

Probability Distribution Based Evolutionary Algorithms

Wei-Neng Chen

South China University of Technology

05-14-2018

华南理工大学
South China University of Technology

广东省计算智能与网络
空间信息重点实验室

Outline

1. Background

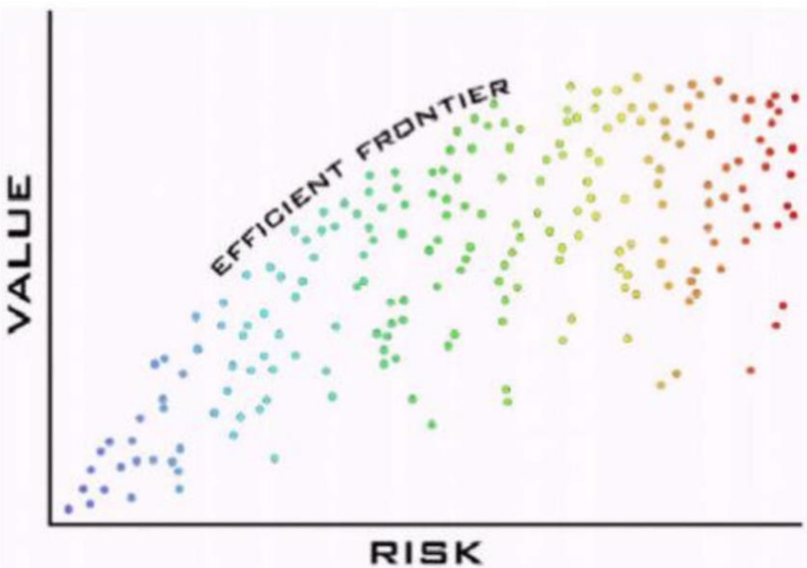
2. Probability Distribution Based EAs for Seeking Multiple Solutions

3. Probability Distribution Based EAs for Discrete Optimization

4. Applications & Future Work

Data-Driven Optimization

- ❑ **In big-data environment, many problems are driven by data**

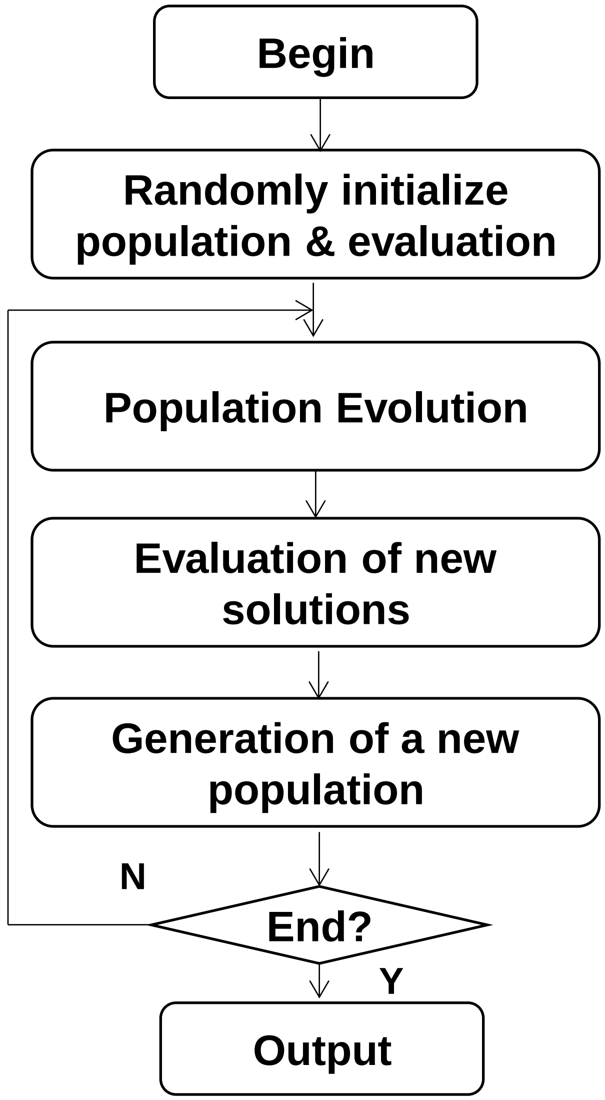


Portfolio Optimization

Order Dispatching

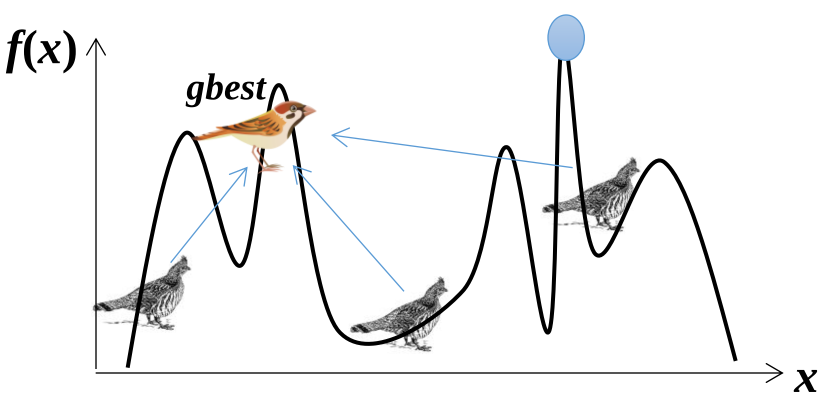
Evolutionary Computation

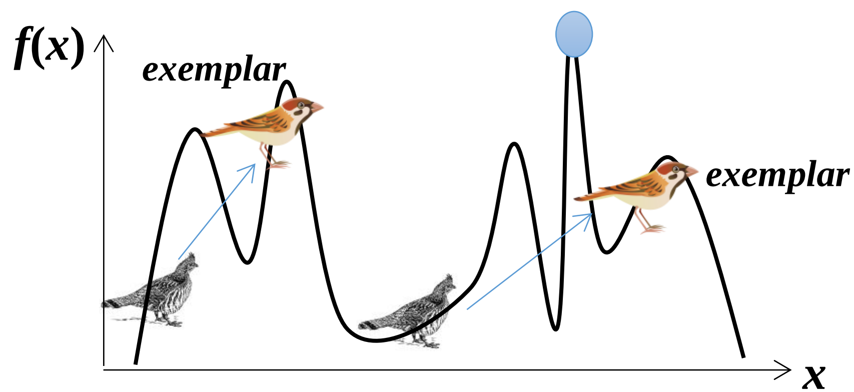
- Population-based stochastic algorithms which simulate intelligent behaviors
- Popular Algorithms
  - Genetic algorithms (GA)
  - Differential evolution (DE)
- Meta-heuristics
  - Particle swarm optimization (PSO)
  - Ant colony optimization (ACO)
  - …
- Advantages
  - Do not make any assumption about the underlying fitness landscape
  - Find approximated solutions within acceptable time



Challenge 1: Premature Convergence

- **How to escape from local optima in multimodal problems has long been a key issue in EC research**
  - **Parameter adaption** **(APSO, Zhan et al., 2009)**
  - **Modified mutation \ update rules** **(Local Topology PSO, Kennedy & Mendes, 2002, CLPSO, Liang et al. 2006, etc.)**
  - **Hybrid approaches**

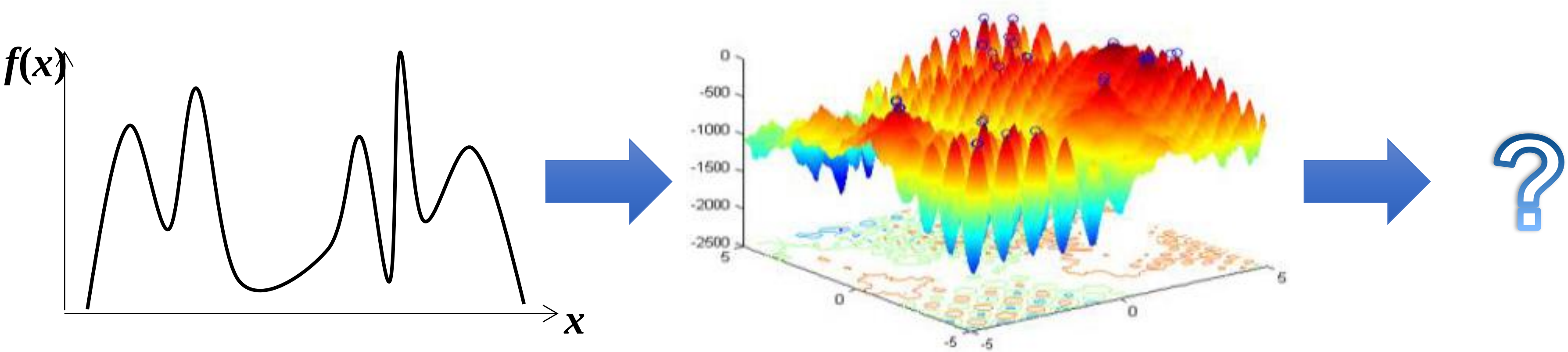




Improved PSO variants:

local neighborhood topology
multi-swarm PSOs
try to introduce more exemplars to improve search diversity

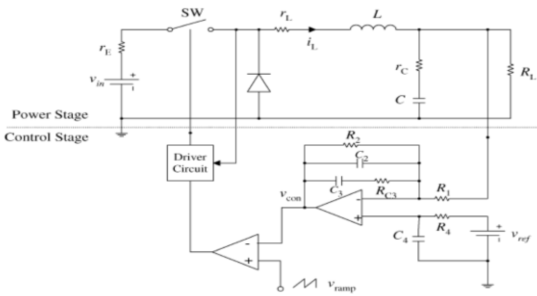
Challenge 2: Curse of Dimensionality

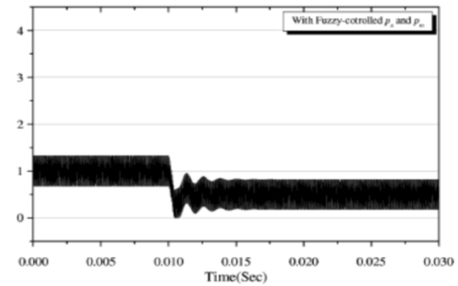


- **The search space is enlarged exponentially**
- **The number of local optima increases rapidly**
- **The execution time becomes too long**

Other Challenges

❑ Mixed Variable Optimization

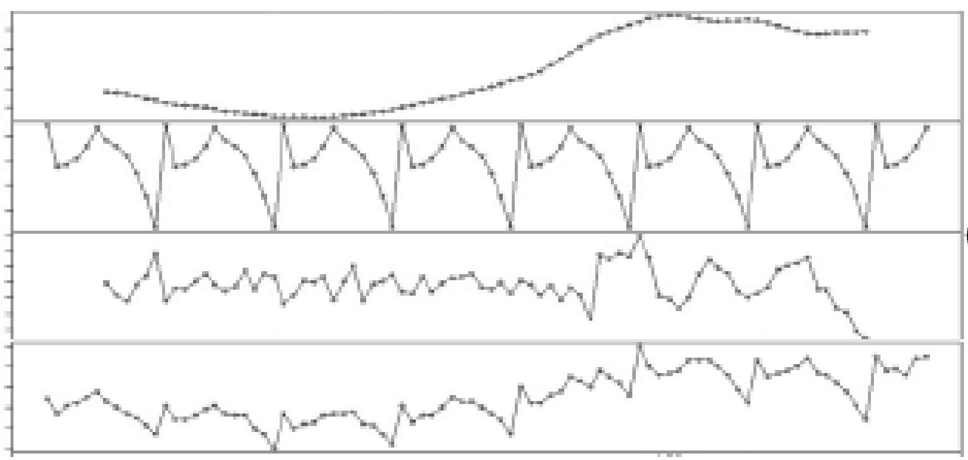




Optimal Design of Power Electronic Circuits

continuous variables
categorical variables

❑ Optimization under Uncertain / Dynamic Environment

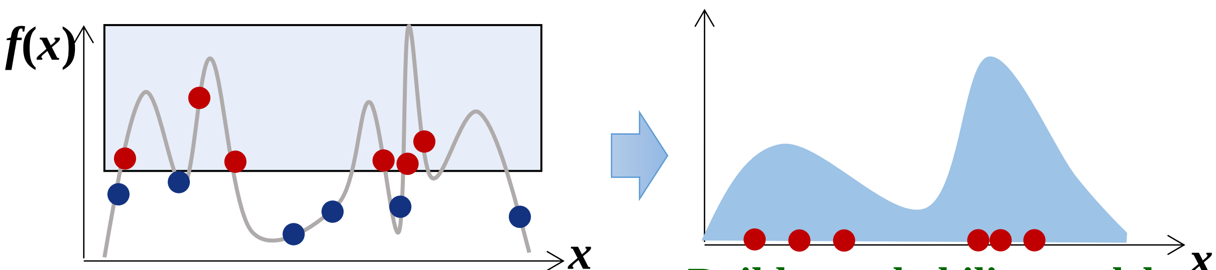


Maximize Profit
Minimize Risk

Robust Optimization

Probability-Based EAs (1)

