Multi-objective Optimization with Estimation of Distribution Algorithms

Pedro Larrañaga

Computational Intelligence Group
Artificial Intelligence Department
Technical University of Madrid

EVOLVE 2012
Mexico City, August 7-9, 2012

Outline

1. Introduction
2. Estimation of Distribution Algorithms
3. Our Proposal
4. Experimental Results
5. Conclusions

The Problem

Multi-objective Optimization Problems (MOPs)

- Multiple objectives should be fulfilled simultaneously

$$min_{\mathbf{v}} \quad \mathbf{o}(\mathbf{v}) = (o_1(\mathbf{v}), \ldots, o_m(\mathbf{v}))$$

$$\text{subject to} \quad \begin{cases} \mathbf{v} \in \mathcal{D} \subseteq \mathbb{R}^r \\ \mathbf{o} \in \mathcal{Q} \subseteq \mathbb{R}^m \end{cases}$$

- A trade-off between objectives: Pareto dominance relation

Our Approach

Multi-objective estimation of distribution algorithms (MOEDAs)

- Multi-objective evolutionary algorithms (MOEAs) based on nature-inspired operators to evolve a population of candidate solutions
- Estimation of distribution algorithms (EDAs) generate new candidate solutions from a probabilistic graphical model (Bayesian network) learnt at each generation from a set of promising solutions
- Multi-objective estimation of distribution algorithms (MOEDAs): MOPs approaches based on EDAs

Our Approach

In this talk

- A new type of MOEDAs where the structure of the Bayesian network facilitates the approximation to the MOP structure
- Discover the relationships among:
  - Objectives (minimum set of objectives)
  - Variables
  - Objectives and variables (which variables have more importance in a concrete objective)
- Experimental results showing the scalability of the approach on the number of objectives, and its competitiveness with respect to state of the art

Outline

1. Introduction
2. Estimation of Distribution Algorithms
   - Example
   - Scheme
   - Bayesian Networks
   - Gaussian Bayesian Networks
   - Simulation
3. Our Proposal
4. Experimental Results
5. Conclusions

Outline

1 Introduction

2 Estimation of Distribution Algorithms
- Example
- Scheme
- Bayesian Networks
- Gaussian Bayesian Networks
- Simulation

3 Our Proposal

4 Experimental Results

5 Conclusions

EDAs. A Toy Example

$$max\ O(\mathbf{x}) = \sum_{i=1}^{6} x_i$$

with $x_i = 0, 1$

EDAs. A Toy Example

$$max\ O(\boldsymbol{x}) = \sum_{i=1}^{6} x_i$$

with $x_i = 0, 1$

	X_1	X_2	X_3	X_4	X_5	X_6	$O(\boldsymbol{x})$
1	1	0	1	0	1	0	3
2	0	1	0	0	1	0	2
3	0	0	0	1	0	0	1
4	1	1	1	0	0	1	4
5	0	0	0	0	0	1	1
6	1	1	0	0	1	1	4
7	0	1	1	1	1	1	5
8	0	0	0	1	0	0	1
9	1	1	0	1	0	0	3
10	1	0	1	0	0	0	2
11	1	0	0	1	1	1	4
12	1	1	0	0	0	1	3
13	1	0	1	0	0	0	2
14	0	0	0	0	1	1	2
15	0	1	1	1	1	1	5
16	0	0	0	1	0	0	1
17	1	1	1	1	1	0	5
18	0	1	0	1	1	0	3
19	1	0	1	1	1	1	5
20	1	0	1	1	0	0	3

EDAs. A Toy Example

$$max\ O(\boldsymbol{x}) = \sum_{i=1}^{6} x_i$$

with $x_i = 0, 1$

	X_1	X_2	X_3	X_4	X_5	X_6	$O(\boldsymbol{x})$
1	1	0	1	0	1	0	3
2	0	1	0	0	1	0	2
3	0	0	0	1	0	0	1
4	1	1	1	0	0	1	4
5	0	0	0	0	0	1	1
6	1	1	0	0	1	1	4
7	0	1	1	1	1	1	5
8	0	0	0	1	0	0	1
9	1	1	0	1	0	0	3
10	1	0	1	0	0	0	2
11	1	0	0	1	1	1	4
12	1	1	0	0	0	1	3
13	1	0	1	0	0	0	2
14	0	0	0	0	1	1	2
15	0	1	1	1	1	1	5
16	0	0	0	1	0	0	1
17	1	1	1	1	1	0	5
18	0	1	0	1	1	0	3
19	1	0	1	1	1	1	5
20	1	0	1	1	0	0	3

EDAs. A Toy Example

Learning the probability distribution from the selected individuals

	X_1	X_2	X_3	X_4	X_5	X_6
1	1	0	1	0	1	0
4	1	1	1	0	0	1
6	1	1	0	0	1	1
7	0	1	1	1	1	1
11	1	0	0	1	1	1
12	1	1	0	0	0	1
15	0	1	1	1	1	1
17	1	1	1	1	1	0
18	0	1	0	1	1	0
19	1	0	1	1	1	1

EDAs. A Toy Example

Learning the probability distribution from the selected individuals

	X_1	X_2	X_3	X_4	X_5	X_6
1	1	0	1	0	1	0
4	1	1	1	0	0	1
6	1	1	0	0	1	1
7	0	1	1	1	1	1
11	1	0	0	1	1	1
12	1	1	0	0	0	1
15	0	1	1	1	1	1
17	1	1	1	1	1	0
18	0	1	0	1	1	0
19	1	0	1	1	1	1

$$p(\boldsymbol{x}) = p(x_1, \ldots, x_6) = p(x_1)p(x_2)p(x_3)p(x_4)p(x_5)p(x_6)$$

EDAs. A Toy Example

Learning the probability distribution from the selected individuals

	X_1	X_2	X_3	X_4	X_5	X_6
1	1	0	1	0	1	0
4	1	1	1	0	0	1
6	1	1	0	0	1	1
7	0	1	1	1	1	1
11	1	0	0	1	1	1
12	1	1	0	0	0	1
15	0	1	1	1	1	1
17	1	1	1	1	1	0
18	0	1	0	1	1	0
19	1	0	1	1	1	1

$$p(\boldsymbol{x}) = p(x_1, \ldots, x_6) = p(x_1)p(x_2)p(x_3)p(x_4)p(x_5)p(x_6)$$

$$p(X_1 = 1) = \frac{7}{10}$$

EDAs. A Toy Example

Learning the probability distribution from the selected individuals

	X_1	X_2	X_3	X_4	X_5	X_6
1	1	0	1	0	1	0
4	1	1	1	0	0	1
6	1	1	0	0	1	1
7	0	1	1	1	1	1
11	1	0	0	1	1	1
12	1	1	0	0	0	1
15	0	1	1	1	1	1
17	1	1	1	1	1	0
18	0	1	0	1	1	0
19	1	0	1	1	1	1

$$p(\boldsymbol{x}) = p(x_1, \ldots, x_6) = p(x_1)p(x_2)p(x_3)p(x_4)p(x_5)p(x_6)$$

$$p(X_1 = 1) = \frac{7}{10} \qquad p(X_2 = 1) = \frac{7}{10} \qquad p(X_3 = 1) = \frac{6}{10}$$

$$p(X_4 = 1) = \frac{6}{10} \qquad p(X_5 = 1) = \frac{8}{10} \qquad p(X_6 = 1) = \frac{7}{10}$$

EDAs. A Toy Example

Learning the probability distribution from the selected individuals

	X_1	X_2	X_3	X_4	X_5	X_6
1	1	0	1	0	1	0
4	1	1	1	0	0	1
6	1	1	0	0	1	1
7	0	1	1	1	1	1
11	1	0	0	1	1	1
12	1	1	0	0	0	1
15	0	1	1	1	1	1
17	1	1	1	1	1	0
18	0	1	0	1	1	0
19	1	0	1	1	1	1

$$p(\boldsymbol{x}) = p(x_1, \ldots, x_6) = p(x_1)p(x_2)p(x_3)p(x_4)p(x_5)p(x_6)$$

$$p(X_1 = 1) = \frac{7}{10} \quad p(X_2 = 1) = \frac{7}{10} \quad p(X_3 = 1) = \frac{6}{10}$$

$$p(X_4 = 1) = \frac{6}{10} \quad p(X_5 = 1) = \frac{8}{10} \quad p(X_6 = 1) = \frac{7}{10}$$

