Evolutionary Computation for Dynamic Optimization Problems

Shengxiang Yang

s.yang@mcs.le.ac.uk

http://www.cs.le.ac.uk/people/syang

Department of Computer Science, University of Leicester

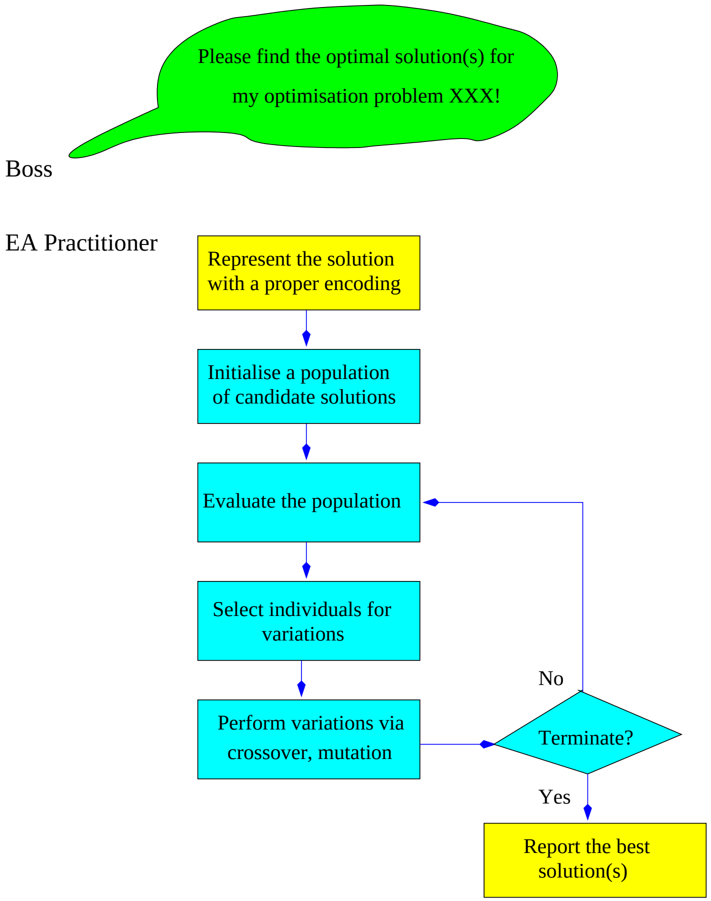
University Road, Leicester LE1 7RH, United Kingdom

Outline of the Talk

- Introduction to evolutionary computation
- Dynamic optimization problems (DOPs)
- Evolutionary algorithms (EAs) for DOPs
- EAs for practical DOPs
- Conclusions

Evolutionary Computation – 1

- Encapsulates a class of stochastic optimisation algorithms — Evolutionary Algorithms (EAs)
- Inspired by principles from natural and biological evolution



- Simple general framework

represent the solution
initialize population
repeat
  evaluation
  selection
  variation (crossover, mutation)
until *termination condition holds*

Evolutionary Computation – 2

- Properties of EAs:
  - Stochastic, global search $\Rightarrow$ (near-) optimal solution
  - Population based search $\Rightarrow$ a set of solutions
  - No strict requirements to problem $\Rightarrow$ easy to use
- Widely used for optimisation and search problems, e.g., in
  - Financial and economical systems, transportation and logistics, industry engineering, automatic programming, art and music design
- Traditionally, research on EAs has focused on static problems
- Recently, a rapidly growing interest in EAs for dynamic optimisation problems (DOPs)

What are DOPs?

- Optimisation problems that are subject to changes over time

$$F = f(\vec{x}, \vec{\phi}, t)$$

$\vec{x}$: decision variable(s), $\vec{\phi}$: parameter(s), t: time

- Change may involve factors like
  - Objectives or problem settings
  - Constraints
  - Environmental parameters
- Key dynamics characteristics
  - Speed of change
  - Severity of change
  - Periodicality of change
  - Detectability or predictability of change

Dynamic Optimisation Test Environments

- Basic principle: to change base static problem(s)
- Binary encoded functions:
  - Switch between ≥ 2 states of a problem: knapsack
  - Use binary masks: XOR generator (Yang and Yao'05)
- Combinatorial optimisation problems:
  - Change decision variables: item weights/profits in the knapsack problem
  - Add/delete decision variables: new cities in TSP, new jobs in scheduling, new nodes in network routing problems
- Real-valued functions:
  - Switch between different functions
  - Move/reshape peaks in the fitness landscape

