Evolutionary Computation for Dynamic Multiobjective Optimization

Shengxiang Yang & Shouyong Jiang

Centre for Computational Intelligence (CCI)
De Montfort University, Leicester LE1 9BH, UK
http://www.tech.dmu.ac.uk/~syang
Email: syang@dmu.ac.uk

Tutorial presented at the 2017 IEEE Ssymposium Series on Computational Intelligence (IEEE SSCI 2017), Honolulu, Hawaii, USA, 27 Nov - 1 Dec, 2017

Presenters

- Shengxiang Yang:
  - Since 2012, Professor in Computational Intelligence (CI) at De Montfort University (DMU), UK
  - Since 2013, Director of Centre for Computational Intelligence (CCI), DMU, UK
  - Research areas: Computational Intelligence (esp. Evolutionary computation (EC)), dynamic optimisation and/or multi-objective optimisation, and real-world applications
  - Over 230 publications and £2M funding for research
  - AE/Editorial Board Member for 8 journals (IEEE Trans Cybern, Evol Comput, Inform Sci, Neurocomputing, and Soft Comput)
  - Chair for 2 IEEE CIS Task Forces (ECiDUE and INS)

- Shouyong Jiang:
  - PhD (2013-2017), De Montfort University, UK
  - Now, postdoc research associate at Newcastle University, UK
  - Research interests: EC for dynamic and/or multi-objective optimisation problems

Centre for Computational Intelligence

- CCI (www.cci.dmu.ac.uk):
  - Mission: Developing fundamental theoretical and practical solutions to real-world problems using a variety of CI paradigms
  - Members: 16 staff, several research fellows, 30+ PhDs, visiting researchers
  - Themes: EC, fuzzy logic, neural networks, data mining, robotics, game ...
- Funding:
  - Research Councils/Charities: EPSRC, EU FP7 & Horizon 2020, Royal Academy of Engineering, Royal Society, Innovate UK, KTP, Innovation Fellowships, Nuffield Trust, etc.
  - Government: Leicester City Council, DTI
  - Industries: Lachesis, EMDA, RSSB, Network Rail, etc.
- Collaborations:
  - Universities: UK, USA, Spain, and China
  - Industries and local governments
- Teaching/Training:
  - DTP-IS: University Doctor Training Programme in Intelligent Systems
  - MSc Intelligent Systems, MSc Intelligent Systems & Robotics
  - BSc Artificial Intelligence with Robotics
- YouTube page: http://www.youtube.com/thecci

Outline of the Talk

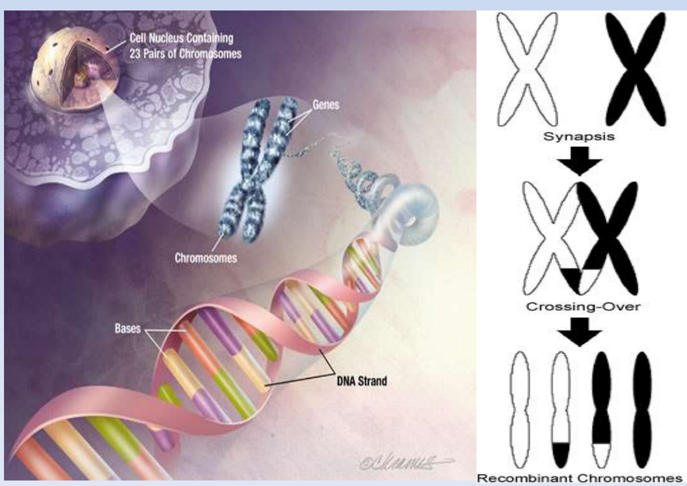
- Part I: Fundamentals
  - Basic Concepts of evolutionary computation (EC)
  - EC for dynamic multiobjective optimization problems (DMOPs): Concept & Motivation
  - Classification, Benchmarks and Test Problems
  - Performance Measures
- Part II: Approaches, Case Studies, Issues and Future Work
  - EC-based Approaches for DMOPs
  - Case Studies
  - Relevant Issues
  - Future Work

What Is Evolutionary Computation (EC)?

EC uses mechanisms inspired by

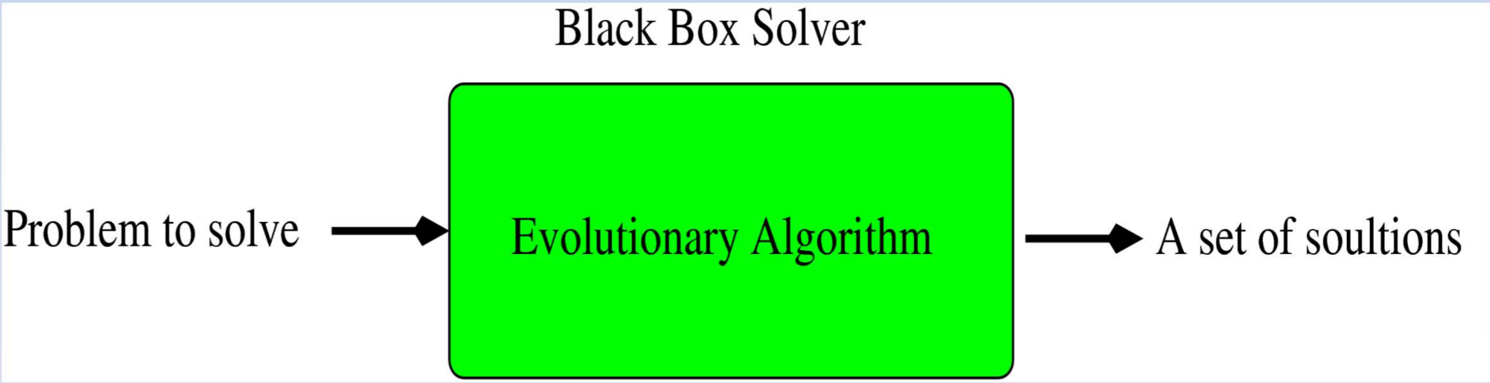
- Biological evolution (e.g., survival of fittest and genetics) or
- Biological behaviour (e.g., ant foraging, bird flocking, animal herding, bacterial growth, fish schooling….)




What Is Evolutionary Computation (EC)?

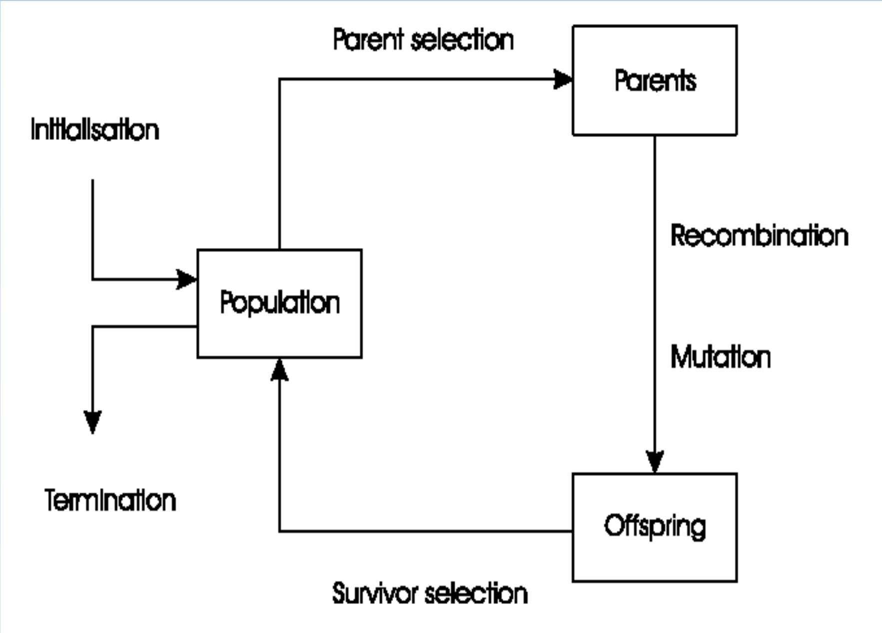
- EC encapsulates a class of stochastic optimization algorithms, dubbed Evolutionary Algorithms (EAs)
- An EA is an optimisation algorithm that is
  - Generic: a black-box tool for many problems
  - Population-based: evolves a population of candidate solutions
  - Stochastic: uses probabilistic rules
  - Bio-inspired: uses principles inspired from biological evolution or biological behaviour



Design and Framework of an EA

- Given a problem to solve, two key things to consider:
  - Representation of solution into individual
    - Binary string, real numbers, or permutation of integers, ……
  - Evaluation or fitness function

- Framework of an EA:
  - Initialization of population
  - Evolve the population
    - Selection of parents
    - Variation operators (recombination, mutation)
    - Selection of offspring into next generation
  - Termination condition: e.g., a given number of generations



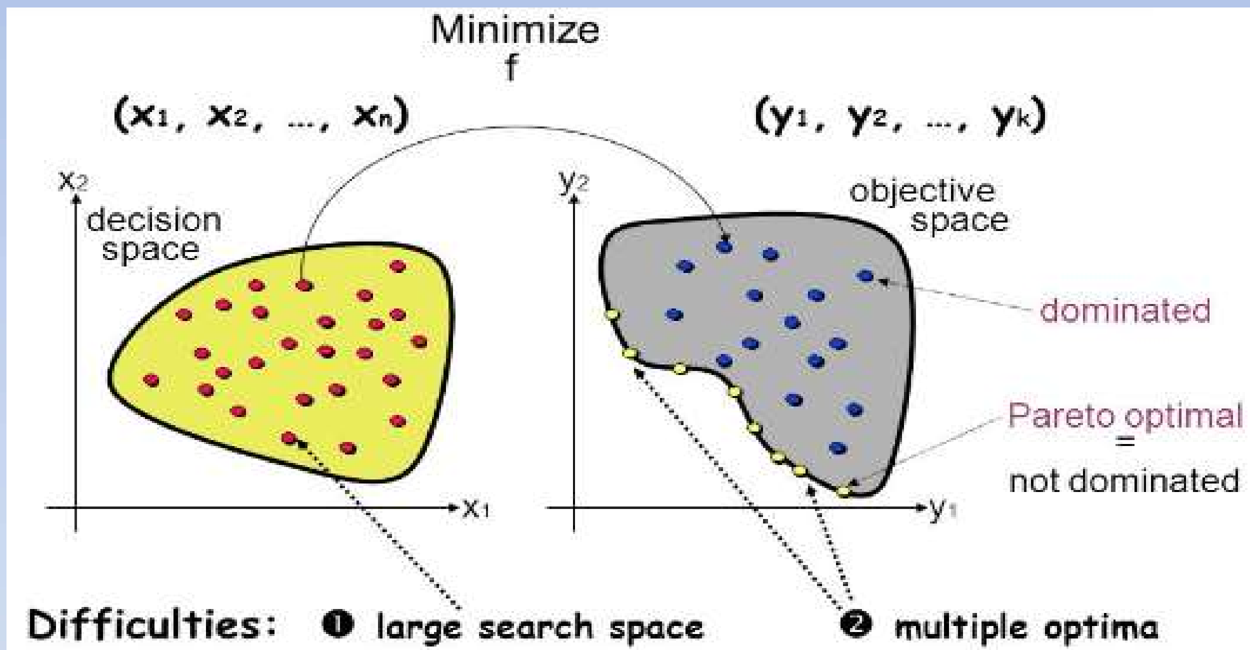
EC Applications

- Advantages of EAs:
  - Multiple solutions in a single run
  - No strict requirements to problems
  - Easy to use

- Widely used for optimisation and search problems
  - Financial and economical systems
  - Transportation and logistics systems
  - Industry engineering
  - Automatic programming, art and music design
  - ......

EC for Optimization Problems

- Traditionally, research on EC has focused on static problems:
  - Single, multiple, and many objectives
  - Aim to find the optimum *quickly* and *precisely*



- But, many real-world problems are dynamic optimization problems, where changes occur over time
  - In transport networks, travel time between nodes may change
  - In logistics, customer demands may change

What Are DMOPs?

- In general terms, "optimization problems that involve multiple conflicting objectives and change over time" are called dynamic or time-dependent multiobjective problems:

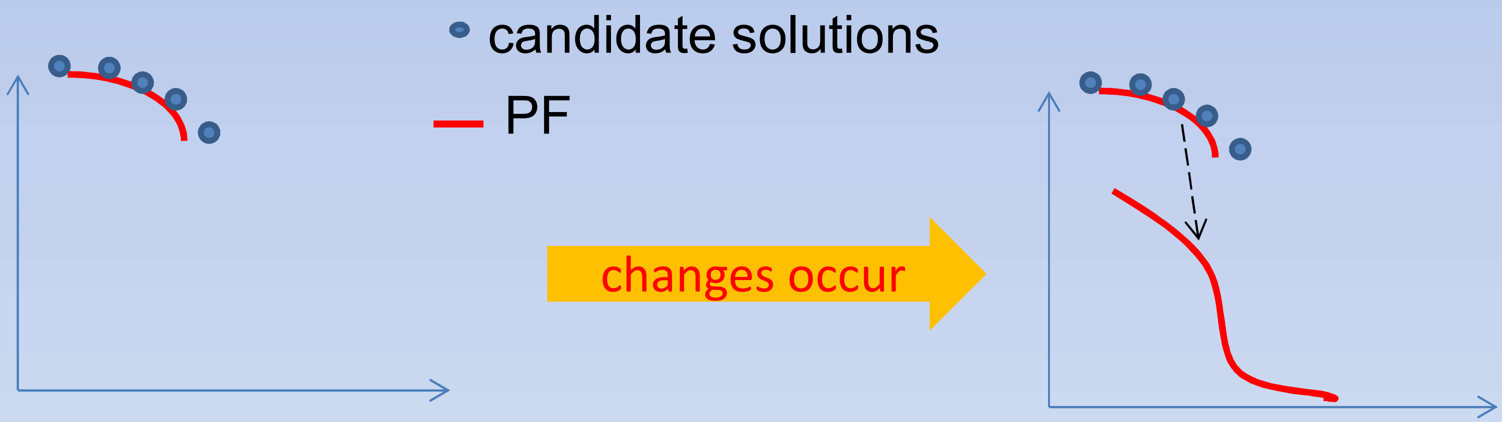
$$F = \left(f_1(x,\varphi,t), f_2(x,\varphi,t), \cdots, f_M(x,\varphi,t) \right)^T$$

  - x: decision variables;
  - φ: parameter;
  - t : time

- DMOPs: a special class of dynamic problems that are solved by an algorithm as time precedes.

Why Are DMOPs Challenging?

- For DMOPs, Pareto fronts (PFs) and/or Pareto sets (PSs) may change over time
  - ➢ Challenge 1: need to track the moving PF/PS over time
  - ➢ Challenge 2: need to re-spread non-dominated solutions



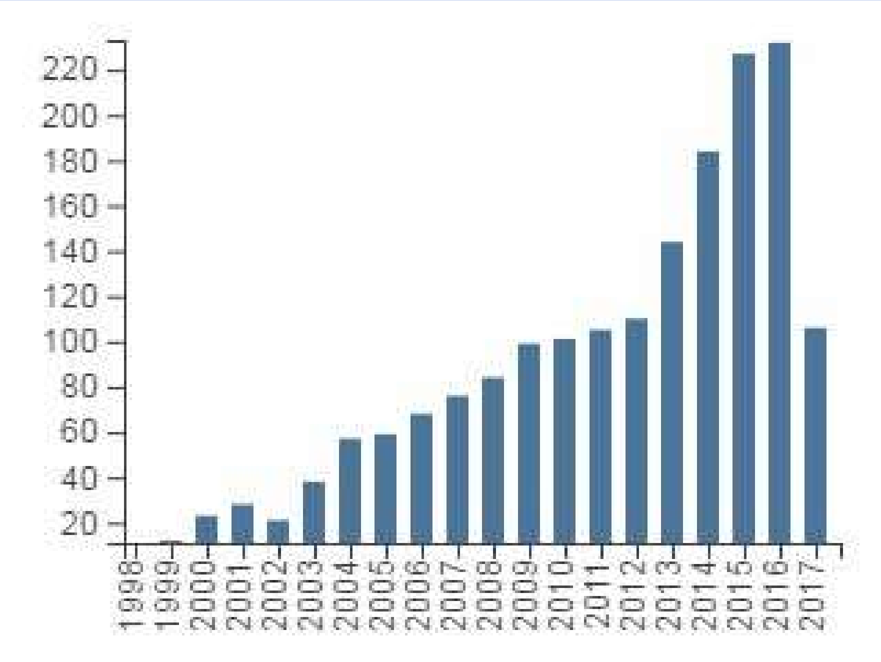
- DMOPs challenge traditional EAs
  - ➢ Limited time to respond to environmental changes.
  - ➢ Once converged, hard to escape from an outdated PF/PS.
  - ➢ Very likely to lose diversity after a changes.

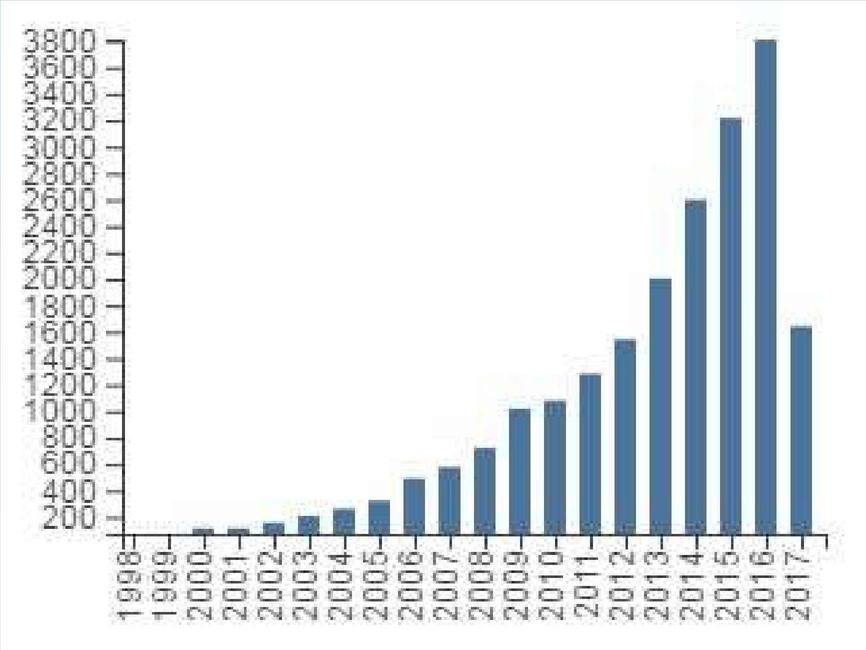
Why EC for DMOPs?