EDAs. A Toy Example

Obtaining the new population by sampling from the probability distribution

$$p(X_1 = 1) = \frac{7}{10}; p(X_2 = 1) = \frac{7}{10}; p(X_3 = 1) = \frac{6}{10}$$

$$p(X_4 = 1) = \frac{6}{10}; p(X_5 = 1) = \frac{8}{10}; p(X_6 = 1) = \frac{7}{10}$$

$$p(\boldsymbol{x}) = p(x_1, \ldots, x_6) = p(x_1)p(x_2)p(x_3)p(x_4)p(x_5)p(x_6)$$

$$\begin{array}{ll}
0.23 & p(X_1 = 1) = \frac{7}{10} > 0.23 \longrightarrow 1 \\
0.65 & p(X_2 = 1) = \frac{7}{10} > 0.65 \longrightarrow 1 \\
0.89 & p(X_3 = 1) = \frac{6}{10} < 0.89 \longrightarrow 0 \\
0.12 & p(X_4 = 1) = \frac{6}{10} > 0.12 \longrightarrow 1 \\
0.48 & p(X_5 = 1) = \frac{8}{10} > 0.48 \longrightarrow 1 \\
0.54 & p(X_6 = 1) = \frac{7}{10} > 0.54 \longrightarrow 1
\end{array}$$

EDAs. A Toy Example

Obtaining the new population by sampling from the probability distribution

	X_1	X_2	X_3	X_4	X_5	X_6	$O(\boldsymbol{x})$
1	1	1	0	1	1	1	5
2	1	0	1	0	1	1	4
3	1	1	1	1	1	0	5
4	0	1	0	1	1	1	4
5	1	1	1	1	0	1	5
6	1	0	0	1	1	1	4
7	0	1	0	1	1	0	3
8	1	1	1	0	1	0	4
9	1	1	1	0	0	1	4
10	1	0	0	1	1	1	4
11	1	1	0	0	1	1	4
12	1	0	1	1	1	0	4
13	0	1	1	0	1	1	4
14	0	1	1	1	1	0	4
15	1	1	1	1	1	1	6
16	0	1	1	0	1	1	4
17	1	1	1	1	1	0	5
18	0	1	0	0	1	0	2
19	0	0	1	1	0	1	3
20	1	1	0	1	1	1	5

Outline

1 Introduction

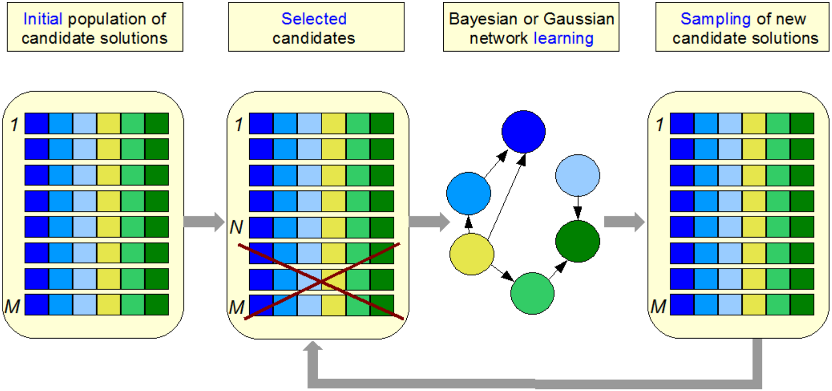
2 Estimation of Distribution Algorithms
- Example
- Scheme
- Bayesian Networks
- Gaussian Bayesian Networks
- Simulation

3 Our Proposal

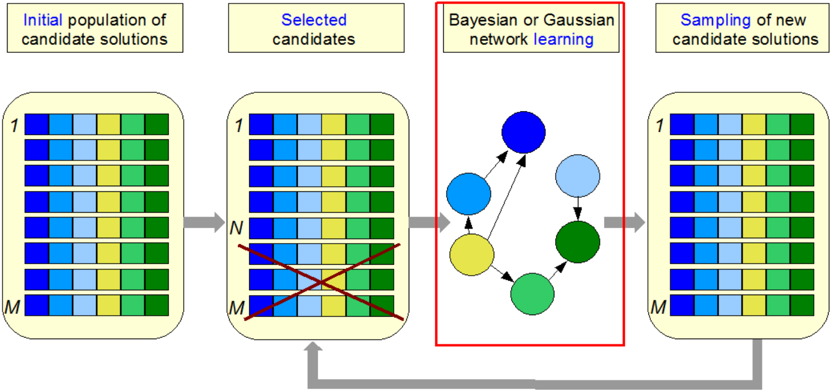
4 Experimental Results

5 Conclusions

Graphical Representation of EDAs



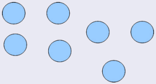
Graphical Representation of EDAs



Directed Probabilistic Graphical Models in EDAs

Univariate EDAs: Univariate Marginal Distribution Algorithm (UMDA). Mühlenbein and Paaß, 1996)

- Probabilistic model: $p_l(\boldsymbol{x}) = \prod_{i=1}^{n} p_l(x_i)$
- Structural learning: not necessary

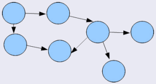


Bivariate EDAs: Mutual Information Maximization for Input Clustering (MIMIC). De Bonet et al., 1997)

- Probabilistic model:
  $p_l^{\pi}(\boldsymbol{x}) = p_l(x_{i_1} \mid x_{i_2}) p_l(x_{i_2} \mid x_{i_3}) \cdots p_l(x_{i_{n-1}} \mid x_{i_n}) p_l(x_{i_n})$
- Structural learning: best permutation (factorization closest to the empirical distribution in the sense of Kullback-Leibler divergence)

Multivariate EDAs: (Etxeberria and Larrañaga, 1999) (EBNA); (Pelikan et al., 1999) (BOA); (Harik et al., 1999) (EcGA); (Mühlenbein and Mahnig, 1999) (LFDA)

- Probabilistic model: $p_l(\boldsymbol{x}) = \prod_{i=1}^{n} p_l(x_i|\boldsymbol{pa}_i)$
- Structural learning: directed acyclic graph



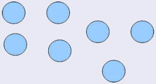
EDAs in continuous domains: Assuming Gaussianity

- Univariate: (Larrañaga et al., 2000) (UMDA$_c^G$)
- Bivariate: (Larrañaga et al., 2000) (MIMIC$_c^G$)
- Multivariate: (Larrañaga et al., 2000) ($EMNA_{global}^G$, $EMNA_{ee}^G$, $EGNA^G$)

Directed Probabilistic Graphical Models in EDAs

Univariate EDAs: Univariate Marginal Distribution Algorithm (UMDA). Mühlenbein and Paaß, 1996)

- Probabilistic model: $p_l(\boldsymbol{x}) = \prod_{i=1}^{n} p_l(x_i)$
- Structural learning: not necessary

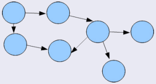


Bivariate EDAs: Mutual Information Maximization for Input Clustering (MIMIC). De Bonet et al., 1997)

- Probabilistic model: $p_l^{\pi}(\boldsymbol{x}) = p_l(x_{i_1} \mid x_{i_2}) p_l(x_{i_2} \mid x_{i_3}) \cdots p_l(x_{i_{n-1}} \mid x_{i_n}) p_l(x_{i_n})$
- Structural learning: best permutation (factorization closest to the empirical distribution in the sense of Kullback-Leibler divergence)