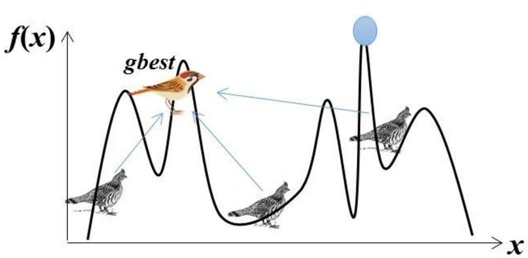
- **Estimation of Distribution Algorithm (EDA, Lozano et al., 2002)**: building and sampling explicit probabilistic models of promising candidate solutions



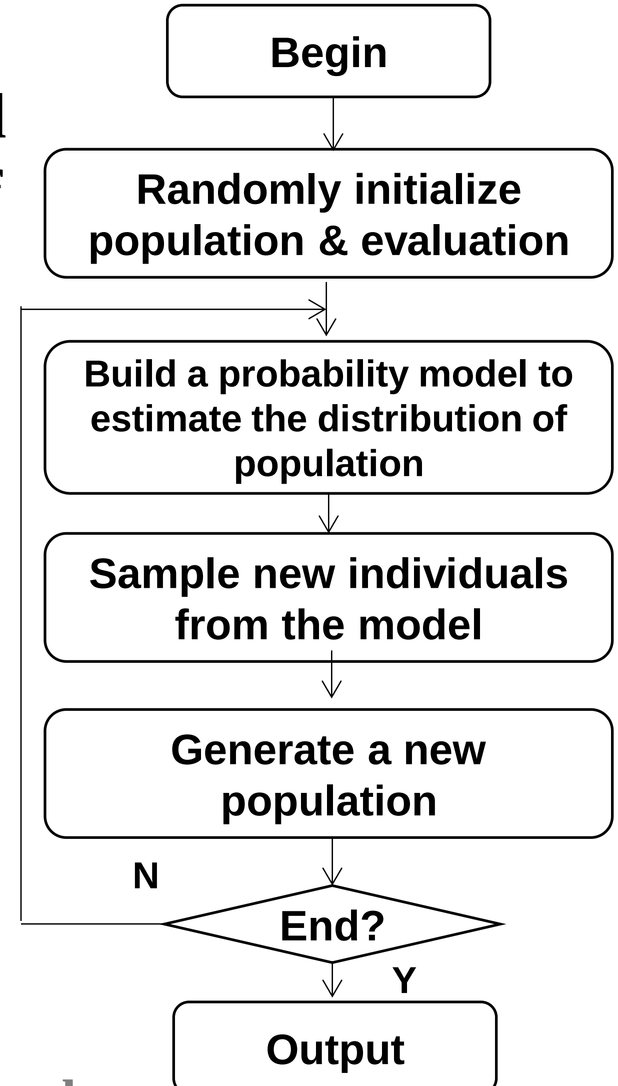
Select promising candidate solutions

Build a probability model to estimate the distribution of candidate solution

EDA: focus on the global fitness landscape

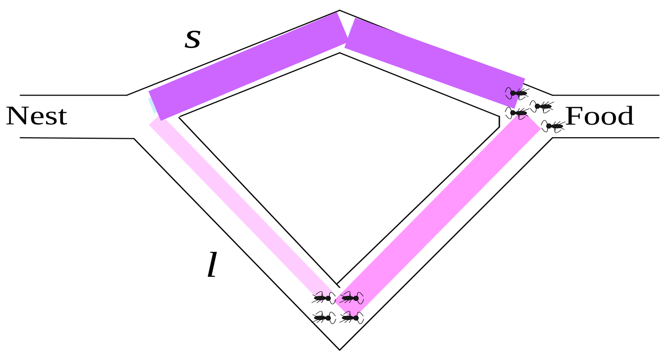


PSO: focus on the local neighborhood around the exemplar



Probability-Based EAs (2)

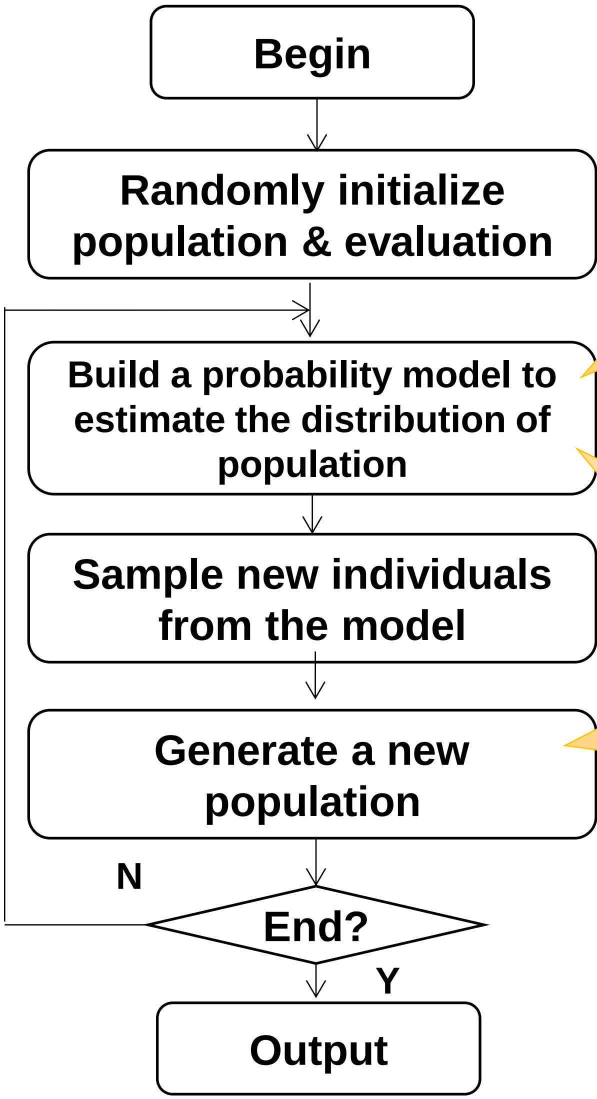
- In a broad sense, there are also some other EC algorithms that implicitly use probability distribution
  - Ant Colony Optimization (ACO, Dorigo et al., 1996)



The pheromone depositing behavior can be viewed as building a probability distribution in the search space.

- Strengths
  - Good at global optimization and diversity preservation
- Weakness
  - Speed, Precision

An Outline of Our Study



1 Probability Distribution Based EAs for Seeking Multiple Solutions

Build a framework with local (niched) probability distribution to improve search diversity of EC

2 Probability Distribution Based EAs for Discrete Optimization

Build a framework for probability distribution based EC algorithms for solving mixed-variable optimization problems

Outline

1. Background

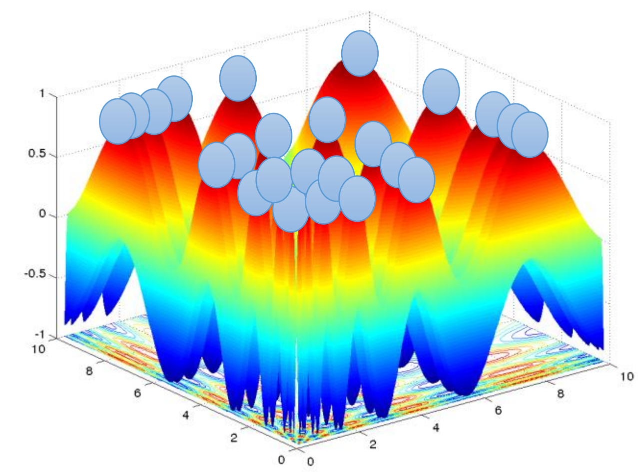
2. Probability Distribution Based EAs for Seeking Multiple Solutions

3. Probability Distribution Based EAs for Discrete Optimization

4. Applications & Future Work

Motivation

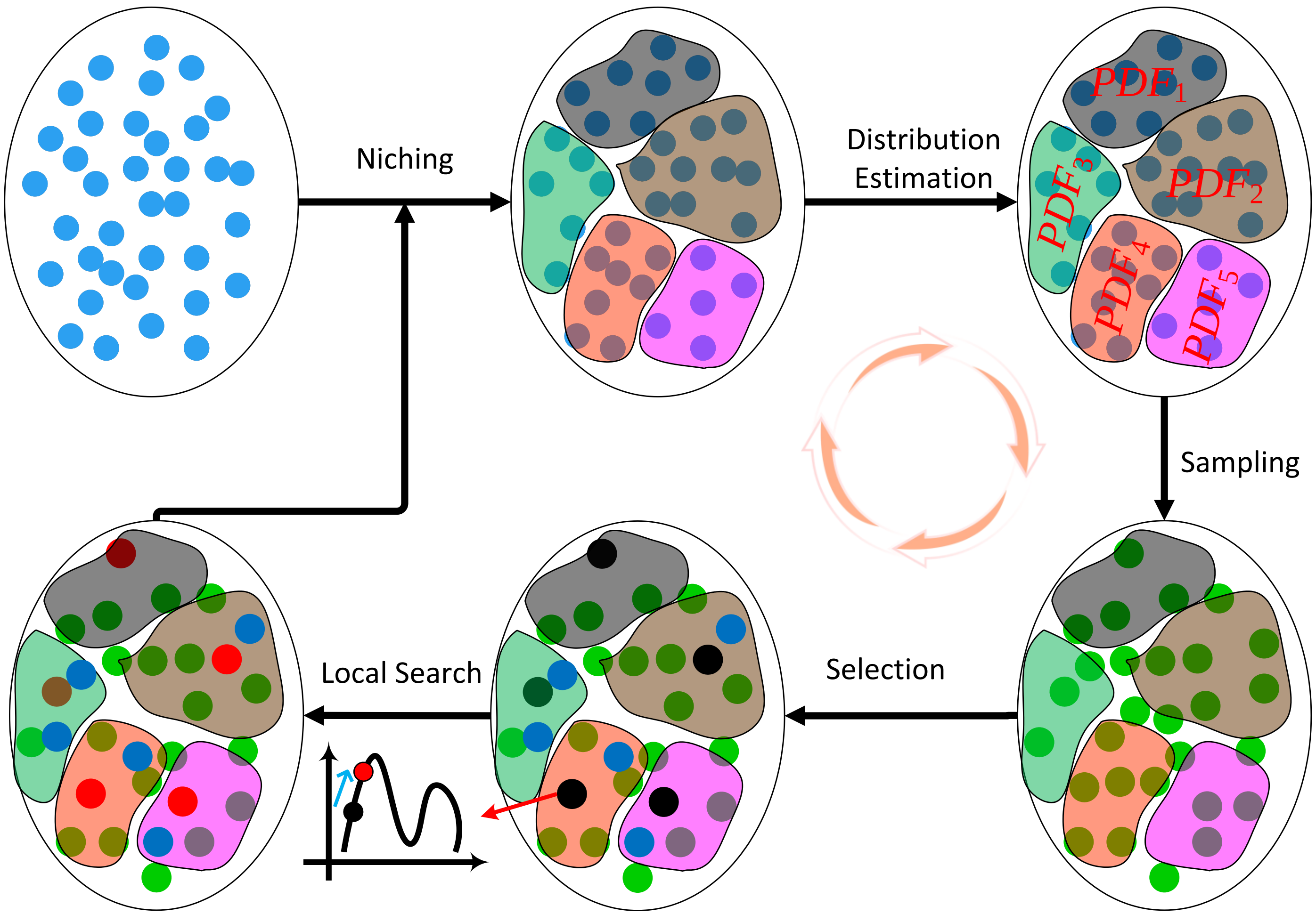
- Seeking multiple optimal solutions requires **extremely high search diversity**
  - By now, almost all approaches to seek multiple solutions are based on PSO or DE
  - On some high dimension problems with many local optima, the detection rates of many approaches are below 15%, or even 0%.



