Unconventional Invited Talks 2008

David Corne

Heriot-Watt University, Edinburgh, UK

dwcorne@macs.hw.ac.uk

Some Predictions for the Future of Optimisation Research

David Corne

Heriot-Watt University, Edinburgh, UK

dwcorne@macs.hw.ac.uk

Special issues on
Swarm Intelligence: The state of the art in theory and practice

A joint **call for papers**

Theoretical Computer Science (TCS) - Elsevier
Natural Computing (NACO) - Springer

The journals *Theoretical Computer Science* (TCS) (Section C) and *Natural Computing* (NACO) are focussed on the study of computing using resources occurring in nature as well as computing techniques that are inspired by nature. In this joint call for papers, the aim is to produce special issues of TCS and NACO that will reflect the state of the art, along with exciting new developments, in, respectively, theoretical issues and practical/empirical issues in swarm intelligence.

The scope of these joint special issues is broad, covering the latest theoretical and empirical research in the many established areas of swarm intelligence (including ant systems, particle swarm optimisation, foraging algorithms, stochastic diffusion search, and so forth), while welcoming newer developments, novel frameworks and synergies, and so on. The guest editors will welcome and quickly respond to informal questions about the scope.

We envision that some of the accepted papers will be suitable for either TCS or NACO, and the guest editors will therefore partition the accepted papers between the two journals in a suitable way that optimizes coherence. However, if authors have a strong preference for one or the other journal, we ask that you indicate this at the time of initial submission.

Please send submissions in PDF format (leaving wide margins) to any one of the guest editors via email. The guest editors will enlist the services of reviewers from both journal boards, and from others, as appropriate. Note that the final versions of accepted papers will be handled by the relevant journal, and prepared according to the instructions of that journal.

We will make every effort to provide notification of acceptance/rejection within fifteen weeks of submission.

Important dates:
Submission deadline: 15th September 2008
Expected Publication date: mid to late 2009

Important dates:
Submission deadline: 15th September 2008
Expected Publication date: mid to late 2009

Guest Editors:

Eric Bonabeau, Icosystem Corporation, Cambridge, MA, USA
eric@icosystem.com

David Corne, Heriot-Watt University, Edinburgh, UK
dwcorne@macs.hw.ac.uk

Riccardo Poli. University of Essex, UK
rpoli@essex.ac.uk

http://www.macs.hw.ac.uk/~dwcorne/siswarm.pdf

- About HW

Eighth oldest educational institution in UK (1821)

First to educate the working classes

First to allow women onto degree studies (1861)

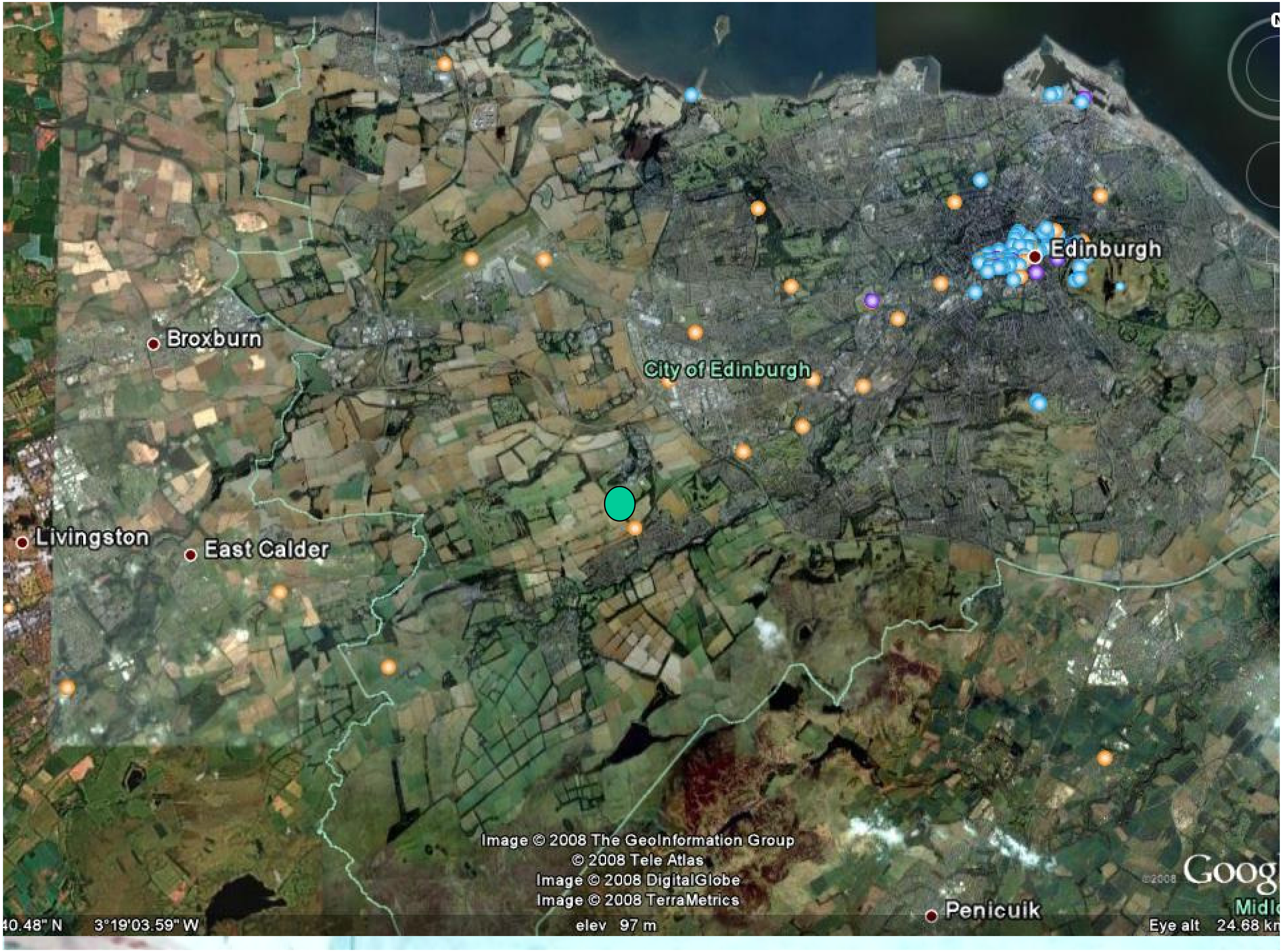
Fourth to employ me.

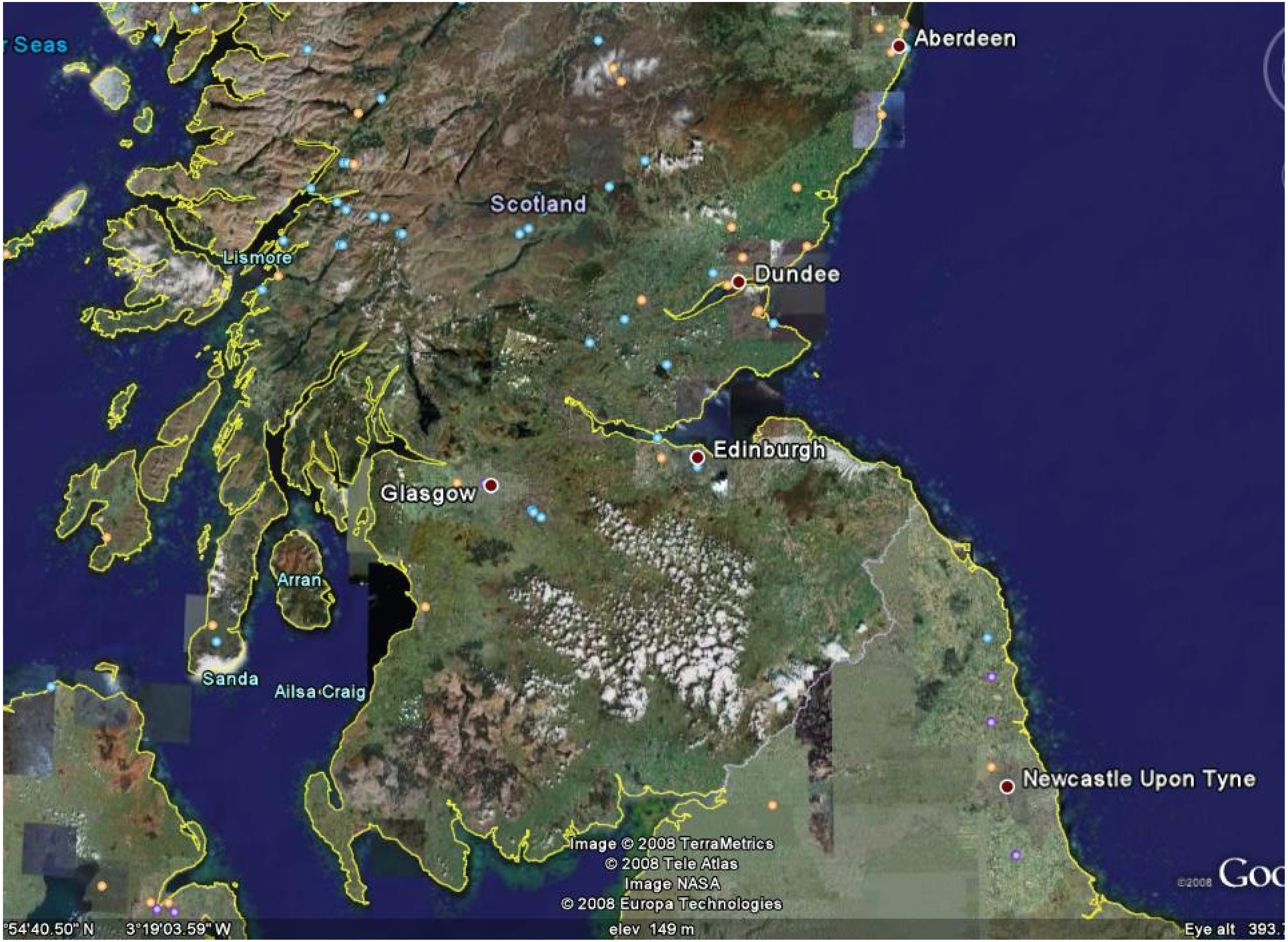
- About DC

Head of ISL (Intelligent Systems Lab) at Heriot-Watt

Aspects and applications of EC, multiobjective optimisation, bioalgorithmics, bioinformatics, data mining, design, applications in medicine, biosciences, logistics, telecomms, web intelligence


Hermiston
Riccarton
Image © 2008 The GeoInformation Group
© 2008 Tele Atlas
54'40.57" N
3°19'03.59" W
elev 93 m
Eye alt


Broxburn
City of Edinburgh
Edinburgh
Livingston
East Calder
Penicuik
Image © 2008 The GeoInformation Group
© 2008 Tele Atlas
Image © 2008 DigitalGlobe
Image © 2008 TerraMetrics
40.48" N
3°19'03.59" W
elev 97 m
Eye alt 24.68 km


Seas
Aberdeen
Scotland
Lismore
Dundee
Edinburgh
Glasgow
Arran
Sanda
Ailsa Craig
Newcastle Upon Tyne
Image © 2008 TerraMetrics
© 2008 Tele Atlas
Image NASA
© 2008 Europa Technologies
°54'40.50" N
3°19'03.59" W
elev 149 m
Eye alt 393.

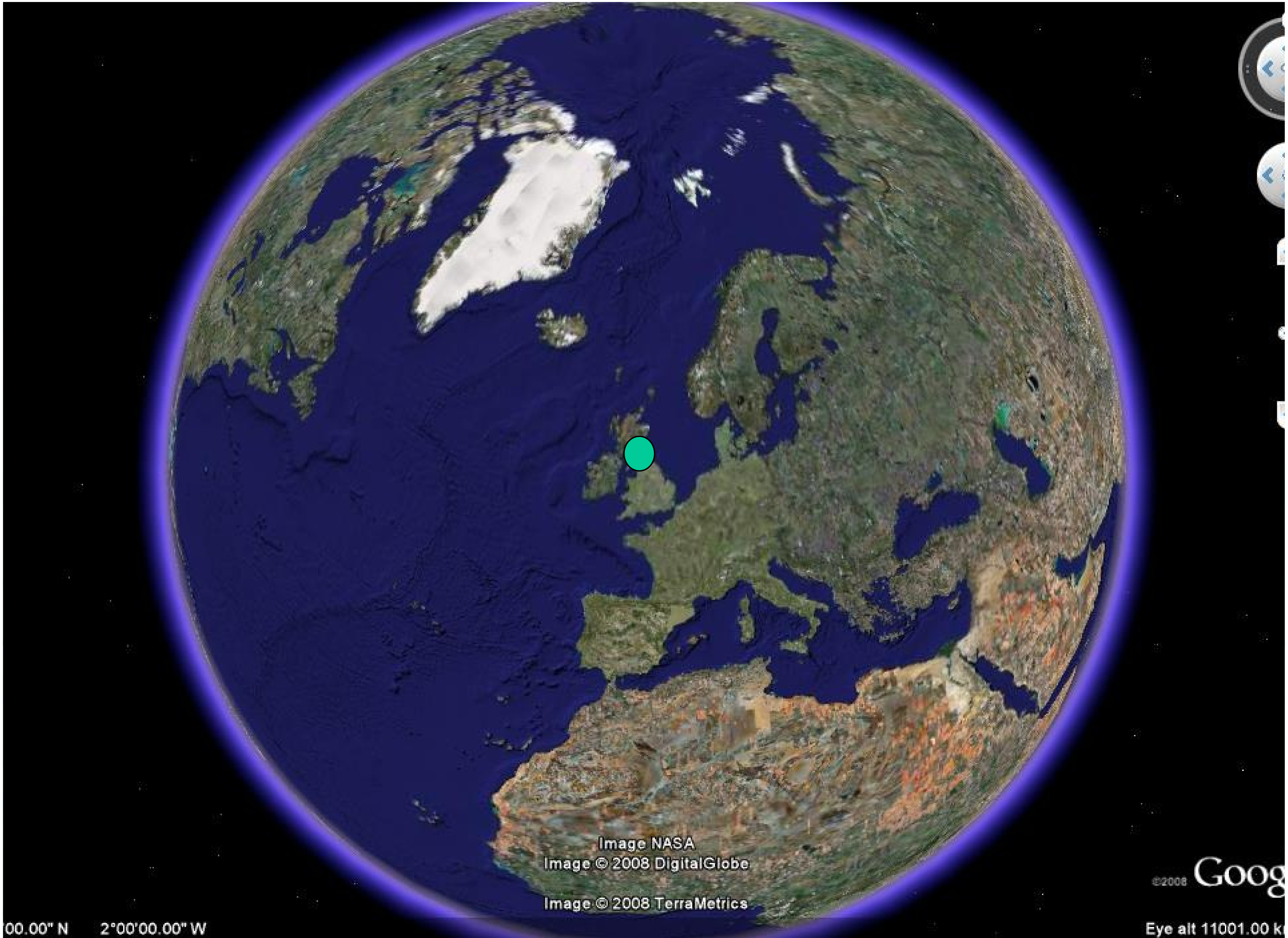
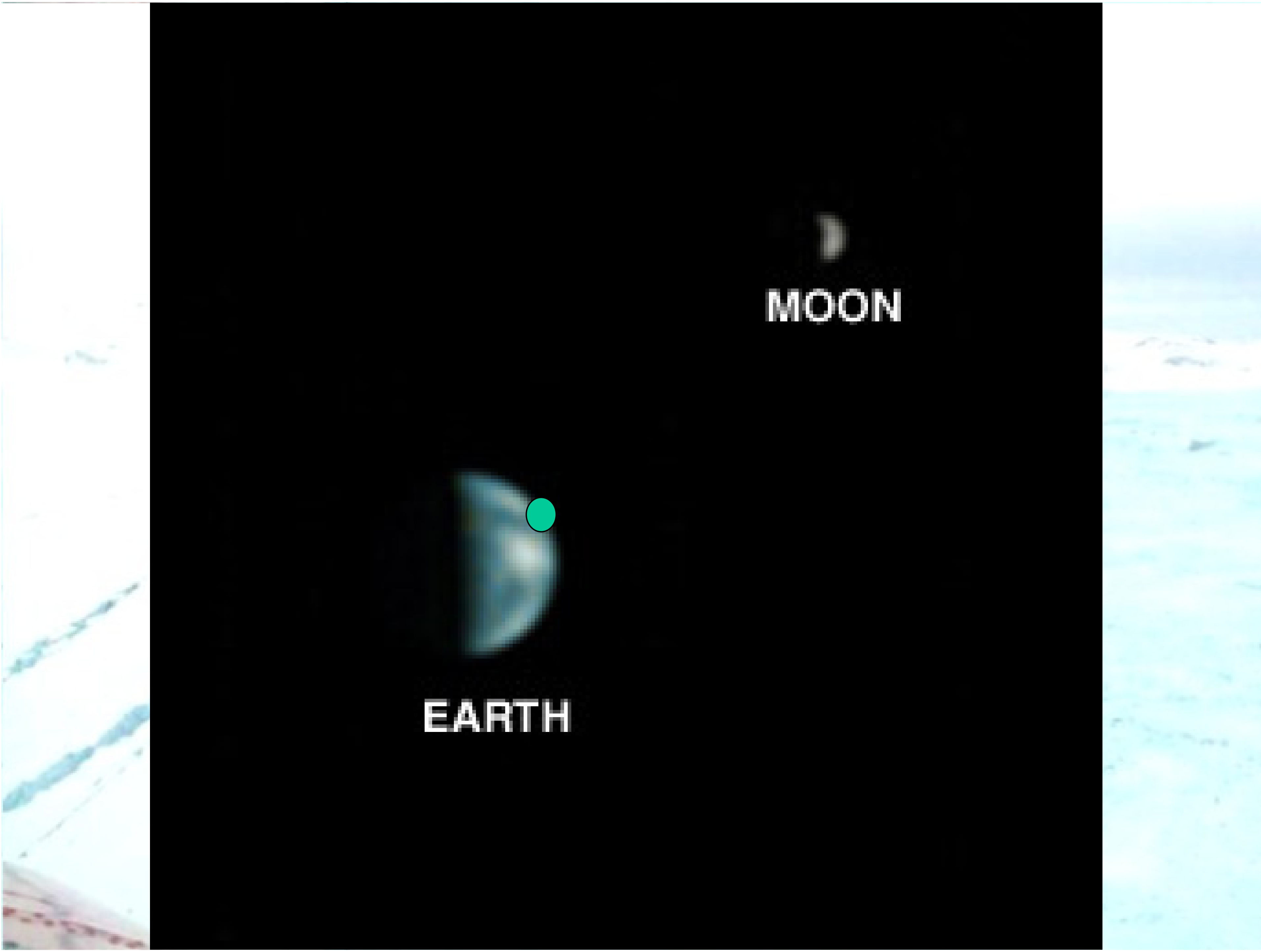