- Many real-life problems are DMOPs
  - Desirable to present a set of diverse solutions to decision makers over time
- EAs, once properly modified/enhanced, are good choice
  - Inspired by biological evolution/behaviour, always in dynamic environments
  - Able to provide multiple solutions at any time
  - Intrinsically, should be fine to deal with DMOPs
- Research on EC for DMOPs rises recently

DMOPs Are Getting Popular

- Web of Science:
  - TS=((dynamic OR time-varying OR time-dependent OR non-stationary) AND multiobjective AND optimization)





publication by year

citation by year

Classification of DMOPs

- Cause-based rules (*Tantar et al. 2011*):
  - Case 1: the decision variables change over time
  - Case 2: the objective functions change over time
  - Case 3: the current values of decision variables or objective functions depend on their previous values
  - Case 4: parts of or the entire environments change over time
- Effect-based rules (*Farina et al. 2004*):
  - Type I: PS changes, PF remains unchanged
  - Type II: Both PS and PF change
  - Type III: PF changes, PS remains unchanged
  - Type IV: Both PS and PF remain unchanged, although objective functions, constraints, etc., change over time
  - Mixed Type (*Jiang & Yang 2017a*): All of the above four types of change can be present, either randomly or in turn

Benchmarking

- Two ideas based on classification rules:
  - Change basic static MOPs to obtain different dynamic effects
  - Introduce novel dynamics that change optimization problems over time
- Real space:
  - Change objective functions with some time-varying factors
  - Dynamically change constraints or the search space
- Combinatorial space:
  - Change decision variables: item weights/profits in multi-objective knapsack problems
  - Add/delete decision variables: nodes added/deleted in network routing problems

Jin-Sendhoff's Framework (2004)

- Main idea: Aggregating several objective functions with time-varying weights
- For example, a tri-objective minimization problem can be easily transformed into a bi-objective dynamic problem with time-dependent weighted aggregation of any two objectives.

Static MOPs

$$\min \quad (f_1, f_2, f_3)^T$$

DMOPs

$$\min \quad \begin{cases} w_t f + (1-w_t) f_2 \\ w_t f + (1-w_t) f_3 \end{cases}$$

w_t: time-dependent weight

- <u>This framework does not provide well-defined test problems</u>

FDA Test Suite by Farina *et al.* (2004)

- 3 ZDT (2-objective) based & 2 DTLZ based (3-objective) problems
- FDA problems based on ZDT
  - Problem definition: $\min F = (f_1(x,t), g(x,t)h(x, f_1(x,t), g(x,t), t))^T$
  - Scenario 1: time-varying $g(x,t) = 1 + \sum (x_i - G(t))^2$
  - Scenario 2: time-varying $h(x, f_1, g, t) = 1 - \left(\frac{f_1}{g}\right)^{(H(t)+\sum(x_i - H(t))^2)^{-1}}$
  - Scenario 3: time-varying $f_1(x,t) = \sum x_i^{F(t)}, \quad F(t) > 0$
  - Scenario 4: change g, h, and f1 functions simultaneously
- FDA problems based on DTLZ
  - Problem definition:

$$
\begin{aligned}
&\text{Min.} && f_1(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos\left(\tfrac{x_1 \pi}{2}\right) \cdots \cos\left(\tfrac{x_{M-1}\pi}{2}\right) \\
&\text{Min.} && f_2(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos\left(\tfrac{x_1 \pi}{2}\right) \cdots \sin\left(\tfrac{x_{M-1}\pi}{2}\right) \\
&\vdots && \vdots \\
&\text{Min.} && f_M(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \sin\left(\tfrac{x_1 \pi}{2}\right) \\
&\text{with} && g(\mathbf{x}_M) = \textstyle\sum_{x_i \in \mathbf{x}_M} (x_i - 0.5)^2 \\
& && 0 \le x_i \le 1, \text{ for } i = 1, 2, \ldots, n
\end{aligned}
$$

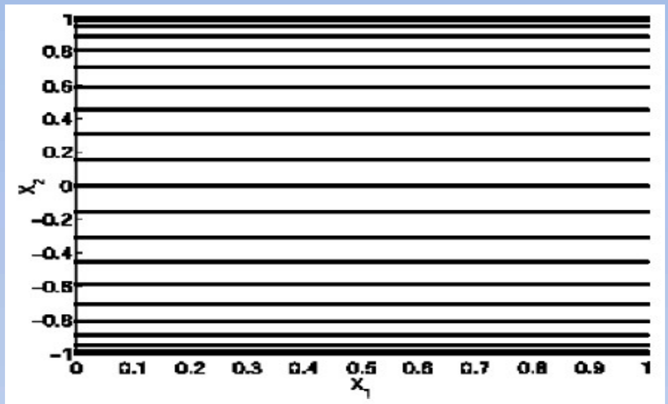
  - Scenario 1: the change of $g(x_M, t) = G(t) + \sum (x_i - G(t))^2, G(t) > 0$
  - Scenario 2: the change of $x_i \rightarrow x_i^{F(x)}$
  - Scenario 3: PF shape variation, the change of $g(x,t) \rightarrow g(x,t) + K_i(t)$ for each objective function

FDA Test Suite by Farina *et al.* (2004) - 2

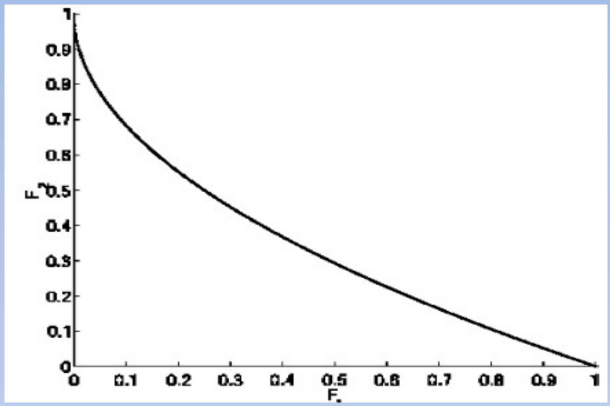
FDA1 (Type I)

$$\begin{cases} f_1(\mathbf{x}_I) = x_1 \\ g(\mathbf{x}_{II}) = 1 + \sum_{x_i \in \mathbf{x}_{II}} (x_i - G(t))^2 \\ h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}} \\ G(t) = \sin(0.5\pi t), \quad t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_T} \right\rfloor \\ \mathbf{x}_I = (x_1) \in [0,1], \quad \mathbf{x}_{II} = (x_2, \ldots, x_n) \in [-1,1] \end{cases}$$

PS



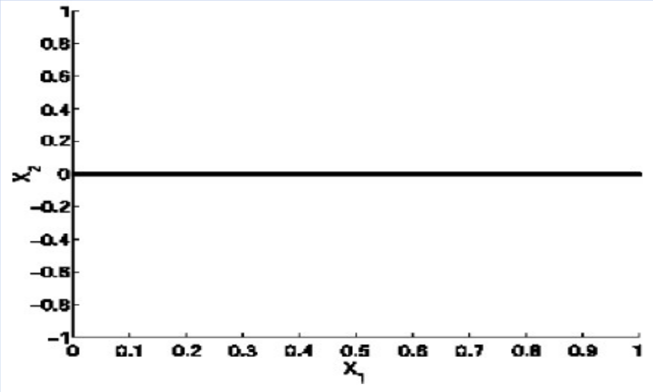
PF



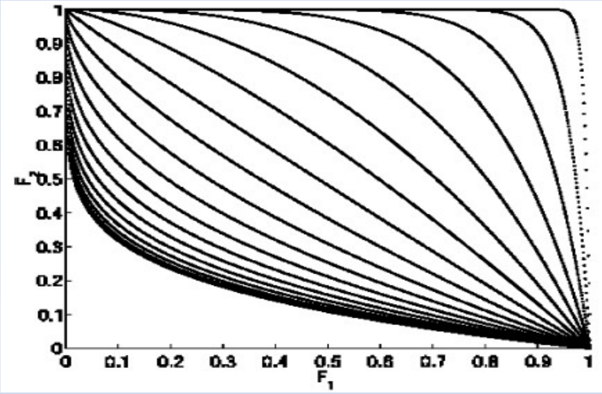
FDA2 (Type III)

$$\begin{cases} f_1(\mathbf{x}_I) = x_1 \\ g(\mathbf{x}_{II}) = 1 + \sum_{x_i \in \mathbf{x}_{II}} x_i^2 \\ h(\mathbf{x}_{III}, f_1, g) = 1 - \left(\frac{f_1}{g}\right)^{\left(H(t) + \sum_{x_i \in \mathbf{x}_{III}} (\mathbf{x}_i - H(t))^2\right)^{-1}} \\ H(t) = 0.75 + 0.7\sin(0.5\pi t), \quad t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_T} \right\rfloor \\ \mathbf{x}_I = (x_1) \in [0,1], \quad \mathbf{x}_{II}, \mathbf{x}_{III} \in [-1,1] \end{cases}$$

PS



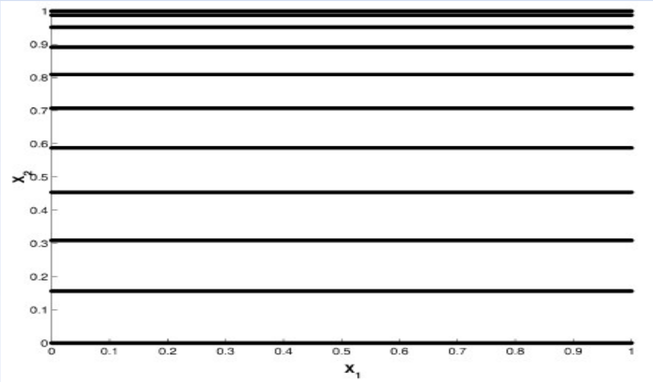
PF



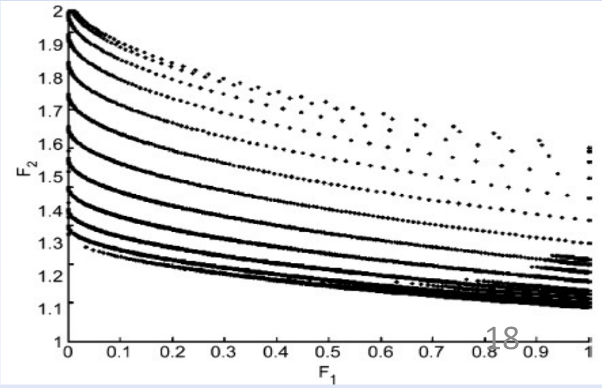
FDA3 (Type II)

$$\begin{cases} f_1(\mathbf{x}_I) = \sum_{x_i \in \mathbf{x}_I} x_i^{F(t)} \\ g(\mathbf{x}_{II}) = 1 + G(t) + \sum_{x_i \in \mathbf{x}_{II}} (x_i - G(t))^2 \\ h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}} \\ G(t) = |\sin(0.5\pi t)| \\ F(t) = 10^{2\sin(0.5\pi t)}, \quad t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_T} \right\rfloor \\ \mathbf{x}_I \in [0,1] \quad \mathbf{x}_{II} \in [-1,1] \end{cases}$$

PS



PF



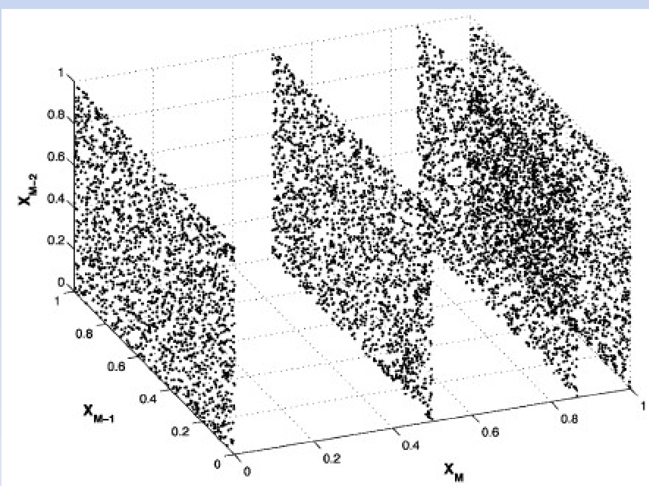
FDA Test Suite by Farina *et al.* (2004) - 3

FDA4 (Type I)

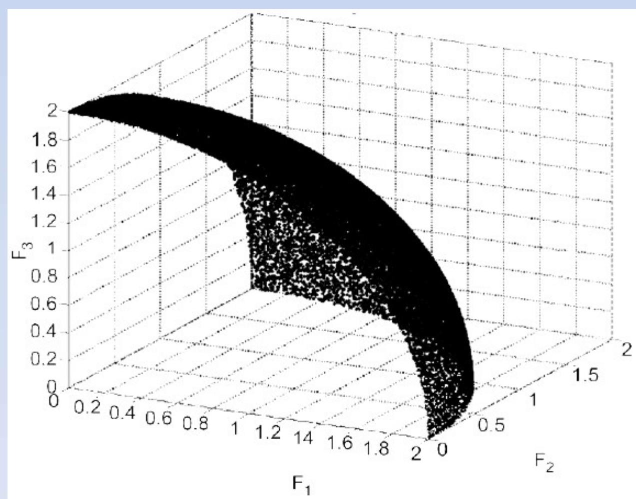
$$
\begin{cases}
\min_{\mathbf{x}} & f_1(\mathbf{x}) = (1+g(\mathbf{x}_{II}))\prod_{i=1}^{M-1}\cos\left(\frac{x_i\pi}{2}\right)\\
\min_{\mathbf{x}} & f_k(\mathbf{x}) = (1+g(\mathbf{x}_{II}))\left(\prod_{i=1}^{M-k}\cos\left(\frac{x_i\pi}{2}\right)\right)\\
 & \sin\left(\frac{x_{M-k+1}\pi}{2}\right), \quad k = 2:M-1\\
\min_{\mathbf{x}} & f_M(\mathbf{x}) = (1+g(\mathbf{x}_{II}))\sin\left(\frac{x_1\pi}{2}\right)\\
\text{where} & g(\mathbf{x}_{II}) = \sum_{x_i\in\mathbf{x}_{II}}(x_i - G(t))^2\\
 & G(t) = |\sin(0.5\pi t)|, \quad t = \frac{1}{n_t}\left\lfloor\frac{\tau}{\tau_T}\right\rfloor\\
 & \mathbf{x}_{II} = (x_M,\ldots,x_n),\\
 & x_i \in [0,1] \quad i = 1:n
\end{cases}
$$

FDA5 (Type II)

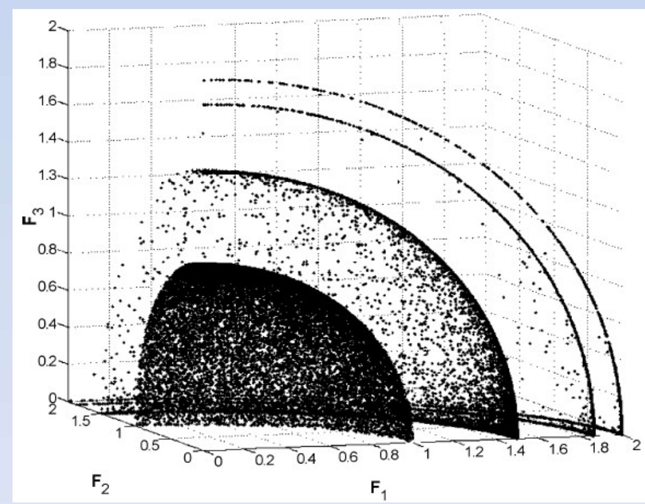
$$
\begin{cases}
\min_{\mathbf{x}} & f_1(\mathbf{x}) = (1+g(\mathbf{x}_{II}))\prod_{i=1}^{M-1}\cos\left(\frac{y_i\pi}{2}\right)\\
\min_{\mathbf{x}} & f_k(\mathbf{x}) = (1+g(\mathbf{x}_{II}))\left(\prod_{i=1}^{M-k}\cos\left(\frac{y_i\pi}{2}\right)\right)\\
 & \sin\left(\frac{y_{M-k+1}\pi}{2}\right) \quad k = 2:M-1\\
\min_{\mathbf{x}} & f_M(\mathbf{x}) = (1+g(\mathbf{x}_{II}))\sin\left(\frac{y_1\pi}{2}\right)\\
\text{where} & g(\mathbf{x}_{II}) = G(t) + \sum_{x_i\in\mathbf{x}_{II}}(x_i - G(t))^2\\
 & y_i = x_i^{F(t)} \quad \text{for } i = 1,\ldots,(M-1)\\
 & G(t) = |\sin(0.5\pi t)|\\
 & F(t) = 1 + 100\sin^4(0.5\pi t)\\
 & t = \frac{1}{n_t}\left\lfloor\frac{\tau}{\tau_T}\right\rfloor\\
 & \mathbf{x}_{II} = (x_M,\ldots,x_n), \quad x_i \in [0,1], \quad i = 1:n
\end{cases}
$$



PS of FDA4 & FDA5



PF of FDA4



PF of FDA5

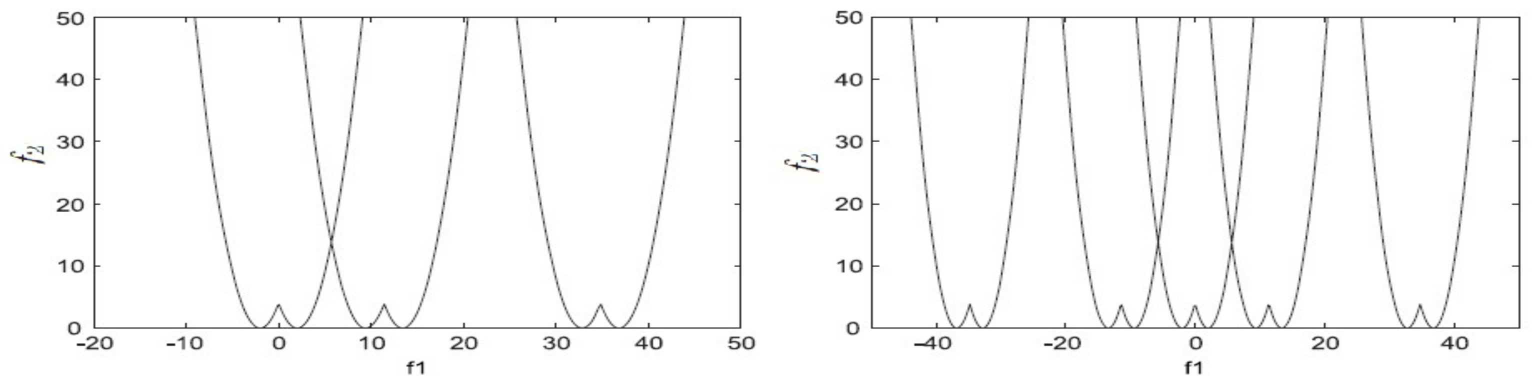
DSW Test Problems by Mehnen *et al.* (2006)

- Motivated by single-objective unimodal sphere models

$$\text{DSW} = \begin{cases} Minimise\colon\ f(\mathbf{x},t) = (f_1(\mathbf{x},t),\ f_2(\mathbf{x},t)) \\ f_1(\mathbf{x},t) = (a_{11}x_1 + a_{12}|x_1| - b_1 G(t))^2 + \sum_{i=2}^{n} x_i^2 \\ f_2(\mathbf{x},t) = (a_{21}x_1 + a_{22}|x_1| - b_2 G(t) - 2)^2 + \sum_{i=2}^{n} x_i^2 \\ where\colon\ G(t) = t(\tau)s,\ t(\tau) = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ \qquad s \text{ representing the severity of change} \end{cases}$$

- Three cases generated by varying a, b parameters
  - DSW1: time-changing continuous PS bounds
  - DSW2: time-changing disconnected PS $[G(t), G(t)+2]$
  - DSW3: time-changing PS and PF $[-G(t)-2, G(t)] \cup [G(t), G(t)+2]$

DSW1: $\mathbf{x} \in [-50, 50]^n, a_{11} = 1, a_{12} = 0, a_{21} = 1, a_{21} = 0, b_1 = b_2 = 1$
DSW2: $\mathbf{x} \in [-50, 50]^n, a_{11} = 0, a_{12} = 1, a_{21} = 0, a_{21} = 1, b_1 = b_2 = 1$
DSW3: $\mathbf{x} \in [-50, 50]^n, a_{11} = 1, a_{12} = 0, a_{21} = 1, a_{21} = 0, b_1 = 0, b_2 = 1$