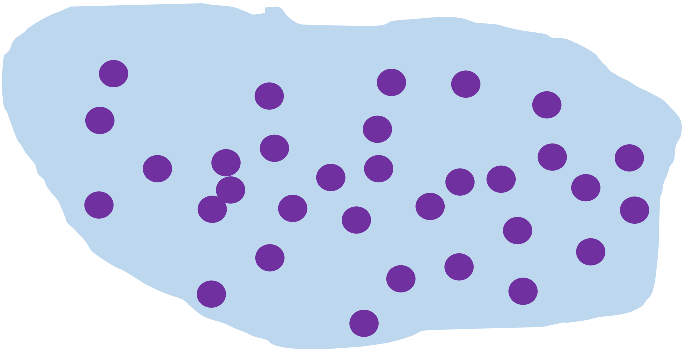
- EDA has shown good search diversity, but the potential of probability distribution based evolution for seeking multiple solutions have not been explored

Framework

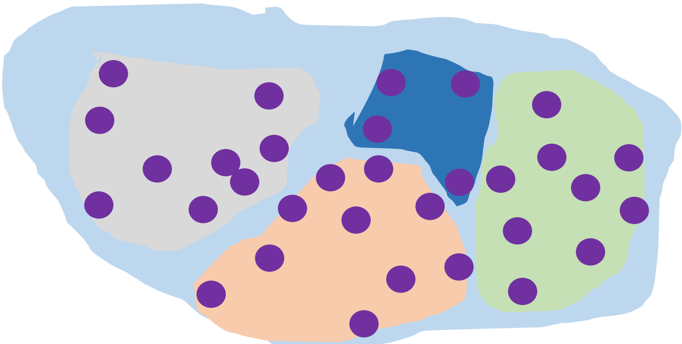
- ❑ **Combine Probability-Distribution Based EC with Niching**



Niching



Population



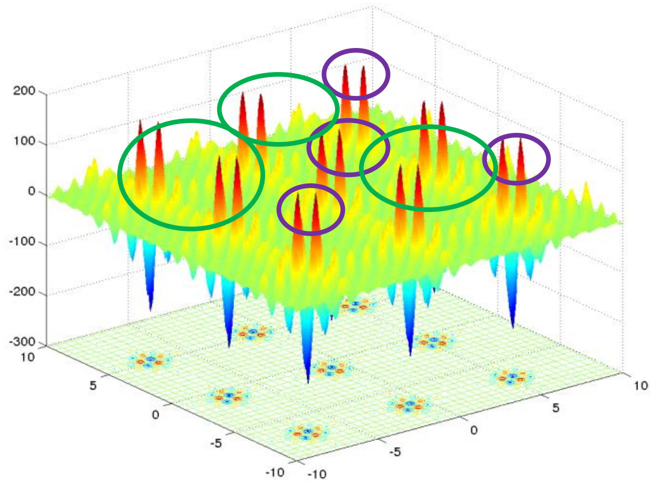
Niching

- Fitness Sharing (Goldberg and Richardson, 1987)
- Crowding (Thomsen, 2004)
- Speciation (Li, 2005)
- Clustering (Gao and Yen, et al., 2014; Qu and Suganthan, et al. 2012)

Multimodal EDA

1) Niching with Dynamic Niche Size

- $\boldsymbol{C}=\{c_1,c_2,\ldots,c_t\}$(different integers)
- Random selection before each generation
- Small size beneficial for exploitation
- Large size profitable for exploration
- Better balance between exploration and exploitation (than with a fixed size)

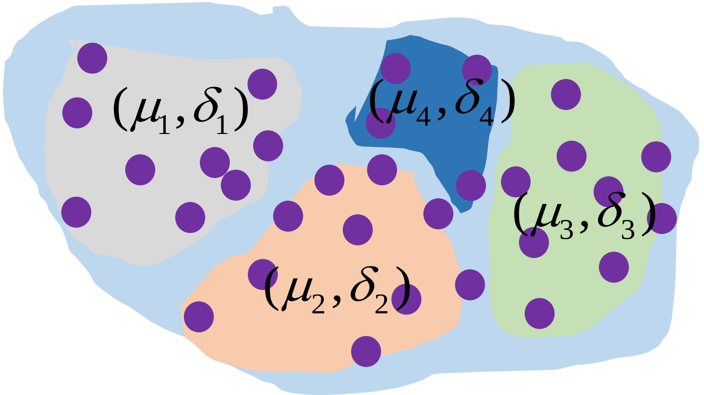


2) Distribution Estimation

- Estimating each niche separately
- All individuals in one niche participating in the estimation to keep high diversity

$$\mu_i^d = \frac{1}{M}\sum_{j=1}^{M} x_j^d$$

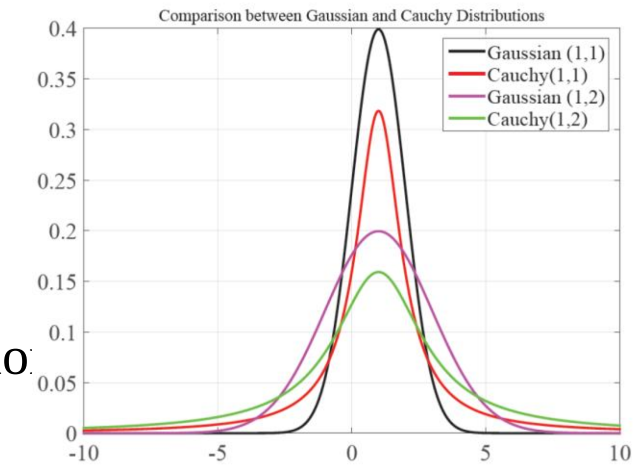
$$\delta_i^d = \sqrt{\frac{1}{M-1}\sum_{j=1}^{M}(x_i^d - \mu_i^d)^2}$$



Multimodal EDA

- **3) Sampling with Cauchy / Gaussian Distribution**

$$C_i = \begin{cases} Cauchy(\mu_i, \delta_i) & if\ rand() \le 0.5 \\ Gaussian(\mu_i, \delta_i) & \text{otherwise} \end{cases}$$



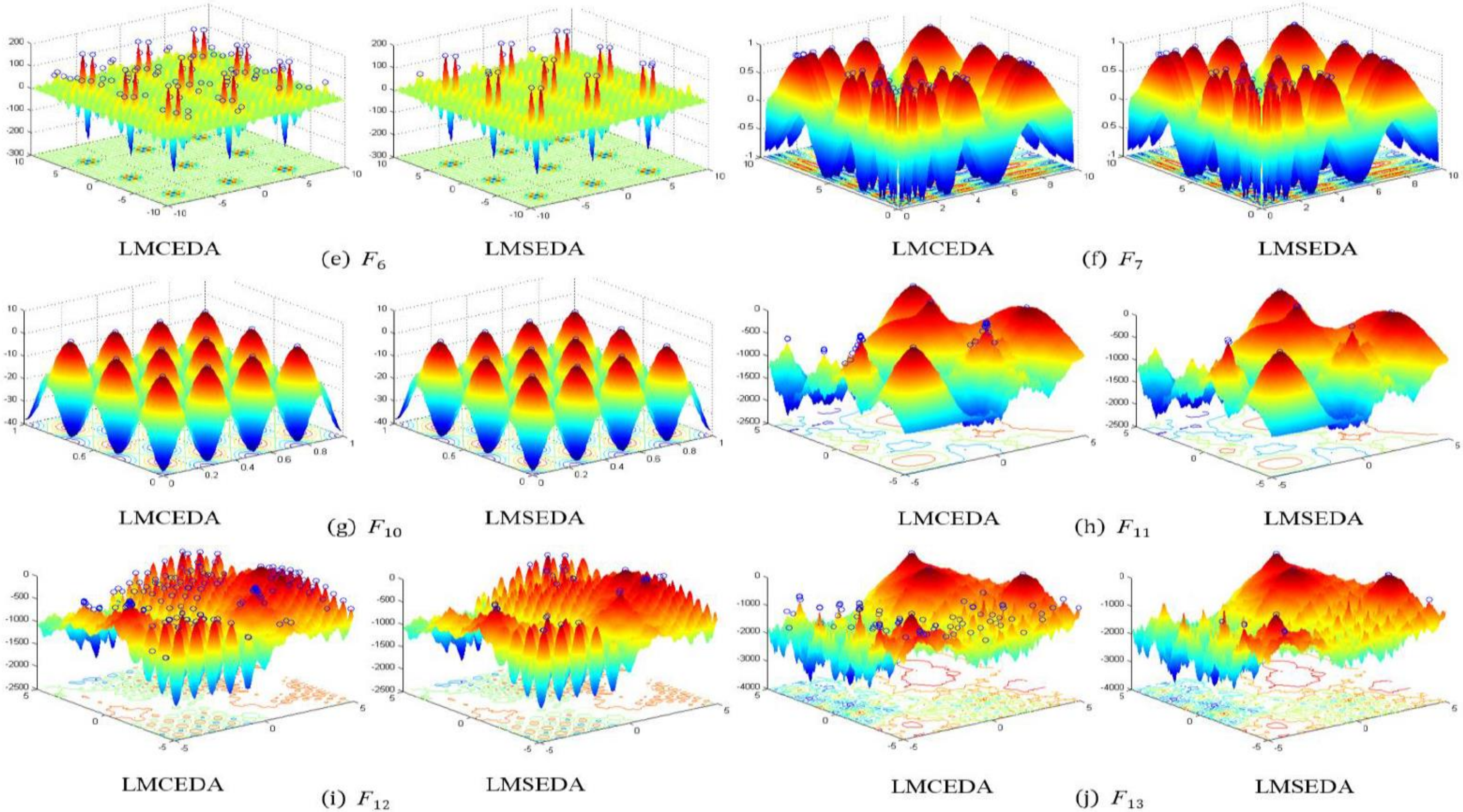
- Cauchy Distribution (long tail, beneficial for explo
- Gaussian Distribution (narrow tail, profitable for exploitation)
- Better balance between exploration and exploration

- **4) Selection with the Neighborhood Replacement Rule**
- **5) Local Search to Accelerate Search**
  - Adaptively conduct around seeds of niches
  - Local search by sampling *N* points using Gaussian Distribution around seeds

