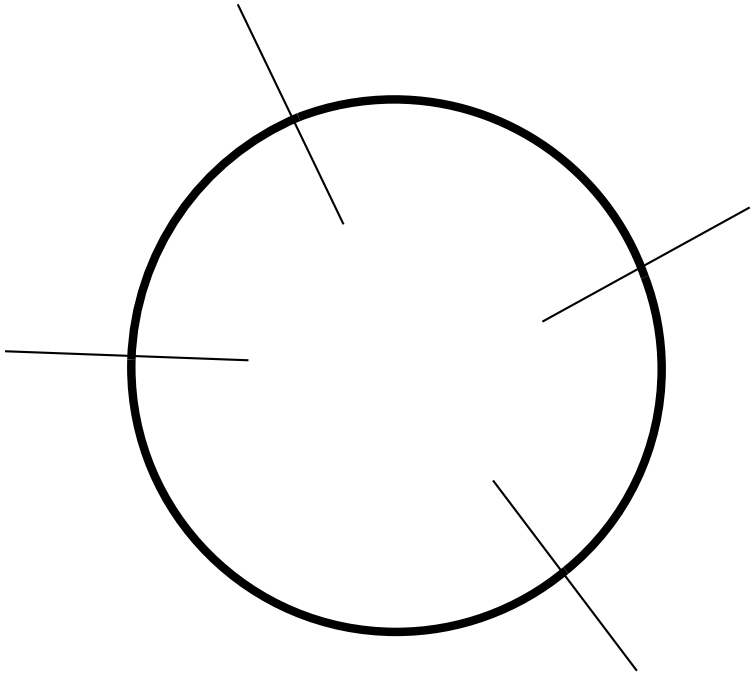
Minimal separation



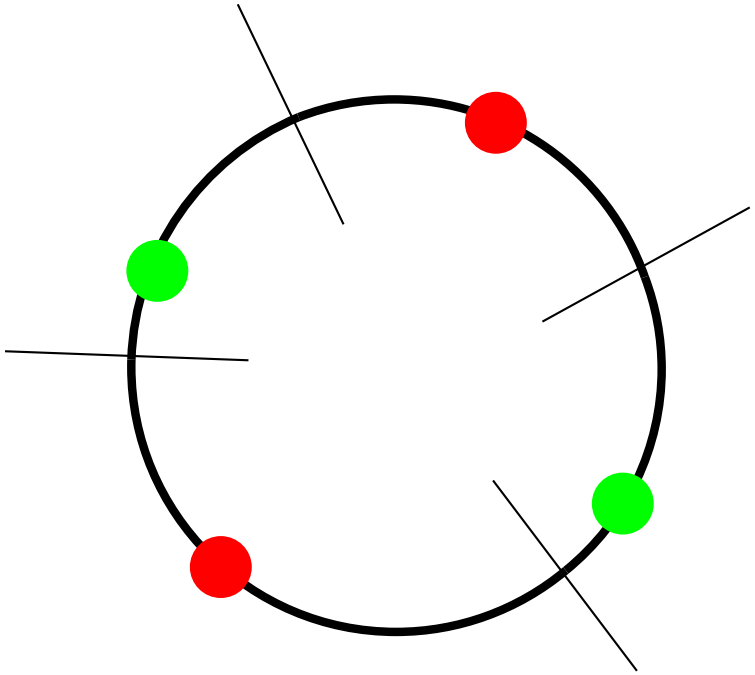
Minimal separation



What we need

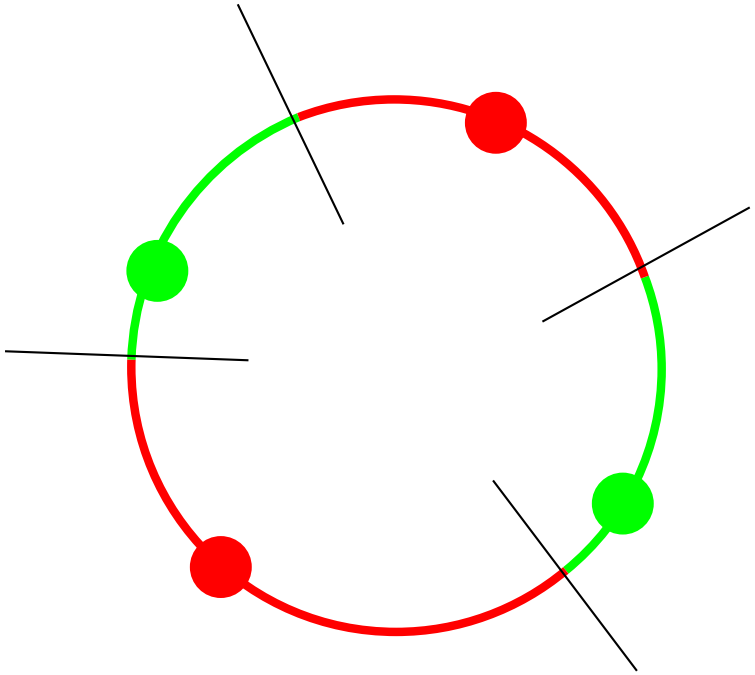


What we want



Then coloring is easy (modulo rounding)

