Multivariate Multi-Model Approach for Globally Multimodal Problems

Chung-Yao Chuang and Wen-Lian Hsu

Institute of Information Science
Academia Sinica
Taiwan

July 10, 2010

Outline

Outline

Problems with Several Global Optima

Globally Multimodal Problems

- More than one global optimum
- Preferable to identify as many as possible

Examples

- Robot path planning[a]: dynamic environment
- Peptide design[b]: estimated docking energy

[a] Hocaoğlu and Sanderson. Multimodal function optimization using minimal representation size clustering and its application to planning multipaths. *Evolutionary Computation*, 5(1):81–104, 1997.

[b] Belda, Madurga, Tarragó, Llorà and Giralt. Evolutionary computation and multimodal search: A good combination to tackle molecular diversity in the field of peptide design. *Molecular Diversity*, 11(1):7–21, 2007.

Issues with Globally Multimodal Problems

To Obtain More Than One Optimum

Plain evolutionary algorithms are ineffective...

- No mechanism to maintain balance between each basin
- Selection randomly gives preference on one global optimum
- Other optima tend to disappear after several generations

Convergence Taking Longer Time

Combining solutions located in different basins

- Usually produces poor solutions[a]
- Until population drifting toward a single basin

[a] M. Pelikan and D. E. Goldberg. Genetic algorithms, clustering, and the breaking of symmetry. In *Proceedings of the 6th International Conference on Parallel Problem Solving from Nature*: 385–394

Estimation of Distribution Algorithms (EDAs)

Characteristics

- Building probabilistic model on promising solutions
- Using built model to sample new candidate solutions
- Recognizing inter-variable relationship by model building

Inter-Variable Relationship

- Essential to address hard optimization problems
- Automatic discovery of such information
- Linkage Problem

Model Building for Globally Multimodal Problems

Previous Works

- UMDA + k-Means: multiple univariate models[a]
- UEBNA: single Bayesian network with cluster variable[b]
- ϕ-PBIL: multiple simple order-2 models[c]

[a] M. Pelikan and D. E. Goldberg. Genetic algorithms, clustering, and the breaking of symmetry. In *Proceedings of the 6th International Conference on Parallel Problem Solving from Nature*: 385–394

[b] J. M. Peña, J. A. Lozano, and P. Larrañaga. Globally multimodal problem optimization via an estimation of distribution algorithm based on unsupervised learning of bayesian networks. *Evolutionary Computation*, 13(1):43–66, 2005.

[c] L. Emmendorfer and A. Pozo. Effective linkage learning using low-order statistics and clustering. *Evolutionary Computation, IEEE Transactions on*, 13(6):1233–1246, Dec. 2009.

In This Work...

- Consider multivariate probabilistic models
- Build multiple models at each generation
- Automate the selection of the number of models to use

Outline

Marginal Product Models (MPMs)

- A product of marginal distributions on a partition of variables
- Subsets of variables can be modeled jointly
- Each subset is considered independent of others

Example

The probability of generating a sample $s_1 s_2 s_3 s_4 = 0101$:

$$\begin{aligned} & P(s_1 s_2 s_3 s_4 = 0101) \\ & = P(s_1 = 0) \times P(s_2 = 1, s_4 = 1) \times P(s_3 = 0) \\ & = 0.4 \times 0.4 \times 0.5. \end{aligned}$$

$[s_1]$	$[s_2\ s_4]$	$[s_3]$
$P(s_1 = 0) = 0.4$	$P(s_2 = 0, s_4 = 0) = 0.4$	$P(s_3 = 0) = 0.5$
$P(s_1 = 1) = 0.6$	$P(s_2 = 0, s_4 = 1) = 0.1$	$P(s_3 = 1) = 0.5$
	$P(s_2 = 1, s_4 = 0) = 0.1$	
	$P(s_2 = 1, s_4 = 1) = 0.4$	

Table: A marginal product model

Exenteded Compact Genetic Algorithm (ECGA)

- Use marginal product models (MPMs).
- Model building is performed in a greedy approach.
- Structure and parameters are searched at the same time.