$$N(\mu, \delta) \quad \mu = seed_i, \delta = 1.0E-4$$

Results of Multimodal EDA

Clustering for Crowding (MCEDA) **Clustering for Speciation (MSEDA)**



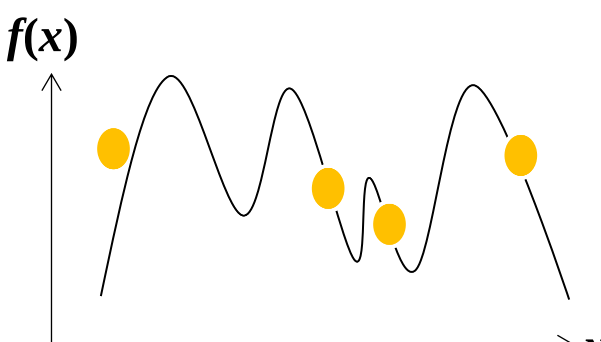
Results of Multimodal EDA

F	Self_CCDE			Self_CSDE			LoICDE			LoISDE			PNPCDE			LMCEDA			LMSEDA		
	PR	SR	CS	PR	SR	CS	PR	SR	CS	PR	SR	CS	PR	SR	CS	PR	SR	CS	PR	SR	CS
F_1	**1.000**	1.000	3.55E+2	**1.000**	1.000	7.15E+2	**1.000**	1.000	1.73E+2	**1.000**	1.000	1.68E+2	**1.000**	1.000	1.62E+2	**1.000**	1.000	3.23E+2	**1.000**	1.000	3.41E+2
F_2	**1.000**	1.000	7.61E+2	**1.000**	1.000	1.03E+3	**1.000**	1.000	1.24E+3	0.486	0.353	3.25E+4	**1.000**	1.000	1.08E+3	**1.000**	1.000	1.07E+3	**1.000**	1.000	8.97E+2
F_3	**1.000**	1.000	5.55E+2	**1.000**	1.000	7.23E+2	**1.000**	1.000	5.84E+2	**1.000**	1.000	7.48E+2	**1.000**	1.000	8.33E+2	**1.000**	1.000	5.06E+2	**1.000**	1.000	4.84E+2
F_4	**1.000**	1.000	7.07E+3	0.907	0.706	2.97E+4	**1.000**	1.000	1.48E+4	0.265	0.020	4.91E+4	**1.000**	1.000	2.28E+4	**1.000**	1.000	1.18E+4	**1.000**	1.000	6.37E+3
F_5	**1.000**	1.000	1.97E+3	**1.000**	1.000	3.38E+3	**1.000**	1.000	1.88E+3	0.814	0.627	1.91E+4	**1.000**	1.000	3.69E+3	**1.000**	1.000	3.78E+3	**1.000**	1.000	2.51E+3
F_6	0.972	0.647	1.37E+5	0.760	0.020	1.97E+5	**1.000**	1.000	9.58E+4	0.056	0.000	2.00E+5	0.806	0.157	1.97E+5	0.990	0.843	1.64E+5	0.973	0.588	1.43E+5
F_7	**0.884**	0.020	1.97E+5	0.696	0.000	2.00E+5	0.858	0.020	2.00E+5	0.029	0.000	2.00E+5	0.875	0.000	2.00E+5	0.782	0.000	2.00E+5	0.712	0.000	2.00E+5
F_8	**0.997**	0.902	2.35E+5	0.695	0.000	4.00E+5	0.000	0.000	4.00E+5	0.012	0.000	4.00E+5	0.000	0.000	4.00E+5	0.352	0.000	4.00E+5	0.622	0.000	4.00E+5
F_9	0.459	0.000	4.00E+5	0.265	0.000	4.00E+5	0.421	0.000	4.00E+5	0.005	0.000	4.00E+5	0.473	0.000	4.00E+5	0.333	0.000	4.00E+5	0.281	0.000	4.00E+5
F_{10}	**1.000**	1.000	8.15E+3	**1.000**	1.000	1.51E+4	**1.000**	1.000	3.04E+4	0.083	0.000	2.00E+5	**1.000**	1.000	2.16E+4	**1.000**	1.000	1.02E+4	0.998	0.980	1.23E+4
F_{11}	0.824	0.255	1.92E+5	0.565	0.000	2.00E+5	0.667	0.000	2.00E+5	0.167	0.000	2.00E+5	0.667	0.000	2.00E+5	0.667	0.000	2.00E+5	0.905	0.451	1.80E+5
F_{12}	0.591	0.000	2.00E+5	0.409	0.000	2.00E+5	0.615	0.000	2.00E+5	0.125	0.000	2.00E+5	0.015	0.000	2.00E+5	0.750	0.000	2.00E+5	**0.990**	0.922	1.09E+5
F_{13}	0.667	0.000	2.00E+5	0.493	0.000	2.00E+5	0.634	0.000	2.00E+5	0.167	0.000	2.00E+5	0.637	0.000	2.00E+5	0.667	0.000	2.00E+5	0.667	0.000	2.00E+5
F_{14}	**0.667**	0.000	4.00E+5	0.500	0.000	4.00E+5	0.663	0.000	4.00E+5	0.167	0.000	4.00E+5	0.592	0.000	4.00E+5	**0.667**	0.000	4.00E+5	**0.667**	0.000	4.00E+5
F_{15}	0.370	0.000	4.00E+5	0.287	0.000	4.00E+5	0.358	0.000	4.00E+5	0.125	0.000	4.00E+5	0.152	0.000	4.00E+5	0.699	0.000	4.00E+5	**0.738**	0.000	4.00E+5
F_{16}	0.663	0.000	4.00E+5	0.232	0.000	4.00E+5	0.621	0.000	4.00E+5	0.167	0.000	4.00E+5	0.010	0.000	4.00E+5	**0.667**	0.000	4.00E+5	**0.667**	0.000	4.00E+5
F_{17}	0.260	0.000	4.00E+5	0.103	0.000	4.00E+5	0.238	0.000	4.00E+5	0.076	0.000	4.00E+5	0.000	0.000	4.00E+5	0.456	0.000	4.00E+5	**0.620**	0.000	4.00E+5
F_{18}	0.353	0.000	4.00E+5	0.016	0.000	4.00E+5	0.222	0.000	4.00E+5	0.157	0.000	4.00E+5	0.160	0.000	4.00E+5	0.657	0.000	4.00E+5	**0.660**	0.000	4.00E+5
F_{19}	0.150	0.000	4.00E+5	0.000	0.000	4.00E+5	0.054	0.000	4.00E+5	0.027	0.000	4.00E+5	0.000	0.000	4.00E+5	0.451	0.000	4.00E+5	**0.458**	0.000	4.00E+5
F_{20}	0.069	0.000	4.00E+5	0.000	0.000	4.00E+5	0.125	0.000	4.00E+5	0.088	0.000	4.00E+5	0.000	0.000	4.00E+5	**0.250**	0.000	4.00E+5	**0.250**	0.000	4.00E+5
bprs		9			5			7			2			6			9			13	

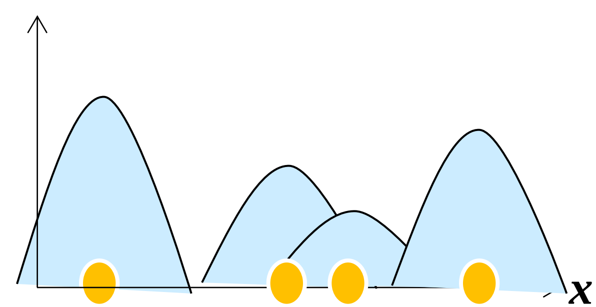
MEDA performs well on high-dimensional multimodal problems with many local optima, as they can locate more globally optimal solutions.

Multimodal ACO

- ACO is developed for discrete optimization (Dorigo et al., 1996)
- By modeling pheromone as probability distribution, ACO can also solve continuous problems (Socha and Dorigo, 2008)
- The capability of EDA is usually limited by the distribution model (usually Cauchy or Gaussian)
- The pheromone depositing behavior of ACO may provide a more flexible probability distribution



Distribution of Individuals



Pheromone Deposition

Multimodal ACO

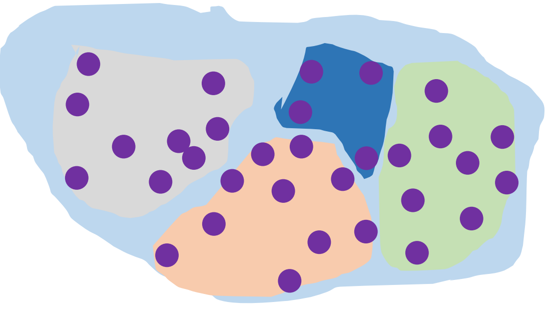
❑ 1) Niching with Dynamic Niche Size

❑ 2) Pheromone Deposition

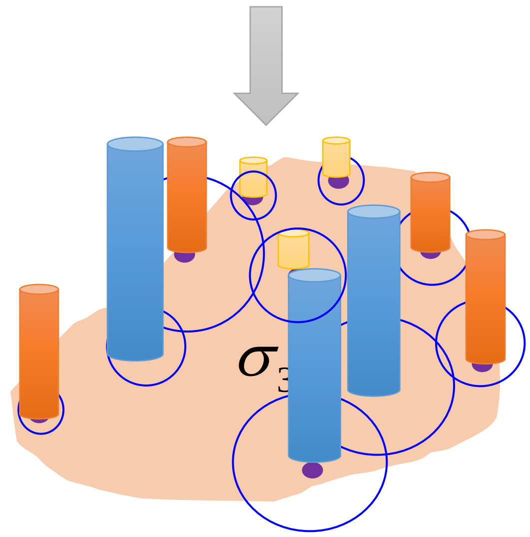
- Each individual deposits pheromone according to its fitness
- The pheromone of an individual determines the probability of sampling new children around this individual
- Adaptive configuration for the parameter σ

$$w_i = \frac{1}{\sigma NP\sqrt{2\pi}} e^{-\frac{(rank(i)-1)^2}{2\sigma^2 NP^2}}$$

$$p_j = \frac{w_j}{\sum_{i=1}^{NP} w_i}$$



Niching



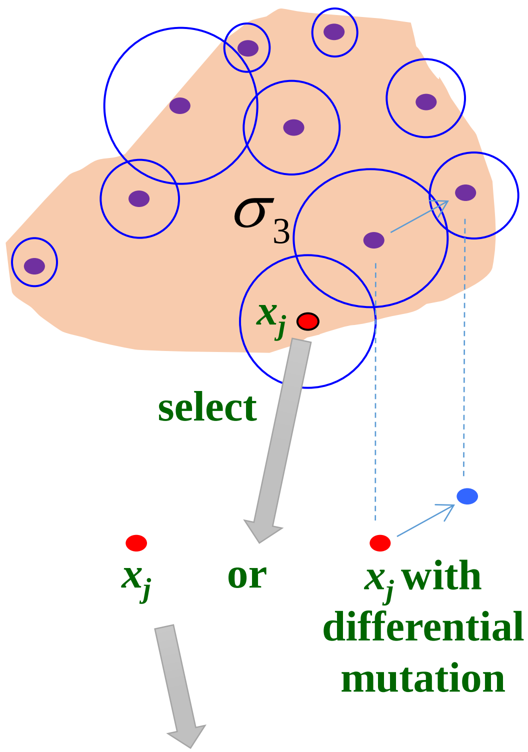
Pheromone Deposition to form a probability distribution for each niche

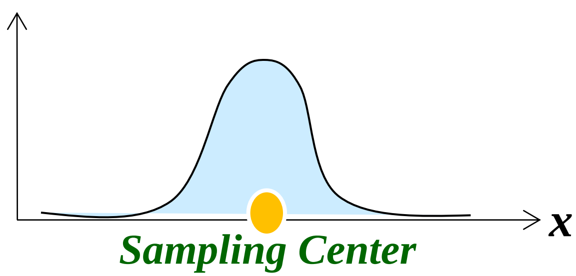
Multimodal ACO

- **3) Sampling new solutions**
  - Step 1. Select an individual according to their pheromone distribution
  - Step 2. Determine the center of sampling

$$\mu = \begin{cases} x_j \text{ (selected solution)} & rand() \le 0.5 \\ u = x_j + F(\mathbf{x}_{seed} - x_j) & \text{otherwise} \end{cases}$$

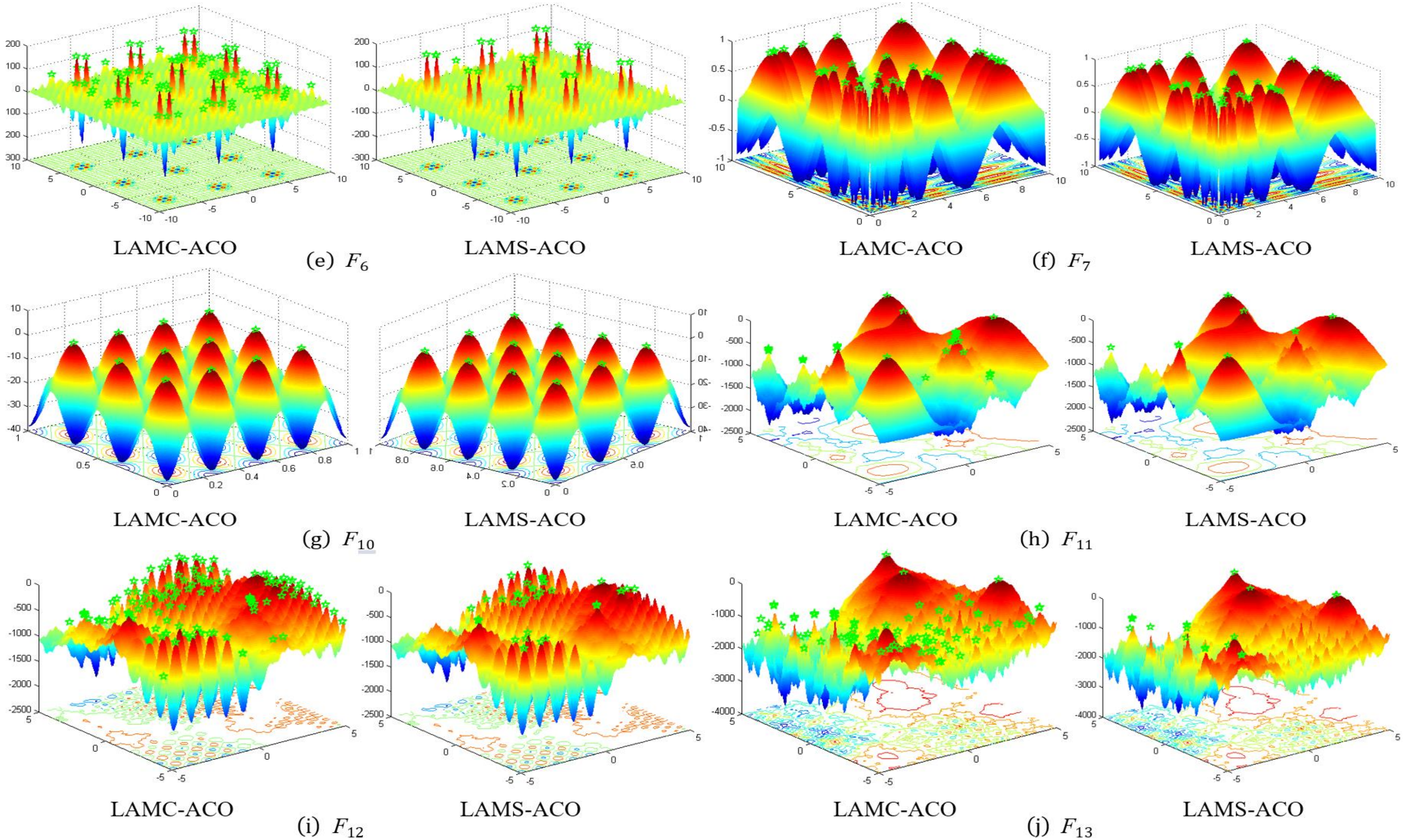
  - Step 3. Sampling with Gaussian Distribution
  - Step 4. Repeat 1-3 until a new generation has been built
- **4) Selection with the Neighborhood Replacement Rule**
- **5) Local Search**