Challenges of DOPs to EAs

- DOPs raise big challenges
  - Conventional EAs are not good for DOPs
    - Aiming to locate static optima quickly and precisely
    - Once converged, hard to escape from old optimum
  - DOPs requires EAs to track the moving optimum
- Why should EA community study DOPs?
  - Many real world problems are DOPs
  - EAs, once properly enhanced, are good choice
    - Inspired by natural evolution (in dynamic environments)
    - Intrinsically, should be fine to deal with DOPs
- Many events have focused on EAs for DOPs in recent years

Recent Events on EAs for DOPs

- Books: Yang, Ong and Jin (2007), Morrison (2004), Weiker (2003), Branke (2002)
- Journal special issues:
  - Neri and Yang, Memetic Computing, expected 2010
  - Yang, Ong and Jin, Genetic Programming & Evolvable Machines, 7(4), 2006
  - Jin and Branke, IEEE Trans Evol Comput, 10(4), 2006
  - Branke, Soft Computing, 9(11), 2005
- Workshops and conference special sessions:
  - EvoSTOC ('04, '05, '06, '07, '08, '09) — part of EvoWorkshops
  - ECiDUE ('04, '05, '06, '07, '08, '09) — part of IEEE CEC
  - EvoDOP ('99, '01, '03, '05, '07) — part of GECCO
- IEEE Symposium on CIDUE, to be held in Singapore, 2011

EAs for DOPs: General Approaches

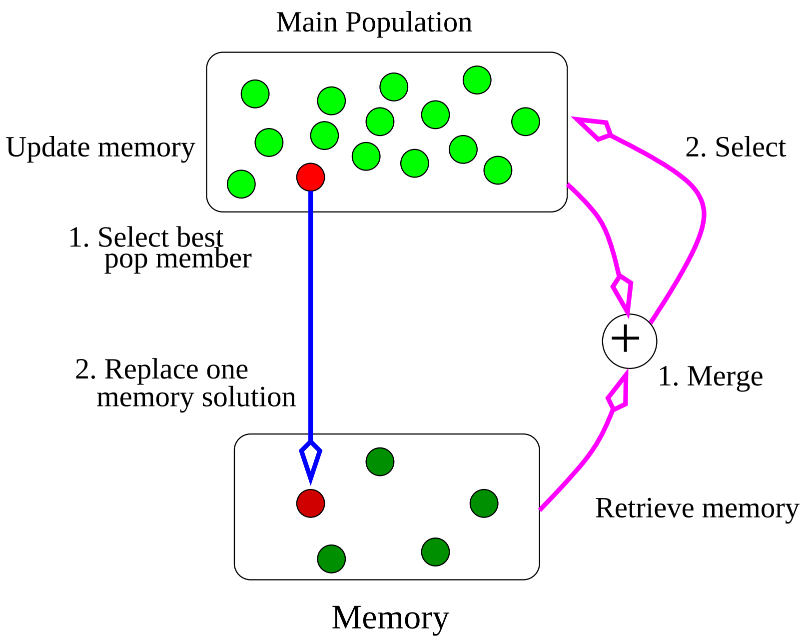
- In recent years, many approaches developed for EAs to address DOPs, inspired from principles in biology/nature
- General approaches:
  - Memory schemes: store and reuse useful information
  - Diversity schemes: handle convergence directly
  - Multi-population schemes: co-operate sub-populations
  - Adaptive schemes: adapt generators and parameters