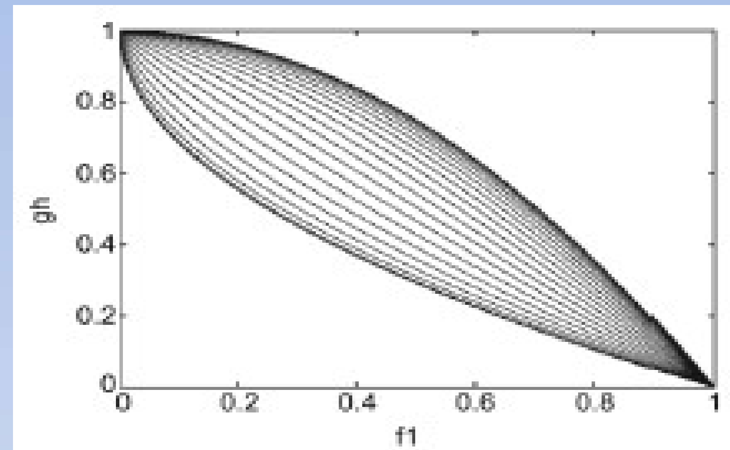


POF of DSW1 POF of DSW2

dMOP Test Suite by Goh & Tan (2007)

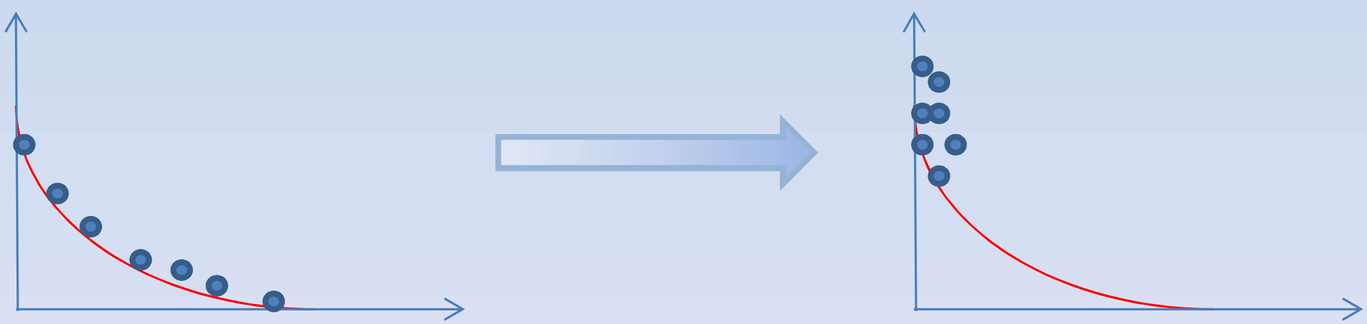
- Derived from ZDT problems and FDA problems
- Three bi-objective problems
  - dMOP1-2 similar to FDA

$$\text{dMOP1} = \begin{cases} \textit{Minimise: } f(\mathbf{x}, t) = (f_1(\mathbf{x}_{\text{I}}), g(\mathbf{x}_{\text{II}}) \cdot h(f_1(\mathbf{x}_{\text{I}}), g(\mathbf{x}_{\text{II}}), t)) \\ f_1(\mathbf{x}_{\text{I}}) = x_1 \\ g(\mathbf{x}_{\text{II}}) = 1 + 9 \sum_{x_i \in \mathbf{x}_{\text{II}}} (x_i)^2 \\ h(f_1, g, t) = 1 - \left(\frac{f_1}{g}\right)^{H(t)} \\ \textit{where:} \\ H(t) = 0.75 \sin(0.5\pi t) + 1.25, \quad t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ x_i \in [0, 1]; \quad \mathbf{x}_{\text{I}} = (x_1); \quad \mathbf{x}_{\text{II}} = (x_2, \ldots, x_n) \end{cases}$$



POF of dMOP1

  - dMOP3 involves randomness and marked diversity loss



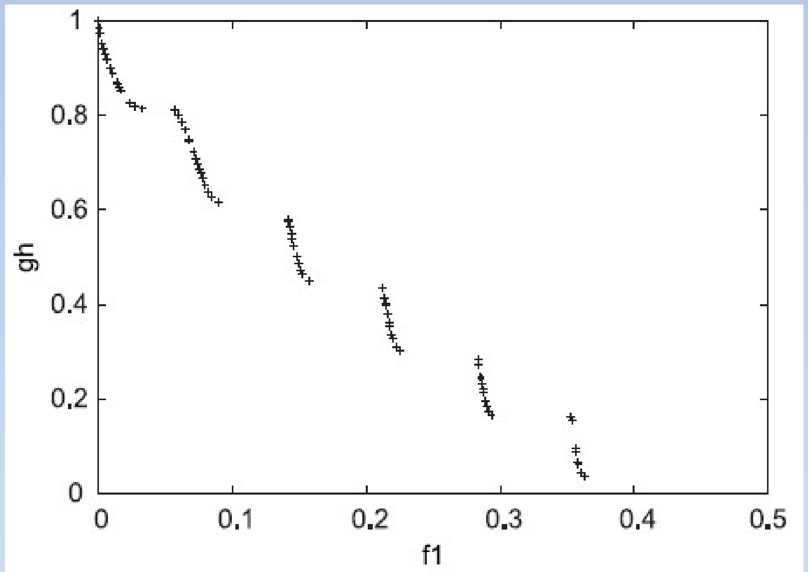
objective space,
population at time t

objective space,
population at time t+1

HE Test Suite by Helbig & Engelbrecht (2013,2014)

- Adding WFG (Huband et al. 2006) characteristics:
  - isolated PFs
  - deceptive PFs
  - ...

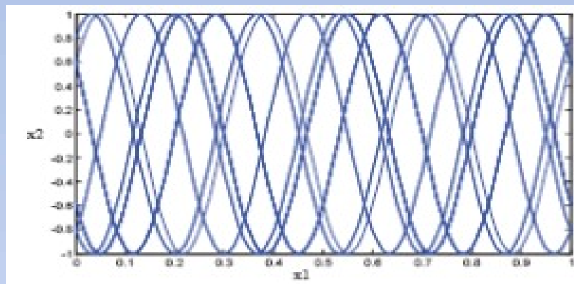
$$
\text{HE1} = \begin{cases}
\text{Minimise: } f(\mathbf{x}, t) = (f_1(\mathbf{x}_{\text{I}}), g(\mathbf{x}_{\text{II}}) \cdot h(f_1(\mathbf{x}_{\text{I}}), g(\mathbf{x}_{\text{II}}), t)) \\
f_1(\mathbf{x}_{\text{I}}) = x_1 \\
g(\mathbf{x}_{\text{II}}) = 1 + \frac{9}{n-1} \sum_{x_i \in \mathbf{x}_{\text{II}}} x_i \\
h(f_1, g, t) = 1 - \sqrt{\frac{f_1}{g}} - \frac{f_1}{g} \sin(10\pi t f_1) \\
\text{where:} \\
t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\
x_i \in [0, 1]; \quad \mathbf{x}_{\text{I}} = (x_1); \quad \mathbf{x}_{\text{II}} = (x_2, \ldots, x_n)
\end{cases}
$$



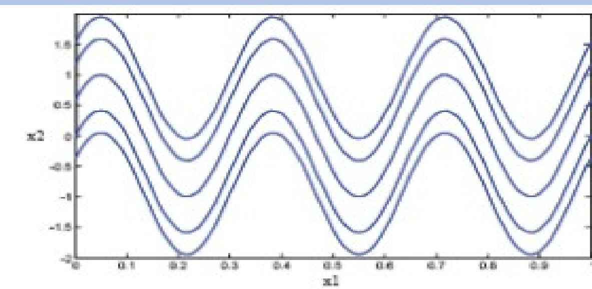
- <u>However, main optimization difficulties come from WFG characteristics instead of introduced dynamics</u>

UDF Test Suite by Biswas et al. (2014)

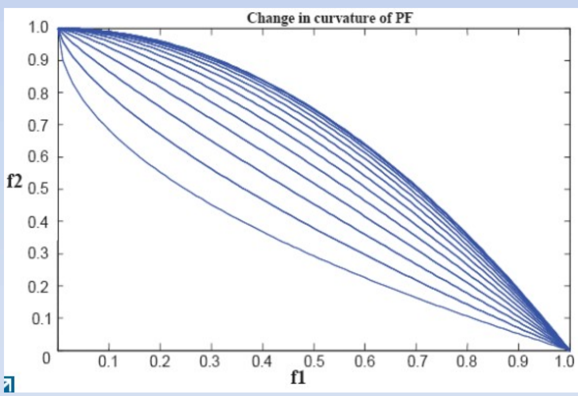
- Based on UF problems (Zhang et al. 2009)
- General techniques to design DMOPs
  - Shifting
  - Shape variation
  - Slope variation
  - Curvature variation
  - ...

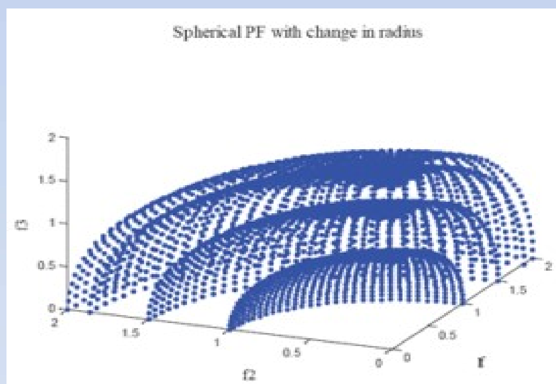


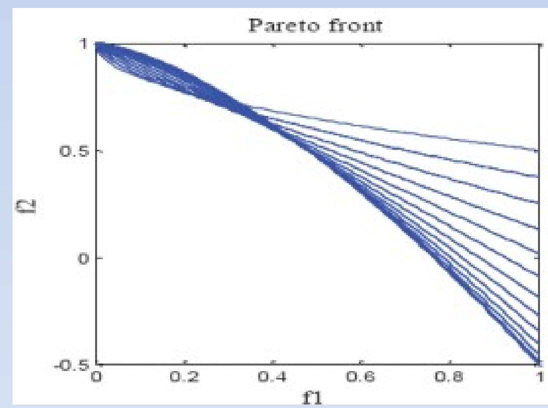
(a) Horizontal Shift of the PS in 2D



(b) Vertical Shift of the PS in 2D







The F (ZJZ) Test Suite by Zhou et al. (2014)

- Based on UF test problems (Zhang et al. 2009)
- F1-F4 are the same as FDA1-FDA4 (Farina et al. 2004)
- Involving strong nonlinear variable linkages

$F5 \quad [0,5]^n$

$f_1(x,t) = |x_1 - a|^H + \sum_{i \in I_1} y_i^2,$
$f_2(x,t) = |x_1 - a - 1|^H + \sum_{i \in I_2} y_i^2,$
$y_i = x_i - b - 1 + |x_1 - a|^{H+\frac{i}{n}},\ H = 1.25 + 0.75\sin(\pi \frac{t}{n_T}),$
$a = 2\cos(\pi \frac{t}{n_T}) + 2,\ b = 2\sin(2\pi \frac{t}{n_T}) + 2,$
$I_1 = \{i | 1 \le i \le n, i \text{ is odd}\},\ I_2 = \{i | 1 \le i \le n, i \text{ is even}\}.$
PS(t): $a \le x_1 \le a+1,\ x_i = b + 1 - |x_1 - a|^{H+\frac{i}{n}}$, for $i = 2, \ldots, n$.
PF(t): $f_1 = s^H,\ f_2 = (1-s)^H,\ 0 \le s \le 1.$

$F6 \quad [0,5]^n$

$f_1(x,t) = |x_1 - a|^H + \sum_{i \in I_1} y_i^2,$
$f_2(x,t) = |x_1 - a - 1|^H + \sum_{i \in I_2} y_i^2,$
$y_i = x_i - b - 1 + |x_1 - a|^{H+\frac{i}{n}},\ H = 1.25 + 0.75\sin(\pi \frac{t}{n_T}),$
$a = 2\cos(1.5\pi \frac{t}{n_T})\sin(0.5\pi \frac{t}{n_T}) + 2,\ b = 2\cos(1.5\pi \frac{t}{n_T})\cos(0.5\pi \frac{t}{n_T}) + 2$
$I_1 = \{i | 1 \le i \le n, i \text{ is odd}\},\ I_2 = \{i | 1 \le i \le n, i \text{ is even}\}.$
PS(t): $a \le x_1 \le a+1,\ x_i = b + 1 - |x_1 - a|^{H+\frac{i}{n}}$, for $i = 2, \ldots, n$.
PF(t): $f_1 = s^H,\ f_2 = (1-s)^H,\ 0 \le s \le 1.$

$F7 \quad [0,5]^n$

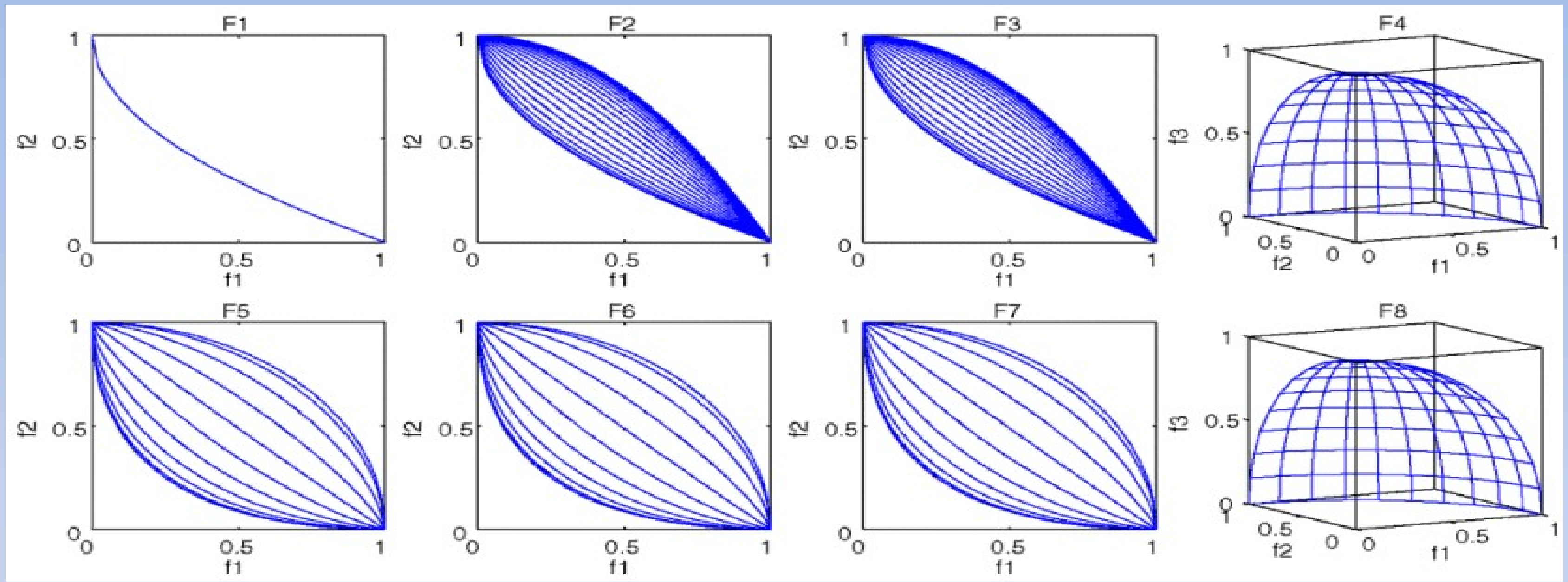
$f_1(x,t) = |x_1 - a|^H + \sum_{i \in I_1} y_i^2,$
$f_2(x,t) = |x_1 - a - 1|^H + \sum_{i \in I_2} y_i^2,$
$y_i = x_i - b - 1 + |x_1 - a|^{H+\frac{i}{n}},\ H = 1.25 + 0.75\sin(\pi \frac{t}{n_T}),$
$a = 1.7(1 - \sin(\pi \frac{t}{n_T}))\sin(\pi \frac{t}{n_T}) + 3.4,\ b = 1.4(1 - \sin(\pi \frac{t}{n_T}))\cos(\pi \frac{t}{n_T}) + 2.1$
$I_1 = \{i | 1 \le i \le n, i \text{ is odd}\},\ I_2 = \{i | 1 \le i \le n, i \text{ is even}\}.$
PS(t): $a \le x_1 \le a+1,\ x_i = b + 1 - |x_1 - a|^{H+\frac{i}{n}}$, for $i = 2, \ldots, n$.
PF(t): $f_1 = s^H,\ f_2 = (1-s)^H,\ 0 \le s \le 1.$

$F8 \quad [0,1]^2 \times [-1,2]^{n-2}$

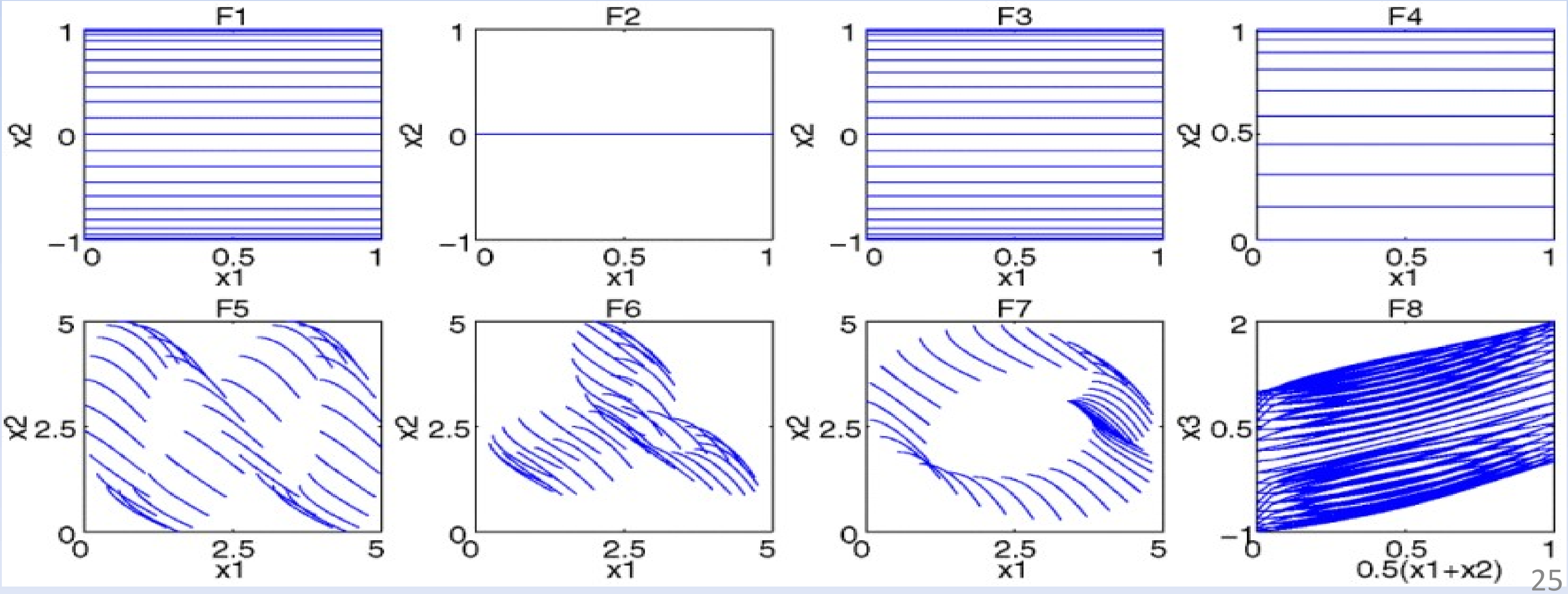
$f_1(x,t) = (1+g)\cos(0.5\pi x_2)\cos(0.5\pi x_1),$
$f_2(x,t) = (1+g)\cos(0.5\pi x_2)\sin(0.5\pi x_1),$
$f_3(x,t) = (1+g)\sin(0.5\pi x_2),$
$g = \sum_{i=3}^{n}(x_i - (\frac{x_1+x_2}{2})^H - G)^2,$
$H = 1.25 + 0.75\sin(\pi \frac{t}{n_T}),\ G = \sin(0.5\pi \frac{t}{n_T}).$
PS(t): $0 \le x_1, x_2 \le 1,\ x_i = (\frac{x_1+x_2}{2})^H + G(t)$, for $i = 3, \ldots, n$.
PF(t): $f_1 = \cos(u)\cos(v),\ f_2 = \cos(u)\sin(v),\ f_3 = \sin(u),\ 0 \le u, v \le \pi/2.$