MPMs are measured by . . .

- Minimum description length (MDL) principle
- How many bits are required to store the model?
- How many bits are required to store the population?
- Model complexity + compressed population complexity

Model Complexity, C_m

Suppose that ...

- The population is of size n.
- The problem is of length ℓ with binary encoding.
- The variables are partitioned into m subsets.
- Each subset is of size k_i, $i = 1 \ldots m$.

Definition

The marginal distribution of the ith variable subset

- require $2^{k_i} - 1$ frequency counts to be completely specified,
- each frequency count is of length $\log_2(n+1)$ bits.

$$C_m = \log_2(n+1) \sum_{i=1}^{m} \left(2^{k_i} - 1\right)$$

Compressed Population Complexity, C_p

Suppose that ...

- The population is of size n.
- The problem is of length ℓ with binary encoding.
- The variables are partitioned into m subsets.
- Each subset is of size k_i, $i = 1 \ldots m$.

Definition

To store the selected population with optimal compression

- each variable subset $\Rightarrow$ a compression block
- optimal compression: probability $p_i \Rightarrow -\log_2 p_i$ bits

$$C_p = n \sum_{i=1}^{m} \sum_{j=1}^{2^{k_i}} -p_{ij} \log_2 p_{ij} \ ,$$

Outline

Partition Solutions into c Subsets

Assume that . . .

We are given

- a set of n solutions, S
- c MPMs, $\{M_y | y \in \{1, 2, \ldots, c\}\}$

and being asked to assign,

- for each solution, the fittest model among these c MPMs

Suitability of Modeling

- Compression performance $\Leftrightarrow$ suitability of modeling
- Compress better $\Rightarrow$ fit better
- Fittest model $\Rightarrow$ encodes the solution to the shortest description

Partition Solutions into c Subsets

Assume that ...

We are given

- a set of n solutions, S
- c MPMs, $\{M_y | y \in \{1, 2, \ldots, \text{c}\}\}$

and being asked to assign,

- for each solution, the fittest model among these c MPMs

Approach

For each solution x, we should choose M_y with the smallest

$$\lambda = \sum_{i=1}^{m} -\log_2 p_{ix_i}$$

- m is the number of marginal distributions in M_y
- x takes the x_ith partial solution in the ith variable subset

Building Multiple Models

procedure BUILDMODELS(c, S)
  Randomly pick a subset $\{d_y | y \in \{1, 2, ..., c\}\}$ from S.
  Estimate $\{M_y | M_y$ is a univariate model based on $d_y\}$.
  for each x in S **do**
    $y_x \leftarrow y$ such that M_y yields smallest λ for x.
  end for
  repeat
    $y'_x \leftarrow y_x$ for each x in S.
    for each y in $\{1, 2, ..., c\}$ **do**
      $M_y \leftarrow$ greedy MPM search on $\{x | y_x = y\}$.
    end for
    for each x in S **do**
      $y_x \leftarrow y$ such that M_y yields smallest λ for x.
    end for
  until $y'_x = y_x$ for all $x \in S$
  return $\{M_y | y \in \{1, 2, ..., c\}\}$ and $\{y_x | x \in S\}$.
end procedure

Outline

Naive Measurement Doesn't Work!

Naive Measurement

To simply sum up all the complexity terms as

$$\sum_y \left(C_m(M_y) + C_p(M_y)\right)$$

$C_m(M_y)$: model complexity of M_y

$C_p(M_y)$: compressed population complexity of solutions associated with M_y

Doesn't Work Because ...

Larger MPM sets will have biased advantage of being able to

- split the population into smaller subpopulations
- build overly-simplified models on the resulting partition

Solution-Model Association Complexity, C_t

The Missing Part . . .