EAs for DOPs: Memory Schemes

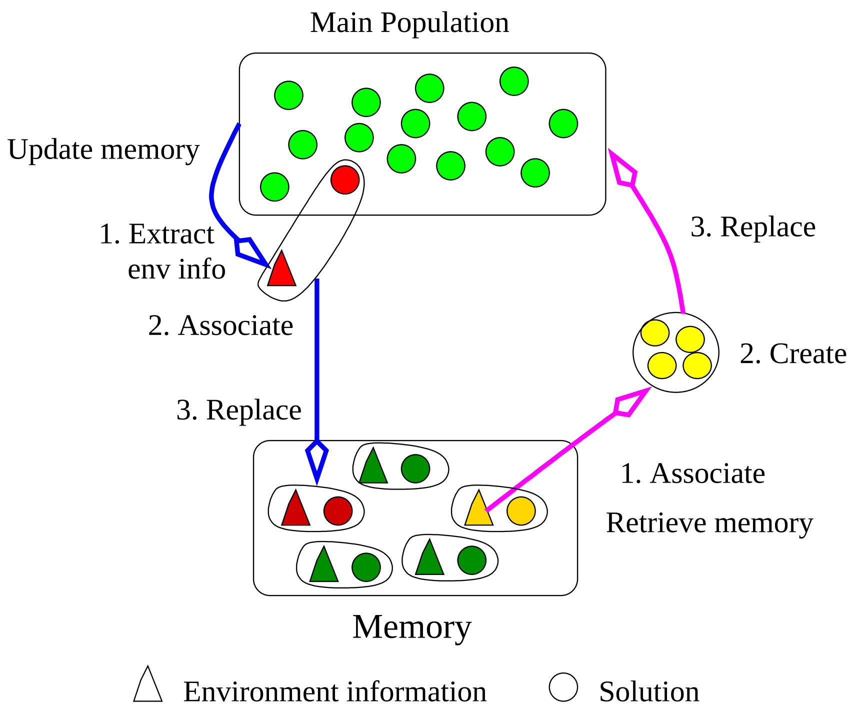
- Works by storing and reusing useful information
- Two classes w.r.t how to store information
- Implicit memory: uses redundant representations
  - Multiploidy and dominance (Ng & Wong'95, Lewis et al '98)
  - Structured encoding (Dasgupta & McGregor'92)
  - Dualism mechanisms (Yang'03, Yang & Yao'05)
- Explicit memory: uses extra space to store information

Explicit Memory Schemes

- Direct memory: good solutions (Branke'99)
- Associative memory: environmental information + good solutions (Yang'06, Yang & Yao'08)