The F (ZJZ) Test Suite by Zhou et al. (2014)

PF:



PS:

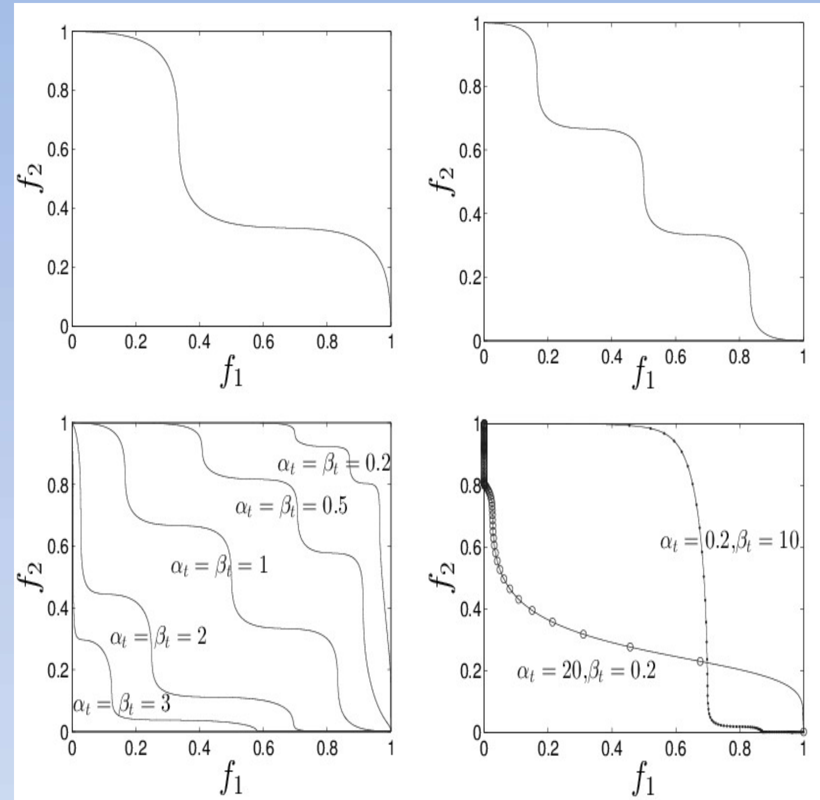


JY Generator by Jiang & Yang (2017a)

- Focusing on dynamics analysis

$$JY=\begin{cases}\min \quad (f_1(x,t), f_2(x,t))^T \\ f_1(x,t)=(1+g(x,t))(h(x)+A_t\sin(W_t\pi h(x)))^{\alpha_t} \\ f_2(x,t)=(1+g(x,t))(1-h(x)+A_t\sin(W_t\pi h(x)))^{\beta_t}\end{cases}$$

PF: $$f_1^{\frac{1}{\alpha_t}}+f_1^{\frac{1}{\beta_t}}=1+2A_t\sin\left(W_t\pi\frac{f_1^{\frac{1}{\alpha_t}}-f_1^{\frac{1}{\alpha_t}}+1}{2}\right)$$



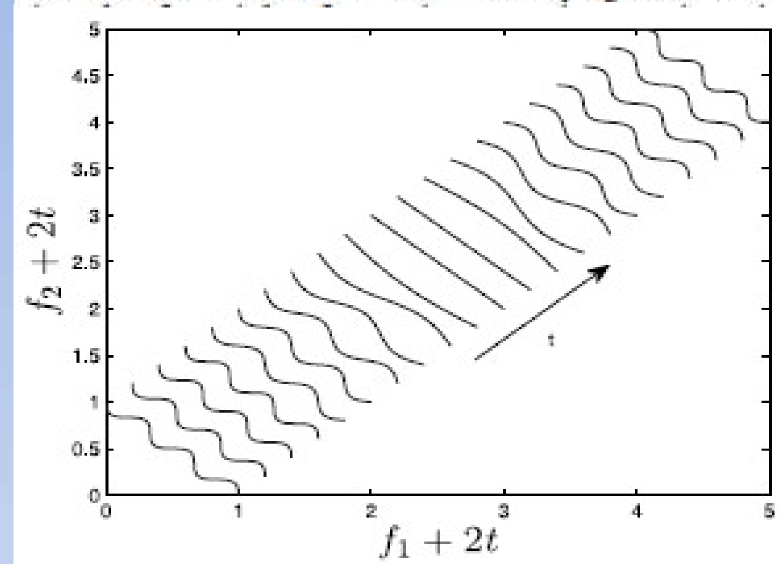
- Characteristics:
  - PF is a sin wave after a clockwise rotation
  - The PF has mixed concave and convex segments
  - Time-varying segments controlled by W_t
  - Time-varying curvature controlled by A_t
  - Various types of problems, e.g. , randomness, knee regions, dis-connectivity
  - Easy to scale up in terms of objectives

JY Generator by Jiang & Yang (2017a) - 2

- JY2: time-changing PS and PF

$$JY2: \begin{cases} \min \quad F(\mathbf{x},t) = (f_1(\mathbf{x},t), f_2(\mathbf{x},t))^T \\ f_1(\mathbf{x},t) = (1+g(\mathbf{x_{II}},t))(x_1 + A_t \sin(W_t \pi x_1)) \\ f_2(\mathbf{x},t) = (1+g(\mathbf{x_{II}},t))(1 - x_1 + A_t \sin(W_t \pi x_1)) \\ g(\mathbf{x_{II}},t) = \sum_{x_i \in \mathbf{x_{II}}} (x_i - G(t))^2, G(t) = \sin(0.5\pi t) \\ A(t) = 0.05, W(t) = \lfloor 6\sin(0.5\pi(t-1)) \rfloor \\ \mathbf{x_I} = (x_1) \in [0,1], \mathbf{x_{II}} = (x_2, \ldots, x_n) \in [-1,1]^{n-1} \end{cases}$$

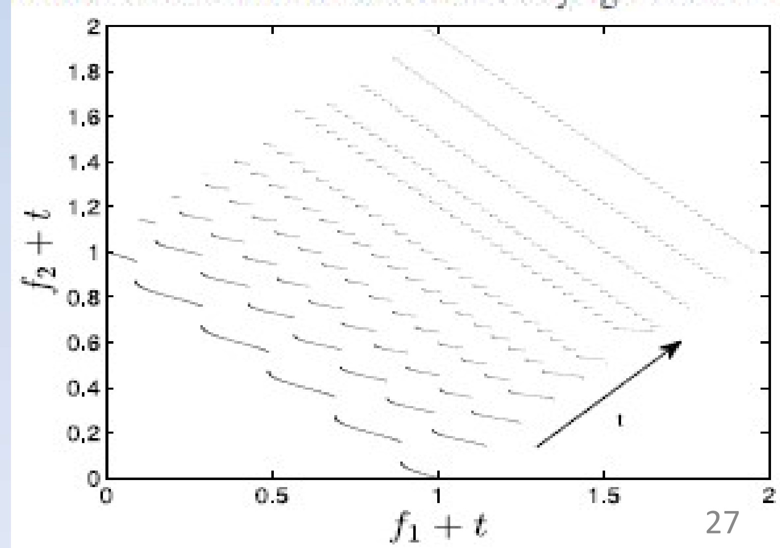
POF of $JY2$ with 21 time windows varying from 0 to 2.



- JY4: time-changing PS and PF, time-changing disconnectivity

$$JY4: \begin{cases} \min \quad F(\mathbf{x},t) = (f_1(\mathbf{x},t), f_2(\mathbf{x},t))^T \\ f_1(\mathbf{x},t) = (1+g(\mathbf{x_{II}},t))(x_1 + A_t \sin(W_t \pi x_1)) \\ f_2(\mathbf{x},t) = (1+g(\mathbf{x_{II}},t))(1 - x_1 + A_t \sin(W_t \pi x_1)) \\ g(\mathbf{x_{II}},t) = \sum_{x_i \in \mathbf{x_{II}}} (x_i - G(t))^2, G(t) = \sin(0.5\pi t) \\ A(t) = 0.05, \quad W(t) = 10^{1+|G(t)|} \\ \mathbf{x_I} = (x_1) \in [0,1], \mathbf{x_{II}} = (x_2, \ldots, x_n) \in [-1,1]^{n-1} \end{cases}$$

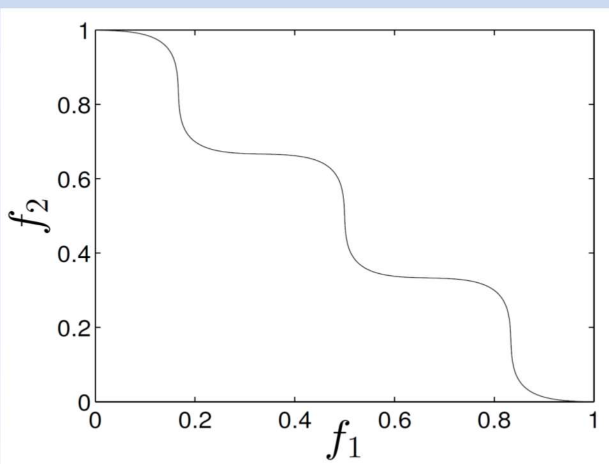
POF of $JY4$ with 11 time windows varying from 0 to 2.



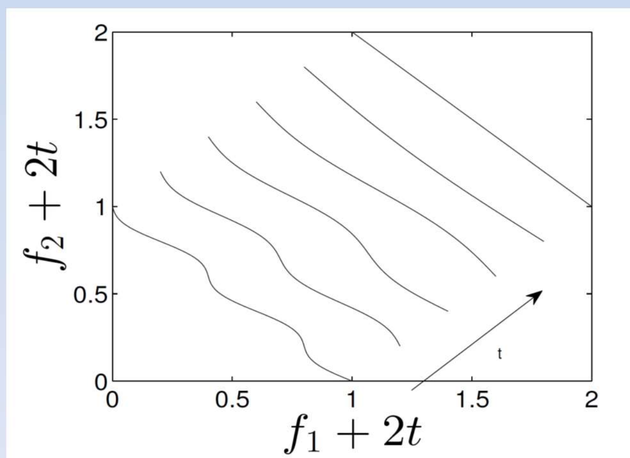
JY Generator by Jiang & Yang (2017a) - 3

- JY10: mixed type, sometimes PS remains static whereas sometimes PS changes over time. PF has the same dynamics

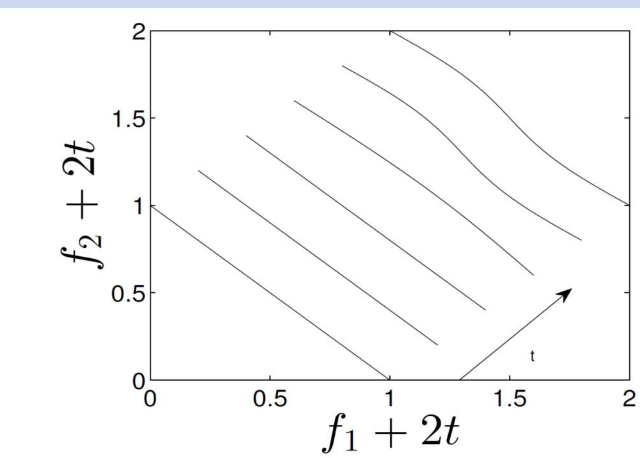
$$
JY10: \begin{cases}
min \quad F(\mathbf{x},t) = (f_1(\mathbf{x},t), f_2(\mathbf{x},t))^T \\
f_1(\mathbf{x},t) = (1+g(\mathbf{x_{II}},t))(x_1 + A_t \sin(W_t \pi x_1))^{\alpha_t} \\
f_2(\mathbf{x},t) = (1+g(\mathbf{x_{II}},t))(1-x_1+A_t \sin(W_t \pi x_1))^{\beta_t} \\
g(\mathbf{x_{II}},t) = \sum_{x_i \in \mathbf{x_{II}}} (x_i + \sigma - G(t))^2, G(t) = |sin(0.5\pi t)| \\
A(t) = 0.05, \quad W(t) = 6 \\
\alpha_t = \beta_t = 1 + \sigma G(t), \sigma \equiv (\lfloor \frac{\tau}{\tau_t \rho_t} \rfloor + R) \pmod 3 \\
\mathbf{x_I} = (x_1) \in [0,1], \mathbf{x_{II}} = (x_2, \dots, x_n) \in [-1,1]^{n-1},
\end{cases}
$$



Static PF, dynamic PS



Dynamic PF, dynamic PS



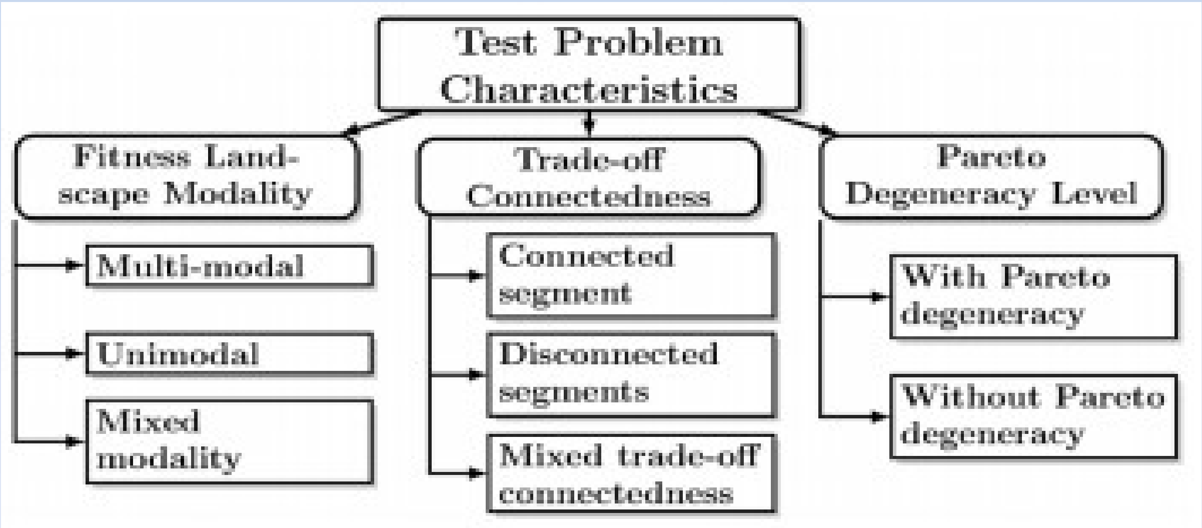
Dynamic PF, static PS

GTA Test Suite by Gee et al. (2017)

- Problems based on the framework by Li and Zhang (2009)

$$\begin{cases} f_1(x,t) = \alpha_{A,1}(x_I,t) + \beta_{A,1}(x_{II} - g_A(x_I,t),t) \\ \vdots \\ f_M(x,t) = \alpha_{A,M}(x_I,t) + \beta_{A,M}(x_{II} - g_A(x_I,t),t) \end{cases}$$

  - $\alpha_{A,i}(x_I,t)$: PF-associated function for objective i
  - $\beta_{A,i}(x,t)$: PS-associated function for objective i
  - $g_A(x_I,t)$: distance-related function to the PF
- Some characteristics generated by changing three functions



Dynamic Multiple Knapscak Problems (DMKPs)

- Static multiple knapsack problems:
  - Given m knapsacks with their own fixed capacities and n items, each item with a weight and a profit to each knapsack, select items to fill up the m knapsacks to maximize the profit vector while satisfying each knapsack's capacity constraint
- The DMKP (Farina et al. 2004):
  - Constructed by changing weights and profits of items, and/or knapsack capacity over time as:

$$\max f_i(x,t) = \sum_{j=1} p_{ij}(t) x_j, \quad i = 1:M$$

$$s.t. \quad \sum_{j=1} w_{ij}(t) x_j \le c_i(t), \quad i = 1:M$$

$$x_i \in \{0,1\}^n$$

- x_i : indicates whether item i is included or not
- p_{ij} : profit and weight of item i to knapsack j at time t
- c_i : the capacity of knapsack i at time t.

DMOPs: Common Characteristics

- Common characteristics of DMOPs in the literature:
  - Most DMOPs are non time-linkage problems
  - For almost all DMOPs, changes are assumed to be detectable (unable to test detection techniques)
  - In most cases, objective functions are changed/optima are shifted
  - Many DMOPs have noise-free changes
  - Most DMOPs have cyclic/recurrent changes
  - Most DMOPs are simple modifications of existing static counterparts

Performance Measures

- For static MOPs, performance measures focus on
  - Convergence: GD, IGD, C-metric...
  - Diversity: Spacing, maximum spread, ...
- For DMOPs, more measured aspects and indicators
  - Averaged measure values of a sequence of static period
    - Mean GD/IGD/SP/HV...
  - Behavior-based performance measures
    - Reactivity
    - Stability
    - Robustness
    - ...