Image NASA
Image © 2008 DigitalGlobe
Image © 2008 TerraMetrics
©2008 Goog
'00.00" N 2°00'00.00" W
Eye alt 11001.00 k


MOON
EARTH

What is my hidden agenda?

Many years ago, Evolutionary Computation conferences (and much of operations research) was all about ***single objective optimisation***

Single objective optimisation was:

- the norm; unquestioned;
- a problem was often formulated in a true many-objective from, then authors/speakers would say "*so, in order to optimise this, we combine these into a scalar objective function like this …*" … and that would be accepted without question.
- tied up with all the machinery being studied in relation to selection methods, EA theory in general.

But now we know this:

Single objective optimisation is:

- wrong/inappropriate; doesn't address the real problem

- biassed and suboptimal

- still researched by much of the evolutionary computation community

What is my other hidden agenda?

In the existing body of published work:

There are *striking* amounts of:

- reinvention of wheels
- ignorance about related literature
- ignorance about what `related' means

And consequently:

- wasted time, wasted intellectual effort
- too slow progress

Of course, none of this is true for UC delegates.

Let’s consider a specific family of optimisation algorithms …

Particle Swarm Optimisation

1. Initialise: generate set of candidate vectors, each has a *position* from some specified distribution, and each has a *velocity*.
2. evaluate them (the *positions*)
3. Each vector updates its position, influenced by:
    1. its **velocity** -- to some extent ***v***
    2. its ***personal best*** position -- to extent ***c*1**
    3. Its ***neighbourhood best*** position – to extent **c2**
4. Return to 2

Optimisation Research

1. Initialise: each individual is a scientist or group; its ***position*** is its latest **piece of research;** its **velocity** is a measure of how much it favours incremental research, exploraty research, etc.
2. Evaluation: ***reviews, impact, citations, esteem***
3. Each individual updates its position (next piece of research), influenced by:
   1. its **velocity** -- to some extent ***v***
   2. its ***personal best*** position -- whatever has seemed to generate most success for this individual
   3. Its ***neighbourhood best*** position – the individual's **assessment of and insights from the state of research in their cloud of fields**
4. Return to 2

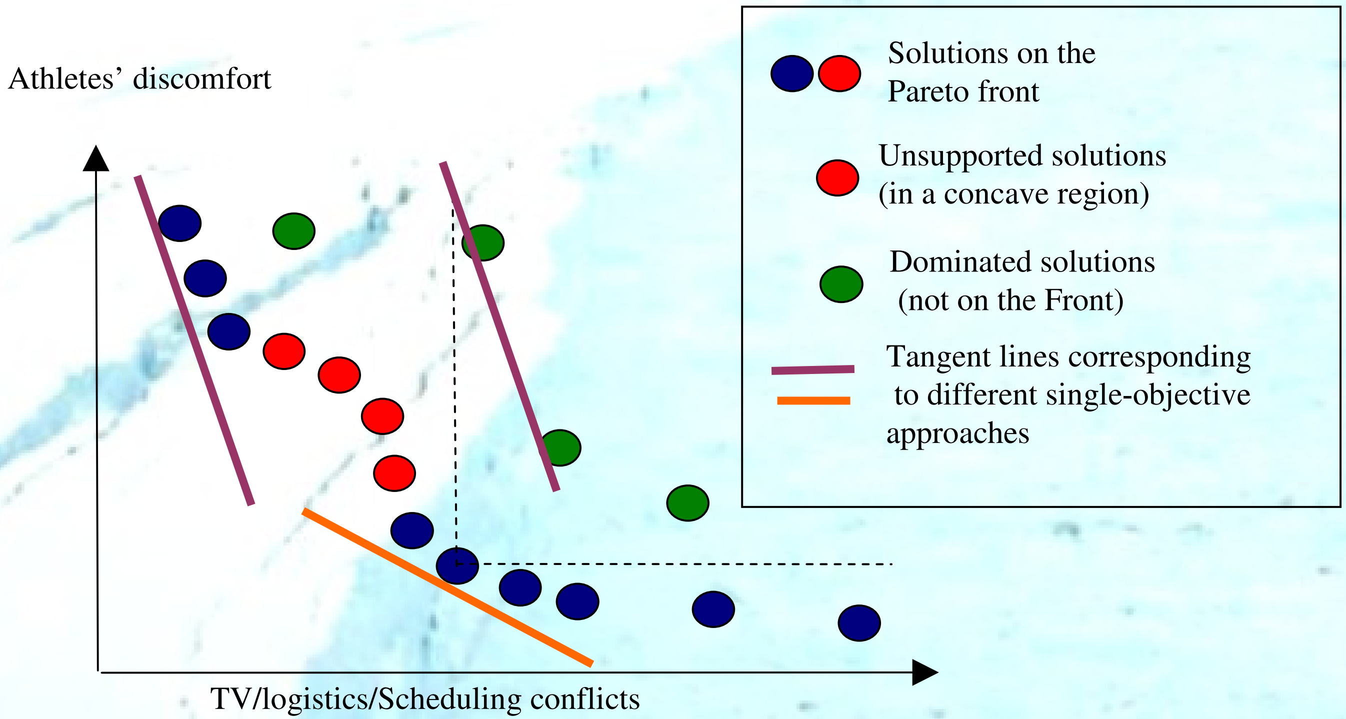
In the world of evolutionary algorithms we know from both theory and (*lots*) of practice that:

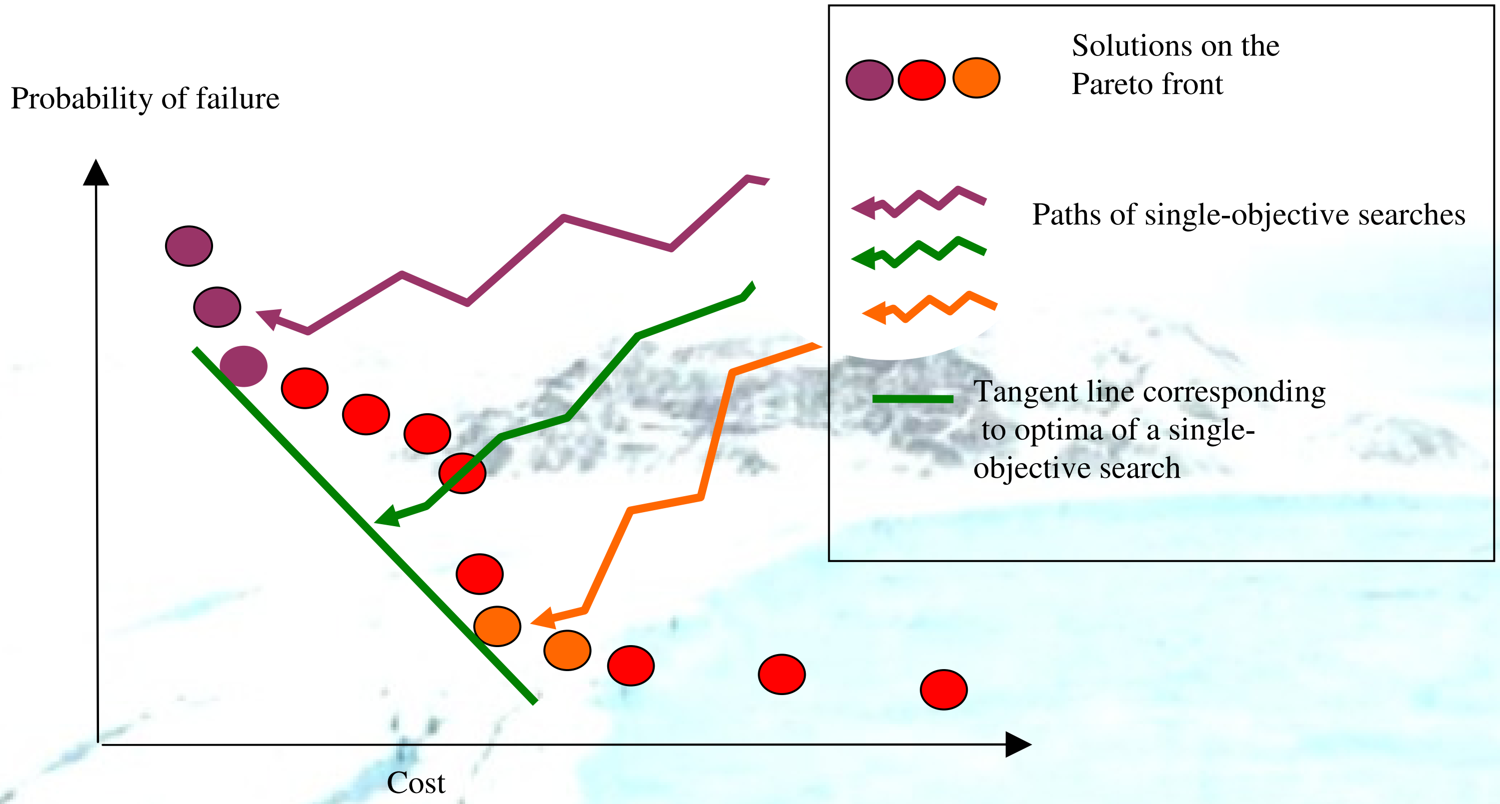
1. Low velocity means slow progress
2. Over-influence of “personal best” means slow overall progress
3. These two combined lead to stagnation in many poor local optima.

But this is exactly what happens in the “science” algorithm

A Very Very Brief Multiobjective Optimisation Primer

[DOC] The good of the many outweighs the good of the one: evolutionary multi-objective optimization
D Corne, K Deb, PJ Fleming - IEEE Connections Newsletter 1 (1) - dbkgroup.org
The **Good** of the **Many Outweighs** the **Good** of the One: Evolutionary Multi-Objective Optimization. David W. Corne, University of Reading, UK. **...**
Cited by 13 - Related Articles - View as HTML - Web Search





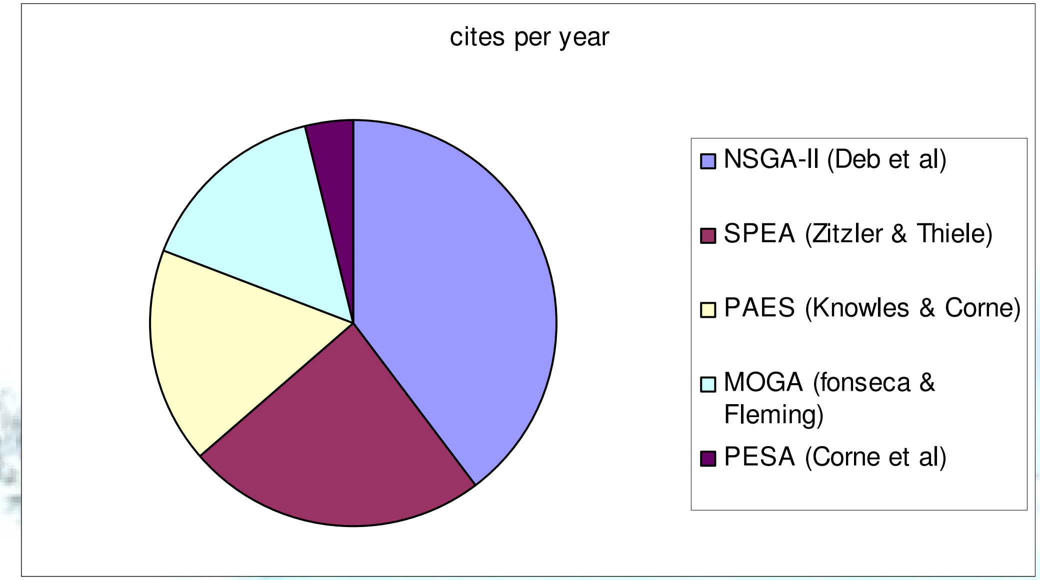
Proper **Multiobjective** search:

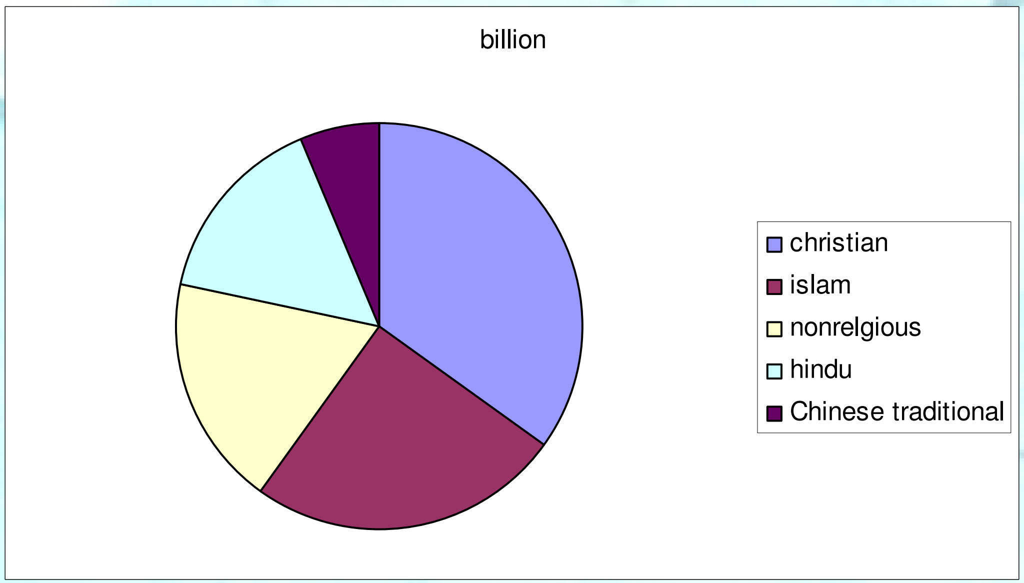
- tries to find the Pareto front (a set of solutions, not a single ‘best’)
- gives the problem solver what s/he wants (and much more)
- typically performs at least as good as SO on the SO criterion
- can now be done efficiently with well-known algorithms
- so, now we solve the <u>‘real’ problem</u>, not a <u>simplification</u> of it