What we want

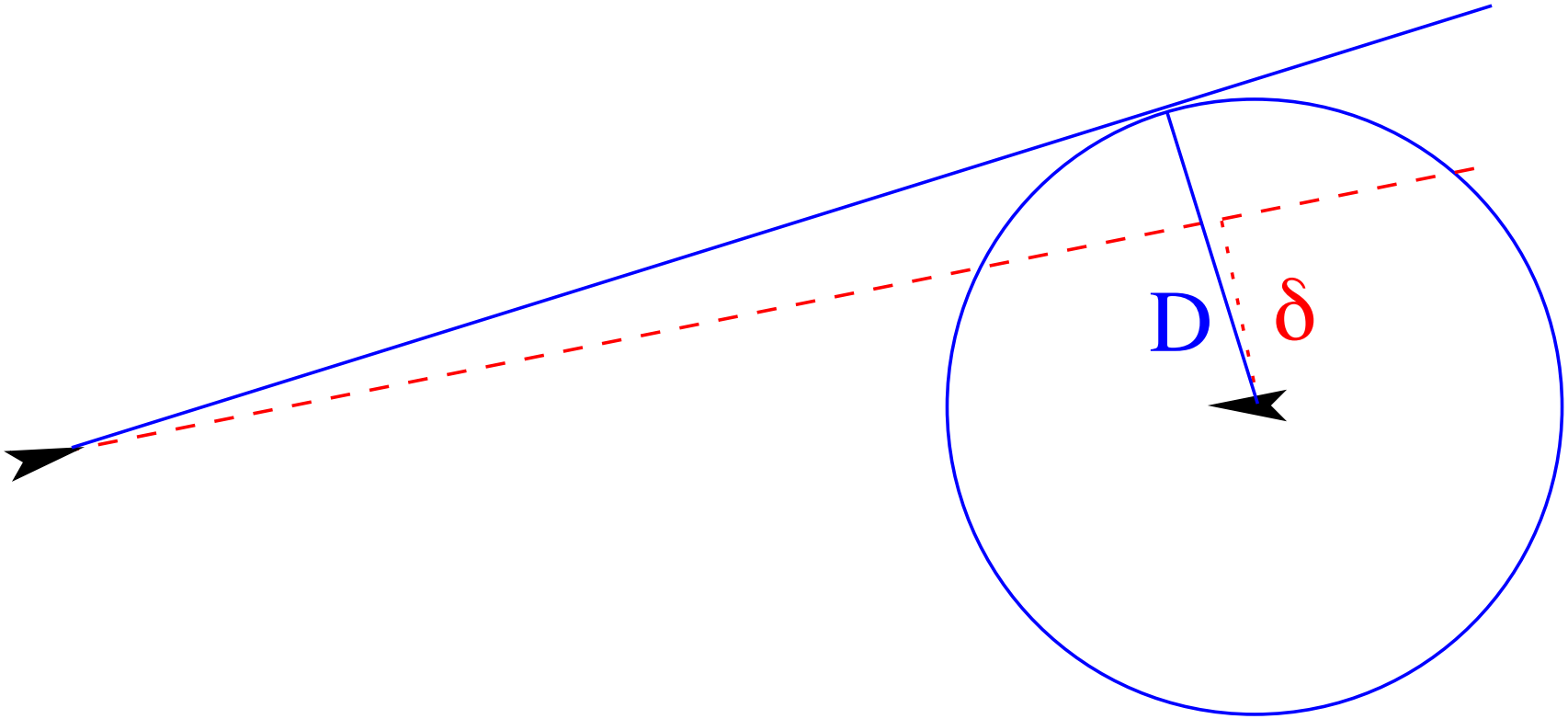


Then coloring is easy (modulo rounding)