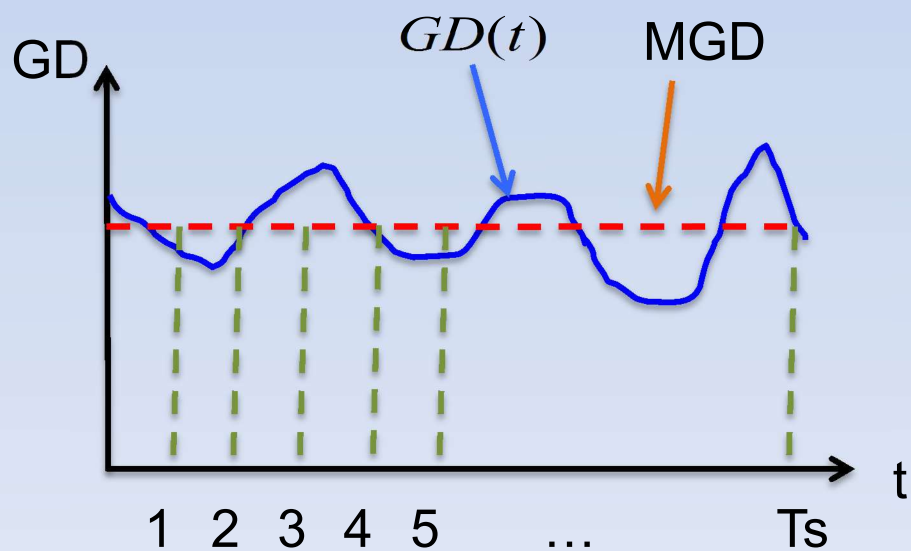
Performance Measures: Examples

- Mean of generational distance (MGD)

$$MGD = \frac{1}{T_s} \sum_{t=1}^{T_s} GD(t)$$

  - T_s: number of time steps
  - $GD(t)$: generational distance value at time t

- Similarly, mean value of other performance measures can be defined



Performance Measures: Examples

- Accuracy: How well an approximation (PF*) represents the true Pareto front (PF)
- Accuracy often accounts for both diversity and convergence
- Hypervolume (HV) is preferred in definition of accuracy, which measures the HV difference between PF* and PF at time t:

$$acc(t) = \left| HV(PF^{*}) - HV(PF) \right|$$

- It can also be defined as the ratio of HV(PF*) to HV(PF):

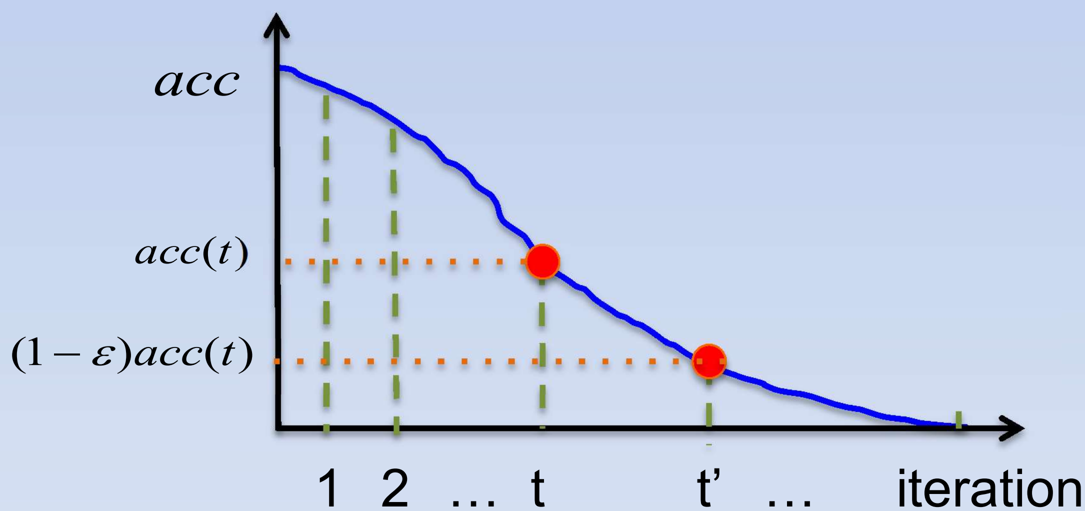
$$acc(t) = \frac{HV(PF^{*})}{HV(PF)}$$

Performance Measures: Examples

- Reactivity: how long it takes to reach a specified accuracy threshold (ε):

$$react(t,\varepsilon) = \min\{t' - t \mid t < t' < t_{max}, acc(t') \geq (1-\varepsilon)acc(t)\}$$

  - t_{max} : the maximum number of iterations/generations



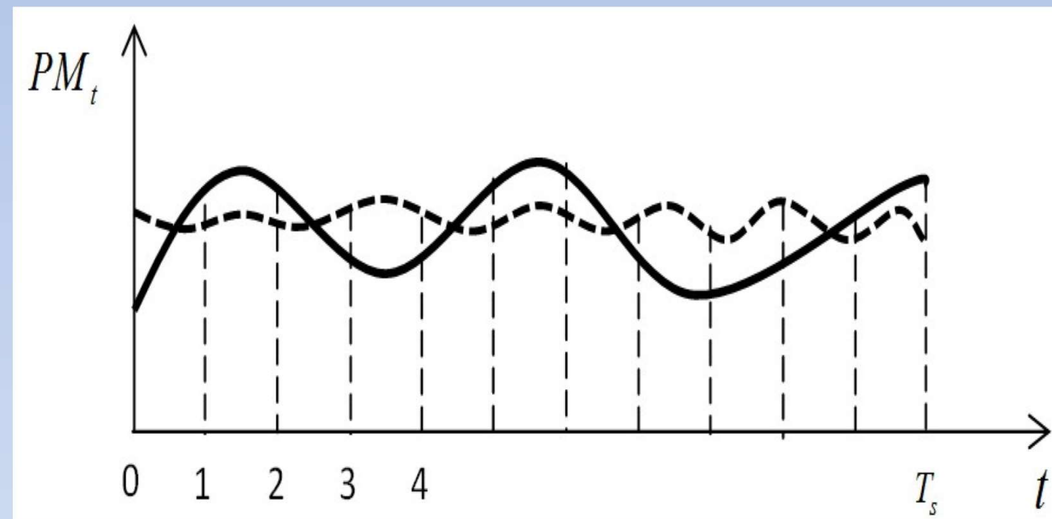
$$acc(t') \geq (1-\varepsilon)acc(t)$$

Performance Measures: Examples

- Robustness: used to describe the stability of the performance of an algorithm in a number of environmental changes, defined as:

$$R(PM) = \sqrt{\frac{1}{T_s - 1}\sum_{t=1}^{T_s}(PM_t - \overline{PM})^2}$$

where PM_t is the value of another performance metric at time t.



Part II: Approaches, Issues & Future Work

- Enhanced EC approaches for DMOPs
- Case studies
- Relevant issues
- Future work

EC for DMOPs: Things to Do

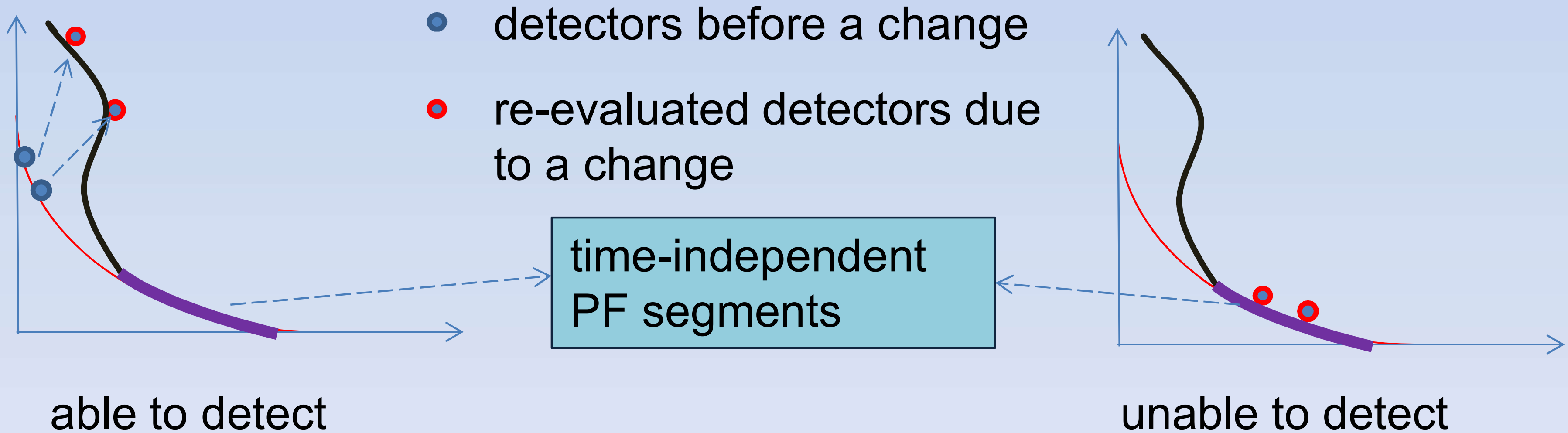
- To detect potential environmental changes
  - Success rate of detection
  - Cost of detection
- To track the changing PS/PF
  - To obtain a set of well-distributed solutions
  - To minimize the gap between approximations and the true PF
- To expect a steady and fast change response
- To reduce the cost of tracking (given the budget limit, i.e., time, memory)

EC for DMOPs: Detection Approaches

- Why is detection important ?
  - When a change occurs, non-dominated solutions in the archive may become dominated
  - EAs would get misled if archived solutions are not re-evaluated in time
  - Detection could help EAs learn more about the environments, and thus store useful information for future use
- Two ways of detecting changes:
  - Individual-level detection: fast but not robust
  - Population-level detection: slow but robust
  - Both methods could fail to detect changes (not 100% guaranteed)

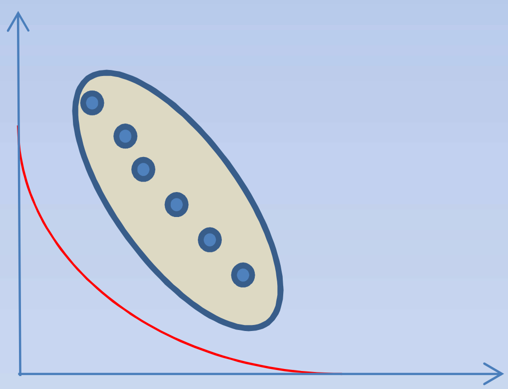
EC for DMOPs: Detection Approaches

- Individual-level detection
  - Re-evaluate some individuals' objective values before using them in every iteration/generation
  - Check the discrepancy between their current and previous objective values
- Success rate of detection depends on
  - Detectability of environmental changes
  - Location of detectors placed

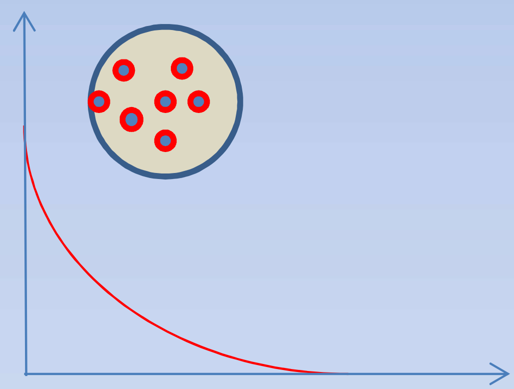


EC for DMOPs: Detection Approaches

- Population-level detection
  - Population-related statistical information, i.e., distribution, is assessed in every generation
  - Check the significance of variation in statistical information



population distribution
before a change



population distribution
after a change

- Less sensitive to noise but possibly higher computational cost

EC for DMOPs: Response Approaches

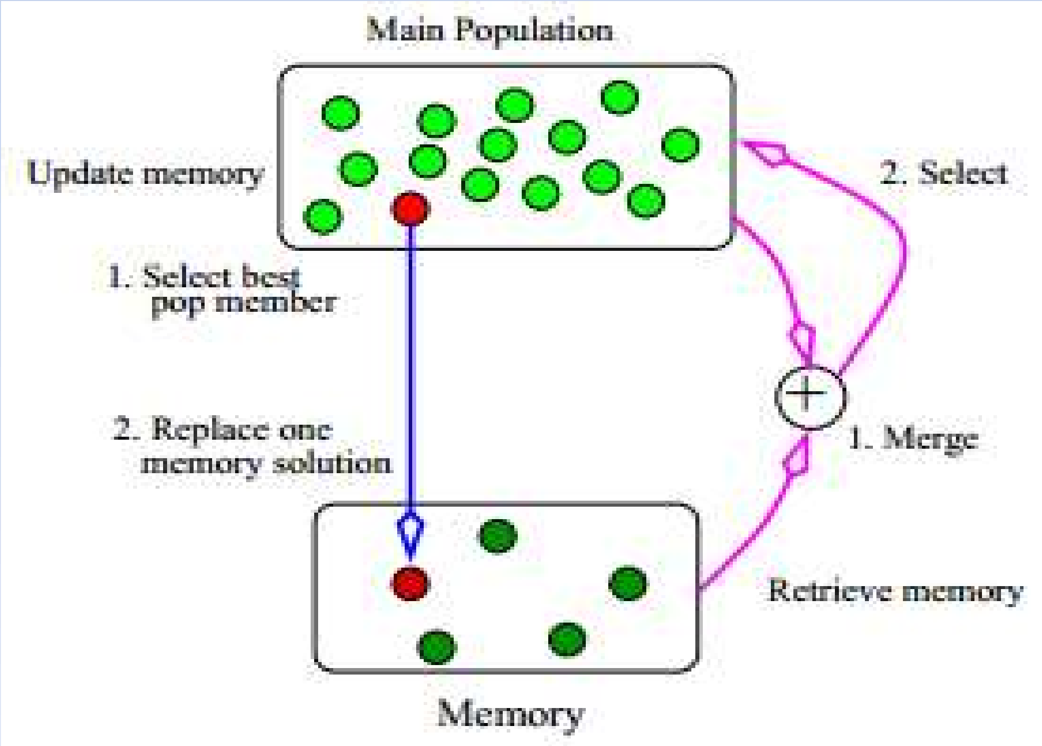
- How about restarting an EA after a change ?
  - Natural and easy choice
  - But, not good choice due to:
    - It may be inefficient, wasting computational resources
    - It may lead to very different solutions before and after a change. For real-world problems, we may expect solutions to remain similar
- Extra approaches are needed to enhance EAs for DMOPs

EC for DMOPs: Response Approaches

- Some approaches developed to enhance EAs for DMOPs
- Typical approaches:
  - Memory: store and reuse useful information
  - Diversity: handle convergence directly
  - Multi-population: co-operate sub-populations
  - Prediction: predict changes and respond in advance
- Their use depends on types of DMOPs
  - Predictability
  - Cyclicality
  - ...

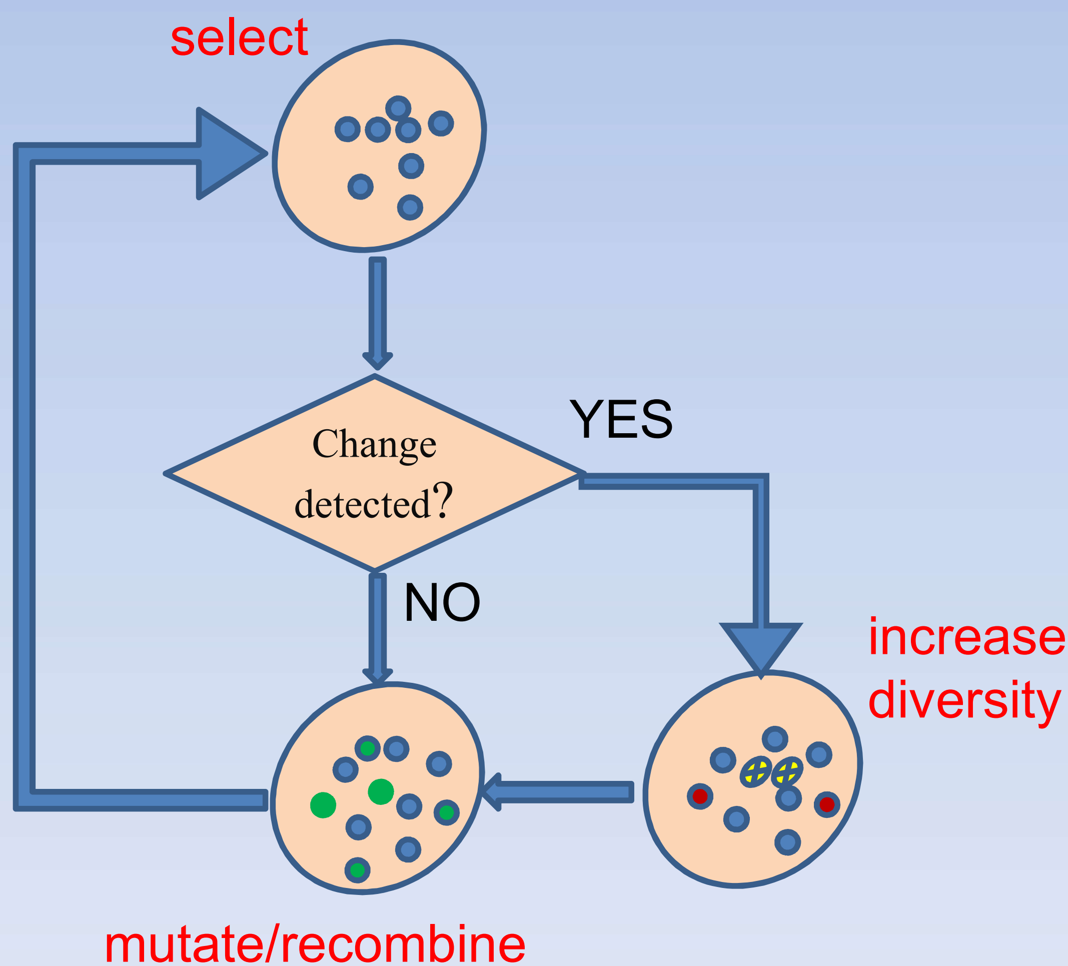
Memory-based Approaches

- For some DMOPs, optimal solutions repeatedly return to previous locations
- Memory: to store history information for future use
- Challenges:
  - What information to store?
  - When and how to retrieve memory?
  - How to update memory?