Single objective optimization is a *crime*

Some of the prominent \`simple'EMO algorithms google scholar mean cites per year (NSGA-II = 212)

(*there are lots more*)





The five most prominent religions, from http://www.adherents.com/Religions_By_Adherents.html (Christianity = 2.1bn)

Past

Formulate a scalar (single objective) function that represents solution quality, and optimise this for a problem instance

- This formulation is an over-simplifcation that prevents solution of the real problem

F(design) = cost + mass + risk + …

Present

Formulate a vector (multiple objective) of scalar functions each for a different quality objective, and aim for the Pareto set of solutions for a problem instance

F(design) = (cost, mass, risk)

Present

Formulate a vector (multiple objective) of scalar functions each for a different quality objective, and aim for the Pareto set of solutions for a problem instance

- This formulation is an over-simplifcation that prevents solution of the real problem,
- **and** our algorithms are too dumb.
  - *we still usually specify too few objectives*
  - *we throw away immense amounts of sampled information that could help solve this instance **and others***

Future

Formulate a (often *many*) component vector of objectives, and search for a useful model (covering a distribution of problem instances) that links design space to objective space

- *solve the `whole' problem*
- *use much cleverer (and more elegant) algorithms that `combine' evolution and learning*
- *in the same process, produce* algorithms *that can quickly solve many instances*

Present

Future

• *we still usually specify too few objectives*
• *we throw away immense amounts of sampled information that could help solve this instance* ***and others***

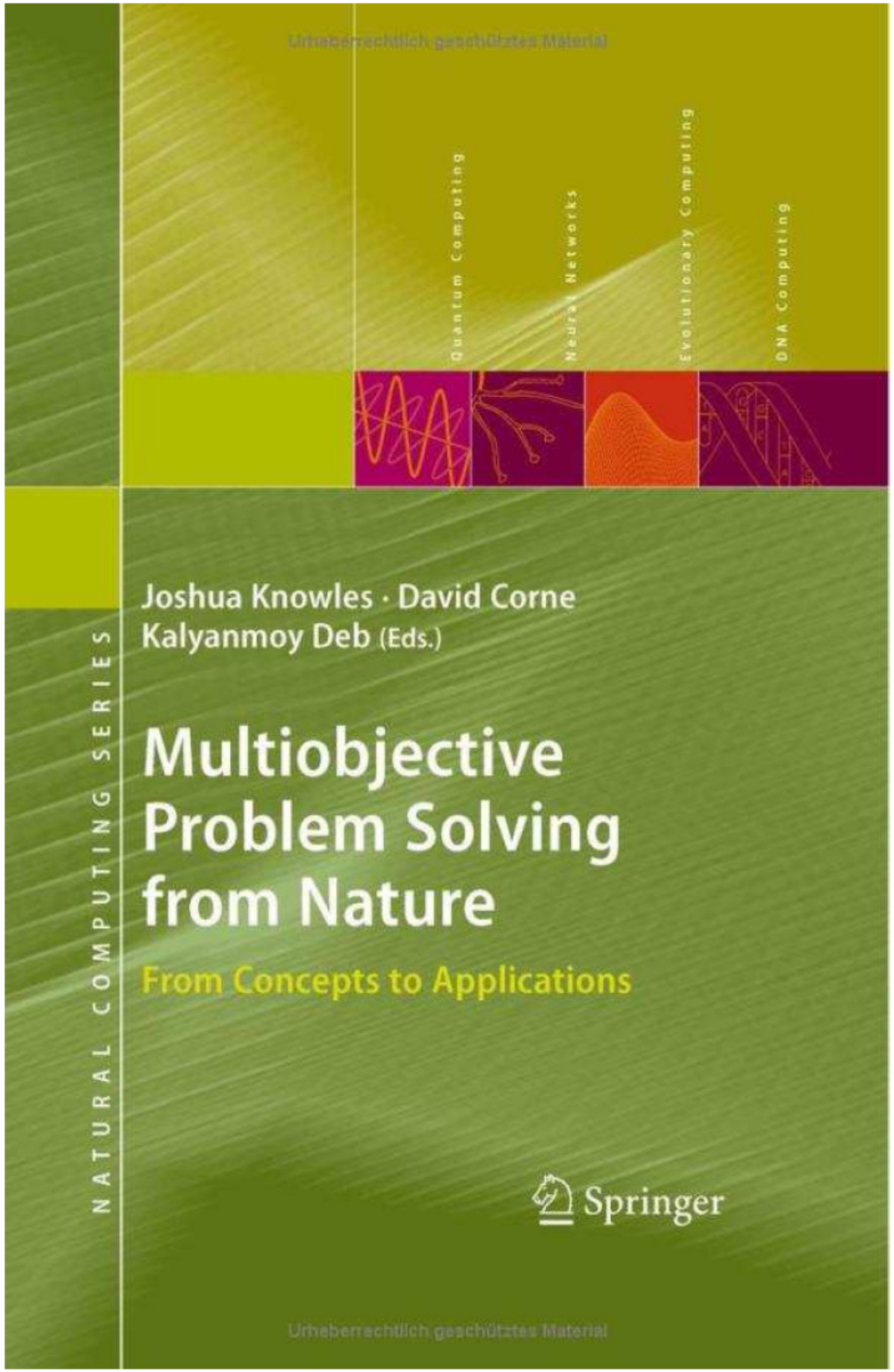
• *solve the `whole' problem*

• *use much cleverer (and more elegant) algorithms that `combine' evolution and learning*

• *in the same process, produce* algorithms *that can quickly solve many instances*

Just as was the case with the "past→present" transition, this transition is possible because <u>we are beginning to discover methods that can do these things well</u>.

Alternative views of the future



18 chapters concerned with using multiobjective techniques for much more than optimisation. E.g.

- *preventing bloat in GP*
- *promoting understandable rules*
- *handling constraints*
- *discovering design principles*
- *single objective optimisation*(!)

…

I could say more, but I prefer it if you simply buy the book

Kalyan Deb's view?

Muiltiobjective Optimization Using Nondominated Sorting in Genetic Algorithms - all 15 versions »
N Srinivas, K **Deb** - Evolutionary Computation, 1994 - MIT Press
Abstract In trylng to solve multiobjective optimization problems, many traditional methods scalar- ize the objective vector into a single objective. In those cases, the obtained solution is highly sensitive to the weight vector **...**
Cited by 1000 - Related Articles - Web Search

NSGA

A fast and elitist multiobjective genetic algorithm: NSGA-II - all 11 versions »
K **Deb**, A Pratap, S Agarwal, T Meyarivan - Evolutionary Computation, IEEE Transactions on, 2002 - ieeexplore.ieee.org
Abstract—Multiobjective evolutionary algorithms (EAs) that use nondominated sorting and sharing have been criti- cized mainly for their: 1) (3) computational complexity (where is the number of objectives and is the **...**
Cited by 1270 - Related Articles - Web Search - BL Direct

NSGA-III

Present

- *we still usually specify too few objectives*
- *we throw away immense amounts of sampled information that could help solve this instance* ***and others***

Future

- *solve the `whole' problem*
- *use much cleverer (and more elegant) algorithms that `combine' evolution and learning*
- *in the same process, produce* algorithms *that can quickly solve many instances*

Just as was the case with the "past→present" transition, this transition is possible because ***we are beginning to discover methods that can do these things well***.

First, Briefly: Many-Objective Methods

Many-O is problematic; many of the traditional EMO methods don't scale well from 2—20.

A good source for showing this is:

On the Evolutionary Optimization of Many Conflicting Objectives
RC Purshouse, PJ Fleming - Evolutionary Computation, IEEE Transactions on, 2007 - ieeexplore.ieee.org
Abstract—This study explores the utility of multiobjective evolu- tionary algorithms (using standard Pareto ranking and diversity- promoting selection mechanisms) for solving optimization tasks with many conflicting ...
Related Articles - Web Search - BL Direct

Approaches include:

- **Simplification** Treat 30-objectives as 2 or 3 (say)
- **Exploit information** (e.g. identify the correlations, study the dominance graph, etc…)

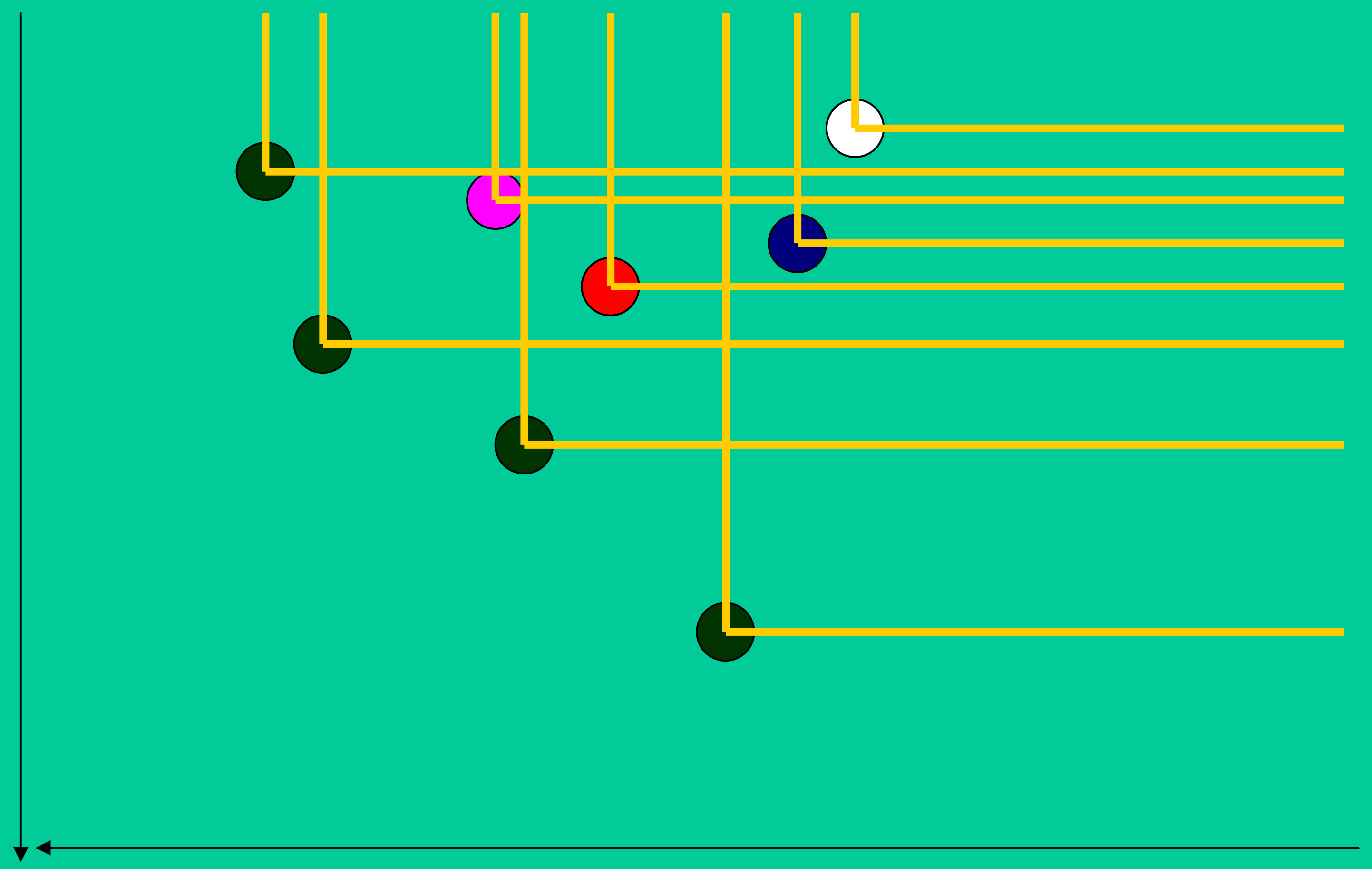
Dimensionality Reduction in Multiobjective Optimization: The Minimu
Problem - all 2 versions »
D Brockhoff, E **Zitzler** - Proc. of Operations Research, 2006 - Springer
Summary. The number of objectives in a multiobjective optimization problei strongly influences both the performance of generating methods and the de making process in general. On the one hand, with more objectives, more ...
Cited by 3 - Related Articles - Web Search

Non-linear Dimensionality Reduction Procedures for Certain Lar
objective …
DK Saxena, **K Deb** - Proceedings of the 4th International Conference on …, 2
Abstract. In our recent publication [1], we began with an understanding that many real-world applications of multi-objective optimization involve a large num- ber (10 or more) of objectives but then, existing evolutionary multi- ...
Cited by 4 - Related Articles - Web Search - BL Direct

- **New/better selection methods**

Techniques for highly multiobjective optimisation: some nondominated points are better than others - all 2 versions »
DW Corne, JD Knowles - Proceedings of the 9th annual conference on Genetic and …, 2007 - portal.acm
ABSTRACT The research area of evolutionary multiobjective optimization (EMO) is reaching better understandings of the properties and capabilities of EMO algorithms, and accumulating much evidence of their worth in practical ...
Cited by 5 - Related Articles - Web Search

Why many-objectives is hard



In few-o, we can rank points according to domination-counts

Quantifying the Effects of Objective Space Dimension in Evolutionary Multiobjective Optimization

J Knowles, D Corne - Proc. 4th Int. Conf. Evol. Multi-Criterion Optim.(EMO 2007) - Springer

... Page 3. **Quantifying** the **Effects** of **Objective Space Dimension** 759 ... Page 5. **Quantifying** the **Effects** of **Objective Space Dimension** 761 ...