Sampling using Gaussian Distribution

Results of AM-ACO



Results of AM-ACO

F	Self_CSDE			LoICDE			LoISDE			PNPCDE			MOMMOP			LAMC-ACO			LAMS-ACO		
	PR	SR	CS	PR	SR	CS	PR	SR	CS	PR	SR	CS	PR	SR	CS	PR	SR	CS	PR	SR	CS
F_1	**1.000**	1.000	7.15E+2	**1.000**	1.000	1.73E+2	**1.000**	1.000	1.68E+2	**1.000**	1.000	1.62E+2	**1.000**	1.000	1.66E+2	**1.000**	1.000	2.59E+2	**1.000**	1.000	2.14E+2
F_2	**1.000**	1.000	1.03E+3	**1.000**	1.000	1.24E+3	0.486	0.353	3.25E+4	**1.000**	1.000	1.08E+3	**1.000**	1.000	1.09E+3	**1.000**	1.000	5.35E+2	**1.000**	1.000	4.90E+2
F_3	**1.000**	1.000	7.23E+2	**1.000**	1.000	5.84E+2	**1.000**	1.000	7.48E+2	**1.000**	1.000	8.33E+2	**1.000**	1.000	9.58E+2	**1.000**	1.000	3.50E+2	**1.000**	1.000	3.53E+2
F_4	0.907	0.706	2.97E+4	**1.000**	1.000	1.48E+4	0.265	0.020	4.91E+4	**1.000**	1.000	2.28E+4	**1.000**	1.000	3.50E+4	**1.000**	1.000	5.04E+3	**1.000**	1.000	3.50E+3
F_5	**1.000**	1.000	3.38E+3	**1.000**	1.000	1.88E+3	0.814	0.627	1.91E+4	**1.000**	1.000	3.69E+3	**1.000**	1.000	1.48E+4	**1.000**	1.000	1.42E+3	**1.000**	1.000	1.10E+3
F_6	0.760	0.020	1.97E+5	**1.000**	1.000	9.58E+4	0.056	0.000	2.00E+5	0.806	0.157	1.97E+5	**1.000**	1.000	5.61E+4	0.999	0.980	1.10E+5	0.990	0.824	1.05E+5
F_7	0.696	0.000	2.00E+5	0.858	0.020	2.00E+5	0.029	0.000	2.00E+5	0.875	0.000	2.00E+5	**1.000**	1.000	6.32E+4	0.789	0.000	2.00E+5	0.716	0.000	2.00E+5
F_8	0.695	0.000	4.00E+5	0.000	0.000	4.00E+5	0.012	0.000	4.00E+5	0.000	0.000	4.00E+5	**1.000**	1.000	2.85E+5	0.680	0.000	4.00E+5	0.782	0.000	4.00E+5
F_9	0.265	0.000	4.00E+5	0.421	0.000	4.00E+5	0.005	0.000	4.00E+5	0.473	0.000	4.00E+5	**1.000**	1.000	2.95E+5	0.348	0.000	4.00E+5	0.295	0.000	4.00E+5
F_{10}	**1.000**	1.000	1.51E+4	**1.000**	1.000	3.04E+4	0.083	0.000	2.00E+5	**1.000**	1.000	2.16E+4	**1.000**	1.000	4.24E+4	**1.000**	1.000	9.47E+3	**1.000**	1.000	9.02E+3
F_{11}	0.565	0.000	2.00E+5	0.667	0.000	2.00E+5	0.167	0.000	2.00E+5	0.667	0.000	2.00E+5	0.938	0.647	1.73E+5	0.683	0.000	2.00E+5	**0.974**	0.843	1.40E+5
F_{12}	0.409	0.000	2.00E+5	0.615	0.000	2.00E+5	0.125	0.000	2.00E+5	0.015	0.000	2.00E+5	0.949	0.627	1.73E+5	0.824	0.098	1.99E+5	**0.983**	0.863	1.03E+5
F_{13}	0.493	0.000	2.00E+5	0.634	0.000	2.00E+5	0.167	0.000	2.00E+5	0.637	0.000	2.00E+5	0.667	0.000	2.00E+5	0.667	0.000	2.00E+5	0.676	0.000	2.00E+5
F_{14}	0.500	0.000	4.00E+5	0.663	0.000	4.00E+5	0.167	0.000	4.00E+5	0.592	0.000	4.00E+5	**0.667**	0.000	4.00E+5	**0.667**	0.000	4.00E+5	**0.667**	0.000	4.00E+5
F_{15}	0.287	0.000	4.00E+5	0.358	0.000	4.00E+5	0.125	0.000	4.00E+5	0.152	0.000	4.00E+5	0.627	0.000	4.00E+5	0.740	0.000	4.00E+5	**0.748**	0.000	4.00E+5
F_{16}	0.232	0.000	4.00E+5	0.621	0.000	4.00E+5	0.167	0.000	4.00E+5	0.010	0.000	4.00E+5	0.650	0.000	4.00E+5	**0.667**	0.000	4.00E+5	**0.667**	0.000	4.00E+5
F_{17}	0.103	0.000	4.00E+5	0.238	0.000	4.00E+5	0.076	0.000	4.00E+5	0.000	0.000	4.00E+5	0.512	0.000	4.00E+5	0.608	0.000	4.00E+5	**0.708**	0.000	4.00E+5
F_{18}	0.016	0.000	4.00E+5	0.222	0.000	4.00E+5	0.157	0.000	4.00E+5	0.160	0.000	4.00E+5	0.497	0.000	4.00E+5	**0.667**	0.000	4.00E+5	**0.667**	0.000	4.00E+5
F_{19}	0.000	0.000	4.00E+5	0.054	0.000	4.00E+5	0.027	0.000	4.00E+5	0.000	0.000	4.00E+5	0.223	0.000	4.00E+5	0.500	0.000	4.00E+5	**0.502**	0.000	4.00E+5
F_{20}	0.000	0.000	4.00E+5	0.125	0.000	4.00E+5	0.088	0.000	4.00E+5	0.000	0.000	4.00E+5	0.125	0.000	4.00E+5	0.272	0.000	4.00E+5	**0.348**	0.000	4.00E+5
bprs		5			7			2			6			11			9			15	

AM-ACO achieved even better results than MEDA. E.g., for the most difficult problem *F*20, AM-ACO yields a detection rate 34.8%, MEDA is 25%, while most other existing approaches are below 12.5%

Conclusion

- Probability Distribution Based EC algorithms are good at maintaining sufficient search diversity
- Combing Probability Distribution Based EC & Niching is promising for problems with high diversity requirement, e.g., seeking multiple solutions in multimodal optimization
- Local search and parameter adaption are helpful to improve search efficiency
- Implicit Probability Distribution Approaches, e.g., ACO, may sometimes provide a more flexible way to build distribution

"Multimodal Estimation of Distribution Algorithms", ***IEEE Transactions on Cybernetics***, vol. 47, no. 3, pp. 636-650, 2017.

"Adaptive Multimodal Continuous Ant Colony Optimization", ***IEEE Transactions on Evolutionary Computation***, vol. 21, no. 2, pp. 191-205, 2017.

Outline

1. Background
   - Multimodal Optimization
   - Probability Distribution Based EAs

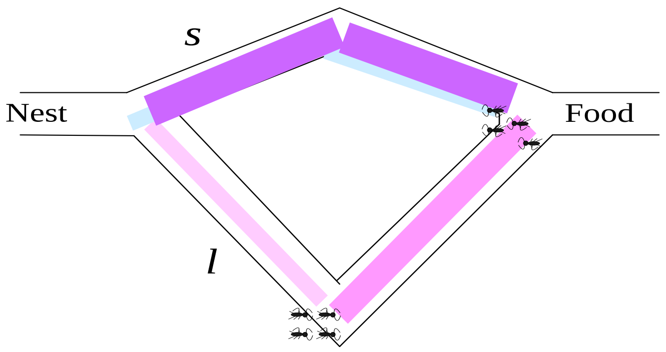
2. Probability Distribution Based EAs for Seeking Multiple Solutions

3. **Probability Distribution Based EAs for Discrete Optimization**

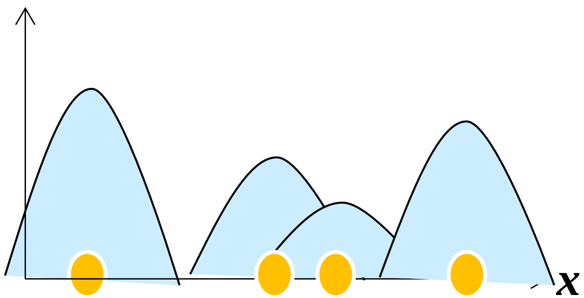
4. Applications & Future Work

Motivation

- ❑ **Probability Distribution is suitable on both continuous and discrete problems**



Probability distribution



Pheromone Deposition

ACO in discrete space **ACO in continuous space**

- ❑ **Some popular EC algorithms, e.g., PSO, are originally defined on continuous real vector space**

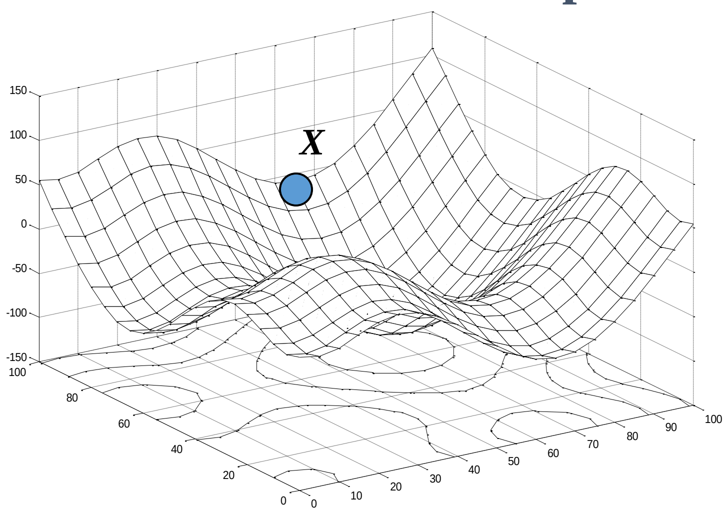
Velocity Update $v_i^j \leftarrow wv_i^j + c_1 r_1^j (pbest_i^j - x_i^j) + c_2 r_2^j (gbest^j - x_i^j), \quad j = 1,2,\ldots,n$

- ❑ **Is it possible to use Probability Distribution to build a more general discrete PSO (DE, etc.) framework?**

Set-Based Representation

COPs can be formulated in the abstract as “find from a set E a subset X that satisfies some constraints and optimizes the objective function f” (Lin and Kernighan 1973)

Continuous Search Space

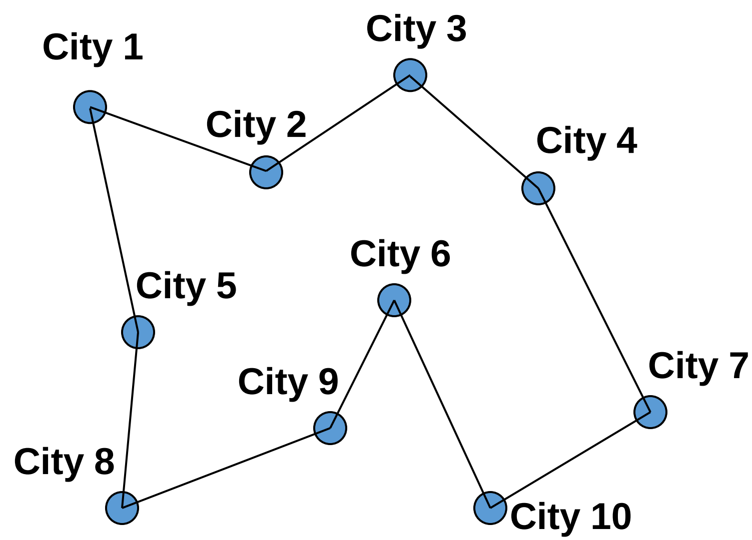


Representation: Real Vector

$$\boldsymbol{x_i}\,(x_i^1, x_i^2, \cdots, x_i^n)$$

?