Multivariate EDAs: (Etxeberria and Larrañaga, 1999) (EBNA); (Pelikan et al., 1999) (BOA); (Harik et al., 1999) (EcGA); (Mühlenbein and Mahnig, 1999) (LFDA)

- Probabilistic model: $p_l(\boldsymbol{x}) = \prod_{i=1}^{n} p_l(x_i | \boldsymbol{pa}_i)$
- Structural learning: directed acyclic graph



EDAs in continuous domains: Assuming Gaussianity

- Univariate: (Larrañaga et al., 2000) (UMDA_c^G)
- Bivariate: (Larrañaga et al., 2000) (MIMIC_c^G)
- Multivariate: (Larrañaga et al., 2000) ($EMNA_{global}^G$, $EMNA_{ee}^G$, $EGNA^G$)

Directed Probabilistic Graphical Models

Qualitative + quantitative parts

A directed probabilistic graphical model, $M = (S, \boldsymbol{\theta}^S)$, (Pearl, 1988; Koller and Friedman, 2009) for $\boldsymbol{X} = (X_1, \ldots, X_n)$ consists of two components:

- A structure S for $\boldsymbol{X}$ is a directed acyclic graph (DAG) that represents a set of conditional (in)dependences between triplets of variables
- A set of local probability distributions $\boldsymbol{\theta}^S = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n)$

Conditional (in)dependences between triplets of variables

Given three disjoints sets of variables, $\boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{W}$, we say that $\boldsymbol{Y}$ is conditionally independent of $\boldsymbol{Z}$ given $\boldsymbol{W}$ if, for any $\boldsymbol{y}, \boldsymbol{z}, \boldsymbol{w}$, we have $p(\boldsymbol{y} \mid \boldsymbol{z}, \boldsymbol{w}) = p(\boldsymbol{y} \mid \boldsymbol{w})$

Factorization of the joint probability distribution

$p(\boldsymbol{x} \mid \boldsymbol{\theta}^S) = \prod_{i=1}^{n} p(x_i \mid \boldsymbol{pa}_i^S, \boldsymbol{\theta}_i)$

Outline

1 Introduction

2 Estimation of Distribution Algorithms
- Example
- Scheme
- Bayesian Networks
- Gaussian Bayesian Networks
- Simulation

3 Our Proposal

4 Experimental Results

5 Conclusions

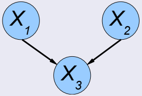
Bayesian Networks

Definition

- X_i discrete variable with $|\Omega_i| = r_i$ for all $i = 1, \ldots, n$
- Local distributions: $p(x_i^k \mid \boldsymbol{pa}_i^{j,S}, \boldsymbol{\theta}_i) = \theta_{x_i^k|\boldsymbol{pa}_i^j} \equiv \theta_{ijk}$
- $\boldsymbol{pa}_i^{1,S}, \ldots, \boldsymbol{pa}_i^{q_i,S}$ denotes the values of $\boldsymbol{Pa}_i^S$ with $q_i = \prod_{X_g \in \boldsymbol{Pa}_i} r_g$

Example

Model structure



Model parameters and local probability distributions

$$
\begin{array}{lll}
\boldsymbol{\theta}_1 = & (\theta_{1\text{-}1}, \theta_{1\text{-}2}) & p(x_1^1 \mid \boldsymbol{\theta}_1), p(x_1^2 \mid \boldsymbol{\theta}_1) \\
\boldsymbol{\theta}_2 = & (\theta_{2\text{-}1}, \theta_{2\text{-}2}) & p(x_2^1 \mid \boldsymbol{\theta}_2), p(x_2^2 \mid \boldsymbol{\theta}_2) \\
\boldsymbol{\theta}_3 = & (\theta_{311}, \theta_{312}) & p(x_3^1 \mid x_1^1, x_2^1, \boldsymbol{\theta}_3), p(x_3^2 \mid x_1^1, x_2^1, \boldsymbol{\theta}_3) \\
 & (\theta_{321}, \theta_{322}) & p(x_3^1 \mid x_1^1, x_2^2, \boldsymbol{\theta}_3), p(x_3^2 \mid x_1^1, x_2^2, \boldsymbol{\theta}_3) \\
 & (\theta_{331}, \theta_{332}) & p(x_3^1 \mid x_1^2, x_2^1, \boldsymbol{\theta}_3), p(x_3^2 \mid x_1^2, x_2^1, \boldsymbol{\theta}_3) \\
 & (\theta_{341}, \theta_{342}) & p(x_3^1 \mid x_1^2, x_2^2, \boldsymbol{\theta}_3), p(x_3^2 \mid x_1^2, x_2^2, \boldsymbol{\theta}_3)
\end{array}
$$

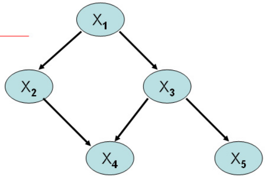
Factorization of the joint probability distribution

$p(\boldsymbol{x} \mid \boldsymbol{\theta_S}) = p(x_1 \mid \boldsymbol{\theta}_1)p(x_2 \mid \boldsymbol{\theta}_2)p(x_3 \mid x_1, x_2, \boldsymbol{\theta}_3)$

Learning Bayesian Networks

Learning structure and parameters

Structure learning



X_1	X_2	X_3	X_4	X_5
1	1	1	1	0
0	1	1	0	1
...	...	...	...	...
...	...	...	...	...
1	1	0	1	1
0	1	1	1	0

$p(X_1=0)=0.60$

$p(X_2=0|X_1=0)=0.10$ $p(X_2=0|X_1=0)=0.70$

$p(X_3=0|X_1=0)=0.40$ $p(X_3=0|X_1=0)=0.90$

$p(X_4=0|X_2=0,X_3=0)=0.60$ $p(X_3=0|X_2=0,X_3=1)=0.20$

$p(X_4=0|X_2=0,X_3=0)=0.60$ $p(X_3=0|X_2=0,X_3=1)=0.20$

$p(X_5=0|X_3=0)=0.60$ $p(X_5=0|X_3=0)=0.60$

Parameter learning

Learning Bayesian Networks

Learning parameters

Given a data set of cases $D = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}\}$ drawn at random from a joint probability distribution $p(x_1, \ldots, x_n)$

- Maximum likelihood estimation: $\widehat{\theta}_{ijk} = p(X_i = x_i^k | \mathbf{Pa}_i = \mathbf{pa}_i^j) = \frac{N_{ijk}}{N_{ij}}$
- Bayesian estimation:
  - It is assumed a prior knowledge expressed by means of a prior joint distribution over the parameters:
    $p(\theta_{ij1}, \theta_{ij2}, \ldots, \theta_{ijr_i}) \rightsquigarrow Dir(\theta_{ij1}, \ldots, \theta_{ijr_i}; \alpha_1, \ldots, \alpha_{r_i}) = \frac{\Gamma(\sum_{w=1}^{r} \alpha_w)}{\prod_{w=1}^{r_i} \Gamma(\alpha_w)} \theta_{ij1}^{\alpha_1 - 1} \ldots \theta_{ijr_i}^{\alpha_{r_i} - 1}$
  - For a multinomial distribution, if the prior is $Dir(\theta_{ij1}, \ldots, \theta_{ijr_i}; \alpha_1, \ldots, \alpha_{r_i})$, then the posterior is $Dir(\theta_{ij1}, \ldots, \theta_{ijr_i}; \alpha_1 + N_{ij1}, \ldots, \alpha_r + N_{ijr_i})$
  - $\widehat{\theta}_{ijk} = p(X_i = x_i^k | \mathbf{Pa}_i = \mathbf{pa}_i^j) = \frac{N_{ijk} + \alpha_k}{N_{ij} + \sum_{w=1}^{r_i} \alpha_w}$, where $\sum_{w=1}^{r_i} \alpha_w$ is called the equivalent sample size (the virtually observed sample)

Learning Bayesian Networks

Learning structures

Finding the best network according to some criterion even with the constraint that each node has no more than K parents is NP-hard (Chickering et al., 1994)