Cited by 1 - Related Articles - Web Search - BL Direct

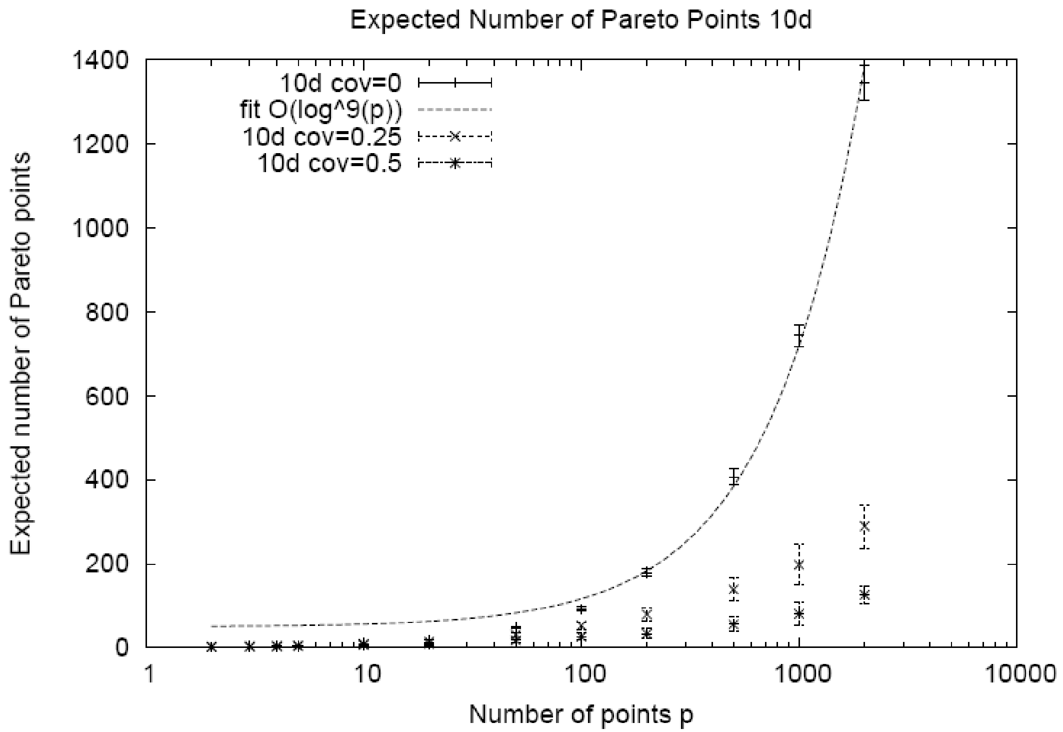
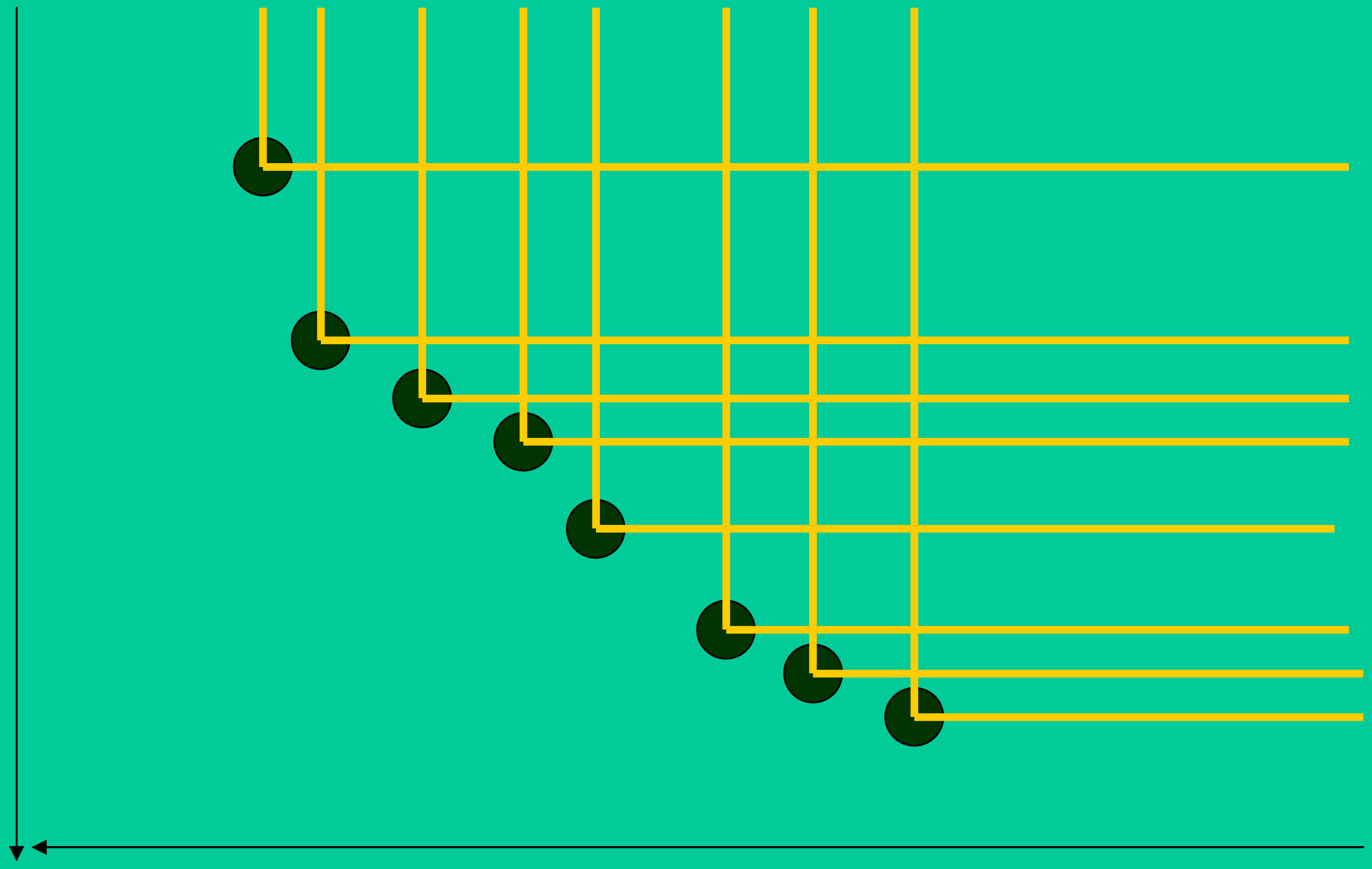


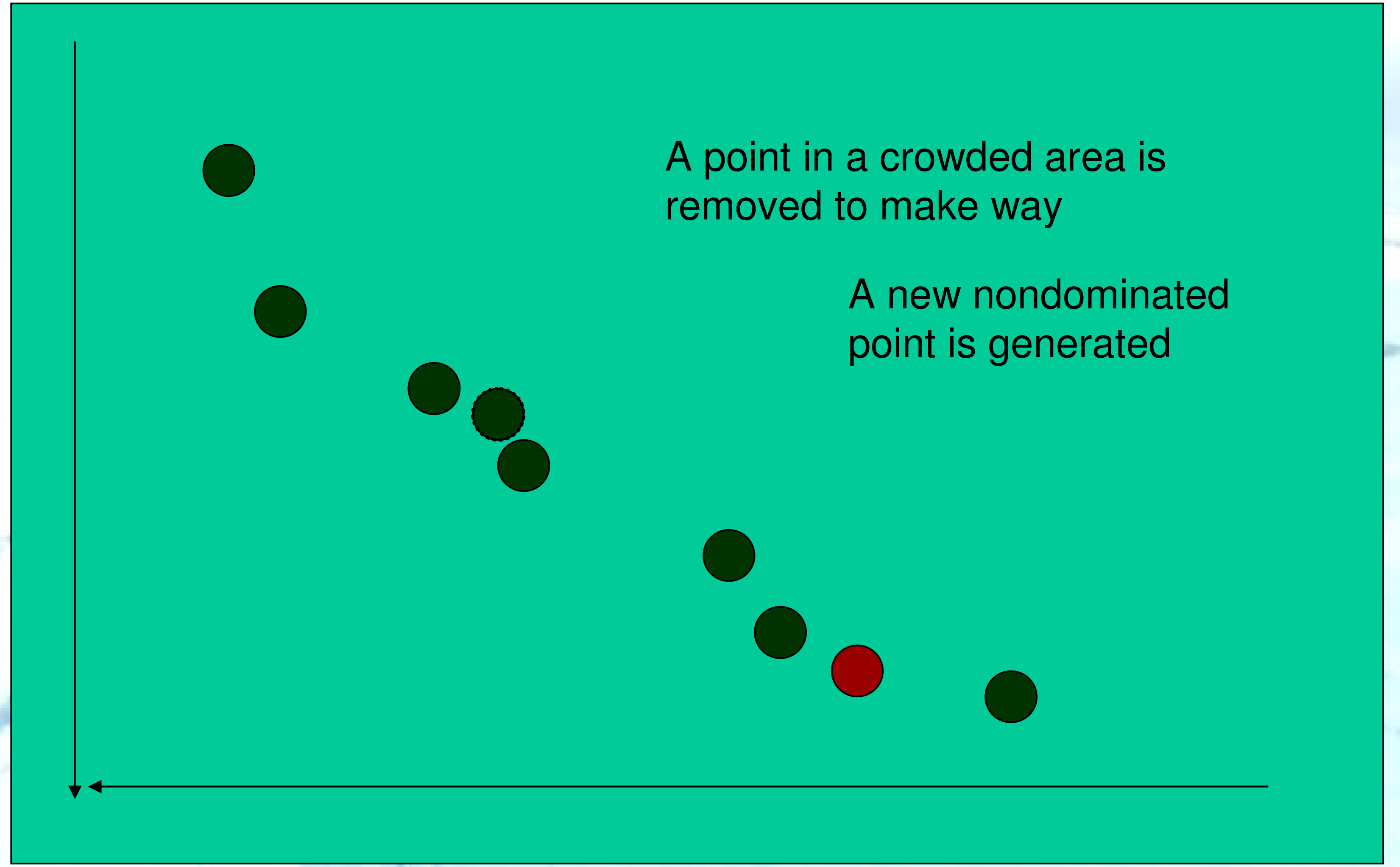
Fig. 1. Empirical distributions of the number of internally nondominated points in a sample of p points for 5 and 10 objectives and three correlations arising from the use of different covariance matrices. For a correlation of 0.0, a curve of $O(\ln^{d-1}(p))$ has been fitted through the largest four values, using least squares estimation

In many-o populations, it's hard to find any dominated points

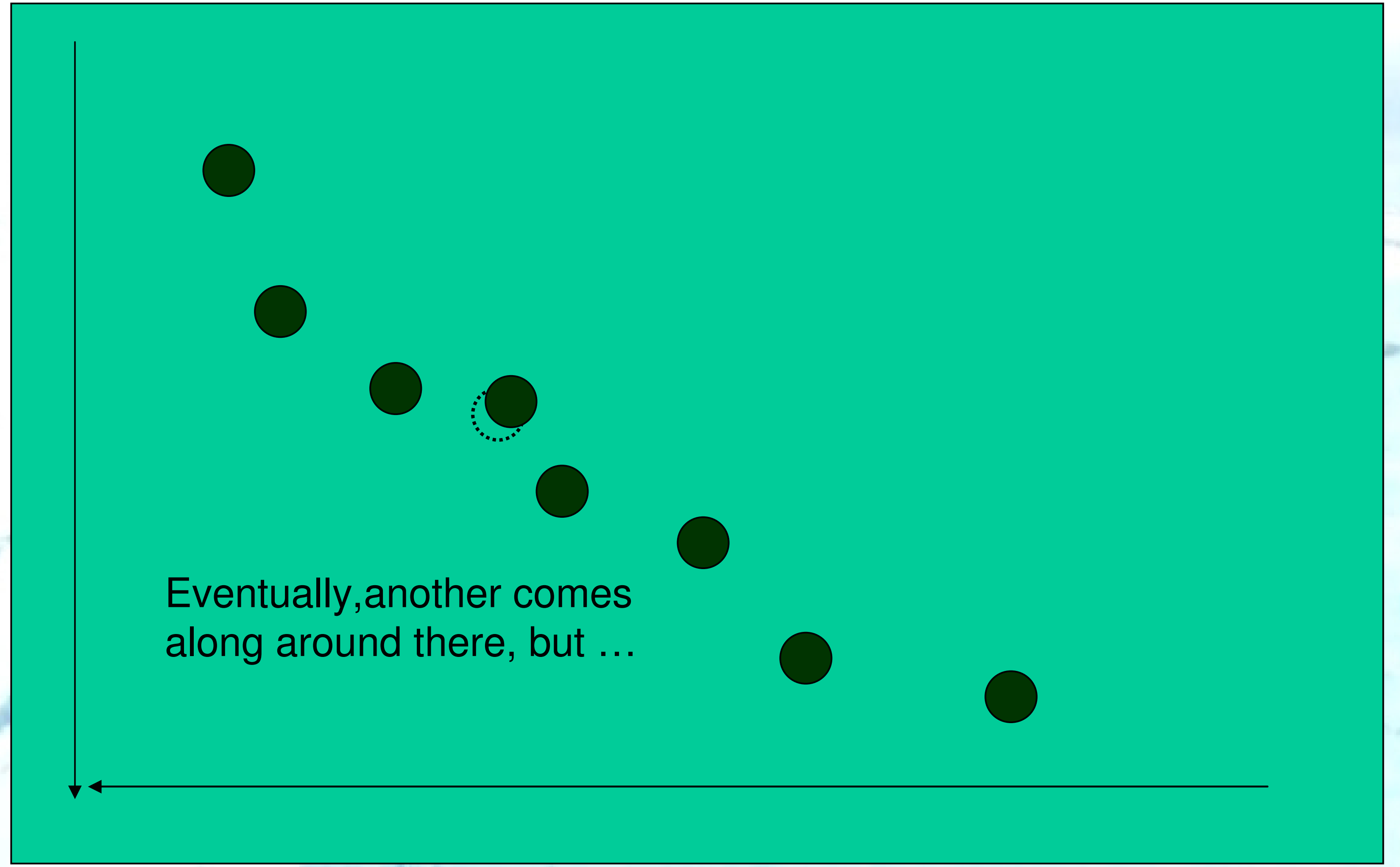


But in many-o, most or all points are non-dominated

Suppose we have an archive size of 8 …



Suppose we have an archive size of 8 …



The problems are:

- A. In many-o, the proportion of nondominated points is generally high, so we have little or nothing to favour one point over another
- B. `Fitness deterioration' (as coined by Hanne) can occur, when archive size is fixed. Also this is one of the two of reasons why there is FL in MO.
- C. Some modern EMOs simply don't scale to many-o since they need data or time that grows quickly with o.

Some progress on many-o – The ARF selection method

Objectives /Correlation	Rank-Ordering (Best … Worst)
TSP 5 / -40	ARF, FR, KO, RF, RR, SO, SR
TSP 5 / -20	ARF, FR, KO, RF, RR, SO, SR
TSP 5 / 0	ARF, KO, FR, RF, RR, SO, SR
TSP 5 / 20	ARF, KO, FR, RF, RR, SO, SR
TSP 5 / 40	ARF, KO, FR, RF, RR, SO, SR
TSP 10 /-40	ARF, KO, RF, RR, FR, SR, SO
TSP 10 /-20	ARF, KO, RF, RR, FR, SR, SO
TSP 10 / 0	ARF, KO, RF, RR, FR, SR, SO
TSP 10 / 20	ARF, KO, RF, RR, FR, SO, SR
TSP 10 / 40	ARF, KO, RF, RR, FR, SR, SO

Best results were obtained by AR for 5-10 objectives.

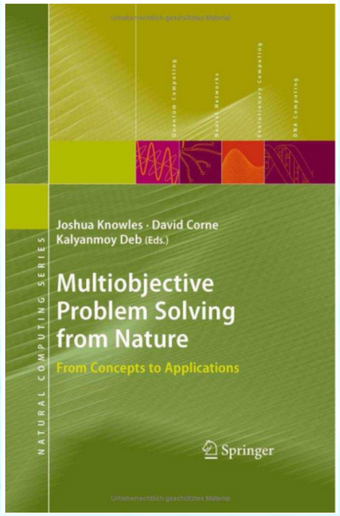
and more objectives … note how the alternatives are often no better than random (RR)

Objectives/ Correlation	Rank-Ordering (Best … Worst)
TSP 15 / -40	ARF, {all others equally rated}
TSP 15 / -20	ARF, {all others equally rated}
TSP 15 / 0	ARF, KO=RF, FR, SR = SO= RR
TSP 15 / 20	ARF, KO=RF, RR=FR, SR=SO
TSP 15 / 40	ARF, KO, RF=RR, FR, SR=SO
TSP 20 / -40	SO=RR=SR, RF, ARF, KO=FR
TSP 20 / -20	ARF, NSO=RR=SR, RF, KO=FR
TSP 20 / 0	ARF, KO=RF=RR, FR=SO=SR
TSP 20 / 20	ARF, KO= RF=RR, FR, SO=SR
TSP 20 / 40	ARF, KO=RF=RR, FR, SO=SR

Won’t say more

Have slides if time left at the end.

Also see Evan Hughes’ chapter in



Interim summary of the many-o bit

- We still simplify problems from many-o to few-o
- This is because we are still trying to find out how best to deal with many-o
- Good such methods are arriving, so in the future we will be avoiding this simplification.