Discrete Search Space

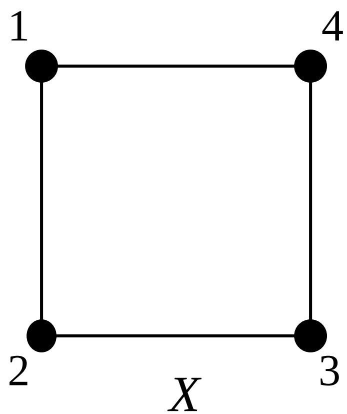


Representation: Set

{(1,2), (2,3), (3,4), …, (1,5)}

Set-Based Representation

- **Position: a crisp set**



$$E = \{(1,2),(1,3),(1,4),(2,3),(2,4),(3,4)\}$$

$$E^1 = \{(1,2),(1,3),(1,4)\} \qquad E^2 = \{(1,2),(2,3),(2,4)\}$$

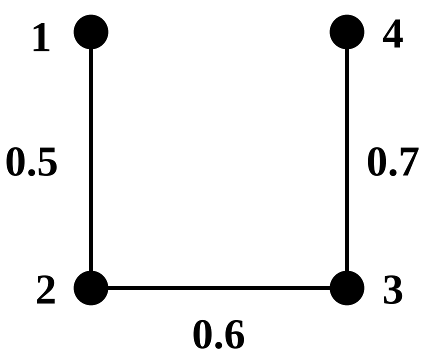
$$E^3 = \{(1,3),(2,3),(3,4)\} \qquad E^4 = \{(1,4),(2,4),(3,4)\}$$

$$X = \{(1,2),(2,3),(3,4),(1,4)\}$$

$$X^1 = \{(1,2),(1,4)\} \qquad X^2 = \{(1,2),(2,3)\}$$

$$X^3 = \{(2,3),(3,4)\} \qquad X^4 = \{(1,4),(3,4)\}$$

- **Velocity: a set with possibilities**



$$V = \{(1,2)/0.5,(2,3)/0.6,(3,4)/0.7\}$$

$$V^1 = \{(1,2)/0.5\}$$

$$V^2 = \{(1,2)/0.5,(2,3)/0.6\}$$

$$V^3 = \{(2,3)/0.6\}$$

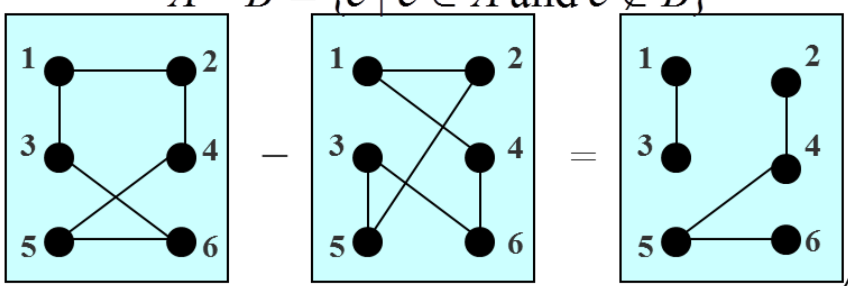
Velocity Update

- Essence of velocity update: build a **probability distribution of the "promising elements"** for particles to learn from

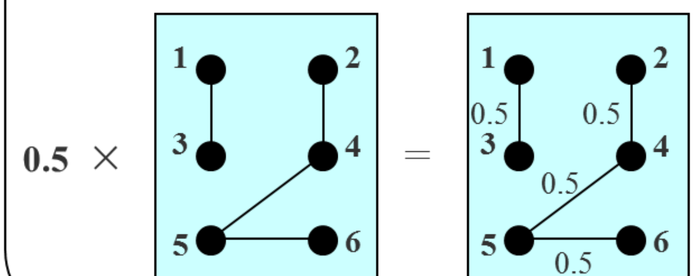
$$V_i^j \leftarrow \omega V_i^j + c_1 r_1^j \boxed{(PBest_i^j - X_i^j)} + \boxed{c_2 r_2^j (GBest^j - X_i^j)}, \qquad j = 1,2,\ldots,n$$

Position – Position
（crisp set – crisp et）

$$A - B = \{e \mid e \in A \text{ and } e \notin B\}$$



Coefficient * Crisp set



Union of two sets with possibilities

$V_1^1=\{(1, 2)/0.6, (1, 4)/0.9\}$ $\qquad$ $V_1^2=\{(1, 2)/0.9, (1, 3)/0.3\}$

$V_1^1 + V_1^2 = \{(1, 2)/0.9, (1, 3)/0.3, (1, 4)/0.9\}$

Position Update

- Particles uses the elements sampling from the probability distribution defined by “velocity” to update positions

Convert the velocity (probability distribution) into a crisp set

$$cut_\alpha(V_i^j) = \{e \mid e / p(e) \in V_i^j \text{ and } p(e) \geq \alpha\}$$

↓

Learn from the elements in $cut_\alpha(V_i^j)$

↓

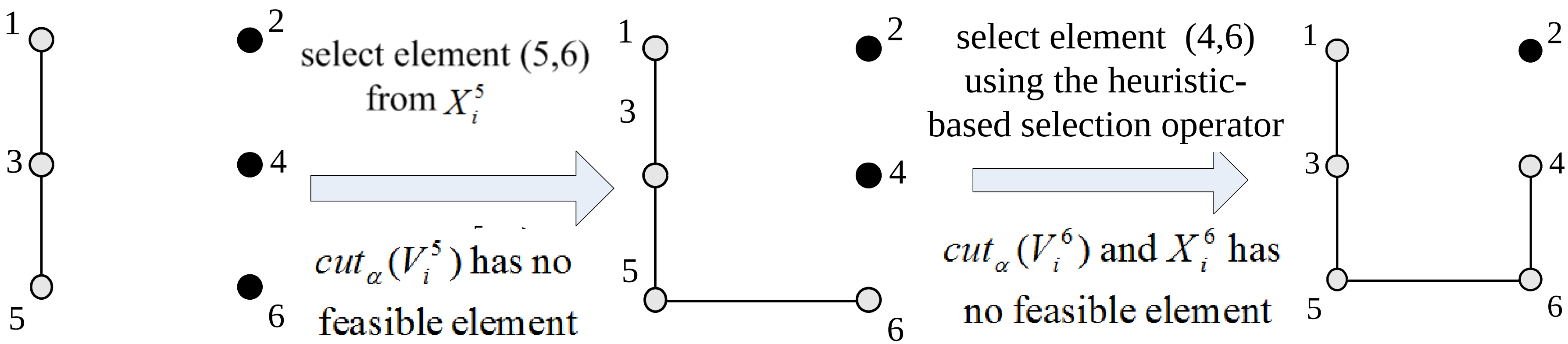
Reuse the elements from the previous position X_i^j

↓

Use other unselected elements to build a complete feasible solution

Constraint Handling

❑ Step-by-Step



❑ Build and Repair

X_i

Item 1	selected
Item 2	unselected
Item 3	selected
Item 4	unselected
Item 5	selected
Item 6	unselected

$cut_\alpha(V_i^j)$

{item 2, item 3, imtem4}

Build

New Position (infeasible)

Item 1	selected
Item 2	selected
Item 3	selected
Item 4	selected
Item 5	selected
Item 6	unselected

repair

Item 1	selected
Item 2	selected
Item 3	unselected
Item 4	selected
Item 5	selected
Item 6	unselected

Search Behavior

Velocity update in GPSO

Redefine **Discrete PSO**

$$v_i^j \leftarrow wv_i^j + c_1 r_1^j (pbest_i^j - x_i^j) + c_2 r_2^j (gbest^j - x_i^j), \quad j = 1,2,\ldots,n$$

→ **S-GPSO**

Velocity update in LPSO

$$v_i^j \leftarrow wv_i^j + c_1 r_1^j (pbest_i^j - x_i^j) + c_2 r_2^j (lbest_i^j - x_i^j), \quad j = 1,2,\ldots,n$$

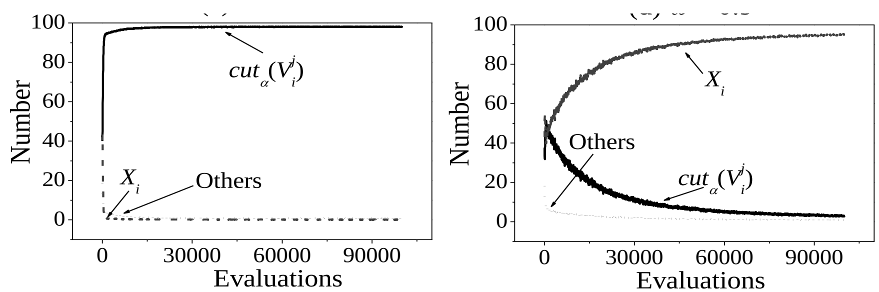
→ **S-LPSO**

Velocity update in CLPSO

$$v_i^j \leftarrow \omega \cdot v_i^j + cr^j (pbest_{f_i(j)}^j - x_i^j), \quad j = 1,2,\ldots,n$$

→ **S-CLPSO**

Different PSO variants can be directly extended to their discrete versions based on the S-PSO.



(e) $\omega = 1$ (f) decreasing ω

Parameter configurations in original PSO can be still used in S-PSO

Conclusion

- Probability Distribution can also be used to extend EC algorithms from continuous space to discrete space
- Different kinds of PSO variants (including multiobjective ones) can be extended to their discrete versions
- First verified on TSP and MKP,
- Solve other discrete optimization problems:
  - Vehicle routing (Gong et al., 2012)
  - Coverage array generation (Wu et al., 2015)

“A novel set-based particle swarm optimization method for discrete optimization problem”, ***IEEE Transactions on Evolutionary Computation***, 2010

"Set-Based Discrete Particle Swarm Optimization Based on Decomposition for Permutation-Based Multiobjective Combinatorial Optimization Problems", ***IEEE Trans. on Cybernetics***, in press

“Set-based discrete particle warm optimization and its applications: a survey”, *Frontiers of Computer Science*, 2018

Outline

1. Background

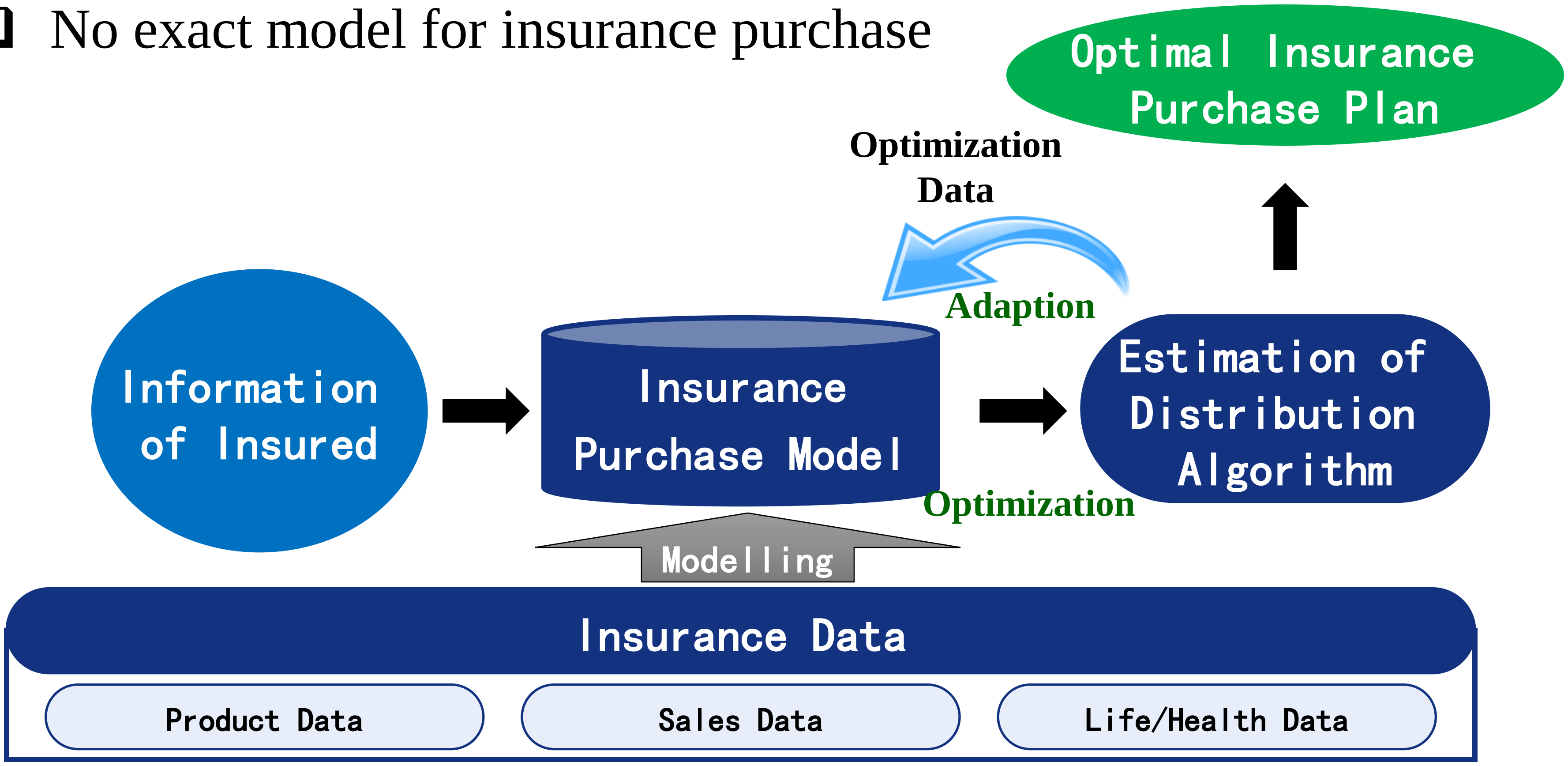
2. Probability Distribution Based EAs for Seeking Multiple Solutions