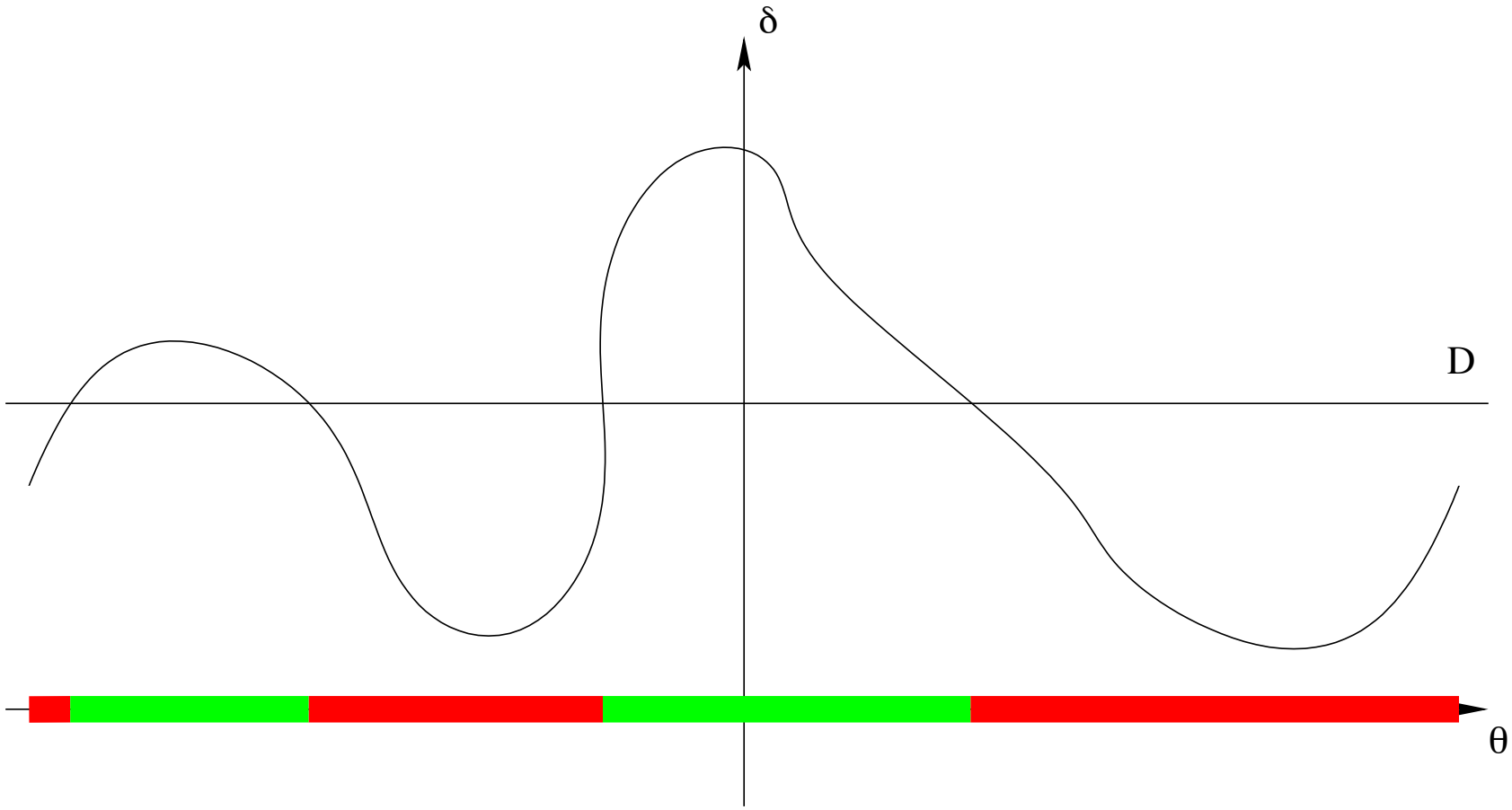
What we need

The values of θ such that $\delta(\theta) = D$

This is the ACCoRD conflict resolution algorithm (reuse)



The intermediate value theorem



requires continuous functions

Is δ continuous in θ?

Separation at time t: $d(\theta, t) = \|\mathbf{s} + \mathbf{v}(\theta)t\|$ continuous

$\delta(\theta) = min_{t \in I} d(\theta, t)$

Is continuity preserved by the min operation?

Unfortunately: not always

Indeed

Slightly slower: no conflict (minimal separation is current)

Slightly faster: conflict (the minimal separation is 0)

Indeed

Slightly slower: no conflict (minimal separation is current)

Slightly faster: conflict (the minimal separation is 0)

Yet: if only slightly slower the violation will be far in the future:

should be eliminated with a finite look-ahead time

A finite look-ahead time helps

$\delta(\theta) = min_{t \in I} d(\theta, t)$

Continuity is preserved by min when I compact $[0, T]$

d continuous on a compact set: uniformly continuous

$\forall t \forall \varepsilon \exists \eta \forall \mathbf{w} \; (\|\mathbf{w} - \mathbf{v}(\theta)\| \leq \eta \Rightarrow \|d(\mathbf{w}, t) - d(\mathbf{v}(\theta), t)\| \leq \varepsilon)$

$\forall \varepsilon \exists \eta \forall \mathbf{w} \; (\|\mathbf{w} - \mathbf{v}(\theta)\| \leq \eta \Rightarrow \forall t \|d(\mathbf{w}, t) - d(\mathbf{v}(\theta), t)\| \leq \varepsilon)$

But...

Do we really want to prove Heine-Cantor theorem in PVS?

Or a simpler solution (bypassing Heine-Cantor theorem)?

$$\delta^2(\mathbf{v}) - D^2 = (\mathbf{s} + min(max(-\frac{\mathbf{s.v}}{\mathbf{v}^2}, 0), T)\mathbf{v})^2 - D^2$$

Difficult to prove continuous but $\mathbf{v}^4(\delta^2(\mathbf{v}) - D^2)$ same sign and obviously continuous

Partial conclusions

(to be contradicted in the second part of the talk)

Air traffic control requires to model not only the algorithm but also its environment (airspace)