Present

- *we still usually specify too few objectives*
- *we throw away immense amounts of sampled information that could help solve this instance*

Future

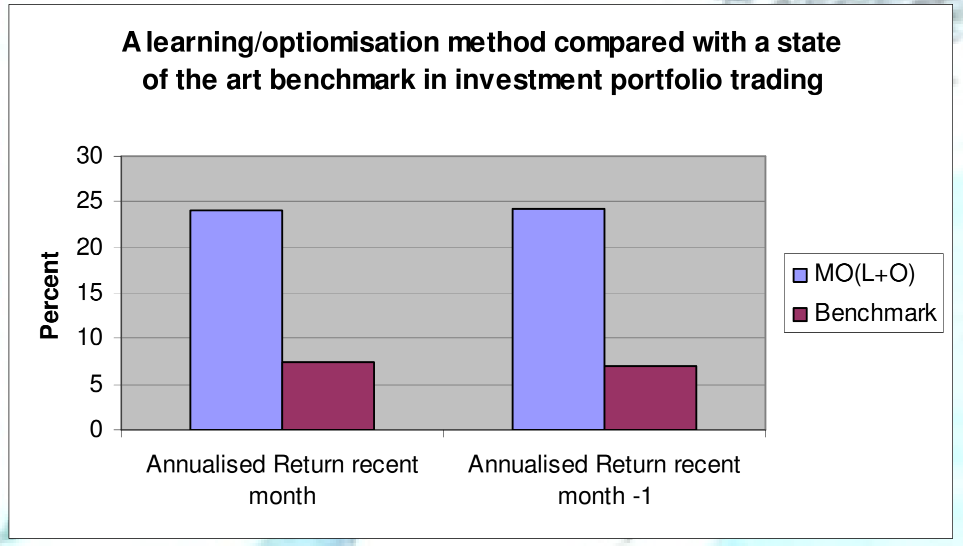
- *solve the `whole' problem*
- *use much cleverer (and more elegant) algorithms that `combine' evolution and learning*
- *in the same process, produce* algorithms *that can quickly solve many instances*

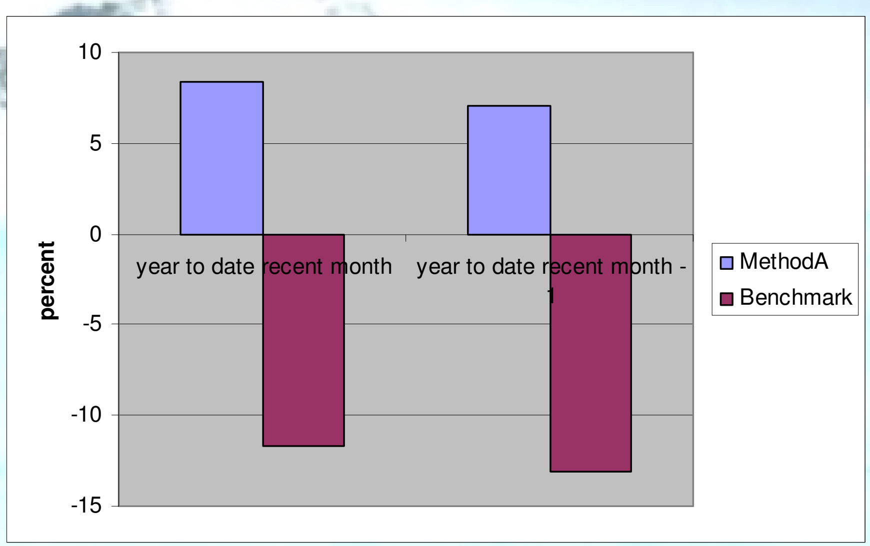
Just as was the case with the "past→present" transition, this transition is possible because <u>we are beginning to discover methods that can do these things well</u>.

The driver: large-scale and important problems

- Some problems are important
- and fitness computation is expensive
- often *very* expensive
- L+O combinations typically achieve significant savings in number of fitness evaluations required
- often *very* significant
- So ManyO(L+O) is clearly the future … (?)

Health, Wealth and Happiness



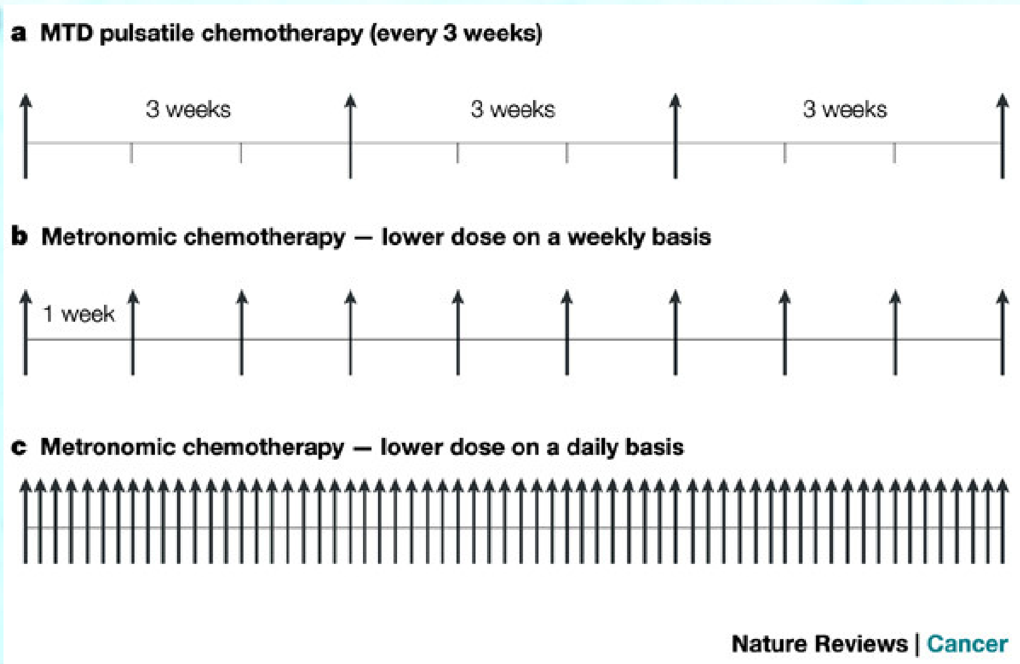


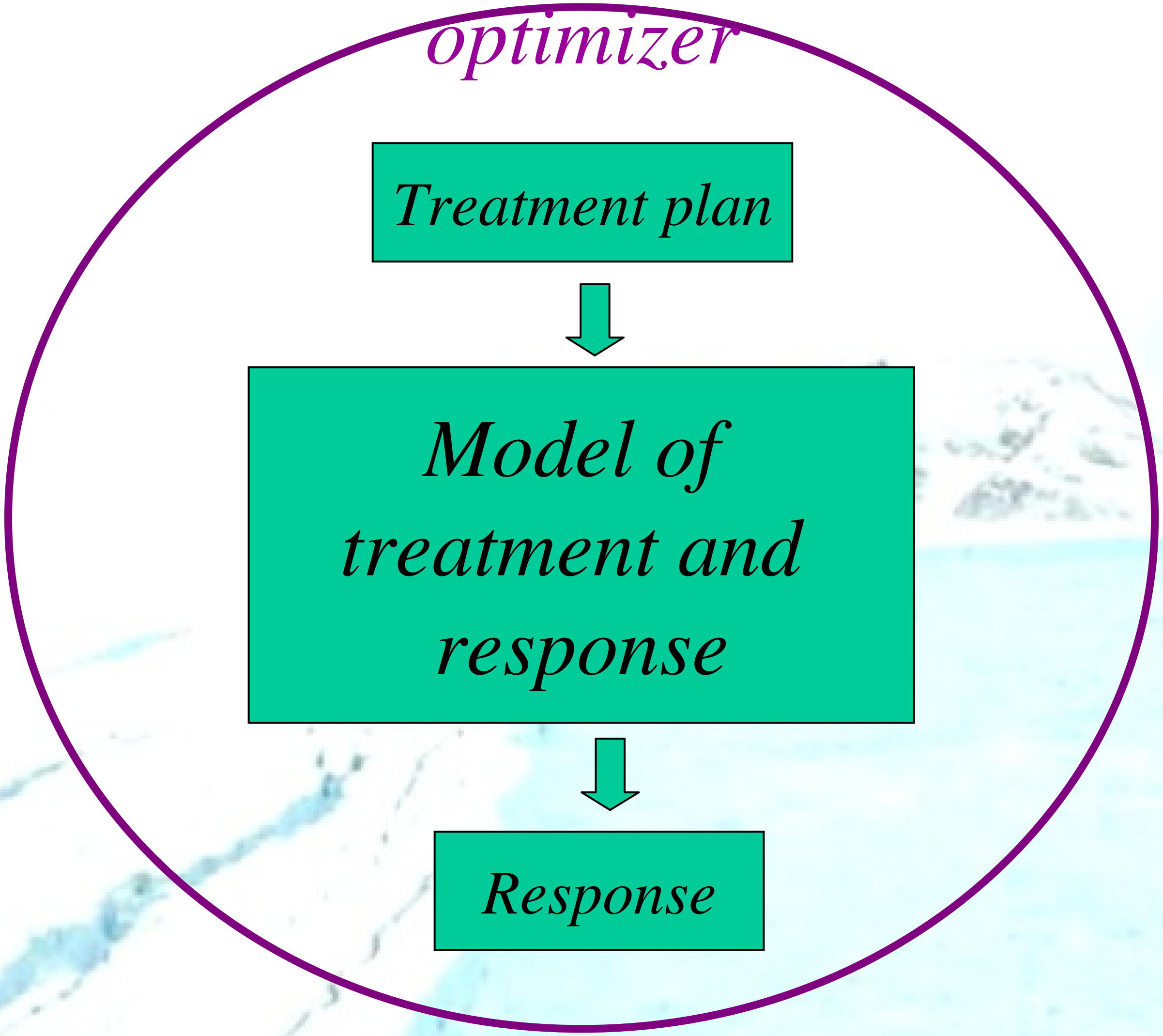
Vaguely, a many-o L+O approach to a problem attracting a large amount of investment, with quite expensive fitness function

Optimisation of medical treatment

Health

E.g. Chemotherapy treatment schedule optimisation





Response = f(Treatment plan)

So we can optimise over the space of treatment plans;

The models can be very compute-intense, e.g. ~1hr

But clinical evaluation of a treatment plan can take years …

HAART Therapy schedule

Cellular Automation model of HIV infection and HAART Therapy response

- *delay onset of AIDs*
- *minimal side effects*

[Evolving Novel and Effective Treatment Plans in the Context of Infectior ...]

R Haines, D **Corne** - LECTURE NOTES IN COMPUTER SCIENCE, 2006 - Springer
... continuous therapy. We Page 2. 414 **R**. **Haines** and D. **Corne** fiind that, insofar **...** ir capturing Page 4. 416 **R**. **Haines** and D. **Corne** the multi-timescale **...**

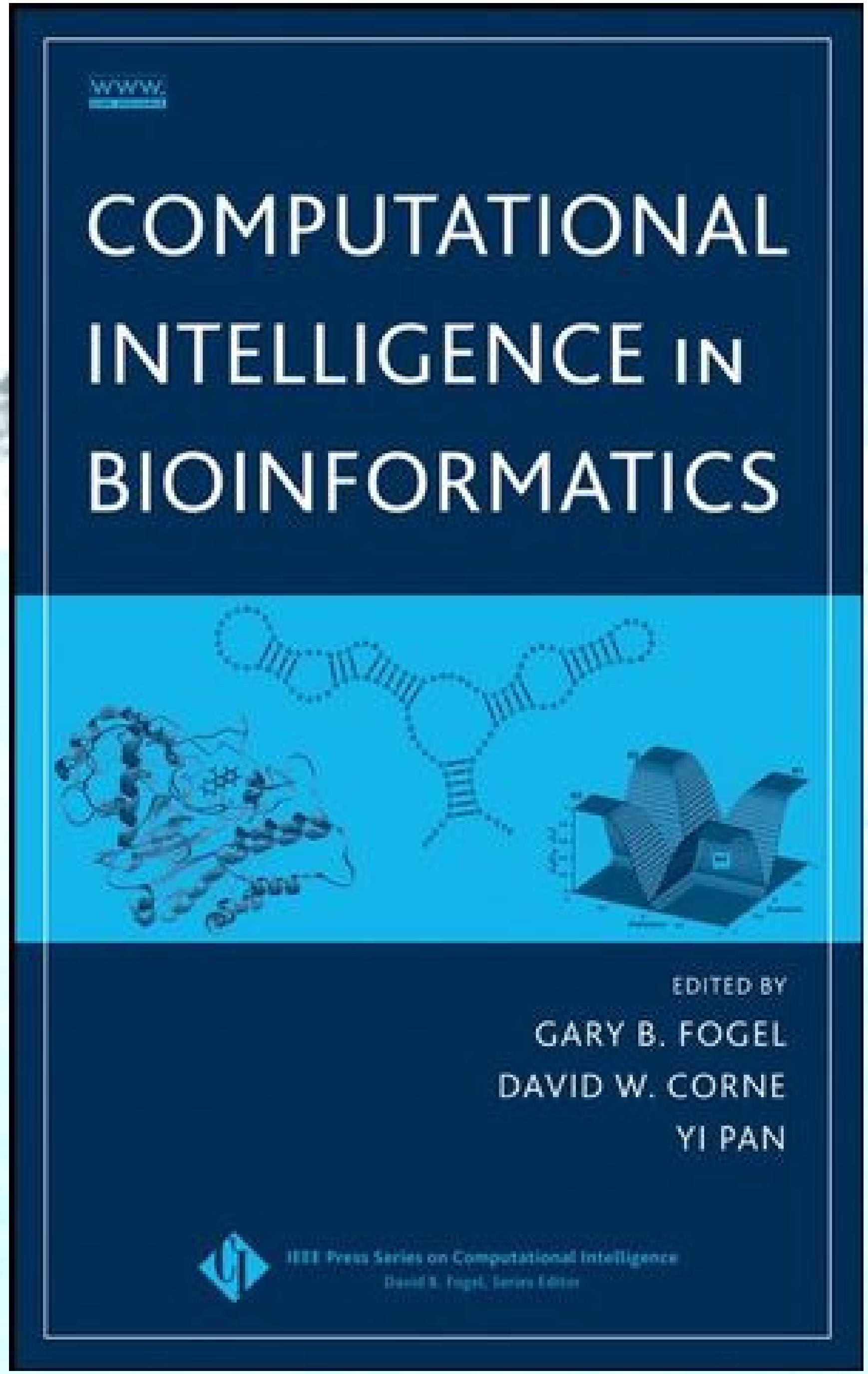
Related Articles - Web Search - BL Direct

Chemotherapy Dose Schedule

Differential equation models of tumour response and toxic effects

- *reduce tumour size*
- *minimal side effects*

McCall, Petrovski, Shakya (2007) Evolutionary algorithms for cancer chemotherapy optimization, in:



Radiotherapy treatment plan
(beam angles, widths, collimation, …)

Monte Carlo simulation using RTP system based on individual patient Scan data

- *Dose to target volume*
- *Organs At Risk stats*

Multiobjective **evolutionary optimization** of the number of beams, their orientations and weights for … - all 7 versions »
E Schreibmann, M Lahanas, L Xing, D Baltas - Physics in Medicine and Biology, 2004 - iop.org
... Abstract We propose a hybrid multiobjective (MO) **evolutionary optimization algorithm** (MOEA) for intensity-modulated **radiotherapy** inverse planning and apply it ...
Cited by 31 - Related Articles - Web Search - BL Direct

Evolutionary drug scheduling models with different toxicity metabolism in cancer chemotherapy

Yong Liang[a,*], Kwong-Sak Leung[b], Tony Shu Kam Mok[c]