3. Probability Distribution Based EAs for Discrete Optimization

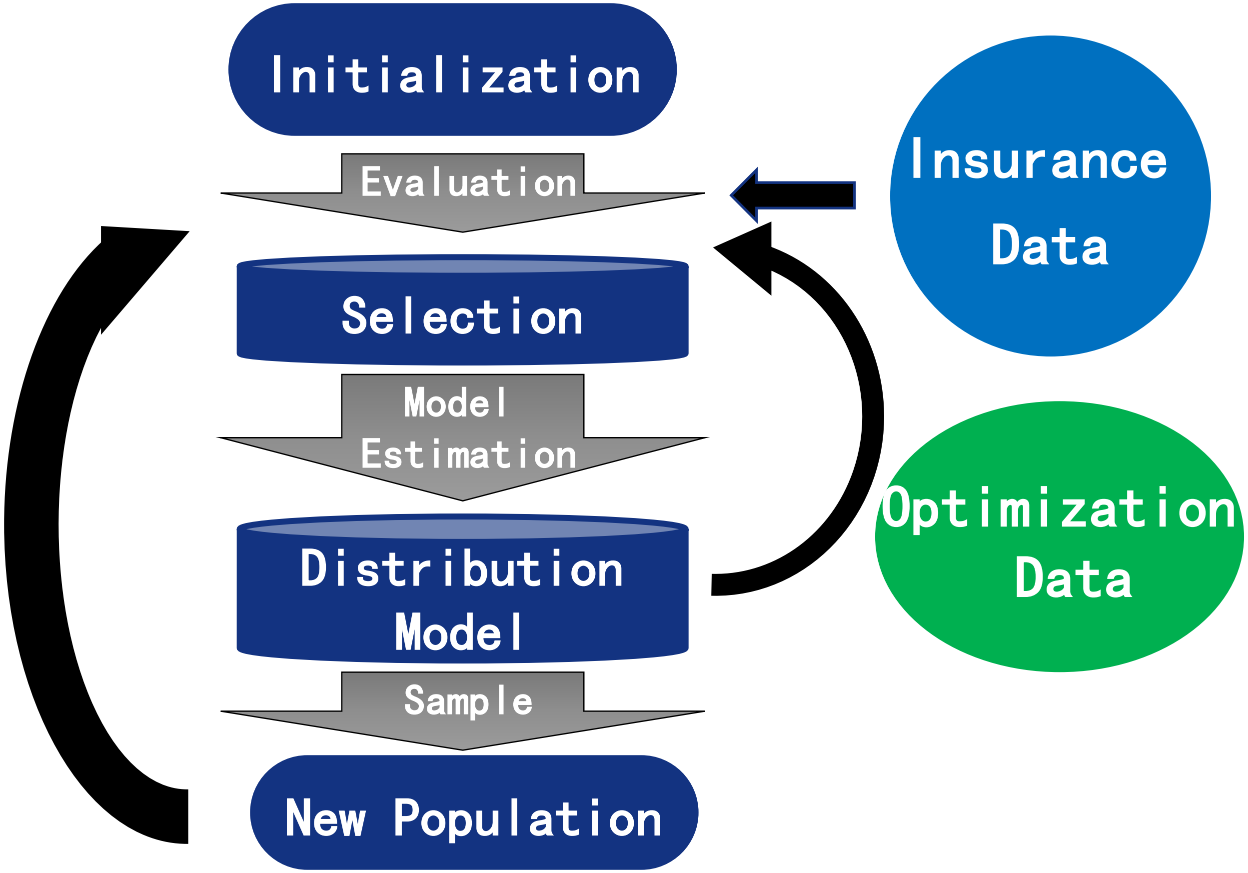
4. Applications & Future Work

Application 1: Insurance Purchase Optimization

- ❑ Insurance is an important way for investment and risk aversion
- ❑ Many insurance products exist
- ❑ No exact model for insurance purchase



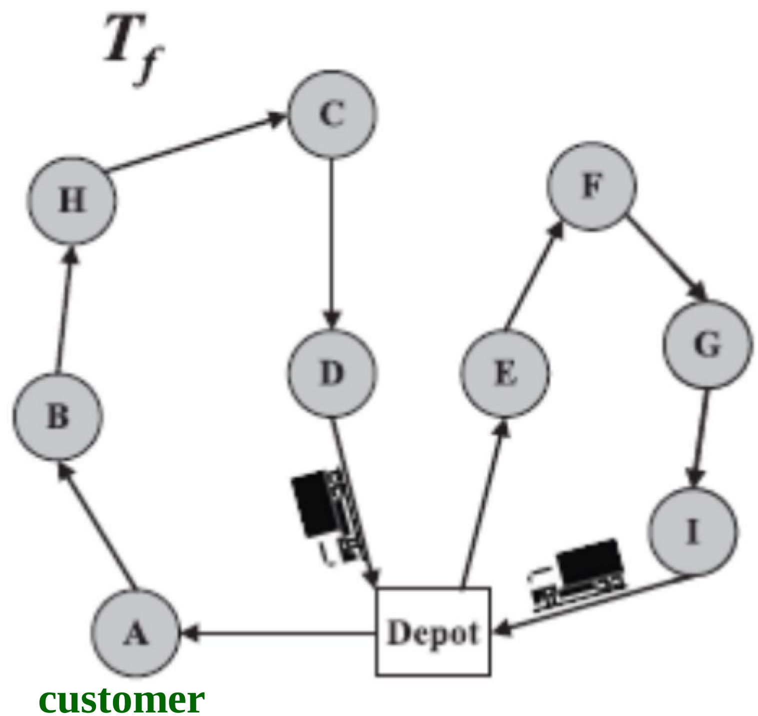
Estimation of Distribution Approach



"An Adaptive Estimation of Distribution Algorithm for Multi-Policy Insurance Investment Planning", ***IEEE Transactions on Evolutionary Computation***, Accepted in Nov. 2017

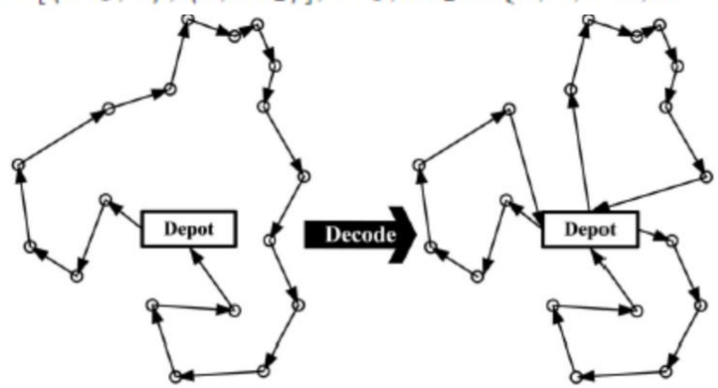
Application 2: Dynamic Vehicle Routing

- **Definition**
  - Designing routes for a fleet of vehicles to serve a set of customers
- **Objective:**
  - Minimize total travel distances subject to various constraints
- **A crucial issue in transportation and logistics systems**
- **NP-hard**



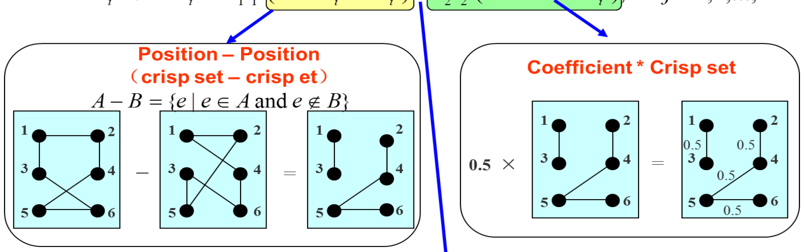
Probability-based EC for Vehicle Routing

$$X_i^d = [\langle nb_1, d\rangle, \langle d, nb_2\rangle], nb_1, nb_2 \in \{0, 1, \ldots, d-1, d+1, n\}$$



Set-based Representation
with a decoding scheme

$$V_i^j \leftarrow \omega V_i^j + c_1 r_1^j (PBest_i^j - X_i^j) + c_2 r_2^j (GBest^j - X_i^j), \quad j = 1,2,\ldots,n$$

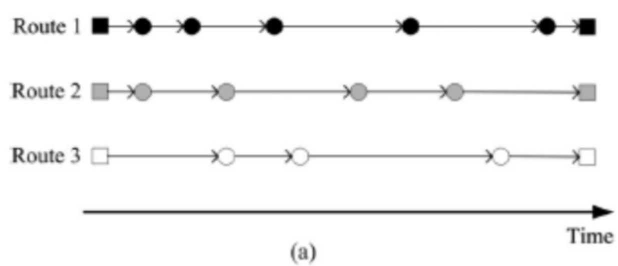


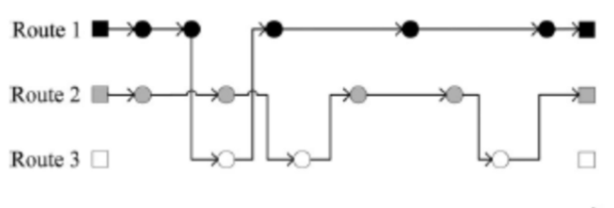
Union of two sets with possibilities

V_1^1={(1, 2)/0.6, (1, 4)/0.9} V_1^2={(1, 2)/0.9, (1, 3)/0.3}

$V_1^1 + V_1^2$= {(1, 2)/0.9, (1, 3)/0.3, (1, 4)/0.9}

Set-based comprehensive learning PSO



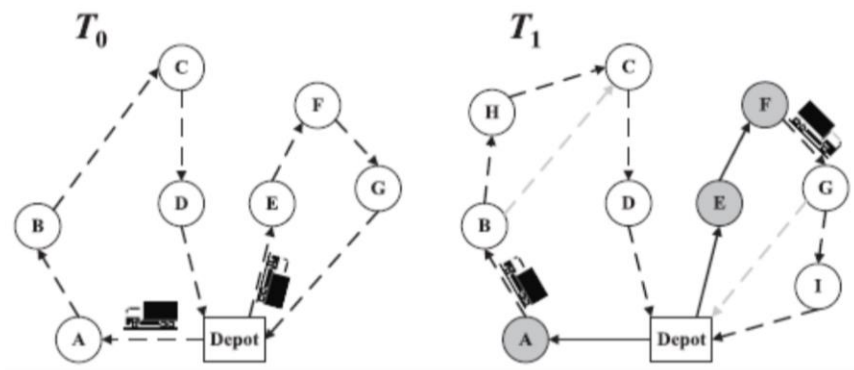


Local search

- ❑ Provide new best-known results on 60+ benchmark instances

Dynamic Logistics Dispatching System

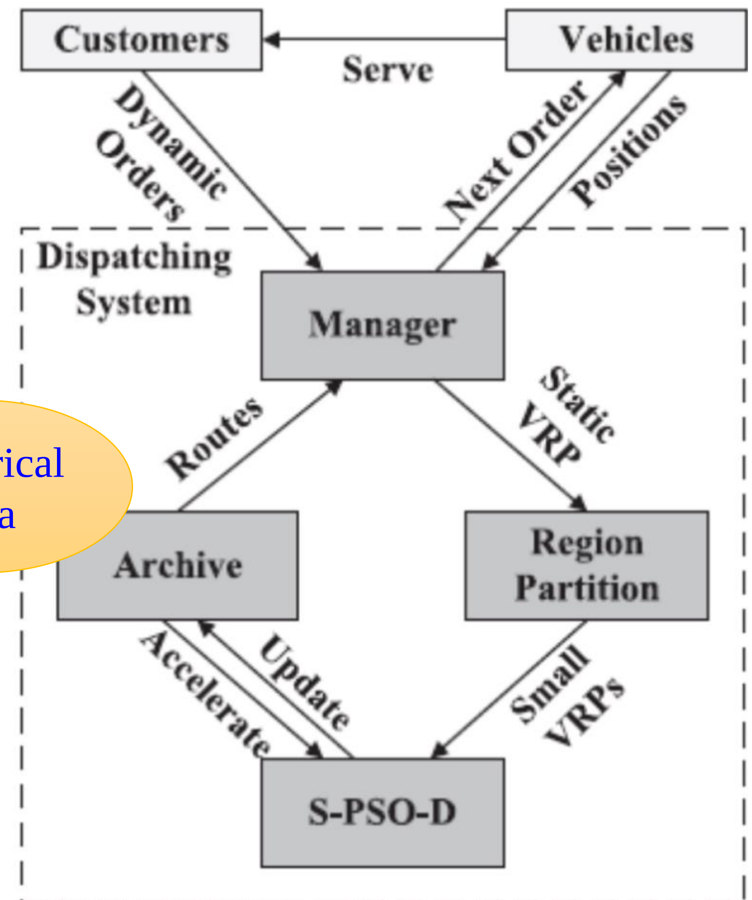
❑ Dynamic orders



❑ Dynamic optimization

- Archive strategy to store historical results and accelerate convergence
- Region partition to cut a large-scale problem into small pieces