Direct Memory



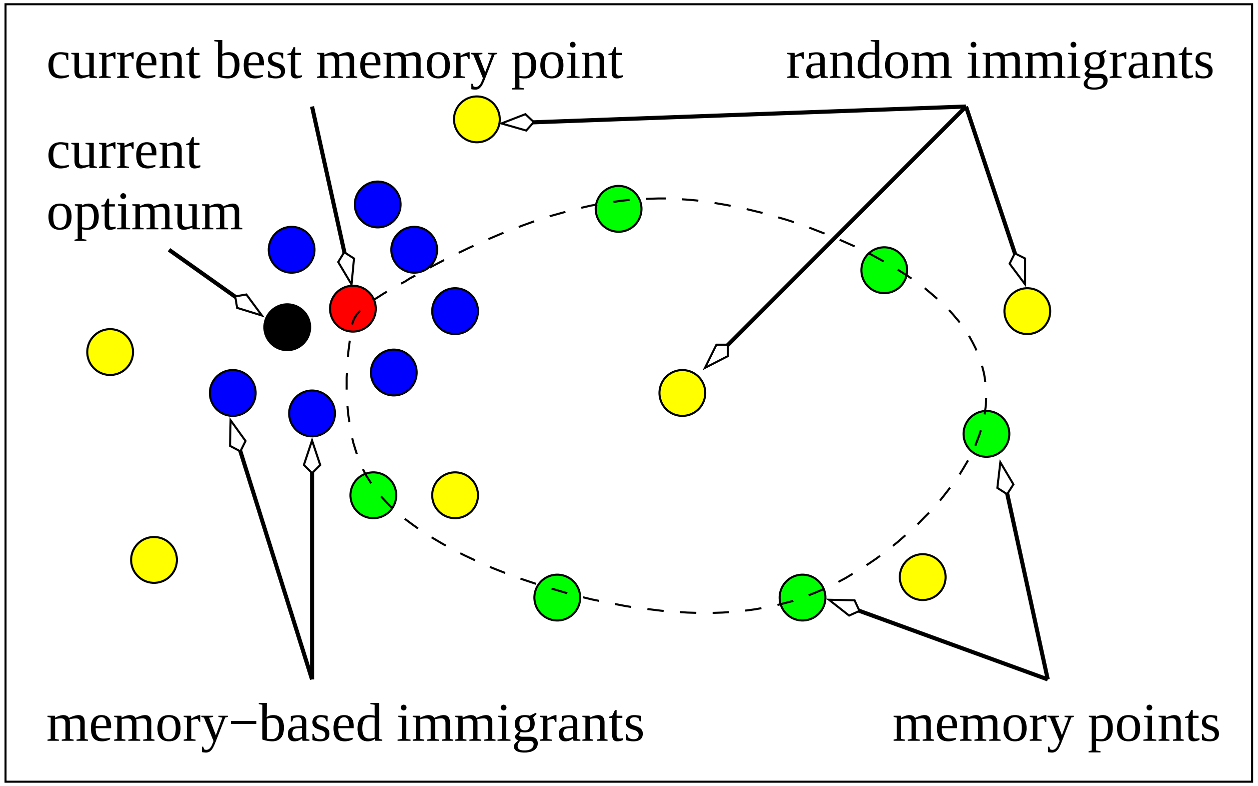
Associative Memory

EAs for DOPs: Diversity Scheme

- Convergence is the main problem for EAs for DOPs
- Random immigrants maintains diversity by replacing random individuals into the population.
  - Problem: random immigrants may be hard to survive
- **Memory-based immigrants**: hybridises memory with random immigrants
  - Re-evaluate the memory every generation
  - Retrieve the best memory point $B_M(t)$ as the base
  - Generate immigrants by bitwise mutating $B_M(t)$ with a prob.
  - Replace worst members in the population by these immigrants
    - S. Yang. *Evol. Comput.*, 16(3): 385-416, 2008

Why Memory-Based Immigrants?

- Key idea: guide immigrants towards current environment

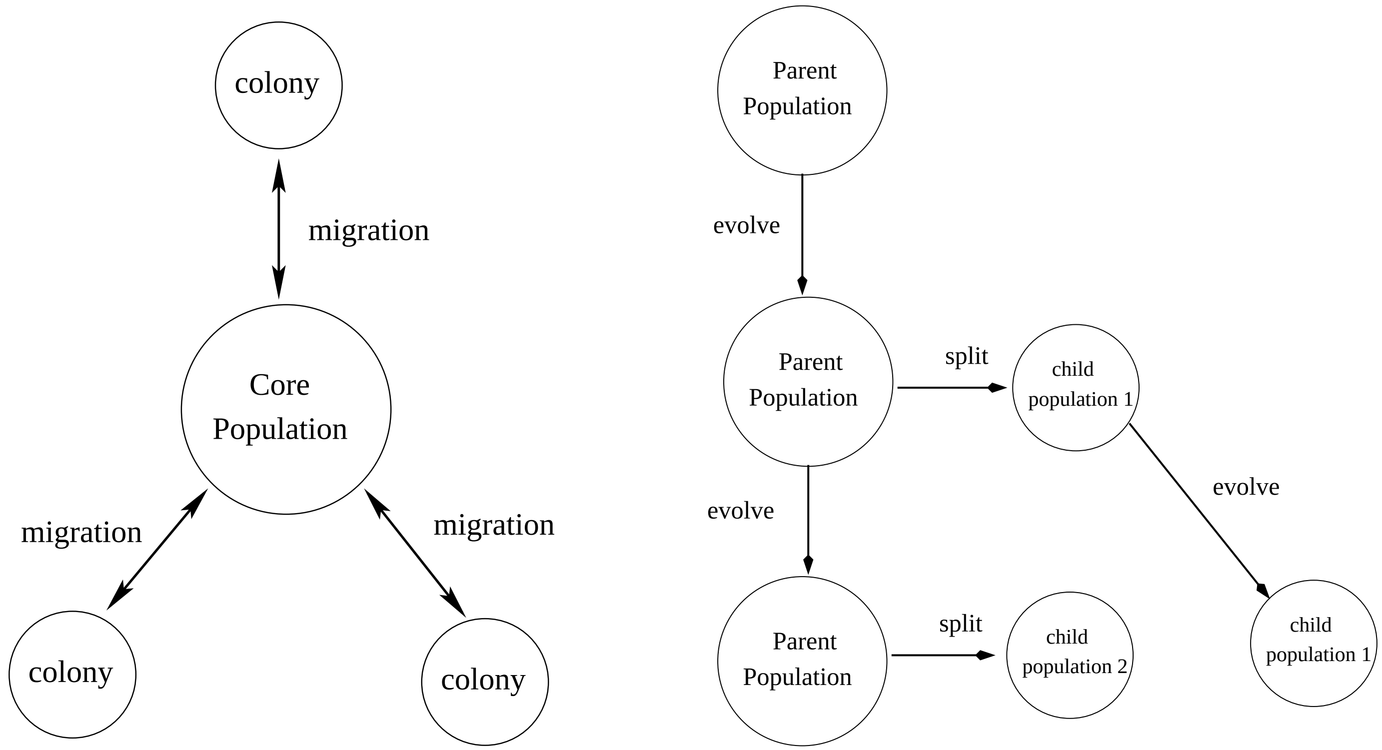


Search Space

Illustration of immigrants schemes.