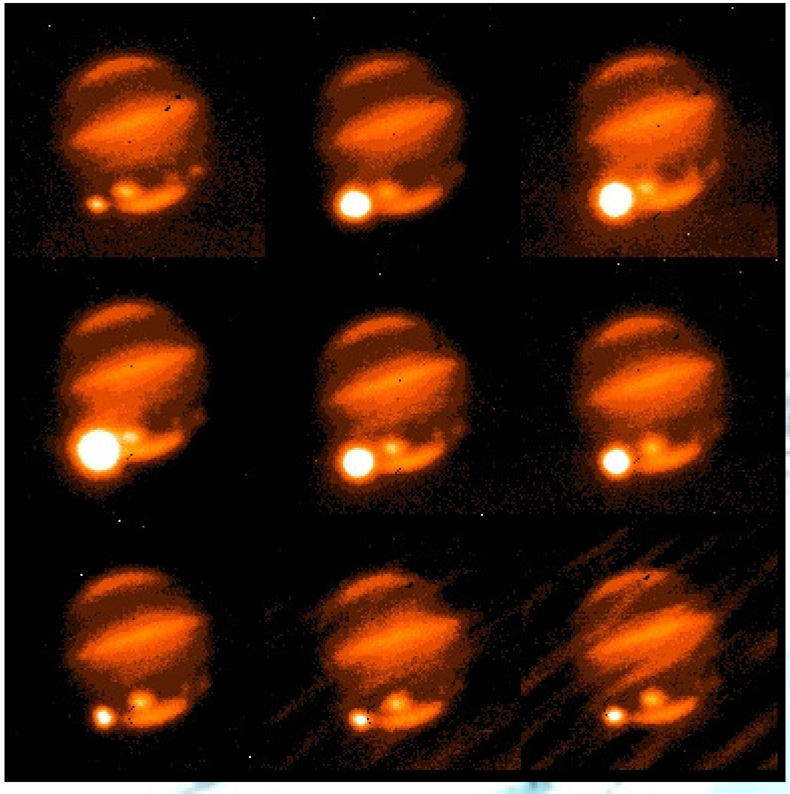
Applied Soft Computing

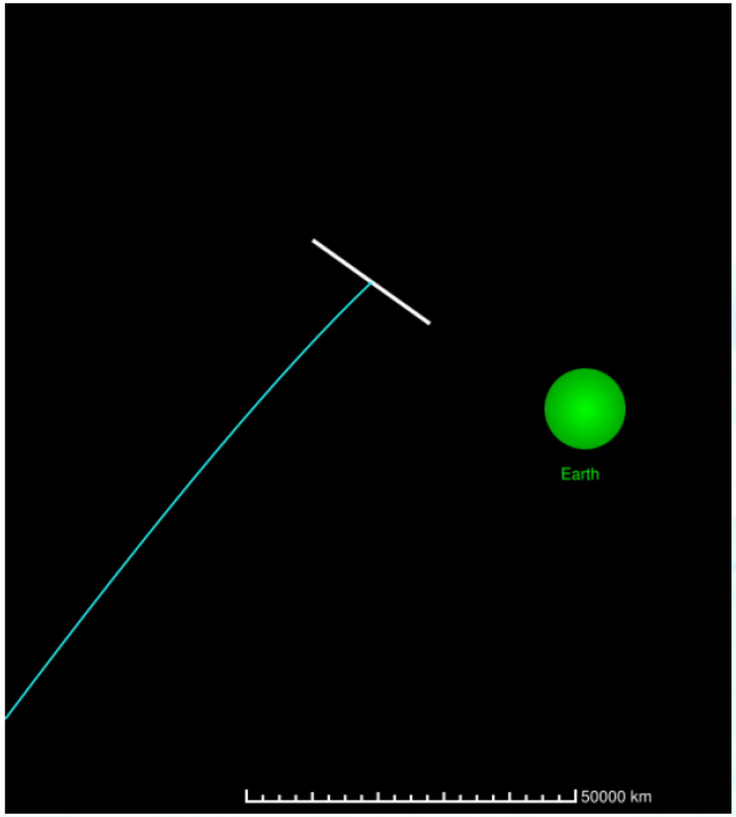
Target: $N = 2.56 \times 10^4$ cells.

Table 4
The most efficient drug scheduling policies obtained by the renewed model with the Gompertz drug toxicity metabolism function

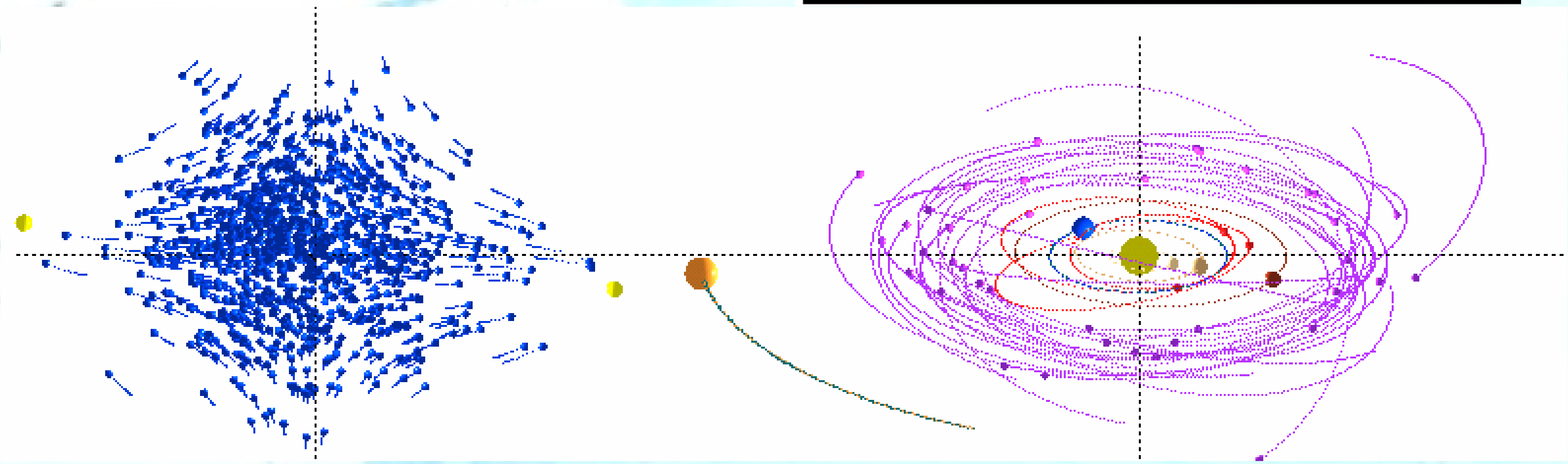
No.	The optimal solutions	Index (x_1)	No. of cells ($\times 10^3$)
(1)	$\{23.03, 13.04, 12.94, 12.78, 12.56, 0, (78 \times \overline{10.02})\}$	22.46	1.76
(2)	$\{53.72, 0, 25.75, 0, 40 \times \overline{(20.03, 0)}\}$	22.43	1.82
(3)	$\{28 \times \overline{(30.02, (2 \times \bar{0}))}\}$	21.99	3.11
(4)	$\{21 \times \overline{(37.68, (3 \times \bar{0}))}\}$	20.65	10.76

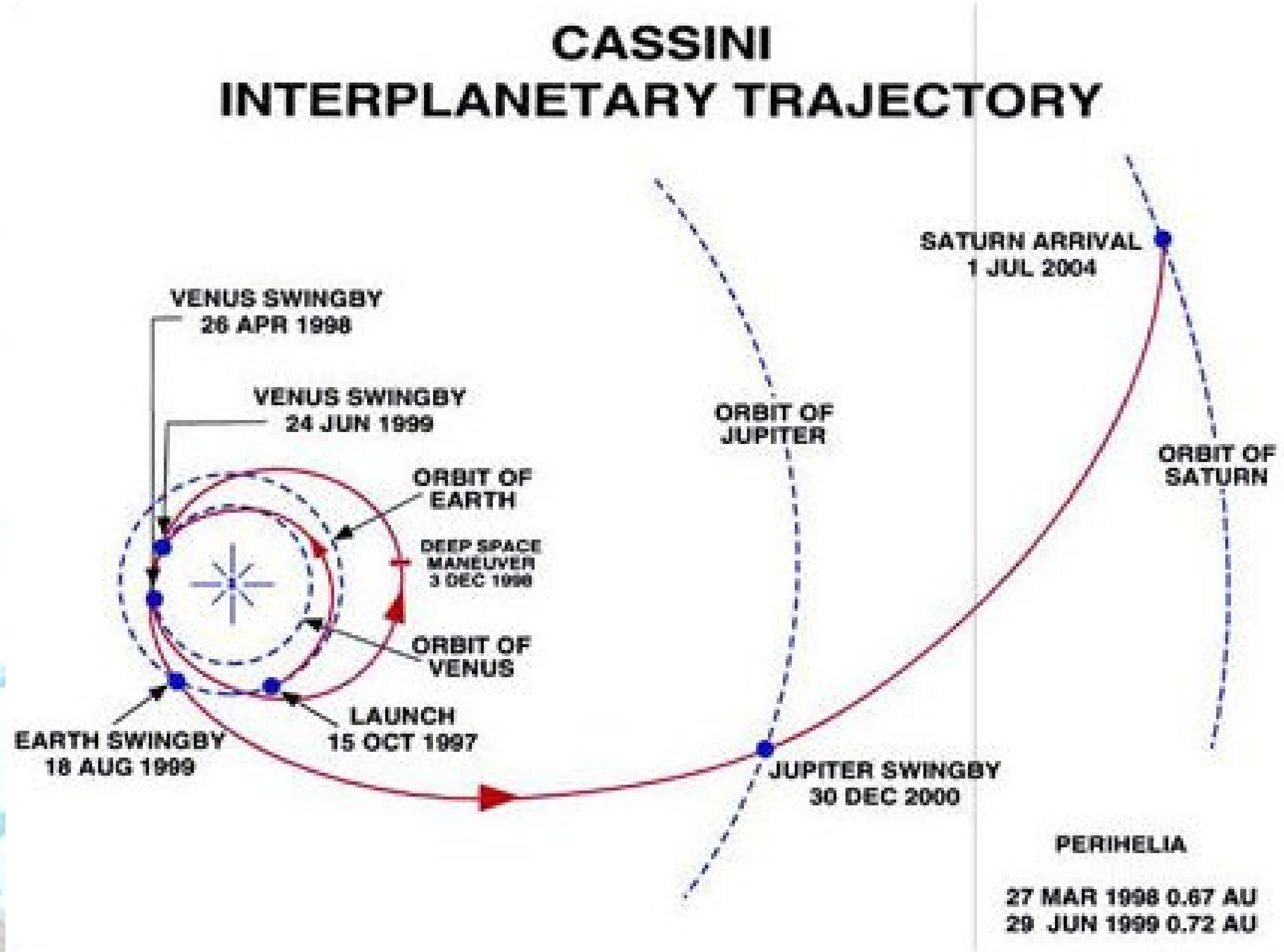
Planetary survival





Earth
50000 km




CASSINI
INTERPLANETARY TRAJECTORY
SATURN ARRIVAL
1 JUL 2004
VENUS SWINGBY
26 APR 1998
VENUS SWINGBY
24 JUN 1999
ORBIT OF
JUPITER
ORBIT OF
SATURN
ORBIT OF
EARTH
DEEP SPACE
MANEUVER
3 DEC 1998
ORBIT OF
VENUS
LAUNCH
15 OCT 1997
EARTH SWINGBY
18 AUG 1999
JUPITER SWINGBY
30 DEC 2000
PERIHELIA
27 MAR 1998 0.67 AU
29 JUN 1999 0.72 AU

EAs are making progress in this area –
M. Vasile, a space scientist who knows about EAs, is the main author. Not yet winning over humans, since the humans are good

Another example of a problem that is:

- naturally highly multiobjective (fuel cost, target distance(s), mission cost, robustness, …)
- for accurate mission design, an n-body model simulation is needed, making fitness evaluation highly expensive.
- important, for things like climate-change and climate-existence

- keep watch at: http://code.google.com/p/nmod/

- a packaged version of this is available that allows EA search for accurate space missions with an accurate simulator; it is continually improving and getting easier for developers.

Of course there are many many more important problems with complex fitness functions; not least:

- in silico drug design; drug docking;
- in vitro protein engineering
- large scale environmental logistics
- … add your own

Present

- *we still usually specify too few objectives*
- *we throw away immense amounts of sampled information that could help solve this instance*

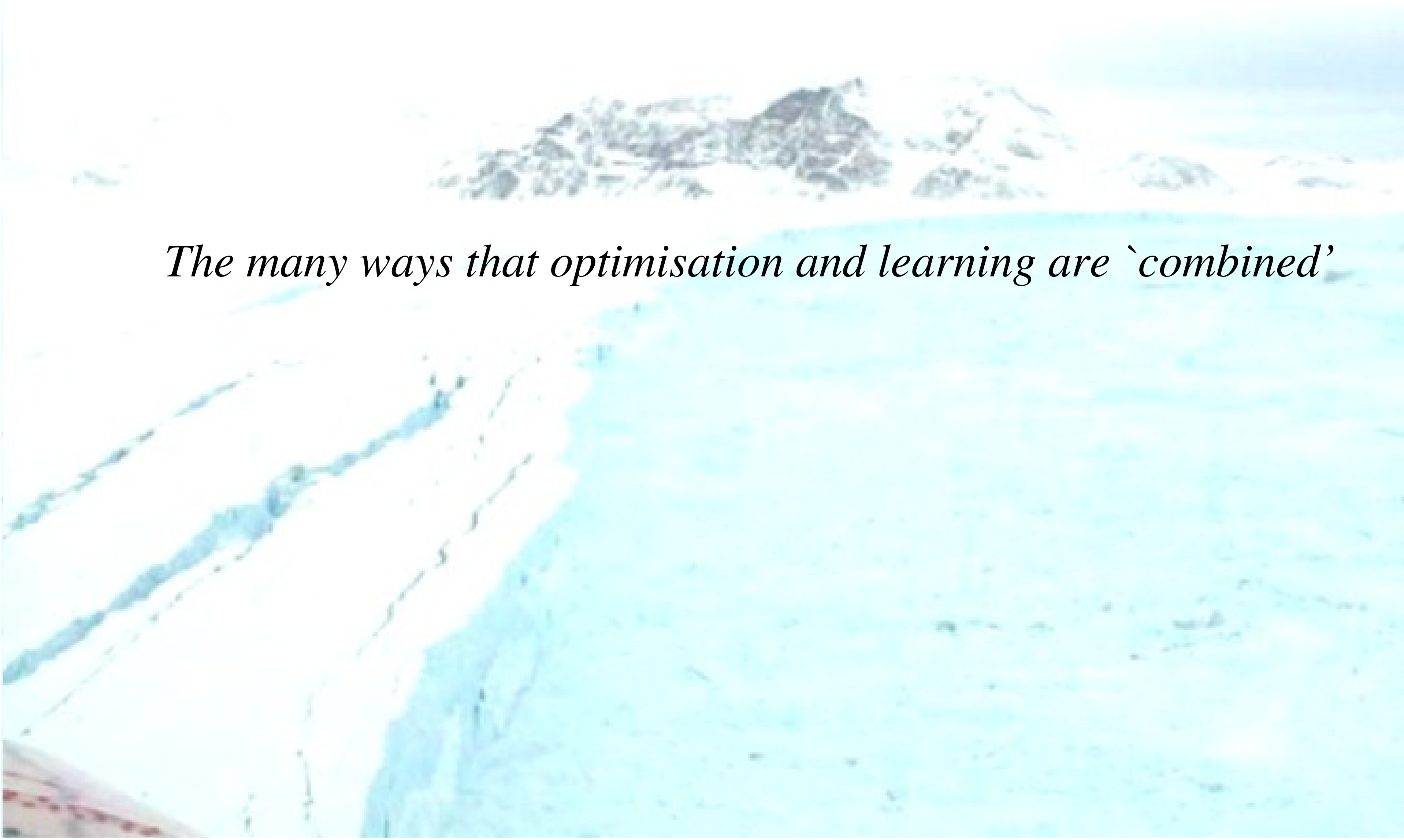
Future

- *solve the `whole' problem*
- *use much cleverer (and more elegant) algorithms that `combine' evolution and learning*
- *in the same process, produce* algorithms *that can quickly solve many instances*

Just as was the case with the "past→present" transition, this transition is possible because <u>we are beginning to discover methods that can do these things well</u>.