Historical data

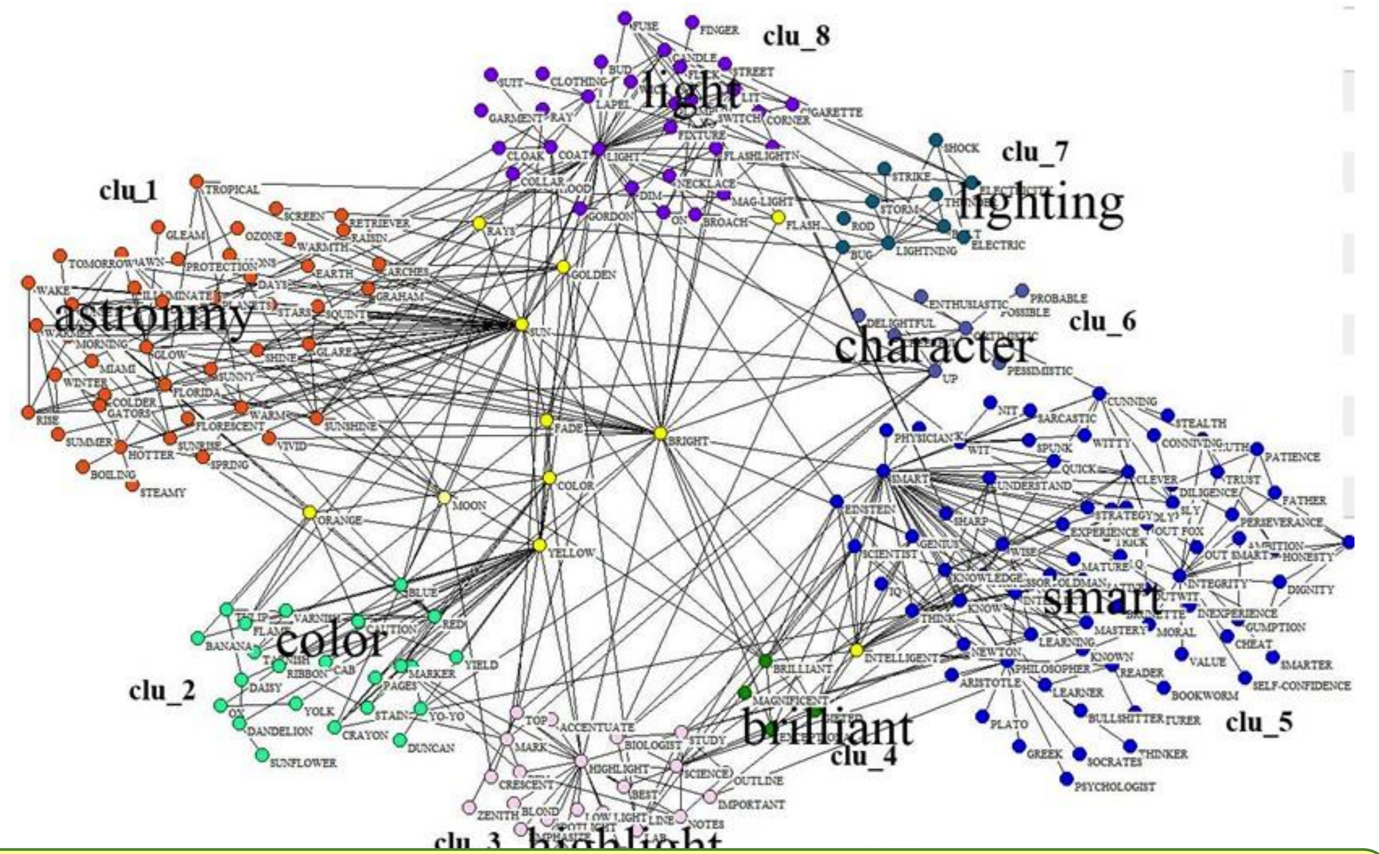


Architecture of the dispatching system

"A Dynamic Logistic Dispatching System With Set-Based Particle Swarm Optimization", ***IEEE TSMC-Systems***, in press, 2017

Application 3: Community Detection

❑ Detection of Overlapping Social Network Communities

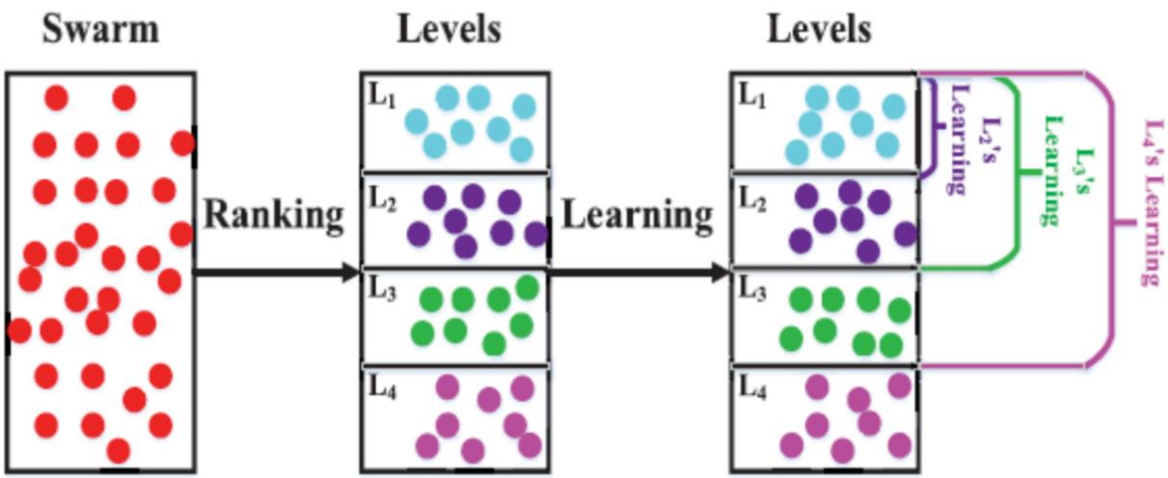


"A Maximal Clique Based Multiobjective Evolutionary Algorithm for Overlapping Community Detection ", ***IEEE Transactions on Evolutionary Computation***, 2017

Potential Future Work

- **Large-scale and high-dimensional problems**
  - **Cooperatively Coevolution** (Li and Yao, 2013 etc., …)
  - **Competitive Swarm Optimizer** (Cheng & Jin, 2015)

Level-based Learning / Segment-Based Predominant Learning Optimizer



Exploration:
- A lot of exemplars

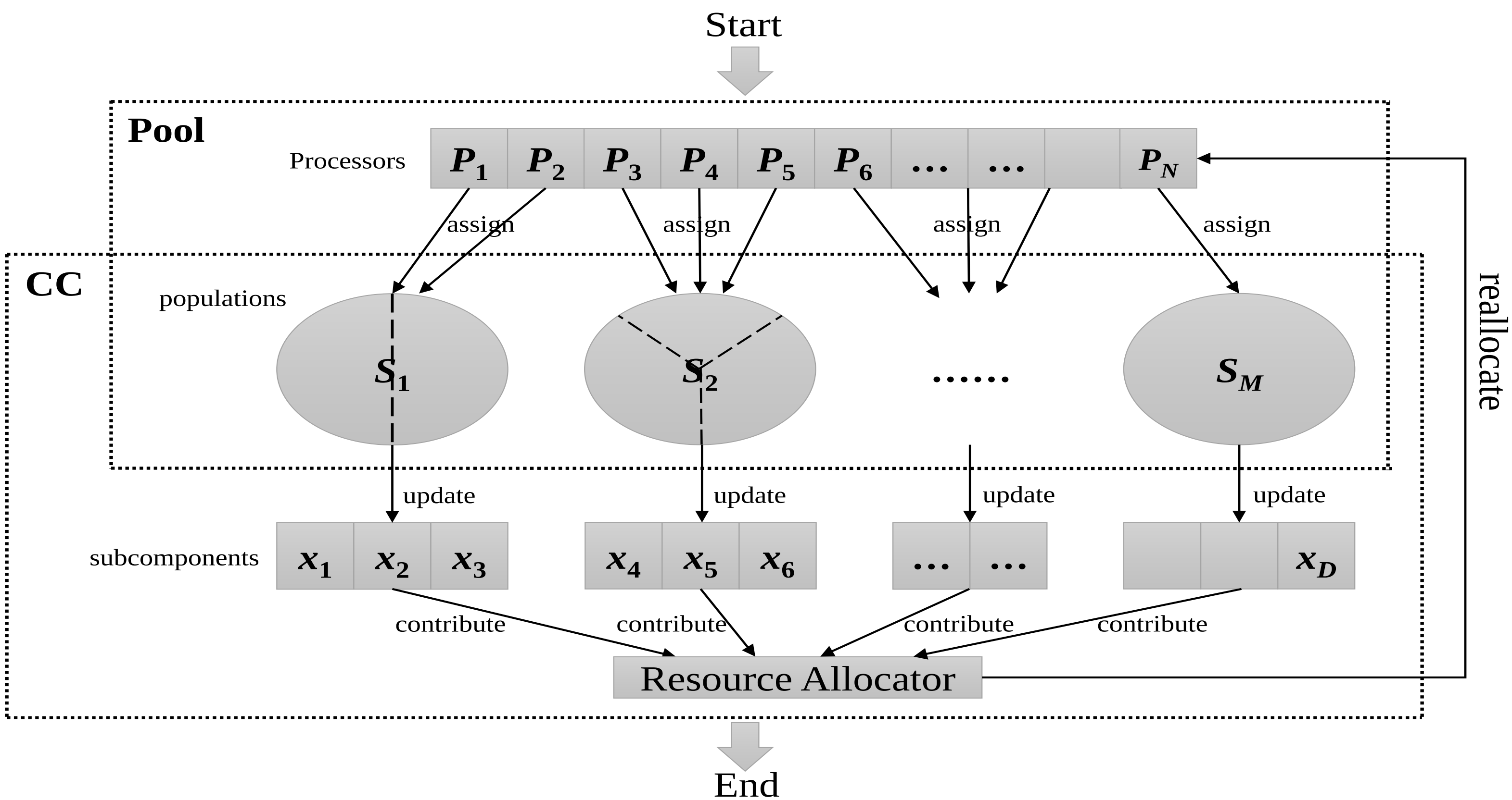
Exploitation:
- Predominant exemplars
- Short distances between learners and exemplars

"A Level-based Learning Swarm Optimizer for Large Scale Optimization", ***IEEE Transactions on Evolutionary Computation***, in press

"Segment-Based Predominant Learning Swarm Optimizer for Large-Scale Optimization," ***IEEE Transactions on Cybernetics***, vol. 47, no. 9, pp. 2896-2910, 2017

Potential Future Work

❑ Distributed Design & Implementation



"Distributed Cooperative Co-evolution with Adaptive Computing Resource Allocation for Large Scale Optimization", ***IEEE Transactions on Evolutionary Computation***, in press

Conclusions

- Probability distribution based EC algorithms are good at preserver search diversity
- Probability distribution is also helpful to extend applicable domain of EC algorithms
- Potential future studies
  - Large-scale optimization
    - Cooperatively coevolution
    - Parallel & distributed design & implementation
  - Optimization under uncertainty
  - Real world applications

Thanks for your attention!