A problem of differential geometry

Requires arbitrarily difficult theorems and concepts

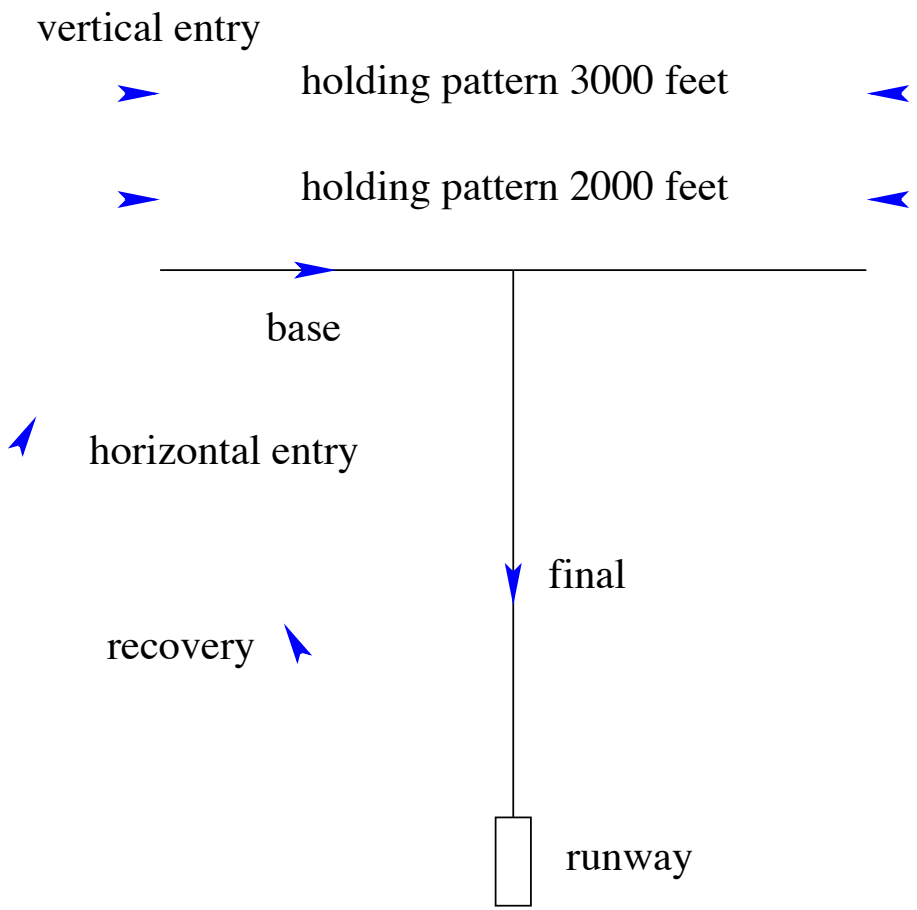
II. Small aircraft transportation system

Many small airports in the world

A few aircraft everyday

Cannot afford someone in the control tower

Can we imagine a protocol for landing and taking off with no human intervention?


vertical entry
holding pattern 3000 feet
holding pattern 2000 feet
base
horizontal entry
final
recovery
runway

Rules: an example

An aircraft can enter vertically in the holding pattern at 3000 feet on the right if

- no aircraft in this zone
- no aircraft in the right missed approached zone
- no aircraft in the right horizontal approach zone
- at most one aircraft in a right zone or with a right missed approach fix

Questions

Conflict (two aircraft in the same zone)?, Deadlock?, etc.

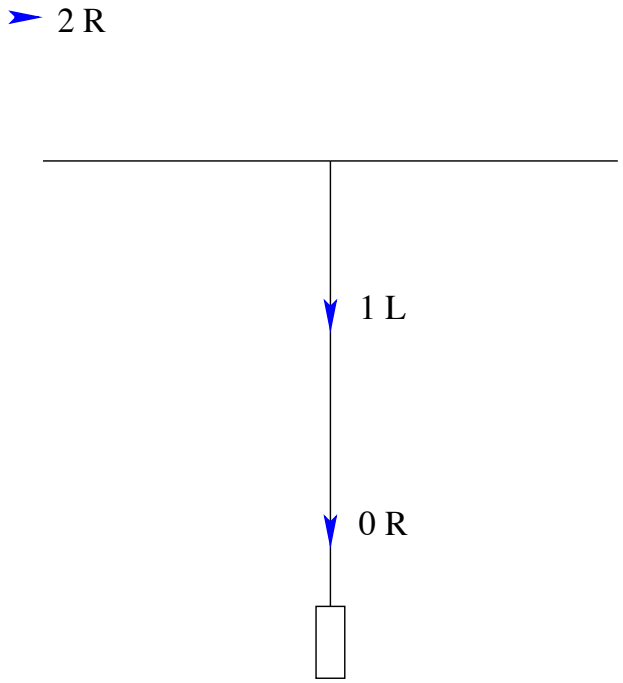
Try and see (but many crashes)

Model and simulate

Model and prove

Explaining the rules to a computer

State: list of aircraft (position, missed approached fix, etc.)




```
[(0,R,final); (1,L,final); (2,R,holding3r)]
```