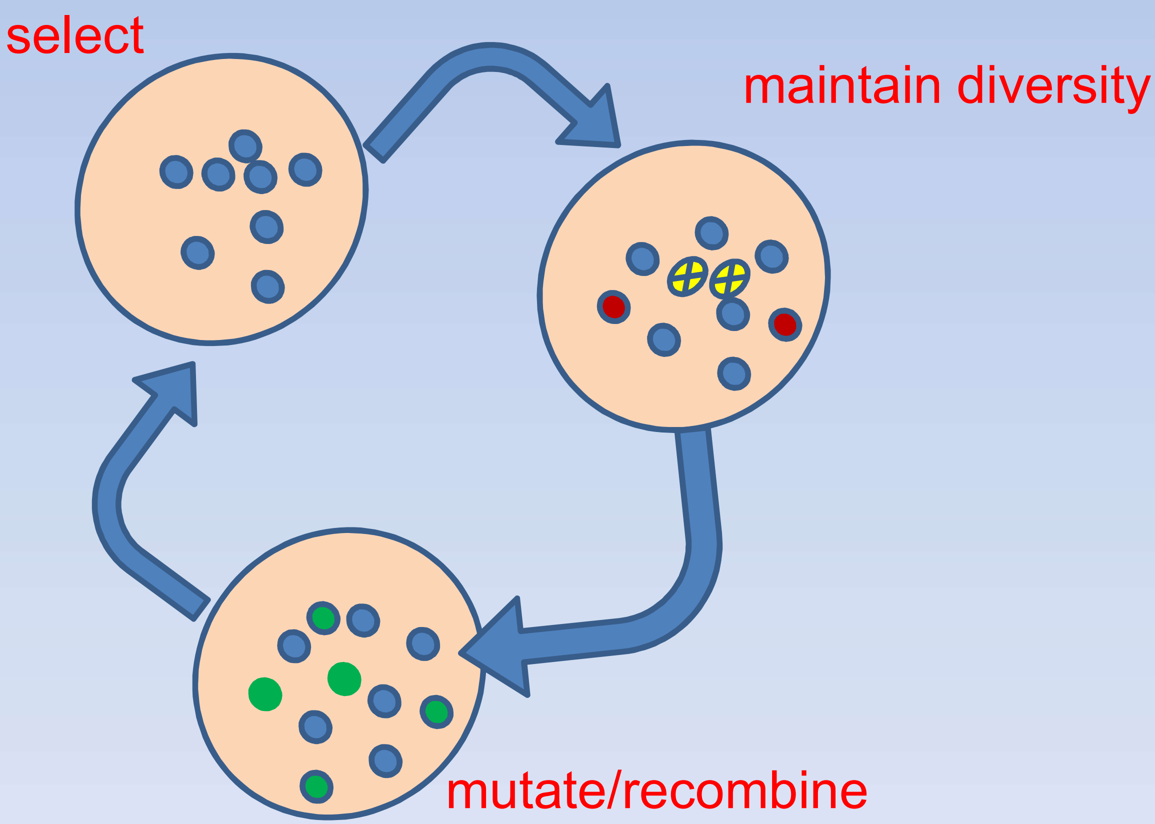
Diversity-based Approaches

- Diversity increase: introduce diversity upon the detection of landscape changes
  - Partially random restart
  - Hypermutation
  - Variable local search



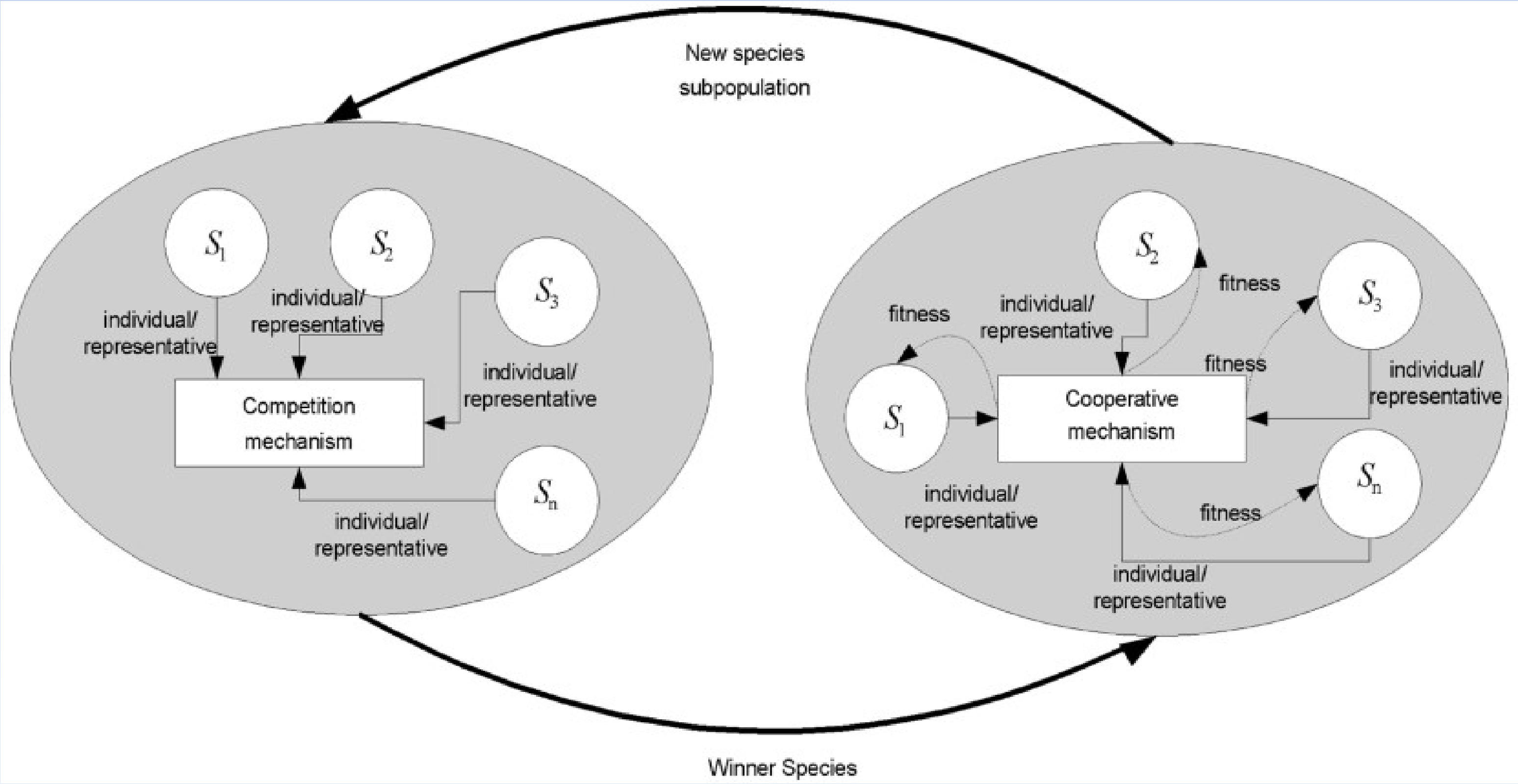
Diversity-based Approaches

- Diversity maintenance: maintain diversity throughout the run (even if no change occurs)
  - Random immigrants



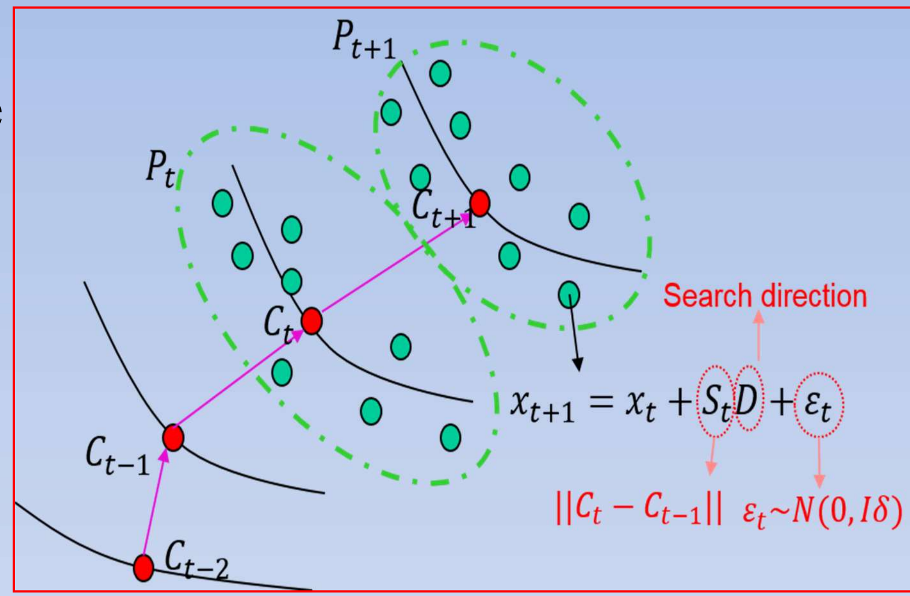
Multi-population Approaches

- Idea: split the population to conduct simultaneous exploration in different regions
- Subpopulations are competitive and/or cooperative (Goh & Tan 2009)
  - Cooperation process generates new species, which are used for the competition process
  - Competition process generates winner, which guides co-evolution of subpopulations



Prediction Approaches

- For some DMOPs, changes exhibit predictable patterns
- Often to predict:
  - The location of new PS after a change
  - When the next change may occur
  - How much a change will be



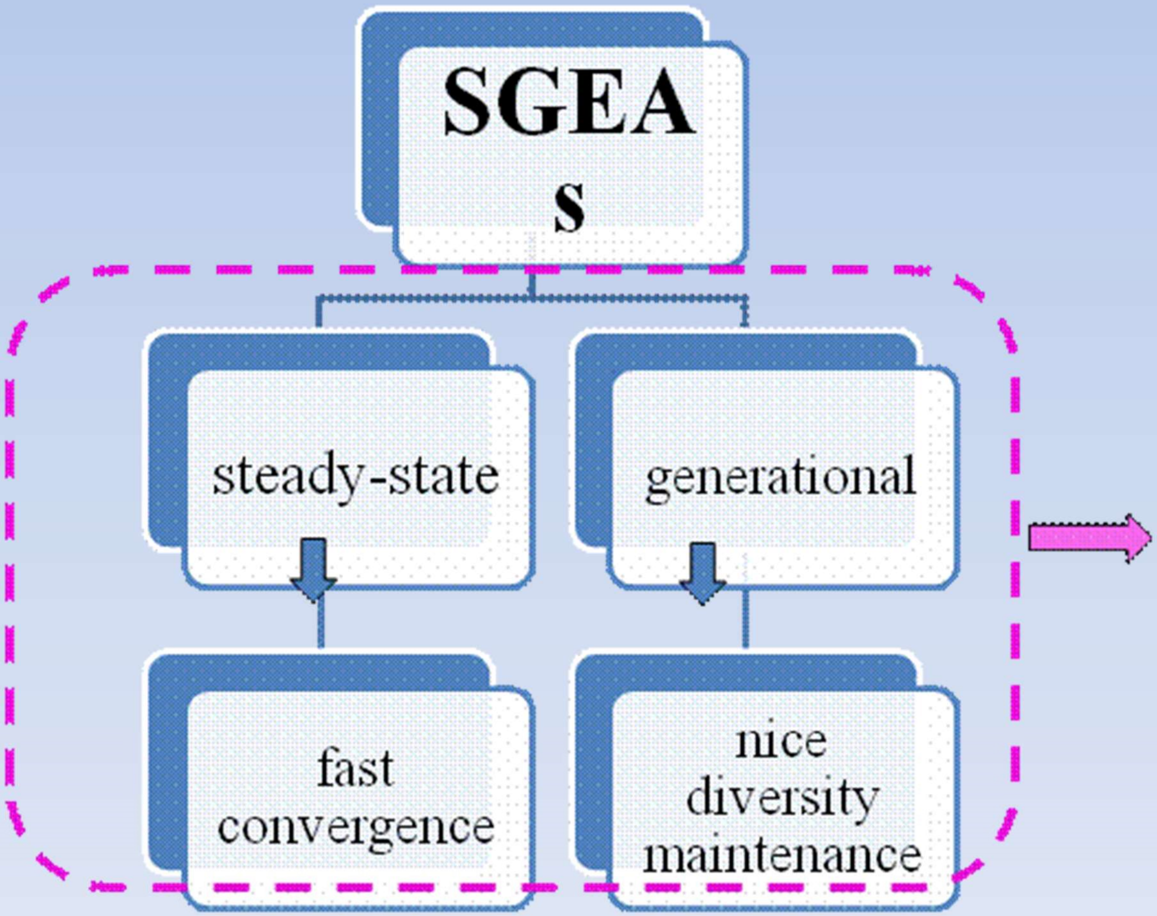
- Techniques:
  - Kalman filter (Muruganantham et al. 2016)
  - Population prediction strategy (Zhou et al. 2014)
  - Feed-forward prediction (Hatzakis & Wallace 2006)
  - Directed search strategy (Wu et al. 2015)
  - Evolutionary gradient search (Koo et al. 2010)
  - Center and knee points prediction (Zou at al. 2017)

Remarks on Enhancing Approaches

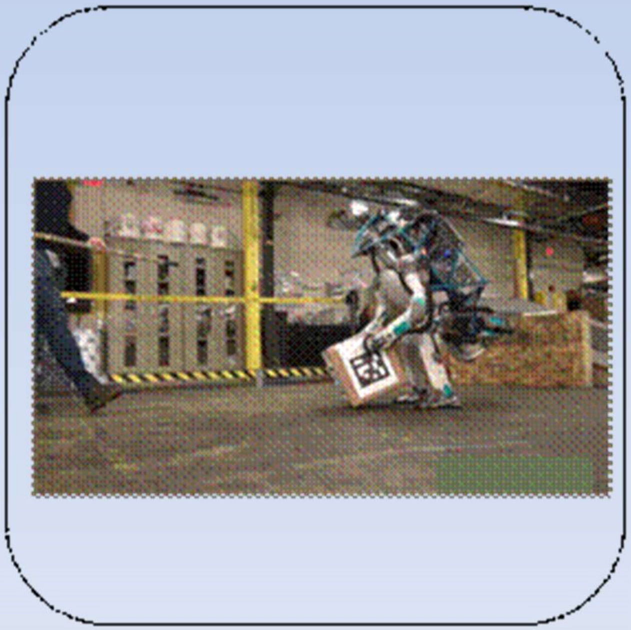
- No clear winner among the approaches
- Memory is efficient for cyclic environments
- Multi-population is good for multimodal problems
  - Able to maintain diversity
  - The search ability will decrease if too many sub-populations
- Diversity schemes are usually useful
  - Guided immigrants may be more efficient
- Thumb of rule: balancing exploration & exploitation over time

Case Study: EA for Continuous DMOPs

- Steady-Generational EA (SGEA)
  - Proposed by Jiang & Yang (2017b)
  - Hybrid of steady-state and generational methods
  - Novel steady-state change detection



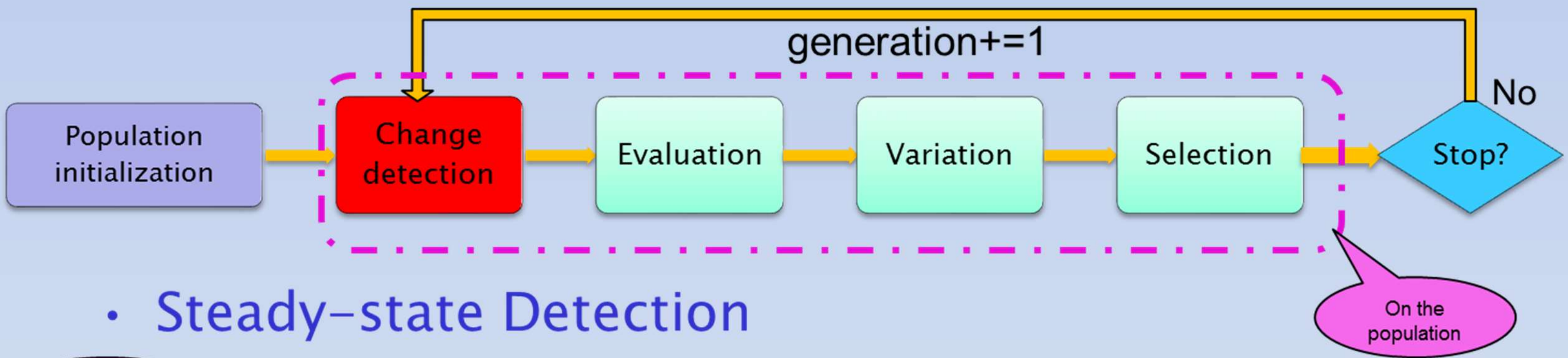
DMOPs



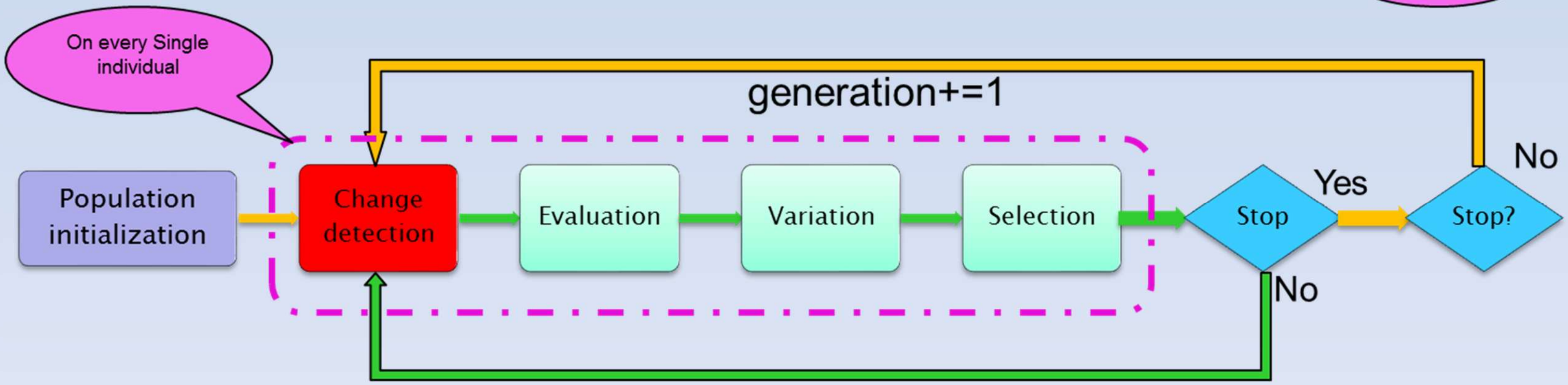
Case Study: EA for Continuous DMOPs

- Steady-state detection in SGEAproblems
  - Can detect changes in the middle of generation
  - Can detect a change immediately
  - Rendering a fast follow-up action