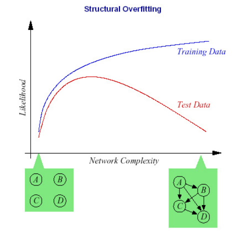
- Based on detecting conditional independencies
  - First: carry out a study of the dependence and independence relationships between the variables by means of statistical tests
  - Second: try to find the structure (or structures) that represents the most (or all) of these relationships
- Based on score + search
  - They try to find the structure that best “fits” the data
  - They need:
    - A score (metric or evaluation function) in order to measure the fitness of each candidate structure
    - A search method (heuristic) to explore in an intelligent manner the space of possible solutions
    - Several types of spaces can be considered

Learning Bayesian Networks

Learning structures (score + search)

Score: Penalized log-likelihood

- Log-likelihood of the data: $\log p(D|S,\widehat{\theta}) = \sum_{i=1}^{n}\sum_{j=1}^{q_i}\sum_{k=1}^{r_i} N_{ijk}\log\frac{N_{ijk}}{N_{ij}}$



- Penalizing the complexity: $\sum_{i=1}^{n}\sum_{j=1}^{q_i}\sum_{k=1}^{r_i} N_{ijk}\log\frac{N_{ijk}}{N_{ij}} - dim(S)pen(N)$
  - $dim(S) = \sum_{i=1}^{n} q_i(r_i - 1)$ model dimension
  - $pen(N)$ non negative penalization function
    - $pen(N) = 1$: Akaike's information criterion (AIC)
    - $pen(N) = \frac{1}{2}\log N$: Bayesian information criterion (BIC) or the minimum description length (MDL) criterion

Learning Bayesian Networks

Learning structures (score + search)

Score: Bayesian scores

$\hat{S} = \arg\max_S p(S|D) \equiv \arg\max_S p(D|S)p(S)$ where $p(D|S)$ denotes the marginal likelihood and $p(S)$ the prior distribution over structures. If $p(S)$ is uniform, $\hat{S} = \arg\max_S p(D|S)$

- K2 score: Assuming that $p(\boldsymbol{\theta}|S)$ is uniform, it is possible to obtain a closed formula for $p(D|S)$ (Cooper and Herskovits, 1992):

$$p(D|S) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}!$$

- BDe score: Assuming that $p(\boldsymbol{\theta}|S)$ follows a Dirichlet distribution, it is possible to obtain a closed formula for $p(D|S)$ (Heckerman et al., 1995):

$$p(D|S) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}$$

  - This score is called Bayesian Dirichlet equivalence metric because it verifies the score equivalence property (two DAGs representing the same set of conditional independencies score the same)

Learning Bayesian Networks

Learning structures (score + search)

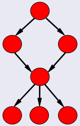
Search: Space of DAGs

- Cardinality of the search space (Robinson, 1977):
  $\mathcal{S}(n) = \sum_{i=1}^{n}(-1)^{i+1}\binom{n}{i}2^{i(n-i)}\mathcal{S}(n-i)$; $\mathcal{S}(0) = 1$; $\mathcal{S}(1) = 1$
- Search algorithms:
  - K2 algorithm (Cooper and Herskovits, 1992):
    - A total ordering between the nodes and an upper bound is set on the number of parents for any node are assumed
    - At each step K2 incrementally adds the parent whose addition provides the best value for $g(X_i, \boldsymbol{Pa}_i) = \prod_{j=1}^{q_i} \frac{(r_i-1)!}{(N_{ij}+r_i-1)!} \prod_{k=1}^{r_i} N_{ijk}!$
    - K2 stops when adding a single parent to any node cannot increase $g(X_i, \boldsymbol{Pa}_i)$
  - B algorithm (Buntine, 1991): insert, delete and invert an arc
  - Tabu search (Bouckaert, 1995)
  - Simulated annealing (Heckerman et al., 1995)

EDAs based on Bayesian Networks

EBNA, BOA, LFDA

- EBNA (Estimation of Bayesian Networks Algorithm) (Etxeberria and Larrañaga, 1999). *II Symposium on Artificial Intelligence*)
  - Detecting conditional independencies: EBNA_{PC}
  - Score: penalized likelihood (EBNA_{BIC} and EBNA_{K2})
  - Search: greedy search starting from the previous generation
- BOA (Bayesian Optimization Algorithm) (Pelikan et al., 1999). *GECCO*)
  - Score: marginal likelihood
  - Search: greedy search starting from scratch at each generation
- LFDA (Learning Factorized Distribution Algorithm) (Mühlenbein and Mahnig, 1999). *Evolutionary Computation*)
  - Score: BIC
  - Search: greedy search starting from scratch at each generation



Outline

1 Introduction

2 Estimation of Distribution Algorithms
- Example
- Scheme
- Bayesian Networks
- Gaussian Bayesian Networks
- Simulation

3 Our Proposal

4 Experimental Results

5 Conclusions

EDAs based on Multivariate Normal Densities

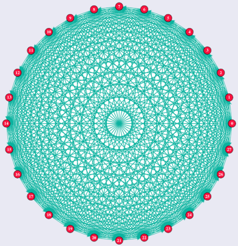
Multivariate normal density

- $f(\boldsymbol{x}) = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^t\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right]$
  - $\boldsymbol{x}$, n dimensional column vector
  - $\boldsymbol{\mu}$, n dimensional mean vector
  - $\boldsymbol{\Sigma}$, $n \times n$ variance-covariance matrix

Estimation of Multivariate Normal Algorithm (EMNA$_{global}$)

Larrañaga et al. (2000). *GECCO*

- Structure of EMNA$_{global}$ in all generations



EDAs based on Sparse Multivariate Normal Densities

Estimation of Multivariate Normal Algorithm by Edge Exclusion (EMNA$_{ee}$)

Larrañaga et al. (2000). *GECCO*

- Based on detecting independencies between pairs of variables
- The learning is carried out by means of $\binom{n}{2}$ tests for arc exclusion
  - X_i and X_j are independent iff the following null hypothesis is accepted (Smith and Whittaker, 1998)

$$\begin{cases} H_0 : w_{ij} = 0 & \text{null hypothesis} \\ H_A : w_{ij} \neq 0 & \text{alternative hypothesis} \end{cases}$$

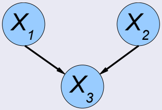
    with w_{ij} elements of the precision matrix $W = \Sigma^{-1}$
  - Likelihood ratio test:

$$T_{lik} = -n \log(1 - r^2_{ij|rest}) \text{ with } r_{ij|rest} = -\hat{w}_{ij}(\hat{w}_{ii}\hat{w}_{jj})^{-1/2}$$

Gaussian Bayesian Networks

Gaussian Bayesian networks

Model structure



Model parameters and local probability density functions

$$\begin{array}{ll} \boldsymbol{\theta}_1 = (m_1, \text{-}, v_1) & p(x_1 \mid \boldsymbol{\theta}_1) \rightsquigarrow \mathcal{N}(x_1; m_1, v_1) \\ \boldsymbol{\theta}_2 = (m_2, \text{-}, v_2) & p(x_2 \mid \boldsymbol{\theta}_2) \rightsquigarrow \mathcal{N}(x_2; m_2, v_2) \\ \boldsymbol{\theta}_3 = (m_3, \boldsymbol{b}_3, v_3) & p(x_3 \mid x_1, x_2, \boldsymbol{\theta}_3) \rightsquigarrow \mathcal{N}(x_3; m_3 + b_{13}(x_1 - m_1) + b_{23}(x_2 - m_2), v_3) \\ \boldsymbol{b}_3 = (b_{13}, b_{23})^t & \end{array}$$

Factorization of the joint density

$p(\boldsymbol{x} \mid \boldsymbol{\theta}^S) = p(x_1 \mid \boldsymbol{\theta}_1)p(x_2 \mid \boldsymbol{\theta}_2)p(x_3 \mid x_1, x_2, \boldsymbol{\theta}_3)$