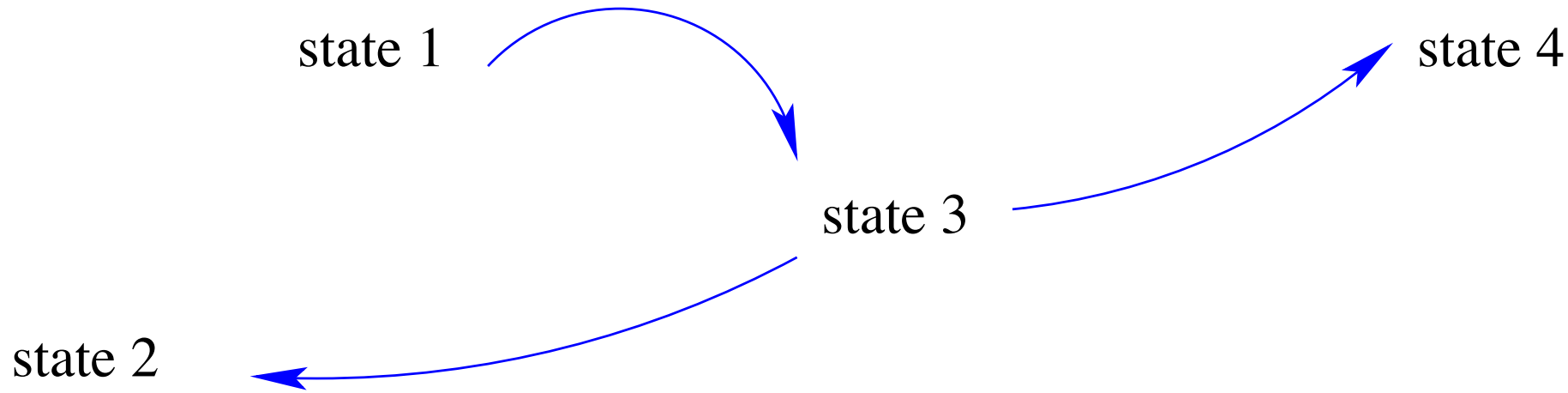
Explaining the rules to a computer

Rule: an algorithm mapping each state to a sequence of states

```
let rule6r s = match s#baser with
| [] -> []
| h::r ->
    let x = order h
    in if x = 0 || mem (x-1) s#final
       then [{<baser = r; final = final@[h]>}]
       else []
```

From simulation to enumeration

States and transitions



A priori, an infinite number of states

But only a finite number of reachable states: enumeration

No conflict but a deadlock

A complex scenario lead to

- 0 (3000 feet)
- 1 (2000 feet)

A modification of one rule made the deadlock disappear

What is a discrete / continuous problem?

Two problems: very different methods

Formal methods: a Swiss knife rather than one size fit all

What is the difference?

Finite / infinite? not quite: infinities in both cases

But...

Modeling a problem as a state / transition system makes enumeration possible (depth first search)

An algorithm can answer the question: is there a deadlock?

A (slightly) unorthodox definition

Poincaré: not all questions need to be answered with a proof, some just require computation

Discrete problem: can be answered by an algorithm

Continuous problem: requires a proof

Two fundamental problems

Design logics that take computation into account

Allow interoperability between formal methods tools as none will ever solve all the problems

Teaching informatics in high-school: what have we learnt?

Gilles Dowek

I. Some history (and geography)

Prehistory

Informatics has been taught (as an option) in French high-schools in the 80's and 90's

At that time a professional organization of teachers:

Enseignement public et informatique (EPI)

Removed twice (too expensive, too elitist, not a real science, ...)

More than ten years with no informatics in high-schools

But a concern

Academy of sciences (Maurice Nivat: March 15th, 2005)

Absurdity of teaching the `while` loop to 19 or 21 y.o. students

Too many people saying too many stupid things (Hadopi)

2007

A letter from EPI (Jacques Baudé and Jean-Pierre Archambault) to all the candidates to the presidential election

An answer from the one who finally was elected

Creation of the ITIC group by EPI and ASTI

Gérard Berry's course in *Collège de France*

Promises are sometimes kept

Réforme Darcos: *Informatique et société numérique*: an optional course in *Seconde* (10th grade) (very few mandatory courses, but a lot of optional courses)