EAs for DOPs: Multi-Population Schemes

- Idea: Use co-operating populations
- Shifting balance GA (SBGA): Oppacher & Wineberg'99
- Self-organizing scouts (SOS) GA: Branke et al (2000)



SBGA SOS

Clustering Particle Swarm Optimisation

- Particle Swarm Optimisation (PSO): motivated by the social behaviour of swarm of animals, e.g., bird flocking and fish schooling
- PSO has been used to address DOPs
- Recently, we developed a Clustering PSO (CPSO) for DOPs
  - First, train the particles properly
  - Then, use a clustering technique to construct sub-swarms
  - Each sub-swarm will search among one peak quickly
  - Strategies to response to problem changes
- More details:
  - Li & Yang. *IEEE Congr on Evol Comput*, 2009

EAs for DOPs: Adaptive Schemes

- Adapt operators/parameters, usually after a change
- Hypermutation (Cobb & Grefenstette'93): raise the mutation rate temporarily
- Hyper-selection (Yang & Tinos'08): raise the selection pressure temporarily
- Hyper-learning (Yang & Richter'09): raise the learning rate for Population-Based Incremental Learning (PBIL) temporarily
- Hyper-selection and hyper-learning should be combined with restart or hypermutation

Dynamic Job Shop Scheduling Problems

- Search space shifts as jobs are completed and new jobs arrive
- The relative importances of jobs also change over time
- A classifier-based memory is devised for abstracting and storing information about schedules
  - Used to build similar schedules at future times
- The memory enhanced EA outperforms standard EA and several EAs with diversity techniques
- More details:
  - Barlow & Smith. *EvoWorkshops'08*, 606-615, 2008

Dynamic Shortest Path Problems

- Static SP problem: find the shortest path between source and destination in a fixed network topology
- However, more and more wireless mobile networks appear where the topology keeps changing
- We model the network dynamics like this:
  - For each change, a number of nodes are randomly selected to sleep or wake up based on their current status
- Dynamic SP problem (DSPP): find a series of shortest paths in a series of highly-related network topologies
- We investigated different EA approaches to solve DSPPs

Dynamic Traveling Salesman Problems

- Dynamic TSP: involving dynamic cost (distance) matrix

  $$D(t) = \{d_{ij}(t)\}_{n*n}$$

  $d_{ij}(t)$: the cost from city i to j, n: the number of cities

- DTSP can be defined as $f(x, \phi, t)$, $\phi = \vec{D}$. The objective is to find a minimum-cost route containing all cities at time t:

  $$f(x, \phi, t) = Min(\sum_{i=1}^{n} d_{T_i, T_{i+1}}(t))$$

  where $T_i \in 1, \cdots, n$. If $i \neq j$, $T_i \neq T_j$, $T_{n+1} = T_1$

- Algorithm:
  - Gene pool, inver-over operator, and dynamic elastic operator

Conclusions and Road Ahead

- EAs for DOPs has attracted a growing interest recently
  - But, far from well-studied
- More challenges needs to be taken
  - New approaches
  - Real world applications: scheduling, communications optimisation, financial and economics problems
    - How to model real world DOPs?
  - Theoretical analysis of EAs for DOPs
- Big question: Which EAs for what DOPs?

Relevant Information

- IEEE CIS Task Force on “EC in Dynamic and Uncertain Environments”, maintained by Yaochu Jin
  - http://www.soft-computing.de/ieee_ecidue.html
- Email list, maintained by Jürgen Branke
  - evodop@aifb.uni-karlsruhe.de
- My “EC in Dynamic and Uncertain Environments” website
  - http://www.cs.le.ac.uk/people/syang/ECiDUE/index.html
- UK EPSRC project “EAs for DOPs” website
  - http://www.cs.le.ac.uk/people/syang/EADOP/index.shtml