EDAs based on Gaussian Bayesian Networks

Gaussian Bayesian networks

- The local density functions follow a linear regression model:

$$p(x_i \mid \boldsymbol{pa}_i^S, \boldsymbol{\theta}_i) \equiv \mathcal{N}(x_i; m_i + \sum_{x_j \in \boldsymbol{pa}_i} b_{ji}(x_j - m_j), v_i)$$

  - b_{ji} strength of the relationship between X_j and X_i ($b_{ji} = 0$ iff there is not an arc from X_j to X_i)
  - v_i variance of X_i conditioned to $\boldsymbol{Pa}_i$
  - $\boldsymbol{\theta}_i = (m_i, \boldsymbol{b}_i, v_i)$ local parameters, $\boldsymbol{b}_i = (b_{1i}, \ldots, b_{i-1i})^t$

Estimation of Gaussian Network Algorithm (EGNA$_{BIC}$)

Larrañaga et al. (2000). *GECCO*

- Score: penalized likelihood (BIC)
- Search: greedy
  - First generation: a disconnected graph
  - The rest of generations: start with the model obtained in the previous one

Outline

1 Introduction

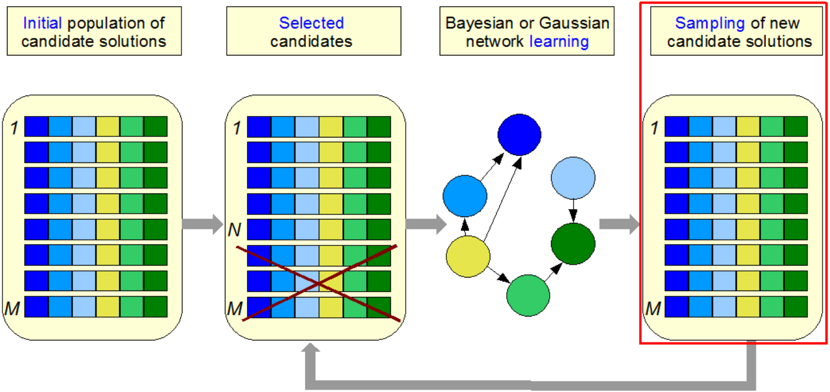
2 Estimation of Distribution Algorithms
- Example
- Scheme
- Bayesian Networks
- Gaussian Bayesian Networks
- Simulation

3 Our Proposal

4 Experimental Results

5 Conclusions

Graphical Representation of EDAs



EDAs

Obtaining the new population by sampling with PLS (Henrion, 1988)

Given an ancestral ordering, π, of the nodes (variables and objectives) in the directed probabilistic graphical model (Bayesian network or Gaussian Bayesian network):

for $j = 1, 2, \ldots, M$

for $i = 1, 2, \ldots, n$

$x_{\pi(i)} \leftarrow$ generate a value from $p(x_{\pi(i)}|\boldsymbol{pa}_{\pi(i)})$

Main scheme of the EDA approach

1. $D_0 \leftarrow$ Generate M individuals randomly
2. $l = 1$
3. **do** $\{$
4. $D_{l-1}^{Se} \leftarrow$ Select $N \leq M$ individuals from D_{l-1} according to a selection method
5. $p_l(\boldsymbol{x}) = p(\boldsymbol{x}|D_{l-1}^{Se}) \leftarrow$ Estimate the joint probability distribution of the selected individuals
6. $D_l \leftarrow$ Sample M individuals (the new population) from $p_l(\boldsymbol{x})$
7. $\}$ **until** A stopping criterion is met

Outline

1. Introduction
2. Estimation of Distribution Algorithms
3. Our Proposal
4. Experimental Results
5. Conclusions

MOEDAs in the literature

Previous MOEDAs

1. Thierens and Bosman (2001) in GECCO: multi-objective mixture-base iterate density estimation evolutionary algorithm (MIDEA)
2. Laumanns and Ocenasek (2002) in PPSN: Bayesian optimization algorithm (BMOA)
3. Costa and Minisci (2003) in EMO: Parzen based estimation of distribution algorithm (MOPED)
4. Li el at. (2004) in ECECCO: hybrid (UMDA + local search) (MOHEDA)
5. Okabe et al. (2004) in CEC: Voronoi-based estimation of distribution algorithm (VEDA)
6. Bosman and Thierens (2005) in IJAR journal: multi-objective mixture-base iterate density estimation evolutionary algorithm (MIDEA)
7. Sastry et al. (2005) in CEC: multi-objective extended compact genetic algorithm (meCGA)

MOEDAs in the literature

Previous MOEDAs

8. Pelikan et al. (2006) chapter in a book: multiobjective hierarchical BOA (mohBOA)
9. Zhong and Li (2007) in CIS: decision trees based multi-objective estimation of distribution algorithm (DT-MEDA)
10. Zhang et al. (2008) in IEEE TEC journal: regularity model-based multiobjective estimation of distribution algorithms (RM-MEDA)
11. Zhang et al. (2009) in IEEE TEC journal: model-based multiobjective evolutionary algorithm (MMEA)
12. Marti et al. (2009) in GECCO: multi-objective neural estimation of distribution algorithm (MONEDA)
13. Gao et al. (2010) in ICMTMA: hybrid (UMDA + PSO)
14. Shim et al. (2012) in EC journal: PSO + likelihood correction + restricted Boltzmann machines in estimation of distribution algorithms (PLREDA)

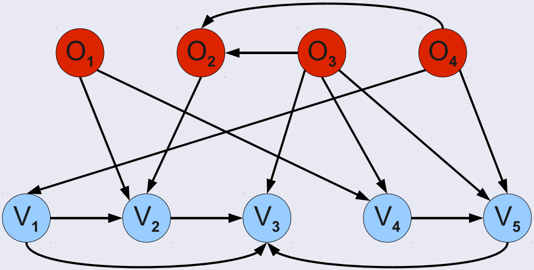
A New MOEDA

Main characteristics

- In standard EDAs the nodes in the probabilistic graphical model structure represent the variables. No node is used for the objective to be optimized
- We propose to represent both, variables and objectives, as nodes in the probabilistic graphical model structure
- The MOEDA, in its evolution, should capture the relationships among objectives, among variables, and also among objectives and variables
- The structure of the probabilistic graphical model structure is a two layer graph
  - First layer: objective nodes
  - Second layer: variables nodes

A New MOEDA

Two Layer Probabilistic Graphical Model

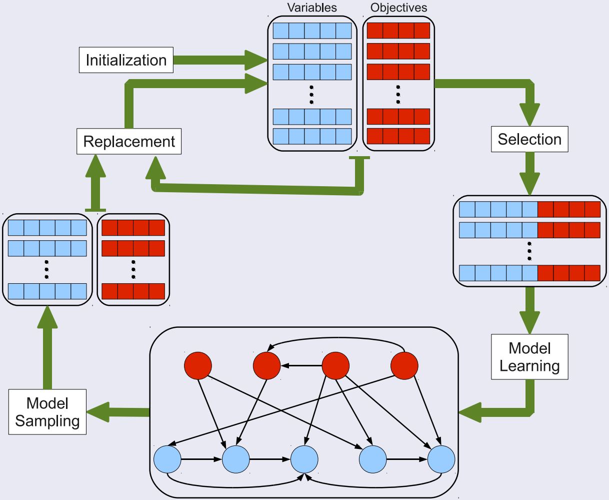


$$p(v_1,\ldots,v_r,o_1,\ldots,o_m)=\prod_{i=1}^{r}p(v_i|pa(V_i))\cdot\prod_{j=1}^{m}p(o_j|pa(O_j)),$$

where $Pa(V_i)\subseteq \mathbf{V}\cup\mathbf{O}\setminus\{V_i\}$ and $Pa(O_j)\subseteq\mathbf{O}\setminus\{O_j\}$

A New MOEDA

General Scheme



A New MOEDA

Instantiation: Multidimensional Bayesian Network based EDA (MBN-EDA)