The Essence of O + L

The many ways that optimisation and learning are `combined'



In a basic EA we find the fitness of each of a population of individuals

Chromosome	Fitness
3,7,4,5,2,1,5,4,3,7,3,2,1,8	2
2,6,4,4,1,1,6,5,3,6,2,2,2,9	3
5,5,6,3,4,3,3,6,1,9,1,4,3,6	1
4,6,5,3,3,4,2,6,2,8,2,4,2,5	2
7,8,6,2,7,7,5,8,3,7,4,1,7,5	4

We then throw most of this information away

Chromosome	**Fitness**
2,6,4,4,1,1,6,5,3,6,2,2,2,9	3
7,8,6,2,7,7,5,8,3,7,4,1,7,5	4

and then proceed, under basic assumptions of smoothness in the landscape ….

In more clever EAs (strategy adaptation, CMA, adaptive operators, …), we learn things about the landscape local to the individuals

Chromosome		**Fitness**
3,7,4,5,2,1,5,4,3,7,3,2,1,8	$\left(\frac{\partial f}{\partial c_1}, \frac{\partial f}{\partial c_2}, \ldots, \frac{\partial f}{\partial c_n}\right)$	2
2,6,4,4,1,1,6,5,3,6,2,2,2,9	$\left(\frac{\partial f}{\partial c_1}, \frac{\partial f}{\partial c_2}, \ldots, \frac{\partial f}{\partial c_n}\right)$	3
5,5,6,3,4,3,3,6,1,9,1,4,3,6	$\left(\frac{\partial f}{\partial c_1}, \frac{\partial f}{\partial c_2}, \ldots, \frac{\partial f}{\partial c_n}\right)$	1
4,6,5,3,3,4,2,6,2,8,2,4,2,5	$\left(\frac{\partial f}{\partial c_1}, \frac{\partial f}{\partial c_2}, \ldots, \frac{\partial f}{\partial c_n}\right)$	2
7,8,6,2,7,7,5,8,3,7,4,1,7,5	$\left(\frac{\partial f}{\partial c_1}, \frac{\partial f}{\partial c_2}, \ldots, \frac{\partial f}{\partial c_n}\right)$	4

So, search effort is more appropriately guided, because we know a little about the shape of the landscape….

In EDAs, we learn probabilistic models of fit solutions, and generate new sample chromosomes from the model

Chromosome	Fitness
3,7,4,5,2,1,5,4,3,7,3,2,1,8	2
2,6,4,4,1,1,6,5,3,6,2,2,2,9	3
5,5,6,3,4,3,3,6,1,9,1,4,3,6	1
4,6,5,3,3,4,2,6,2,8,2,4,2,5	2
7,8,6,2,7,7,5,8,3,7,4,1,7,5	4

So, information from the whole population is compiled into the model, with less loss (similar effect in ACO, PSO)

In EDAs, we learn probabilistic models of fit solutions, and generate new sample chromosomes from the model

Chromosome	**Fitness**
Probabilistic model of good solutions 0.3, 0.8, 0.2, 01, 0.4, 0.2, … Maybe gene probabilities; or bivariate/multivariate Or a Bayesian network, etc…	*good*

So, information from the whole population is compiled into the model, with less loss (similar effect in ACO, PSO)

In LEM, we learn a model that predicts whether candidate solutions are good, OK, or bad.

Chromosome	**Fitness**
3,7,4,5,2,1,5,4,3,7,3,2,1,8	2
2,6,4,4,1,1,6,5,3,6,2,2,2,9	3
5,5,6,3,4,3,3,6,1,9,1,4,3,6	1
4,6,5,3,3,4,2,6,2,8,2,4,2,5	2
7,8,6,2,7,7,5,8,3,7,4,1,7,5	4

So, depending on the learning method, very useful information can be gleaned that will influence the search

In LEM, we learn a model that predicts whether candidate solutions are good, OK, or bad.

Chromosome	**Fitness**

If(c5 == c6) then GOOD
 	
If(c12>c13) AND (c13<c14) then BAD
	Classified into discrete groups

So, depending on the learning method, very useful information can be gleaned that will influence the search

[The LEM3 implementation of learnable evolution model and its testing on complex function …](#) - all 7 versions »
J Wojtusiak, RS Michalski - Proceedings of the 8th annual conference on Genetic and …, 2006 - portal.acm.org
Page 1. The **LEM3** Implementation of Learnable Evolution Model ... 2. DESCRIPTION OF **LEM3** This section describes the top-level structure of **LEM3**. ...
Cited by 9 - Related Articles - Web Search

Table 2: Comparison of LEM3 with EDA on the Rastrigin, Griewangk, and Rosenbrock functions.

Function # vars.	Method	Best fitness Value	Evolution Length	LEM3/EDA Speedup
Griewangk 10 vars.	LEM3	0	1,305	**~ 231**
	EDA	0.051166	301,850	
Griewangk 50 vars.	LEM3	0	4,005	**~ 54**
	EDA	8.7673E-6	216,292	
Rosenbrock 10 vars.	LEM3	1.2	1,389	**~ 118**
	EDA	8.6807	164,519	
Rosenbrock 50 vars.	LEM3	46.74	7,875	**~ 15**
	EDS	48.8234	275,663	

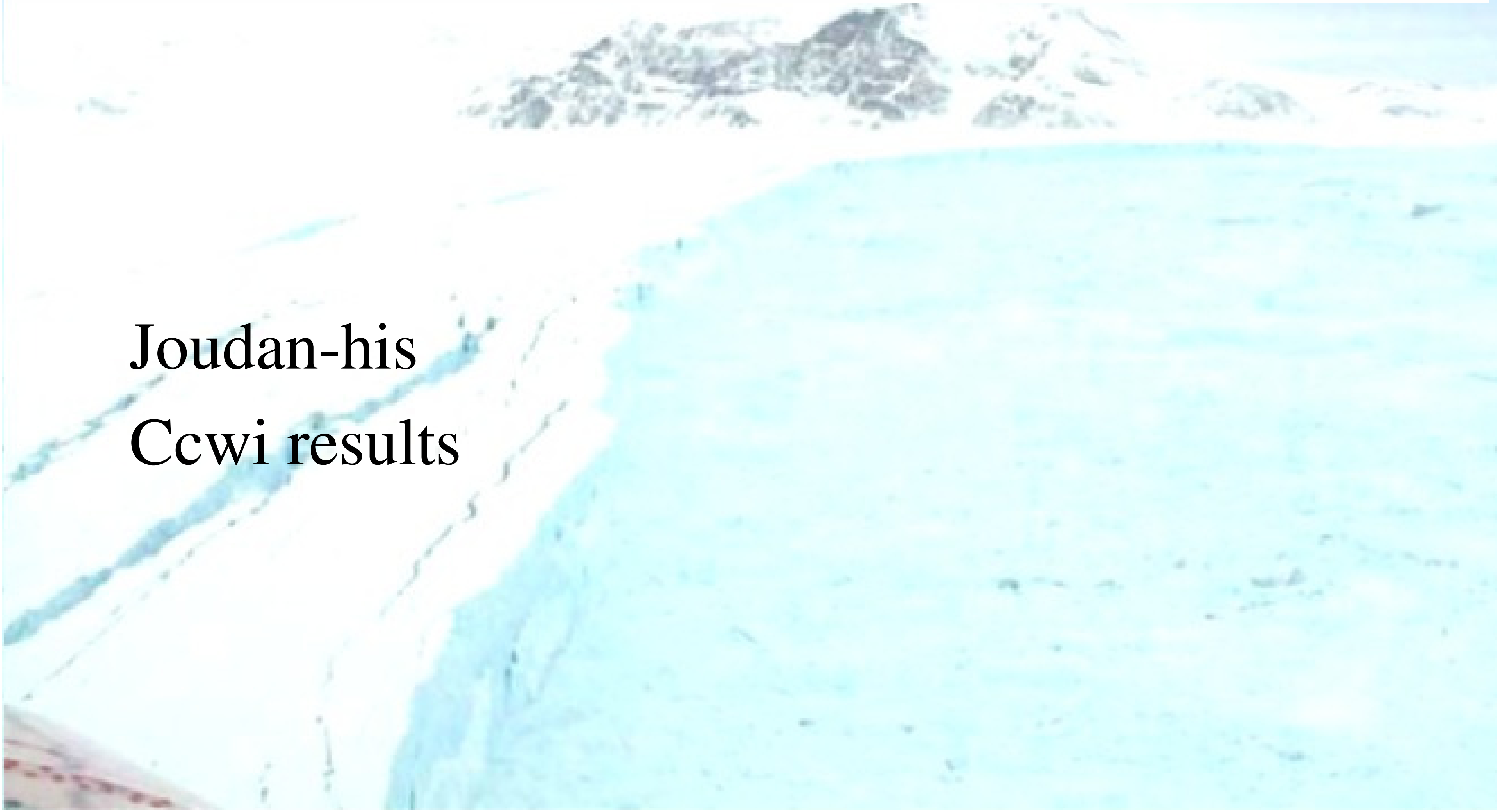
$$EDA = EMNA_{global}$$

Preliminary Investigation of the 'Learnable Evolution Model'for Faster/Better Multiobjective Water … - all 3 versions »

L **Jourdan**, D **Corne**, D Savic, G Walters - Proceedings of The Third Int. Conference on Evolutionary … - Springer

... Evolution Model' for Faster/Better Multiobjective Water Systems Design Laetitia **Jourdan**, David **Corne**, Dragan Savic, and Godfrey Walters **...**

Cited by 8 - Related Articles - Web Search



Joudan-his

Ccwi results

- Integrating our EA with some form of learning is almost always significantly better
- **Almost ALWAYS significantly better**
- There is great momentum in neighbouring communities towards combinations of learning and optimisation.
- From the machine learning community, there is LEM.
- From the operations research community, there is the Cross Entropy method
- From the statistical physics and game theory communities there is Probability Collectives
- From the EA community there is `super'-heuristics
- From the EA community, of course, there is EDA
- There are more, but that's enough for present purposes
- Let's have a closer look at CE, EDA, LEM, PC, SH …

The **Cross-Entropy Method** for Combinatorial and Continuous Optim
»
R Rubinstein - Methodology and Computing in Applied Probability, 1999 - Springer
... Manufactured in The Netherlands. The **Cross-Entropy Method** for Combinatoria
Continuous Optimization REUVEN RUBINSTEIN ... THE **CROSS-ENTROPY METH**
Cited by 103 - Related Articles - Web Search - BL Direct

Arguably, from the probability/OR communities

Discrete, continuous, and constrained optimization using **collectives** - all 7 versions »
S Bieniawski, DH Wolpert, I Kroo - Proceedings of 10th AIAA/ISSMO Multidisciplinary Analysis ..., 2004
pdf.aiaa.org
... have been used for a number of distributed optimization problems in computer science,
recent developments based upon **Probability Collectives** (PC) theory ...
Cited by 20 - Related Articles - Web Search

Arguably, from the statistical physics / control / game theory

LEARNABLE EVOLUTION MODEL: Evolutionary Processes Guided by Machine Learning
all 7 versions »
RS **Michalski** - Machine Learning, 2000 - Springer
... given an H-group and an L-group, a machine learning method generates a description
that discriminates between these groups (**Michalski**, 1983). In **LEM**, one can ...
Cited by 70 - Related Articles - Web Search - BL Direct

Arguably, from (symbolic) machine learning / AI

The Equilibrium Genetic Algorithm and the Role of Crossover - all 2 versions »
A **Juels**, S Baluja, A Sinclair - Unpublished manuscript, 1993 - citeseer.ist.psu.edu
... Algorithm and the Role of Crossover (1993) (Make Corrections) **Ari Juels**, Shumeet
Baluja ... of the GA, which we call the Equilibrium Genetic Algorithm (**EGA**). ...
Cited by 6 - Related Articles - Cached - Web Search

Arguably, from evolutionary computation

The Cross-Entropy method

practical tool for solving NP-hard problems.

The CE method involves an iterative procedure where each iteration can be broken down into two phases:

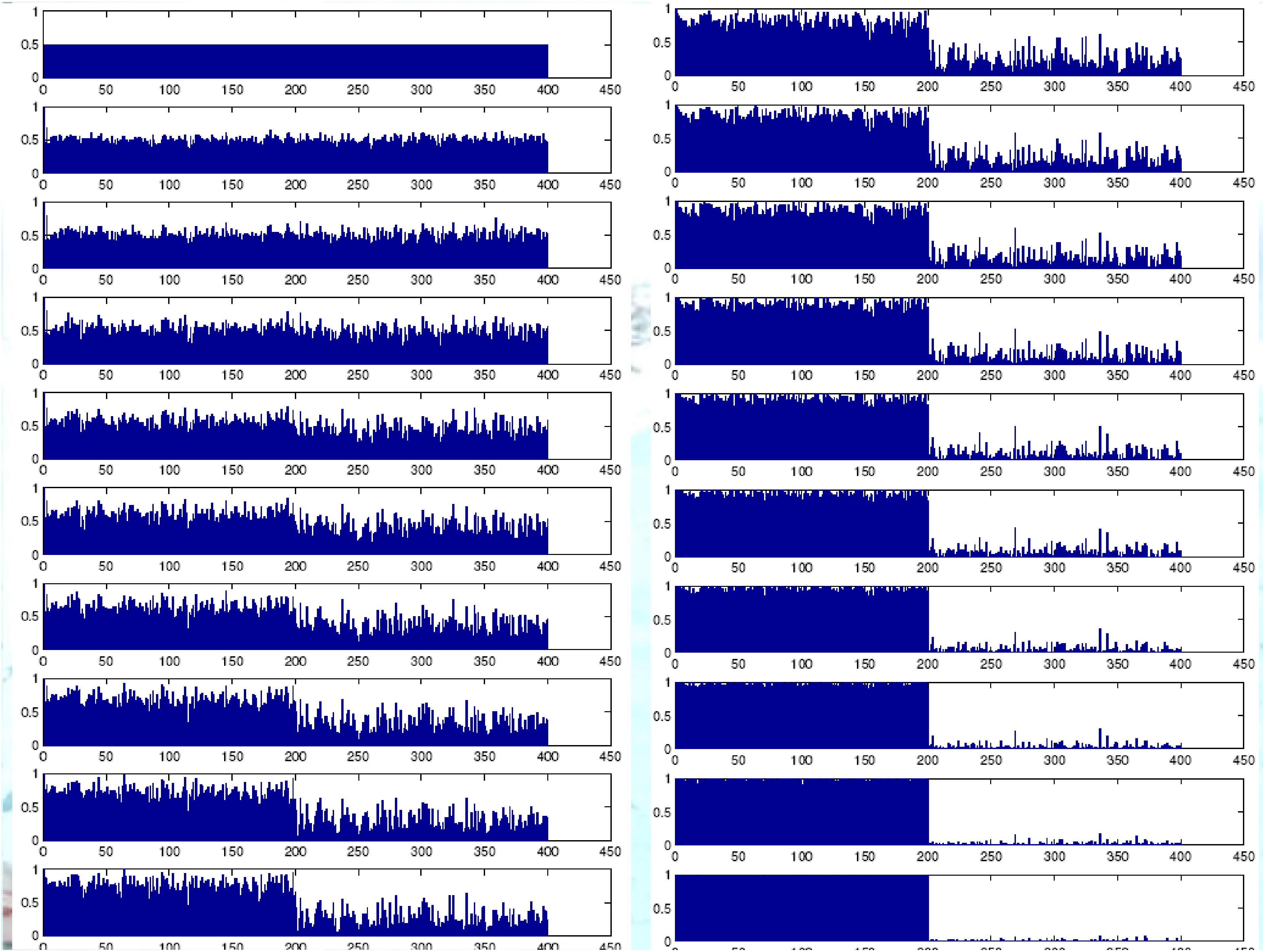
1. Generate a random data sample (trajectories, vectors, etc.) according to a specified mechanism.