- **Generational Detection**

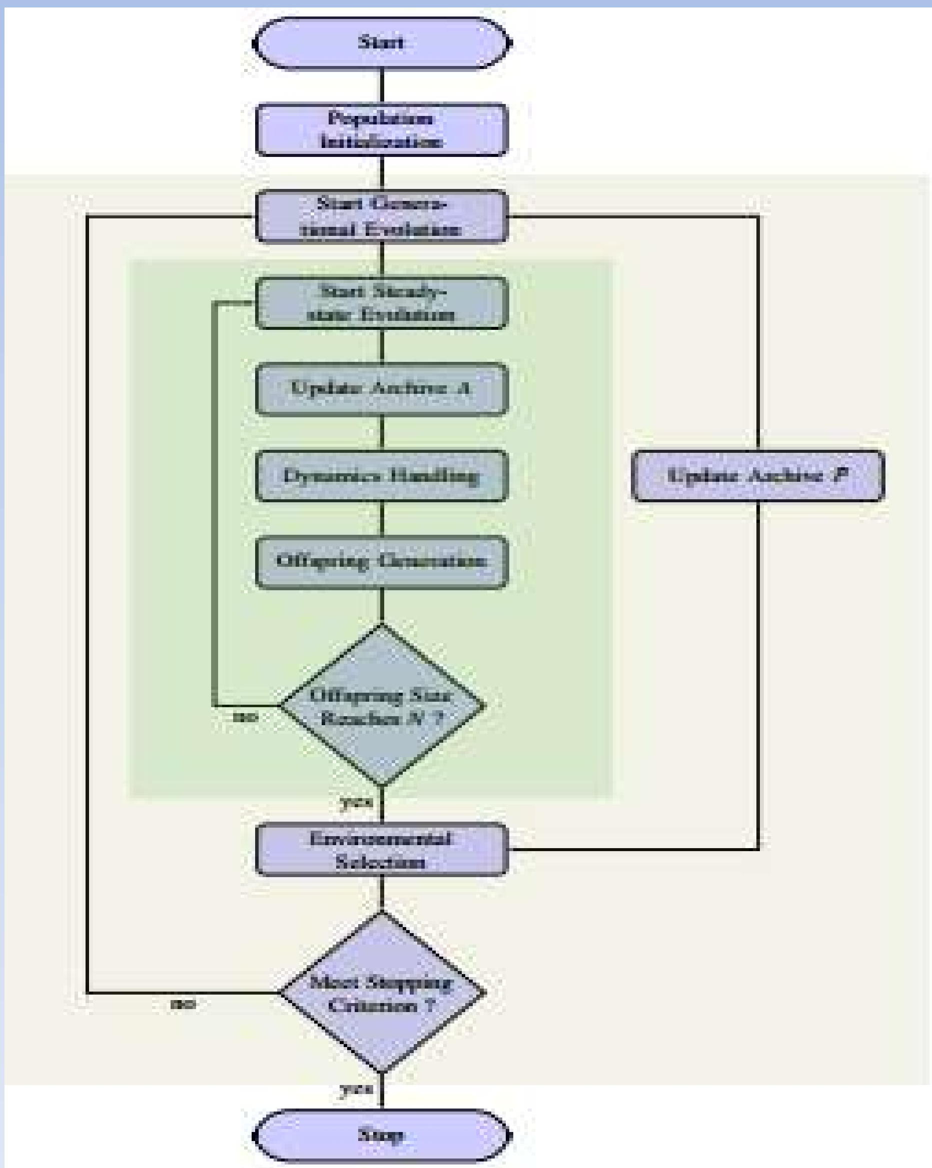


- Steady-state Detection



Case Study: EA for Continuous DMOPs

- Framework of SGEA



Algorithm 1 Framework of SGEA

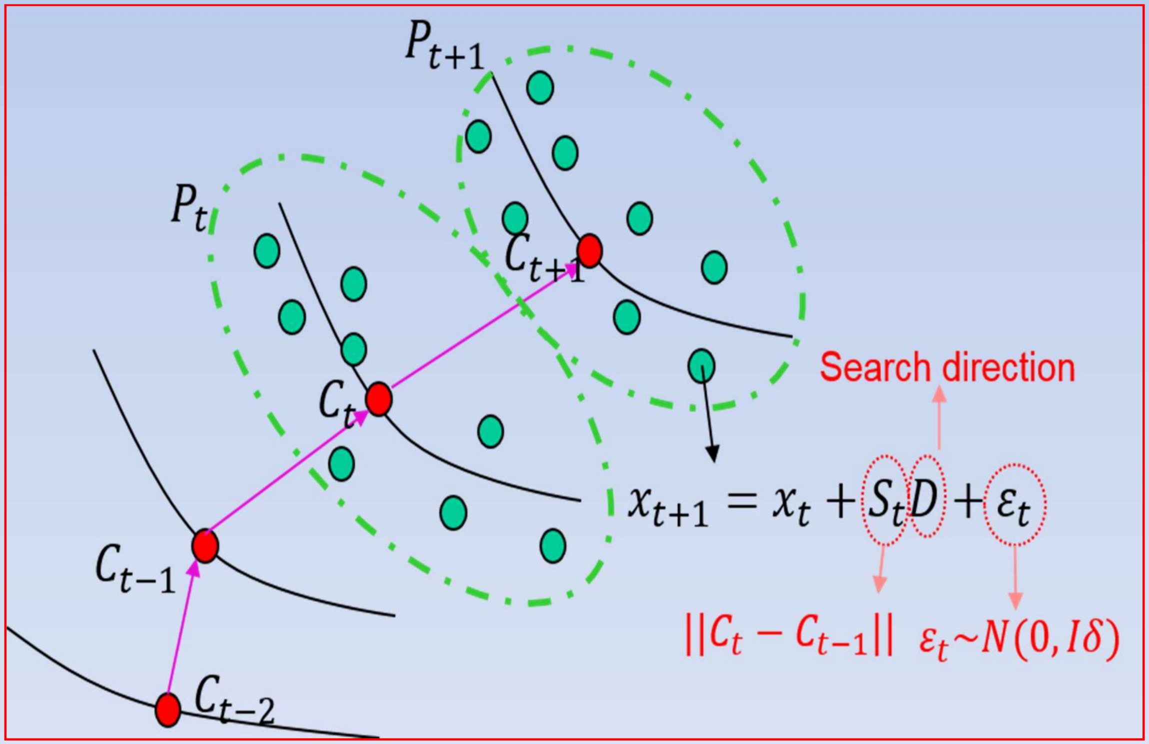
Input: N (population size)
Output: a series of approximated POFs
Create an initial parent population $P := \{x_1, \ldots, x_N\}$;
$(A, \overline{P}) :=$ EnvironmentSelection(P);
while *stopping criterion not met* **do**
  for $i := 1$ to N **do**
    if *change detected and not responded* **then**
      ChangeResponse();
    end if
    $y :=$ GenerateOffspring(P, A);
    $(P, A) :=$ UpdatePopulation(y);
  end for
  $(A, \overline{P}) :=$ EnvironmentSelection$(P \cup \overline{P})$;
  Set $P := \overline{P}$;
end while

steady-state

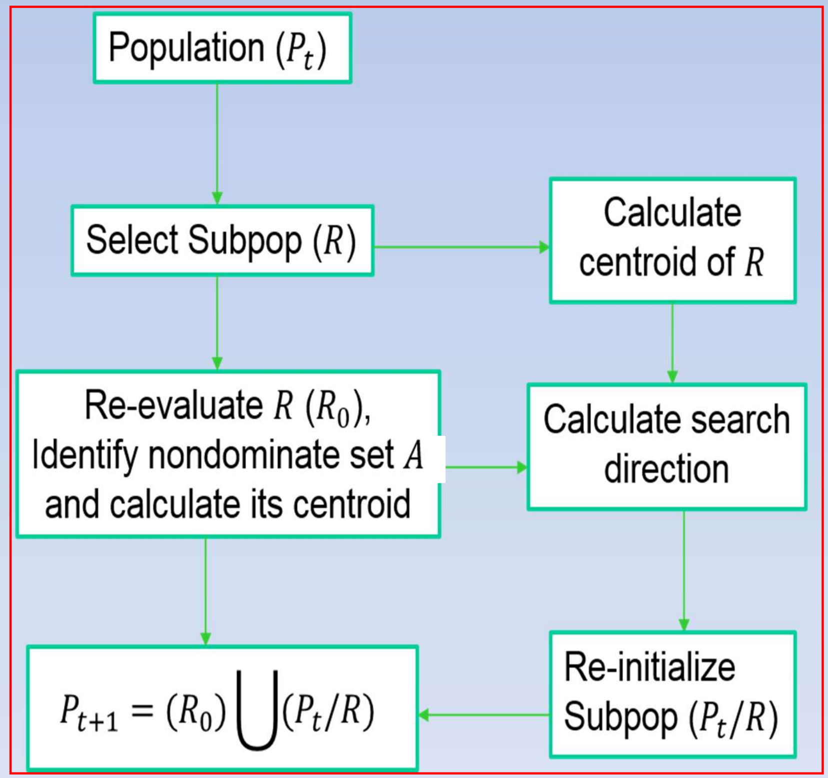
generational

Case Study: EA for Continuous DMOPs

- Change response in SGEA:
  - Split pop into two subpops
  - Re-evaluate subpop1 (R) and keep its solutions
  - Re-initialize subpop2 by prediction methods



Movement of population



Procedure of change reaction

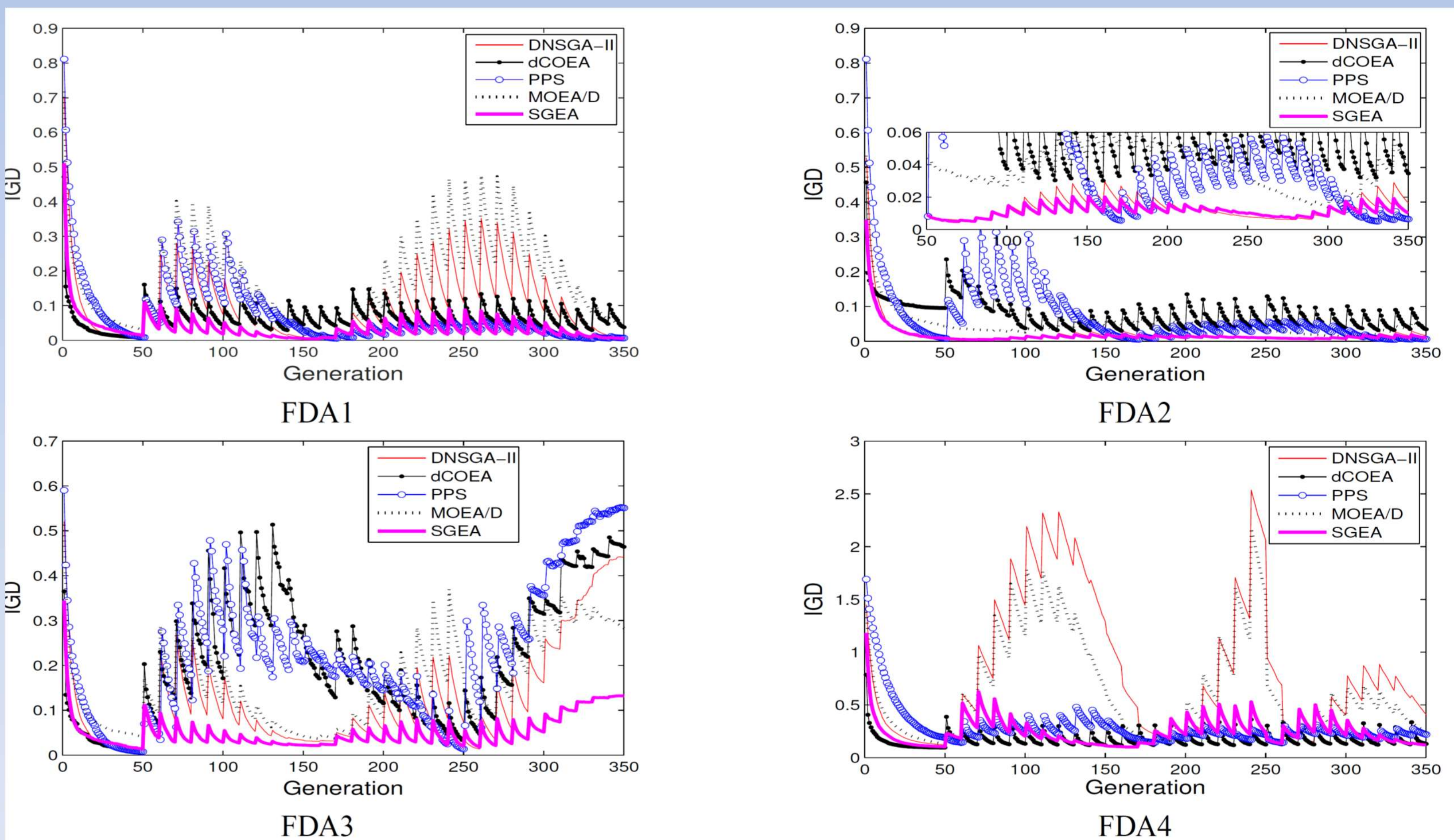
Case Study: EA for Continuous DMOPs

- Empirical Study of SGEA
  - Test problems: FDA, dMOP, UDF, ...
  - Frequency of change: every 5, 10, 20 generations
  - Compared algorithms:
    - DNSGA-II: dynamic NSGAII (Deb et al. 2007)
    - dCOEA: Multi-population approach (Goh & Tan 2009)
    - PPS: population prediction strategy (Zhou et al. 2014)
    - MOEA/D: decomposition-based method (Zhang & Li 2007)

- Main findings:
  - Better tracking results in less frequently changing environments
  - SGEA shows high performance & outperforms the others
  - But, SGEA fails in severe diversity loss due to changes
  - However, introducing some random solutions can avoid diversity loss

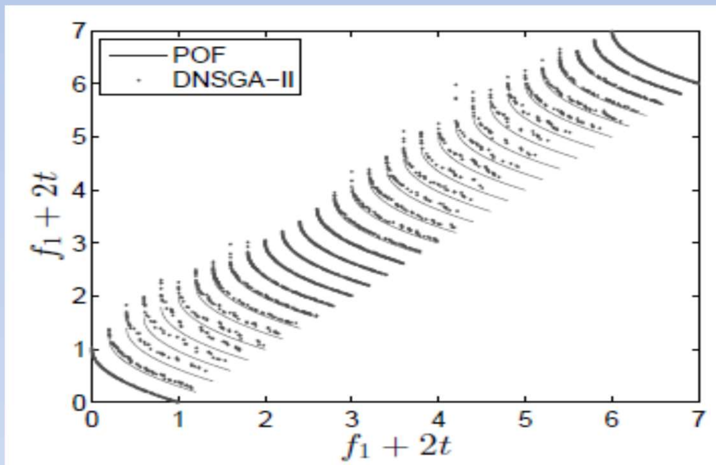
Case Study: EA for Continuous DMOPs

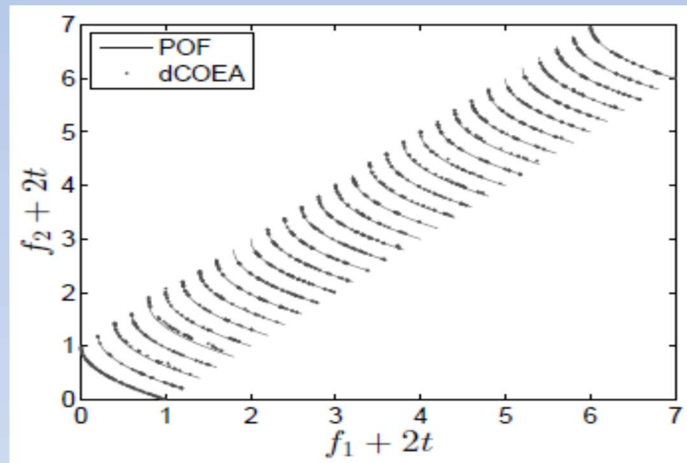
- Some results for FDA problems
  - Performance measure: IGD
  - SGEA is robust

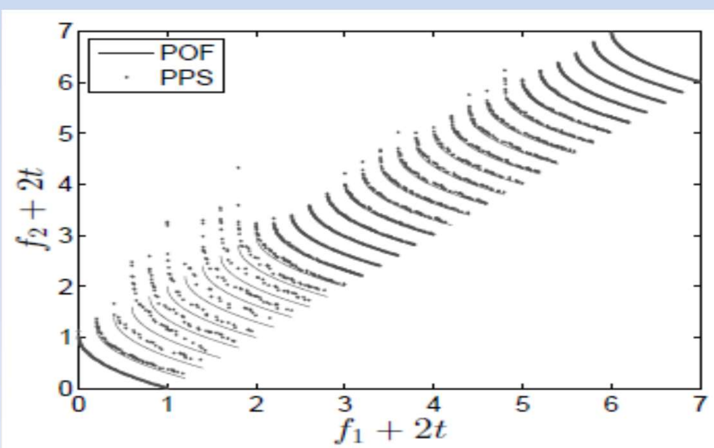


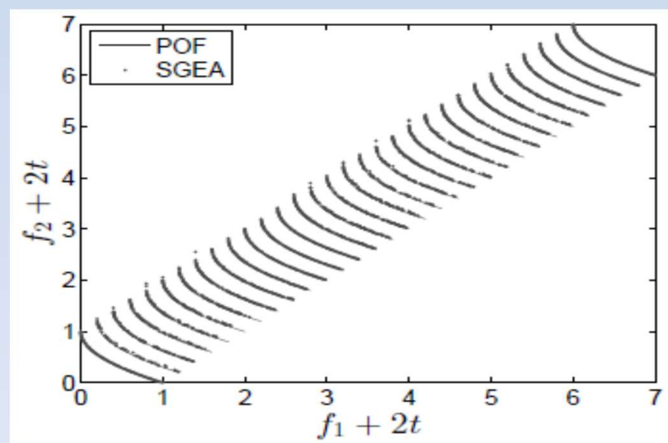
Case Study: EA for Continuous DMOPs

- Some results for FDA1
  - PF approximations obtained by algorithms
  - SGEA is able to track every change







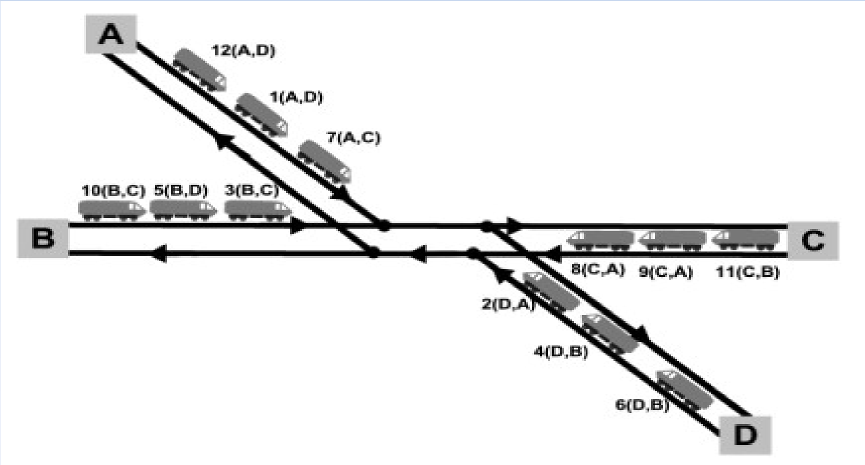


Case Study: Ant Colony Optimization (ACO) for DMOPs

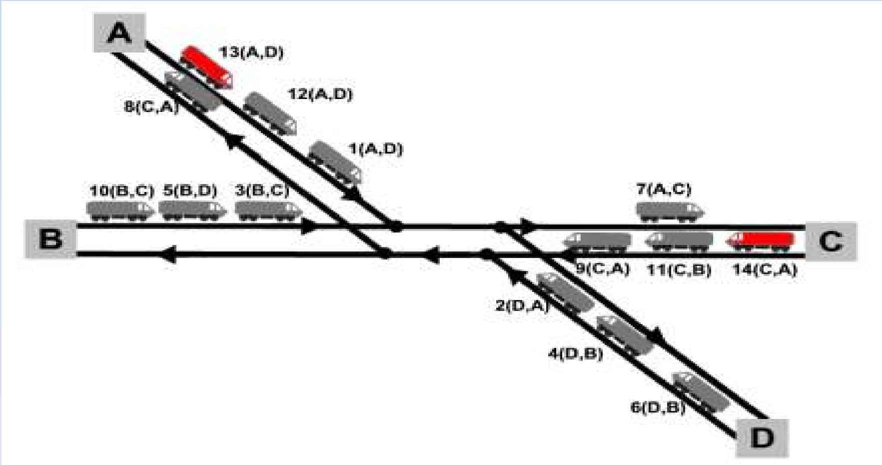
- ACO mimics the behaviour of ants searching for food
- ACO was first proposed for travelling salesman problems (TSPs) (Dorigo *et al.*, 1996)
- Generally, ACO was developed to be suitable for graph optimization problems, such as TSPs and vehicle routing problems (VRPs)
- The idea: let ants "walk" on the arcs of graph while "reading" and "writing" pheromones until they converge into a path
- Standard ACO consists of two phases:
  - Forward mode: Construct solutions
  - Backward mode: Pheromone update
- Conventional ACO cannot adapt well to DMOPs due to stagnation behaviour
  - Once converged, it is hard to escape from the old optimum

Case Study: ACO for DM-RJRP by Eaton et al. (2017)

- Dynamic multi-objective railway junction re-scheduling problem (DM-RJRP):
  - ➢ To find a sequence of trains to pass through two junctions (North Stafford and Stenson) on the Derby to Birmingham line under delays
  - ➢ Two objectives:
    - Minimising timetable deviation
    - Minimising additional energy expenditure
  - ➢ Dynamic:
    - As trains are waiting to be rescheduled at the junction, more timetabled trains will be arriving, which will change the nature of the problem