- Continuous variables and objectives: Gaussian Bayesian networks
- Learning of the Gaussian Bayesian network by a greedy local search with the penalized likelihood (BIC) as score
- Four ranking methods $G : \mathcal{Q} \subseteq \mathbb{R}^m \mapsto \mathcal{T} \subseteq \mathbb{R}$
  1. Weighted sum: $G_{\mathrm{WS}}(\boldsymbol{o}) = \sum_{i=1}^{m} w_i o_i$
  2. Profit of gain: $G_{\mathrm{PG}}(\boldsymbol{o}) = max_{\boldsymbol{r} \in F_t, \boldsymbol{r} \neq \boldsymbol{o}} \mathrm{gain}(\boldsymbol{o}, \boldsymbol{r}) - max_{\boldsymbol{r} \in F_t, \boldsymbol{r} \neq \boldsymbol{o}} \mathrm{gain}(\boldsymbol{r}, \boldsymbol{o})$
     with $\mathrm{gain}(\boldsymbol{q}, \boldsymbol{r}) = \sum_{i=1}^{m} max\{0, r_i - q_i\}$
  3. Global detriment: $G_{\mathrm{GD}}(\boldsymbol{o}) = \sum_{\forall \boldsymbol{r} \in F_t, \boldsymbol{r} \neq \boldsymbol{o}} \mathrm{gain}(\boldsymbol{r}, \boldsymbol{o})$
  4. Distance to best: $G_{\mathrm{DB}}(\boldsymbol{o}) = \mathrm{d}(\boldsymbol{b}, \mathbf{0})$
     where $\boldsymbol{b} = (b_1, \ldots, b_m)$ denotes the best objective values.
     If $\boldsymbol{b}$ is known: $b_i = min_{\boldsymbol{o} \in F_t}\{o_i\}$

Outline

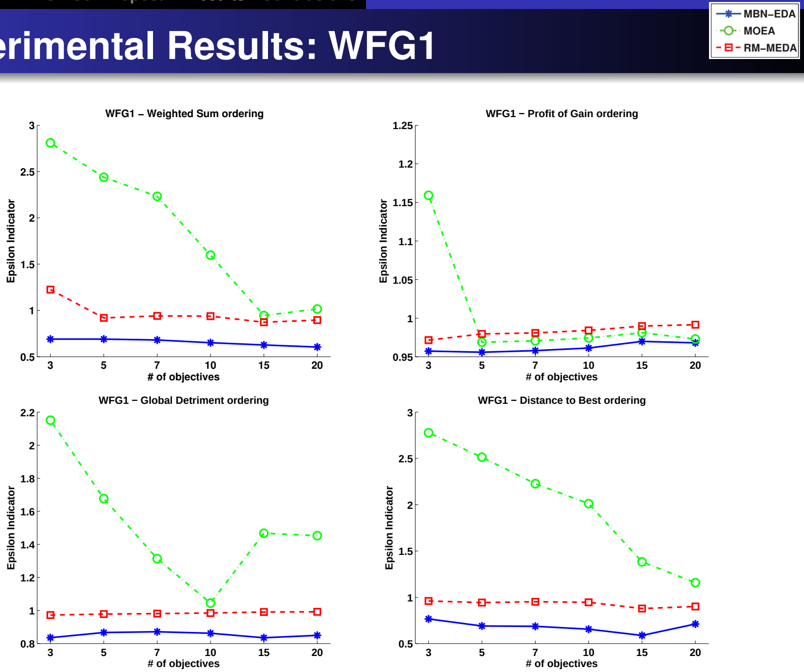
1. Introduction
2. Estimation of Distribution Algorithms
3. Our Proposal
4. Experimental Results
5. Conclusions

Experimental Results

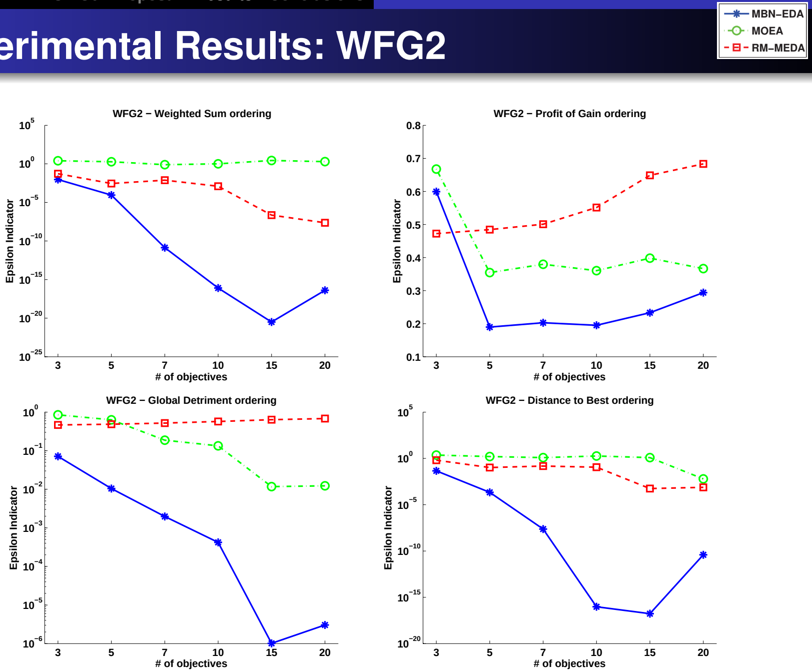
Characteristics of the Empirical Comparison

- Walking Fish Group (WFG) problems: WFG1, WFG2, WFG3, WFG4, WFG5, WFG6, WFG7, WFG8, WFG9
  - Number of objectives: $m \in \{3, 5, 7, 10, 15, 20\}$
  - Number of variables: $r = 16$
- Population size: $M \in \{50, 100, 150, 200, 250, 300\}$ (depending on m)
- Selection rate: 50 %
- Ranking methods: a) Weighted sum; b) Profit of gain; c) Global detriment; d) Distance to best
- The additive epsilon indicator value to measure the quality of the Pareto set approximations is averaged over 20 runs
- Algorithms to be compared:
  - MBN-EDA: our approach
  - MOEA: simulated binary crossover (Deb and Agrawal, 1995) and polynomial mutation (Deb and Goyal, 1996)
  - RM-MEDA: regularity-model based multi-objective EDA (Zhang et al., 2008)
- Matlab toolbox for EDAs (MatEDA) (Santana et al., 2010)