But ... no *réforme Darcos*

Réforme Chatel: *Informatique et sciences du numérique*: a speciality in *Terminale S* (12th grade with a scientific flavor)

Four (tiny) victories

2012: *Informatique et sciences du numérique* in *Terminale S*

10 000 students

2012: hiring of an *Inspecteur Général* who has a PhD in informatics

2013: *Informatique et sciences du numérique* in preparatory classes

2013: test of an option (mostly for *Terminales L* and *ES*) in the Academy of Montpellier (and possibly Versailles)

A battle still to be won

Hiring teachers with a relevant training

Currently: teachers of another topic (maths, physics, engineering, ...) and continued learning (from 10 to 300 hours)

Outside France

A very diverse situation (even in a single country where *Länders*, states, *cantons*, ... and even high-schools may have different policies)

In some places informatics continuously taught from the 80's

In others: teaching of informatics gradually been replaced by ICT in the 90's (text processing systems rather than `while` loop, then the web revolution) and a recent second enlightenment: UK, Switzerland, ...

II. What have we learnt?: what is informatics?

United (e.g.) physics

A very diverse science: mechanics, electromagnetism, quantum physics, statistical physics, atomic physic, astrophysics, biophysics, ...

Power games

But physicists always united when speaking about teaching physics in high-school

Informatics: a different situation

Informatics is what I do: data bases, lambda-calculus, ...

But more dangerous:

A science, nothing to do with technology: I cannot fix your computer (I cannot change a wheel), astronomy and telescopes

A part of mathematics (constructive, discrete, ...)

It exists only as an industry

It is not autonomous: always coupled with a mechanical system (e.g. in a car) ... complex systems

Uniting informatics

Informatics as structured around four concepts: language, information, machine, algorithm (do not change every morning)

Informatics has both scientific and technological aspects: answering questions (is the halting problem decidable?) and building objects (programs, machine, ...) with a purpose

Informatics is a branch of no other science or technology, but has interfaces with other sciences mathematics, physics, ... and is one of the many technologies used to build objects (e.g. aircraft)

Learning informatics in three steps

More unity needed

Robots: machines, process information (through sensors, ...), implement original algorithms (feed-back, analog to digital transformations, ...), led to (reactive) programming languages

Thus a part of informatics

Yet, some think robotics is not part of informatics

Interface with mechanics: an autonomous science

Same for numerical analysis

Broader numerical sciences (while informatics broad enough)

III. What have we learnt?: who are our enemies?

Do we have enemies?

Our arguments are sharp, but things evolve slowly

Motion in a highly viscous medium

Useful to know who makes it viscous and why (very different people and very different reasons)

Teachers of other sciences (e.g. math)

Why?: one minute for informatics is one minute less for the others

Strategy: informatics is the science of algorithms, there have been algorithms in math for ever (Euclide's algorithm, Gaussian elimination, Pascal's triangle, ...), hence informatics (i.e. algorithmic, i.e. the study of algorithms operating on mathematical objects) is a part of math

What about routing algorithms?

Teaching informatics will bring more students to all sciences

Those who believe they know what informatics is

My kid has no problems with informatics: he spends a lot (too much?) time on his computer

I can use Google and I needed no training

I use computer everyday and I hate the IT guys

Those that fight against sciences and technology in general

New topics in middle- and high-school: *Éducation civique, juridique et sociale*, *Morale laïque*, *Histoire des arts*, *Droit et grands enjeux du monde contemporain* + more history and geography

Room for everything except science and technology

Ideological base: science and technology are not the way to know the world, just another narration

Consequence: sciences and technology are for lower-class boys (engineers) Others should focus on higher forms of knowledge (social sciences, art, religion, ...)

Include informatics in curricula, but empty it from any scientific / technological content (remember *Réforme Darcos*)

Need for a unity of scientists and a defense of scientific values

Things change slowly but they change

Informatique et Sciences du numérique exists in *Terminale* and *Classes préparatoires*

The Academy of sciences published a report about teaching informatics at all levels

Industry is starting to consider the problem as central

Parents get worried about the illiteracy of their children

Many countries in the world are taking a similar direction

Many master students in informatics want to be teachers