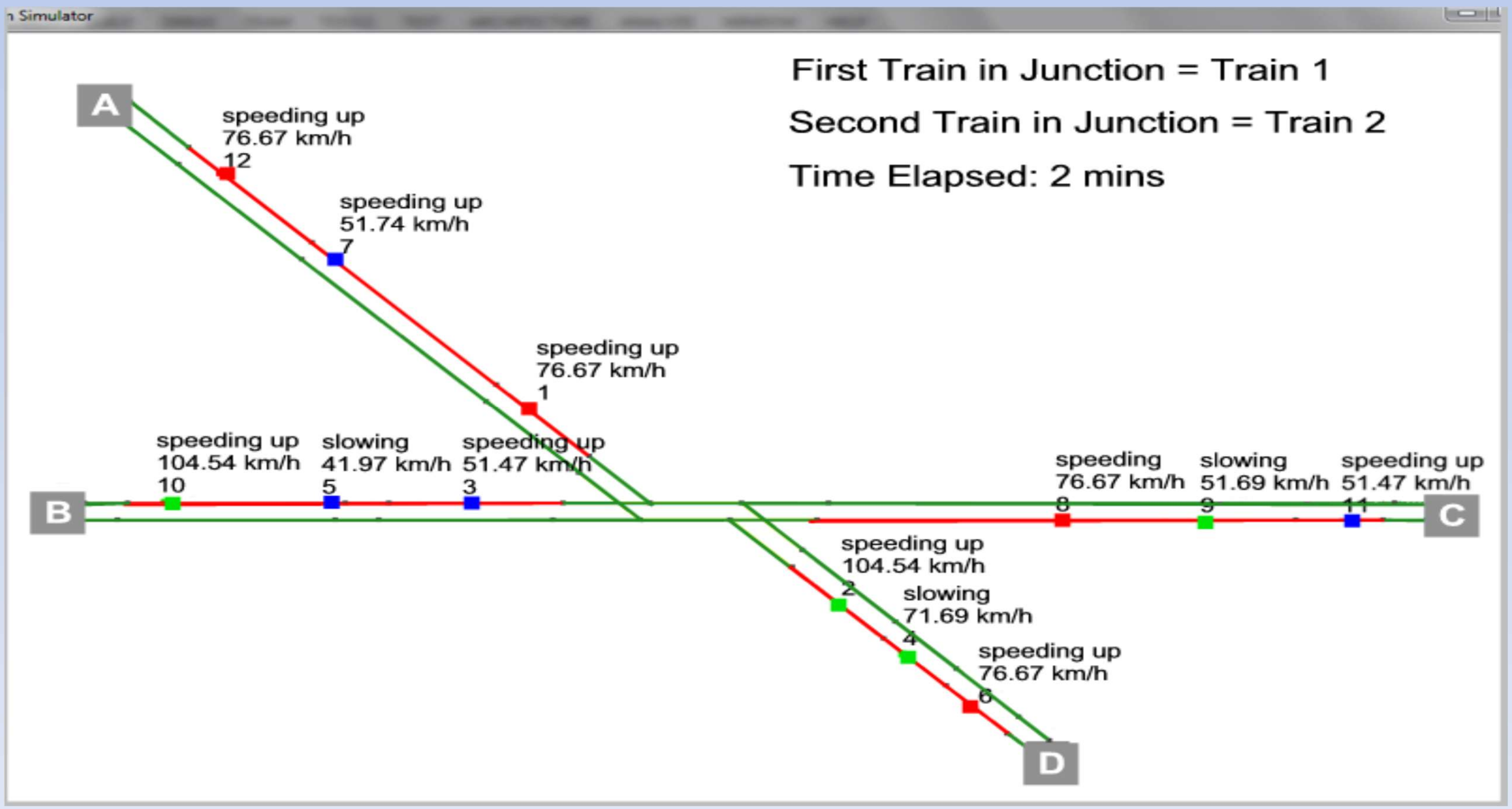
Junction before a change



Junction after a change

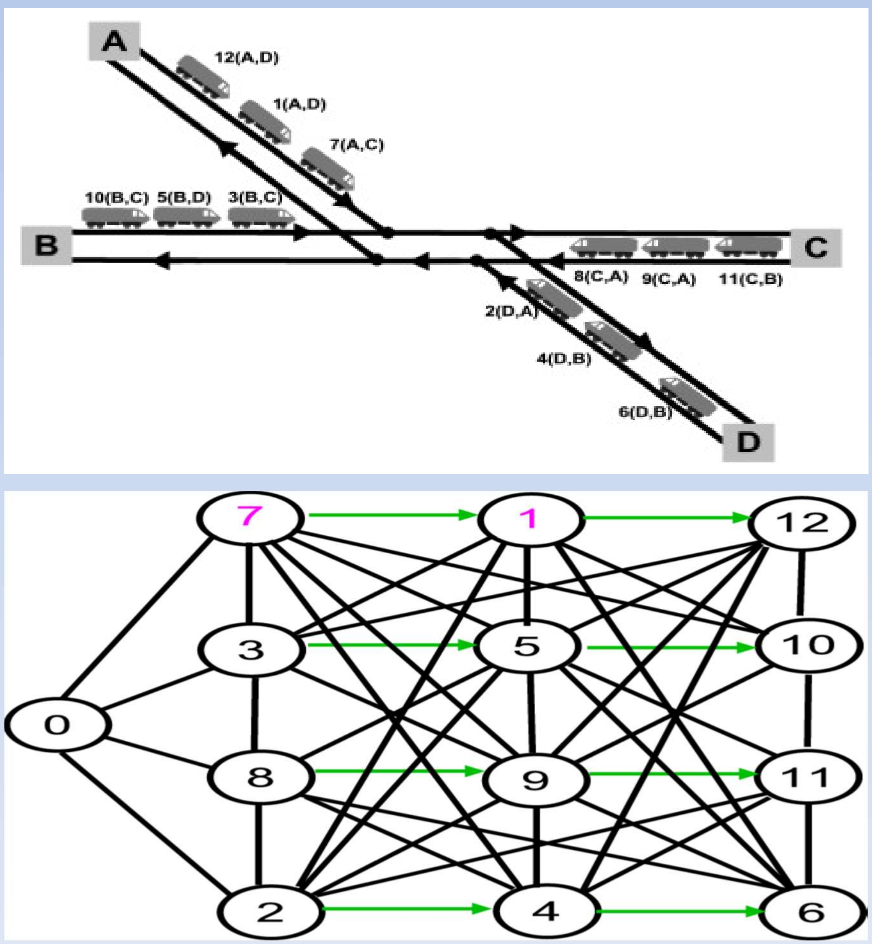
Case Study: ACO for DM-RJRP by Eaton et al. (2017)

- The North Stafford and Stenson junctions train simulator:
  - Developed using C++ Visual Studio 2012
  - Dynamism:
    - Introduced to the simulator by adding *m* trains at a time interval *f* (minutes), where *m* represents the magnitude of change and *f* the frequency of change

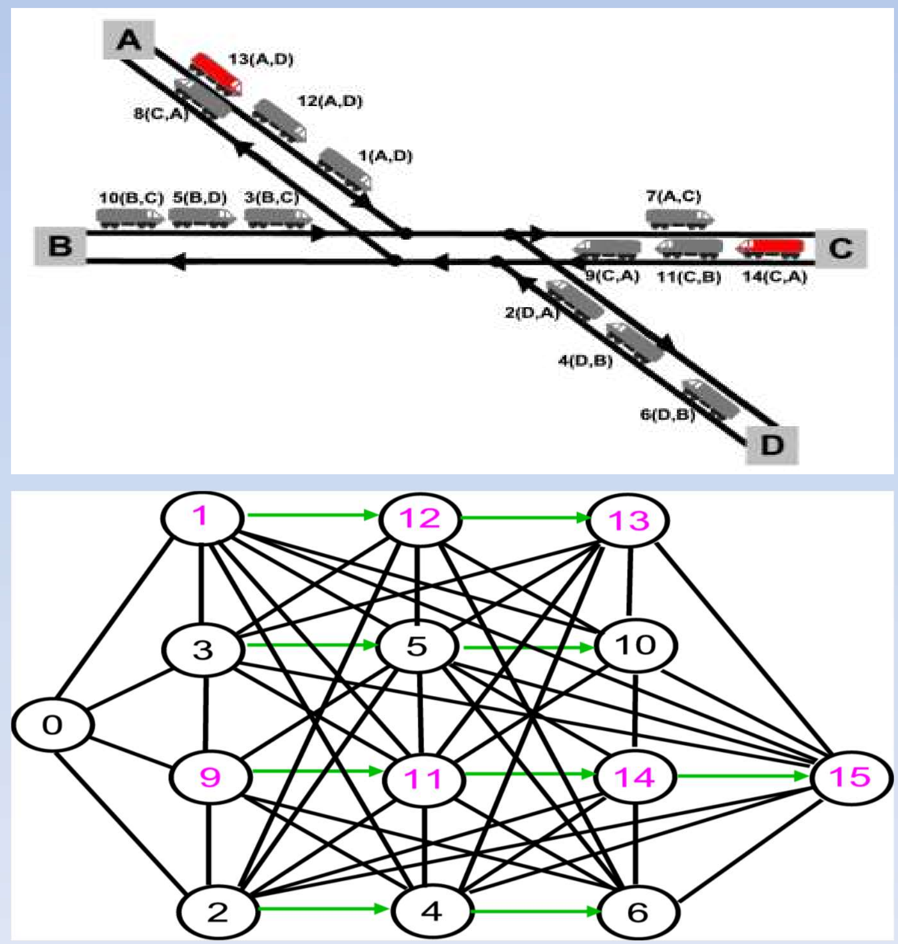


Case Study: ACO for DM-RJRP by Eaton et al. (2017)

- ACO for DM-RJRP: a graphical representation
  - A fully connected, partially one-directional, weighted graph
  - Each node represents a train
- All ants are initially placed at an imaginary start node (zero)



Node matrix before a change



Node matrix after a change

Case Study: ACO for DM-RJRP by Eaton et al. (2017)

- DM-PACO: a new version of P-ACO for DM-RJRP
  - A pheromone and heuristic matrix for each objective
  - An archive to store non-dominated solutions (repaired after a change)
  - A memory: created from the archive and re-created after a change
- DM-MMAS: a new version of Max-Min Ant System (MMAS)
  - A pheromone matrix for each objective
  - An archive to store non-dominated solutions
  - Four designs based on clearing archive or pheromones after a change

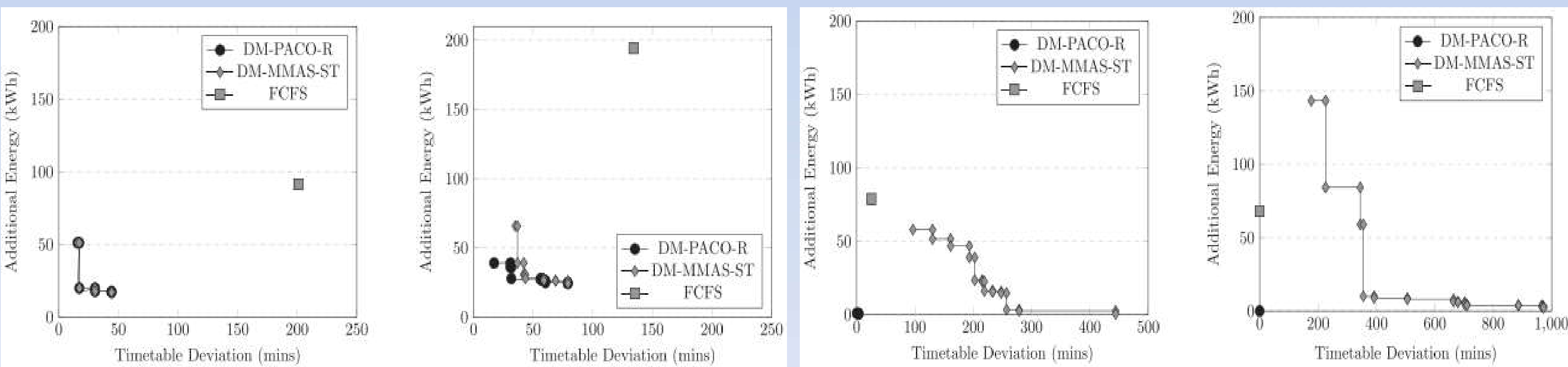
FOUR DIFFERENT VERSIONS OF THE DM-MMAS ALGORITHM

	Clear Pheromones	Retain Pheromones
Clear Archive	DM-MMAS-SC	DM-MMAS-ST
Retain Archive	DM-MMAS-NC	DM-MMAS-NT

- Peer algorithms: NSGA-II and FCFS

Case Study: ACO for DM-RJRP by Eaton et al. (2017)

- Findings:
  - All ACO algorithms can find a POS of solutions for the DM-RJRP
  - DM-PACO outperformed DM-MMAS algorithms
  - DM-PACO also outperformed NSGA-II and FCFS
  - For large and frequent changes:
    - Good to retain an archive of non-dominated solutions
    - Good to update pheromones for new environments
  - Interaction between objectives are more complex than expected



(a) Change 0 (b) Change 1 (c) Change 4 (d) Change 6

Challenging Issues

- Detecting changes:
  - Most studies assume that changes are easy to detect or visible to an algorithm whenever occurred
  - In fact, changes are difficult to detect for many DMOPs
- Understanding the characteristics of DMOPs:
  - What characteristics make DMOPs easy or difficult?
  - Little work, needs much more effort
- Analysing the behaviour of EAs for DMOPs:
  - Requiring more theoretical analysis tools
  - Addressing more challenging DMOPs and EC methods
  - Big question: Which EC methods for what DMOPs?
- Real world applications:
  - How to model real-world DMOPs?
  - How to extend the applicability of EC methods?

Future Work

- The domain has attracted a growing interest recently
  - But, far from well-studied
- New approaches needed: esp. hybrid approaches
- Theoretical analysis: greatly needed
- EC for DMOPs: deserves much more effort
- Real world applications: also greatly needed
  - Fields: logistics, transport, MANETs, data streams, social networks, ...

Summary

- EC for DMOPs: challenging but important
- The domain is still young and active:
  - Benchmarking
  - Optimization approaches
  - Theoretic study
  - Real-world applications
- More young researchers are greatly welcome!

Acknowledgements

- Two EPSRC funded projects on EC for DOPs
  - "EAs for DOPs: Design, Analysis and Applications"
    - Linked project among Brunel Univ. (Univ. of Leicester before 7/2010), Univ. of Birmingham, BT, and Honda
    - Funding/Duration: over £600K/3.5 years (1/2008–7/2011)
    - http://gtr.rcuk.ac.uk/project/B807434B-E9CA-41C7-B3AF-567C38589BAC
  - "EC for Dynamic Optimisation in Network Environments"
    - Linked project among DMU, Univ. of Birmingham, RSSB, and Network Rail
    - Funding/Duration: ~£1M/4.5 years (2/2013–8/2017)
    - http://gtr.rcuk.ac.uk/project/C43F34D3-16F1-430B-9E1F-483BBADCD8FA
- Research team members:
  - Research Fellows: Dr. Hui Cheng, Dr. Crina Grosan, Dr. Changhe Li, Dr. Michalis Mavrovouniotis, Dr. Yong Wang, etc.
  - PhD students: Changhe Li, Michalis Mavrovouniotis, Shouyong Jiang, Jayne Eaton, etc.
- Research co-operators:
  - Prof. Xin Yao, Prof. Juergen Branke, Dr. Renato Tinos, Dr. Hendrik Richter, Dr. Trung Thanh Nguyen, Dr. Juan Zou, etc

Relevant Information

- IEEE CIS Task Force on EC in Dynamic and Uncertain Environments
  - http://www.tech.dmu.ac.uk/~syang/IEEE_ECIDUE.html
  - Maintained by Shengxiang Yang
- Source codes:
  - http://www.tech.dmu.ac.uk/~syang/publications.html

References-1

- S. Biswas, S. Das, P.N. Suganthan, C.A. Coello Coello (2014). Evolutionary multiobjective optimization in dynamic environments: A set of novel benchmark functions, IEEE CEC, pp. 3192-3199
- K. Deb, N. U. B. Rao, S. Karthik (2007). Dynamic multi-objective optimization and decision-making using modified NSGA-II: A case study on hydro-thermal power scheduling. in Evolutionary Multi-Criterion Optimization, pp. 803-817
- J. Eaton, S. Yang, M. Gongora (2017). Ant colony optimization for simulated dynamic multi-objective railway junction rescheduling. IEEE Trans Intell Transport Syst, 18(11): 2980-2992
- M. Farina, K. Deb, P. Amato (2004). Dynamic multiobjective optimization problems: test cases, approximations, and applications. IEEE Trans Evol Comput, 8(5): 425–442
- S. B. Gee, K. C. Tan, H. A. Abbass (2017). A Benchmark Test Suite for Dynamic Evolutionary Multiobjective Optimization. IEEE Trans on Cybern, 47(2): 461-472
- C. Goh, K. C. Tan (2009). A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization. IEEE Trans Evol Comput, 13(1): 103–127
- I. Hatzakis, D. Wallace (2006). Dynamic multi-objective optimization with evolutionary algorithms: A forward-looking approach. Proc 8th Annual Conf Genet Evol Comput (GECCO), pp. 1201-1208
- M. Helbig, A. P. Engelbrecht (2014). Benchmarks for dynamic multi-objective optimisation algorithms. ACM Comput Surv 46(3): 37:1–37:39

References-2

- S. Huband, P. Hingston, L. Barone, L. While (2006). A review of multiobjective test problems and a scalable test problem toolkit. IEEE Trans Evol Comput 10(5): 477-506
- S. Jiang, S. Yang (2017a). Evolutionary dynamic multi-objective optimization: benchmarks and algorithm comparisons. IEEE Trans Cybern, 47(1): 198-211
- S. Jiang, S. Yang (2017b). A steady-state and generational evolutionary for dynamic multi-objective optimization. IEEE Trans Evol Comput, 21(1): 65-82
- Y. Jin, B. Sendhoff (2004). Constructing dynamic optimization test problems using the multi-objective optimization concept. in Applications of Evol. Comput., pp. 525-536
- Y. Jin, J. Branke (2005). Evolutionary optimization in uncertain environments–A survey. IEEE Trans Evol Comput, 9(3): 303–317
- H. Li and Q. Zhang (2009). Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II, IEEE Trans Evol Comput 13(2): 284–302,2009
- W. T. Koo, C. Goh, K. Tan (2010): A predictive gradient strategy for multi-objective evolutionary algorithms in a fast changing environment. Memet Comput 2(2): 87–110
- J. Mehnen, G. Rudolph, T. Wagner (2006). Evolutionary optimization of dynamic multiobjective functions. Tech Report CI-204/06. Universit¨at Dortmund, Germany
- A. Muruganantham (2017) Dynamic Multiobjective Optimization Using Evolutionary Algorithms. PhD Thesis.
- M. Orouskhani (2017). Dynamic Multiobjective Optimization Using Hybrid Algorithm of Cat Swarm Optimization and Borda Count Method. PhD Thesis

References-3

- G. Ruan, G. Yu, J. Zheng, J. Zou, S. Yang (2017). The effect of diversity maintenance on prediction in dynamic multi-objective optimization. Appl Soft Comput, 58: 631-647
- Y. Wang, B. Li (2009). Investigation of memory-based multi-objective optimization evolutionary algorithm in dynamic environment. IEEE CEC, pp. 630-637
- Y. Wu, Y. Jin, X. Liu (2015). A directed search strategy for evolutionary dynamic multiobjective optimization. Soft Computing 19(11): 3221-3235
- S. Yang, X. Yao (2013). Evolutionary Computation for Dynamic Optimization Problems. Springer
- A. Zhou, Y. Jin, Q. Zhang, B. Sendhoff, E. Tsang (2007). Prediction-based population re-initialization for evolutionary dynamic multi-objective optimization. Evolutionary Multi-Criterion Optimization, pp. 832-846
- A. Zhou, Y. Jin, Q. Zhang (2014). A population prediction strategy for evolutionary dynamic multiobjective optimization. IEEE Trans Cybern 44(1): 40-53
- J. Zou, Q. Li, S. Yang, H. Bui, J. Zhen (2017). A prediction strategy based on center points and knee points for evolutionary dynamic multi-objective optimization. Appl Soft Comput 61: 806-818