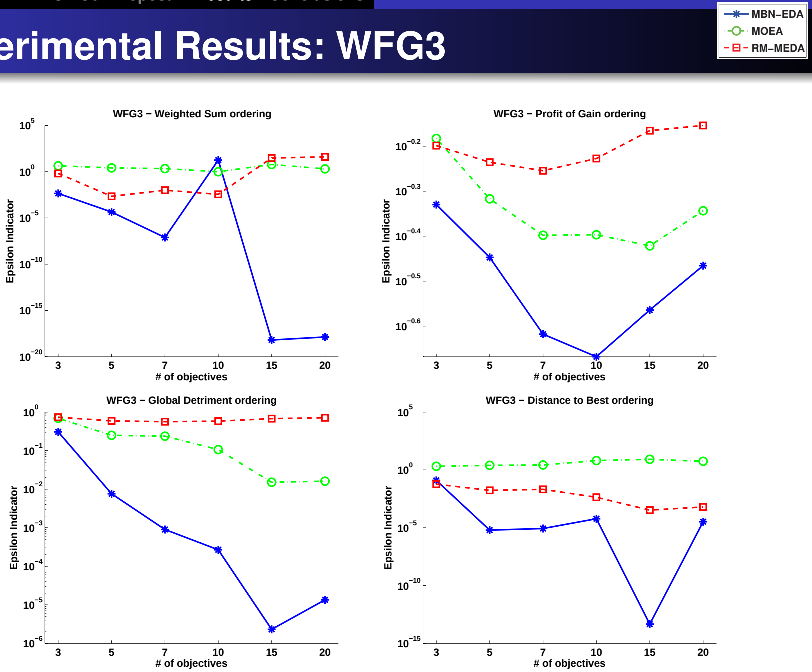
Experimental Results: WFG1



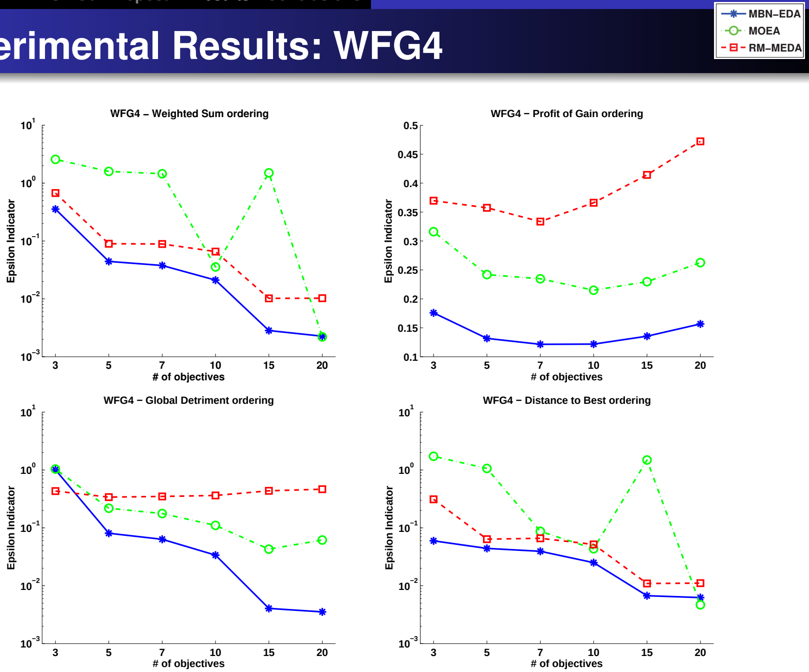
Experimental Results: WFG2



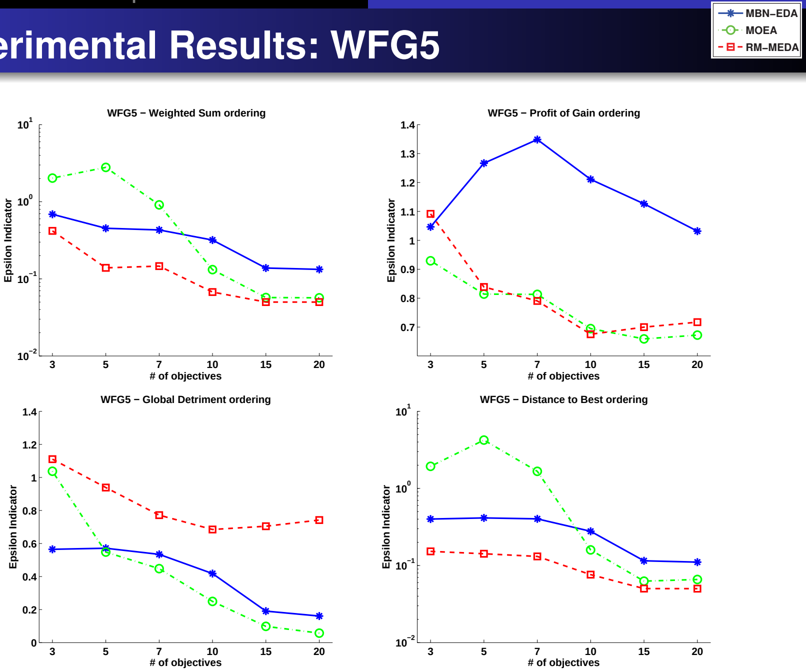
Experimental Results: WFG3



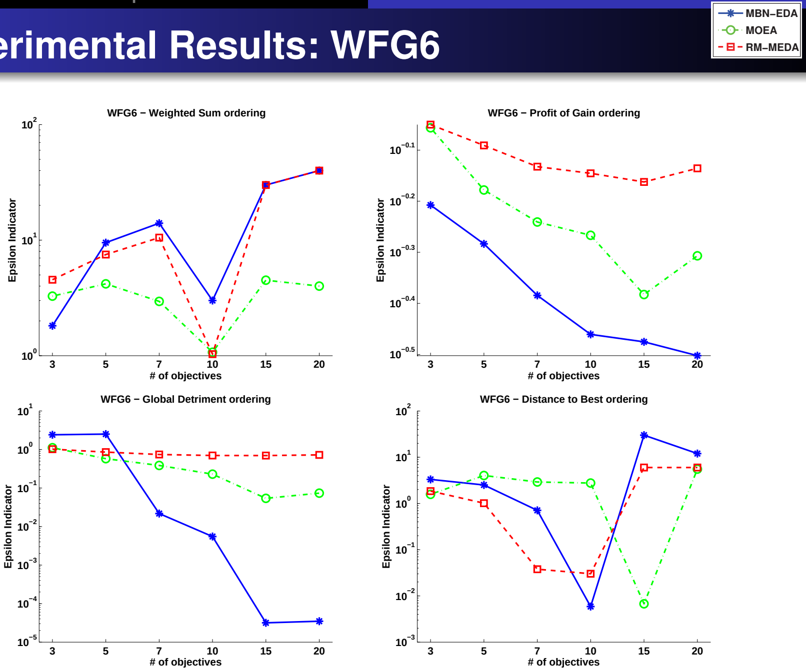
Experimental Results: WFG4



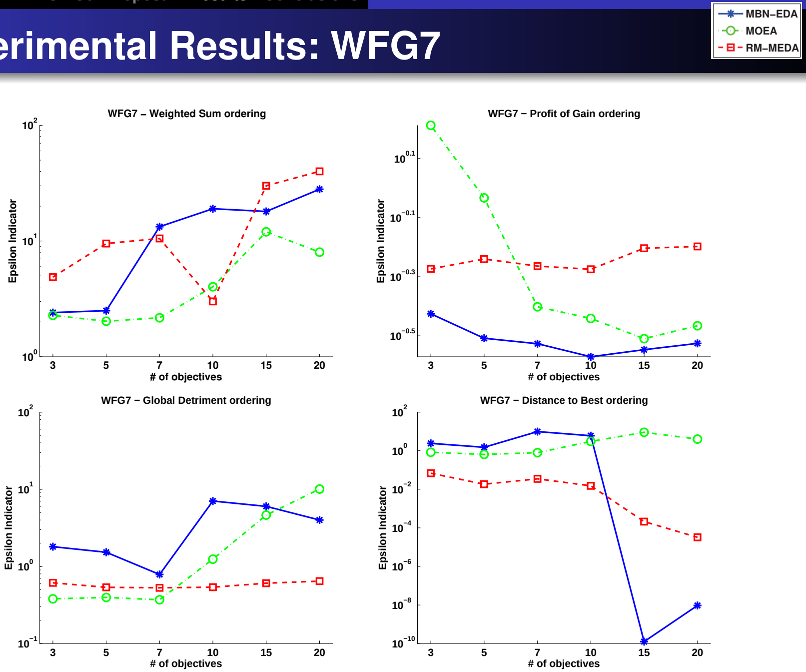
Experimental Results: WFG5



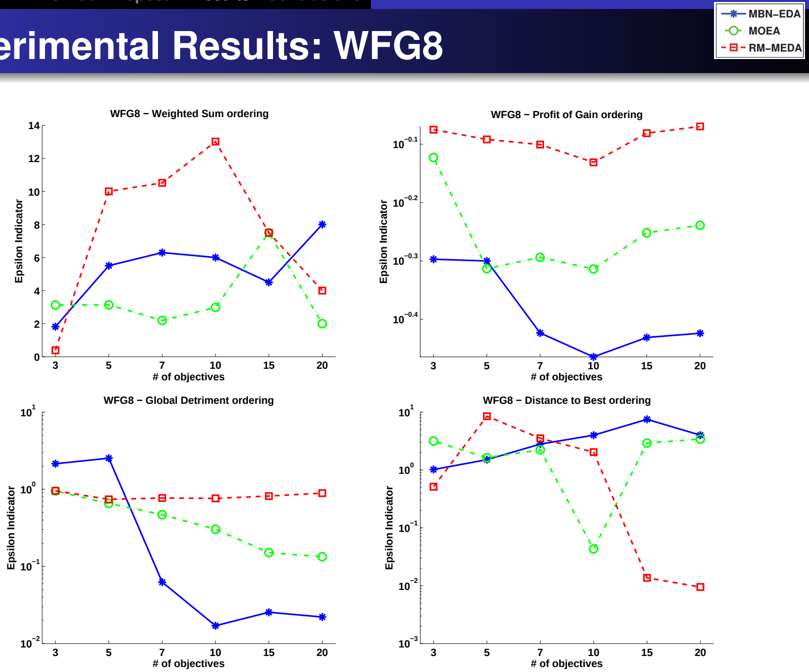
Experimental Results: WFG6



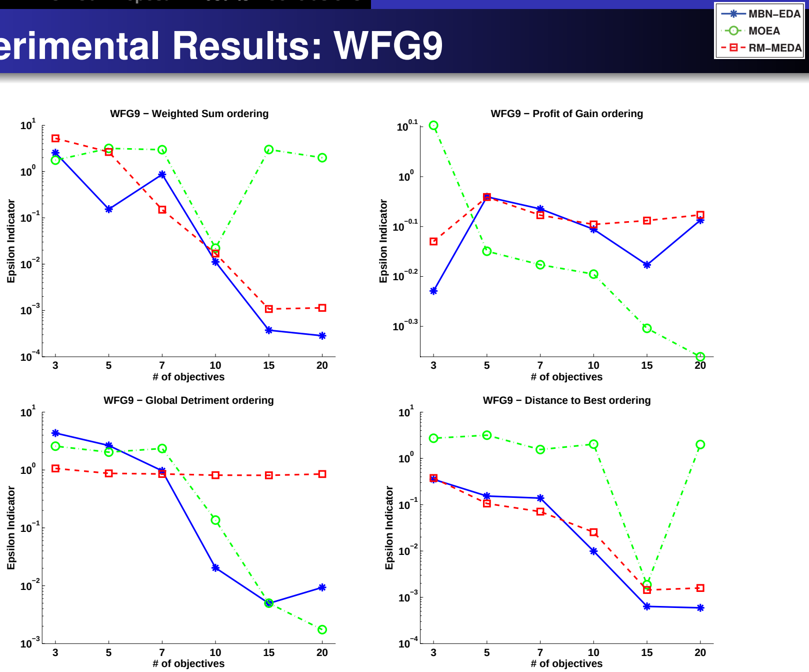
Experimental Results: WFG7



Experimental Results: WFG8

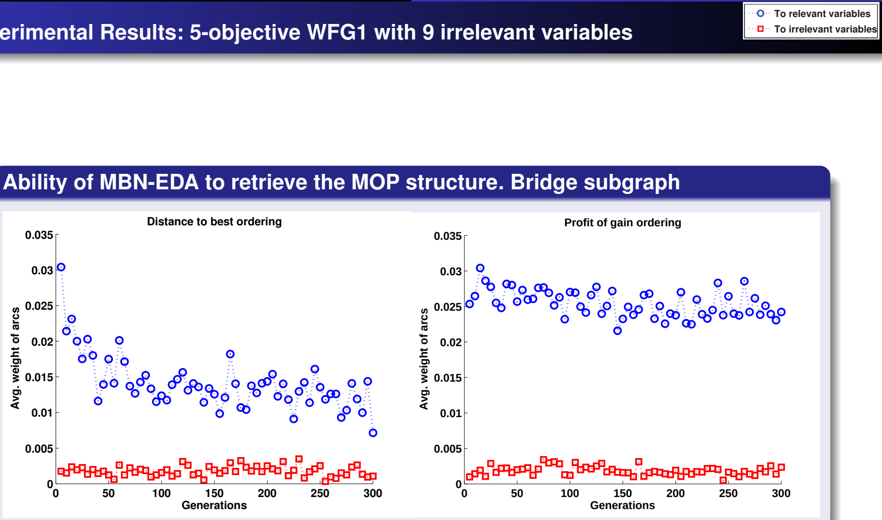


Experimental Results: WFG9



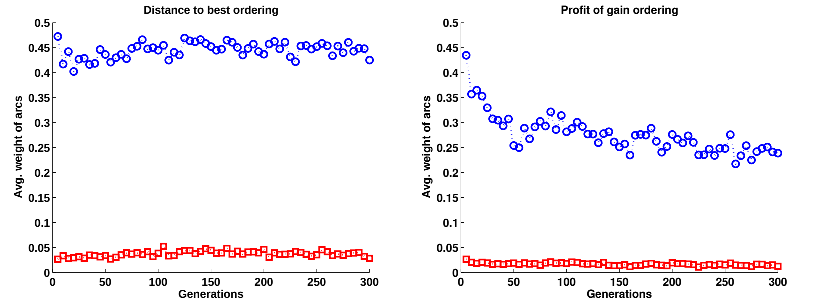
Experimental Results: 5-objective WFG1 with 9 irrelevant variables

Ability of MBN-EDA to retrieve the MOP structure. Bridge subgraph



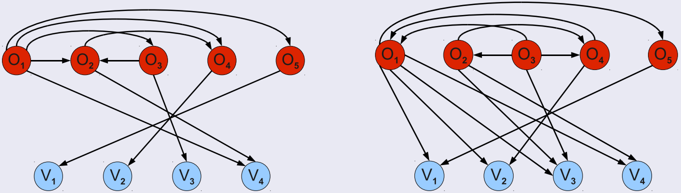
Experimental Results: 8-objective WFG1 with three pairs of similar objectives

Ability of MBN-EDA to retrieve the MOP structure



Experimental Results: 5-objective WFG1 simplified version

Ability of MBN-EDA to retrieve the MOP structure. Two layer structure (most significant arcs)



(a) Distance to best ordering (b) Profit of gain ordering

A simplified version of the 5-objective WFG1 problem