2. Update the parameters of the random mechanism based on the data to produce "better" sample in the next iteration.

The significance of the CE method is that it defines a precise mathematical framework for deriving fast, and in some sense "optimal" updating/learning rules, based on advanced simulation theory. Other well-known randomized

From

A Tutorial on the **Cross-Entropy Method** - all 17 versions »
PT de Boer, DP Kroese, S Mannor, RY Rubinstein - Annals of Operations Research, 2005 - Springer
... A Tutorial on the **Cross-Entropy Method** ... Keywords: **cross-entropy method**, Monte-Carlo simulation, randomized optimization, machine learning, rare events ...
Cited by 93 - Related Articles - Web Search


1
0.5
0
0 50 100 150 200 250 300 350 400 450

Estimation of Distribution Algorithms

- We've already seen it!
- CE has its roots in methods to improve analysis of rare events / this is tweaked in CE towards optimal updating of the model towards capturing the distribution of good solutions
- In EDAs, there are now a variety of ways to update the model, sometimes theoretically justified in some way, sometimes not

Probability Collectives

- Fortunately, this is what David Wolpert chose to talk about at CEC'05

Discrete, continuous, and constrained optimization using collectives - all 8 versions »
S Bieniawski, DH **Wolpert**, I Kroo - Proceedings of 10th AIAA/ISSMO Multidisciplinary Analysis …, 2004
pdf.aiaa.org
Page 1. **Discrete**, Continuous, and Constrained **...** Stanford University, Stanford, CA 94305
David H. **Wolpert** ‡ NASA Ames Research Center, Moffett Field, CA 94035 **...**
Cited by 20 - Related Articles - Web Search

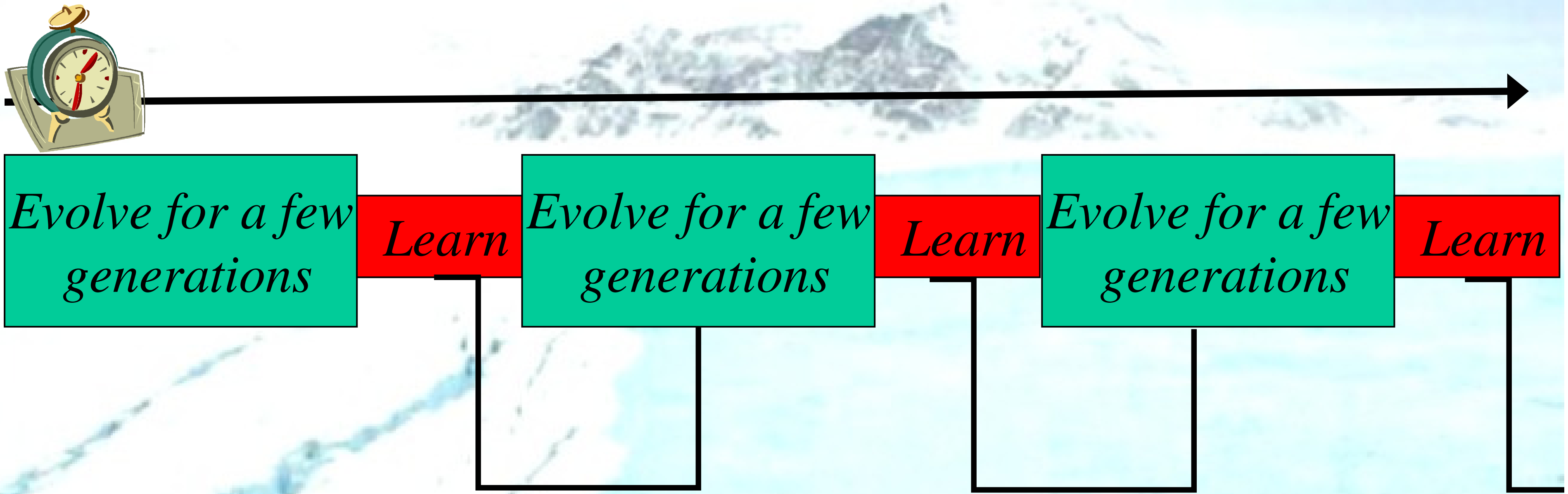
1. Start with initial uniform probability distribution over gene values.
2. Sample and evaluate individuals from the distribution and calculate certain information theoretic measures for each gene.
3. Update the distribution
4. Return to 2.

Step 2 is sophisticated and steeped in information-theoretic game theory. It is maximising a function of the ***distribution*** *concerned competition between genes -- prevents overfitting?*

Step 2 seems heavy on maths, which puts many people off.

But, we have important problems to solve …

Learnable Evolution Model



-Learning influences production of next generation
-variety of ways

Any learning strategy: C4.5, KNN, Evolving rules, AQ, LCS, Naïve Bayes, etc …

Present

Future

- *we still usually specify too few objectives*
- *we throw away immense amounts of sampled information that could help solve this instance and others*

- *solve the `whole' problem*
- *use much cleverer (and more elegant) algorithms that `combine' evolution and learning*
- *in the same process, produce* algorithms *that can quickly solve many instances*

Just as was the case with the "past→present" transition, this transition is possible because we are beginning to discover methods that can do these things well.

Hyper-Heuristics / Super-Heuristics

The idea emerges from a kind of encoding in evolutionary computation:
Example in talk-timetabling:

4,8,2,9,6,…

Means "first talk in session 4, second talk in session 8, etc …"

This is a simple `hyper-heuristic' way:
1 = earliest fit, 2 = best fit in terms of room capacity, 3 = best fit in terms of talker preference, 4 = etc …

4,8,2,9,6, …

Means "use heuristic 4 to schedule the first talk, 8 for the second, etc…"

This is (almost) an algorithm that can be applied to any problem instance

Hyper-Heuristics vs Super-Heuristics

HH is often used in EA applications, since it typically provides better solutions to individual problem instances.

In this sense it is really just another encoding, and many other encodings are of the same nature but have not been called HH.

The much more interesting use of this idea is to evolve algorithms on a **set** of problem instances, where the fitness of an algorithm is its performance on a **different** set of instances.

Think about this for a moment; if it works, this means you can evolve a constructive algorithm based on (for example) the last ten daily job scheduling problems at your factory. Maybe it takes a few hours. At the end, you have an algorithm that solves tomorrows problem well, and very fast. The same algorithm solves the next day's problem, and the next, and the next…

Well, it seems to work! Partly I think this is because it is a tight integration of learning and evolution, which reduces the overfitting that happens in any `ordinary' approach.

SuperID3

ID3 builds a decision tree by repeatedly adding nodes/splits to the tree, until all data have been classified. It uses the Information Gain heuristic (G) to decide what data attribute to use in the next node. An alternative version of ID3 uses Information Gain Ratio (GR) instead of G.

Super-ID3 (my curent PhD student Alan Vella)

[G] [3]	**[GR] [2]**	**[GR] [1]**	**[G] [2]**	**…**	**[GR] [1]**
100%	**99%**	**98%**	**97%**		**1%**

This is basically a set of rules, indicating what criterion (G or GR) to use for choosing the data attribute, when a given percentage of the dataset remains to be classified.

Almost the simplest way to produce a Superheuristic data mining algorithm

First Results Snapshot

	cars		derma		flags		spect	
ID3 (gain)	94.94%	0.48%	88.47%	1.48%	79.64%	1.74%	74.75%	2.65%
ID3 (gain ratio)	95.10%	0.57%	90.03%	1.35%	78.66%	1.81%	75.46%	2.13%
Single Data Set Exp.	CBF1810 InitB [GR]		GA 1P Init VF [GR]		GA 5P Init VF [G][GR]		GA 5P NoInit VF [GR]	
	94.87%	0.55%	92.19%	1.92%	81.07%	3.00%	76.94%	2.52%
Super-Heuristics 3 Data Set Exp.	GA 1P Init [GR]		GA 1P Init [GR]		-		GA 1P Init [GR]	
	93.60%	1.36%	**91.60%**	1.22%	-	-	75.71%	1.86%
Super-heuristics 4 Data Set Exp.	HC1810 5P InitB [G][GR]		HC1810 5P InitB [G][GR]		HC1810 5P InitB [G][GR]		HC1810 5P InitB [G][GR]	
	95.14%	0.49%	88.64%	1.69%	**81.59%**	3.95%	77.49%	2.13%

[PDF] Hyper-heuristics: learning to combine simple heuristics in **bin-packing** problems - 16 versions »
P **Ross**, S Schulenburg, JG Marin-Blazquez, E Hart - Proceedings of the Genetic and Evolutionary Computation ..., 2002 - cs.bham.ac.uk
Hyper-heuristics: learning to combine simple heuristics in **bin-packing** problems
Peter **Ross** School of Computing Napier University Edinburgh EH10 5DT peter@dcs ...

A chromosome is composed of blocks, and each block j contains six numbers $h_j, l_j, m_j, s_j, i_j, a_j$. The first five essentially represent an instance of the problem state. Here h_j corresponds to the proportion of huge items that remain to be packed, and similarly l_j, m_j and s_j refer to large medium and small items, and i_j corresponds to the proportion of items remaining to be packed. The sixth number, a_j, is an integer in the range $0 \cdots 7$ indicating which heuristic is associated with this instance. An example of a set of 12 rules obtained with the GA can be seen in figure 1.

Fig. 1. Example of a final set with 12 rules

```
0.70 -2.16 -1.10  1.55  1.81 --> 1  |   2.34  0.67  0.19  1.93  2.75 --> 1
0.12  1.37 -0.54  1.12  0.58 --> 6  |  -1.93 -2.64 -1.89  2.17 -1.46 --> 3
0.13  1.43 -1.27  0.13 -2.18 --> 2  |  -1.30  0.11  2.00 -1.85  0.84 --> 4
1.87 -0.91  1.30 -1.34  1.93 --> 3  |   0.32  1.94  2.24  0.99 -0.53 --> 0
2.60  1.30 -0.54  1.12  0.58 --> 6  |   0.58  0.87  0.23 -2.11  0.47 --> 1
0.25  2.09 -1.50 -1.46 -2.56 --> 0  |   1.21  0.11  2.00  0.09  0.84 --> 4
```

Table 2. Extra bins compared to best of four heuristics (BFH)

	HH Methods						Heuristics							
	GA		XCSs		XCSm		LFD		NFD		DJD		DJT	
Bins	Trn	Tst	Trn	Tst	Trn	Tst	Trn	Tst	Trn	Tst	Trn	Tst	Trn	Tst
-4		0.4												
-3	0.3	0.8												
-2	1.3	1.2	0.3	0.5	0.3	0.9								
-1	4.2	5.5	2.7	2.2	2.3	3.6								
0	98.3	97.6	98.3	97.3	98.8	97.3	71.1	73.9			91.2	91.7	95.4	94.1
1	100	100	100	100	100	100	83.8	82.6	0.1		97.3	97.6	99.7	99.6
2							88.9	88.5	0.1		98	98.4	100	100
3							91.9	92.5	1.1	2	99.6	98.8		
4							93.8	93.3	3.7	4	100	99.6		
5							95.8	96.1	7.2	5.9		100		
10							97.4	96.8	25.3	24.5				
20							99.7	99.6	48.1	47.8				
30							100	100	61.1	60.5				

So, MO → SHManyO(L+O)

We are already seeing MO-EDA; MO-LEM; probably SH-MO somewhere etc…

IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, VOL. 12, NO. 1, FEBRUARY 2008 41

RM-MEDA: A Regularity Model-Based Multiobjective Estimation of Distribution Algorithm

Qingfu Zhang, *Senior Member, IEEE*, Aimin Zhou, and Yaochu Jin, *Senior Member, IEEE*

Hybrid Estimation of Distribution Algorithm for Global Optimization

Qingfu Zhang, Jianyong Sun, Edward Tsang and John Ford *
Department of Computer Science, University of Essex,
Wivenhoe Park, Colchester, CO4 3SQ, U K
E-mail: qzhang@essex.ac.uk

The argument

Future of Optimisation = Many-objective superheuristics that tightly integrate evolution and learning

- from few-o to many-o (less simplification)
- using better learning strategies to get faster and better solutions
- using super-heuristics approaches to get even better solutions, and much faster solutions to future instances
- Driven by the fact that methods are slowly maturing for each of the above, and by the needs of important optimisation challenges.

That’s the future;

- I *knew* someone would ask that question
  - *Yes*, I would agree that there are a small number of problems that are **really** single objective; so, SO can happily remain as a subfield of EC. But, even these may often be better addressed with a multiobjective formulation – see the new book.

Ways to rank nondominated points

Four nondominated 5-objective points

A:	0, 10, 5, 5, 3
B:	7, 7, 7, 7, 7
C:	10, 4, 4, 3, 8
D:	1, 2, 3, 4, 8

Single-Objective Sum (SO)

Couldn't be simpler:

A: 0, 10, 5, 5, 3 -- rank = 23

B: 7, 7, 7, 7, 7 -- rank = 35

C: 10, 4, 4, 3, 8 -- rank = 29

D: 1, 2, 3, 4, 8 -- rank = 18

The favour relation (Dreschler[2])

Let X beat Y on x *objectives*

Let Y beat X on y objectives

X is favoured over *Y* iff $x > y$

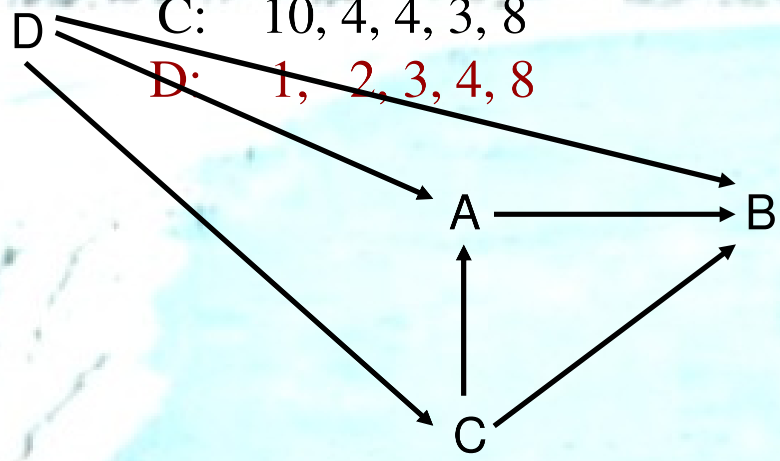
A:	0, 10, 5, 5, 3
B:	7, 7, 7, 7, 7
C:	10, 4, 4, 3, 8
D:	1, 2, 3, 4, 8

Let A $\longrightarrow$ B mean A is favoured over B

We can then draw a graph …

The favour relation (Dreschler^2)

A: 0, 10, 5, 5, 3
B: 7, 7, 7, 7, 7
C: 10, 4, 4, 3, 8
D: 1, 2, 3, 4, 8



From this we can get a rank-ordering: D, C, A, B

NOTE: this differs from the ranking with SO

NOTE: <u>there may be cycles in this graph – it may partition the points into anything from 1 to numpoints ranks.</u>

K – Optimality (di Pierro)

If X is on the PF for each z-dimensional subset of the objectives, then it is *<u>efficient order z</u>*

If k *is the smallest value for which this is true, X is* <u>k-*optimal*</u>

By definition, these are all efficient of order 5.

A is efficient order 4 (check it), but not order 2
(it is dominated by C and D for objectives 2 and 3)

A: 0, 10, 5, 5, 3 -- rank = 4
B: 7, 7, 7, 7, 7 -- rank = 5
C: 10, 4, 4, 3, 8 -- rank = 5
D: 1, 2, 3, 4, 8 -- rank = 3

NOTE: This is relatively time-intensive to calculate.

Many Objectives

Four non-dominated five-objective points

A:	10, 0, 5, 5, 7	--- 27	(Rank 2: 2nd)
B:	3, 3, 3, 3, 3	--- 15	(Rank 4: Worst)
C:	0, 6, 6, 8, 2	--- 22	(Rank 3: 3rd)
D:	9, 10, 8, 6, 2	--- 35	(Rank 1: Best)

Modified for the maximization problem from Corne's GECCO 2007

Unmodified from Ishibuchi's CEC 2007

Combining(?) learning and optimisation: assorted notes

- Optimisation and learning (in the usual CI sense) are the same thing.
- Learning means optimising a predictive model.
- The difference between optimising a predictive model, and optimising a function is one of degree.
- When we call it learning, this is because **we don't know the fitness function, we only know an approximation based on the training set**
- When we call it optimisation, we think we know the fitness function exactly – but actually we rarely do
- Even if we know it, we still overfit, and we call that premature convergence

Single objective optimization is a *crime*