$$
\begin{aligned}
o_1(\mathbf{v}) &= a + 2 \cdot h_1(g_2(v_1), g_2(v_2), g_2(v_3), g_2(v_4)) \\
o_2(\mathbf{v}) &= a + 4 \cdot h_2(g_2(v_1), g_2(v_2), g_2(v_3), g_2(v_4)) \\
o_3(\mathbf{v}) &= a + 6 \cdot h_3(g_2(v_1), g_2(v_2), g_2(v_3)) \\
o_4(\mathbf{v}) &= a + 8 \cdot h_4(g_2(v_1), g_2(v_2)) \\
o_5(\mathbf{v}) &= a + 10 \cdot h_5(g_2(v_1))
\end{aligned}
$$

where $a = g_1(v_5, \ldots, v_{16})$

Outline

1. Introduction
2. Estimation of Distribution Algorithms
3. Our Proposal
4. Experimental Results
5. Conclusions

Conclusions

Conclusions

- MOEDA based on joint modeling of variables and objectives with a two layer structure in the probabilistic graphical model
- Able to discover the structure of the problem
  - Links among variables, objectives and variables and objectives
  - Relevant and irrelevant variables for each of the objectives
- Competitive results with state of the art MOEAs

Many Thanks to

- C. Bielza
- H. Karshenas
- R. Santana

MULTI-OBJECTIVE OPTIMIZATION WITH ESTIMATION OF DISTRIBUTION ALGORITHMS

Pedro Larrañaga

Computational Intelligence Group
Artificial Intelligence Department
Technical University of Madrid

EVOLVE 2012
Mexico City, August 7-9, 2012