- Additional information that maps each solution to its model
- Should be included when measuring complexity

Definition

Additional bits required to tag each solution to its associated model

$$C_t = n \sum_{y=1}^{c} -p_y \log_2 p_y \ ,$$

n : the number of solutions

c : the number of models

p_y : the frequency of assigning a solution to the yth model

Complexity Measure for an MPM Set

The complexity of an MPM set $\{M_y | y \in \{1, 2, ..., c\}\}$ on modeling a given set of solutions, S

$$C = C_t(\{y_x | x \in S\}) + \sum_{y=1}^{c} (C_m(M_y) + C_p(M_y))$$

- y_x : the assignment of x to its most suitable model
- $C_t(\{y_x\})$: solution-model association complexity
- $C_m(M_y)$: model complexity of M_y
- $C_p(M_y)$: compressed population complexity of solutions associated with M_y

Outline

Overall Procedure

Initialize a population P with n solutions.
while the stopping criteria are not met **do**
    Evaluate the solutions in P.
    $S \leftarrow$ apply selection on P.
    $c \leftarrow 1$.
    $\mathcal{M}', \mathcal{Y} \leftarrow$ BUILDMODELS(c, S).
    $\mathcal{C}' \leftarrow$ calculate complexity based on $\mathcal{M}'$ and $\mathcal{Y}$.
    repeat
        $\mathcal{M} \leftarrow \mathcal{M}'$.
        $\mathcal{C} \leftarrow \mathcal{C}'$.
        $c \leftarrow c + 1$.
        $\mathcal{M}', \mathcal{Y} \leftarrow$ BUILDMODELS(c, S).
        $\mathcal{C}' \leftarrow$ calculate complexity based on $\mathcal{M}'$ and $\mathcal{Y}$.
    until $\mathcal{C}' \geq \mathcal{C}$
    $O \leftarrow \emptyset$.
    for each model M_y in $\mathcal{M}$ **do**
        $O' \leftarrow$ generate new solutions by sampling M_y.
        $O \leftarrow O \cup O'$.
    end for
    Incorporate O into P.
end while

Outline

Constructing Test Problems

Subproblems

k-bit trap function : $f_t^{(k)}(s_1 s_2 \cdots s_k) = \begin{cases} k, & \text{if } u = k \\ k - 1 - u, & \text{otherwise} \end{cases}$

k-bit inverse trap function : $\bar{f}_t^{(k)}(s_1 s_2 \cdots s_k) = \begin{cases} k, & \text{if } u = 0 \\ u - 1, & \text{otherwise} \end{cases}$

where u is the number of ones in the binary string $s_1 s_2 \cdots s_k$

The Plan . . .

Design test problems that

- assign different region of search space to different combination of $f_t^{(k)}$ and $\bar{f}_t^{(k)}$

Introduce switch variables

- a set of problem variables
- its values specify the combination of $f_t^{(k)}$ and $\bar{f}_t^{(k)}$ to be used to evaluate the corresponding solution

F_1: 2 Optima, Homogeneous Linkage

Concatenating ten $f_t^{(4)}$ or $\bar{f}_t^{(4)}$ and one switch variable, s_{41}

$$F_1(s_1 s_2 \ldots s_{41}) = \begin{cases} G_0(s_1 s_2 \ldots s_{40}), & \text{if } s_{41} = 0 \\ G_1(s_1 s_2 \ldots s_{40}), & \text{if } s_{41} = 1 \end{cases}$$

where G_0 and G_1 are defined as

$$G_0(s_1 s_2 \ldots s_{40}) = \sum_{i=0}^{9} \bar{f}_t^{(4)}(s_{4i+1} s_{4i+2} s_{4i+3} s_{4i+4}) \ ,$$

$$G_1(s_1 s_2 \ldots s_{40}) = \sum_{i=0}^{9} f_t^{(4)}(s_{4i+1} s_{4i+2} s_{4i+3} s_{4i+4}) \ .$$

Note

- Split the search space into 2 equal halves
- Each has a different optimum (all 1's and all 0's)

F_1: Results

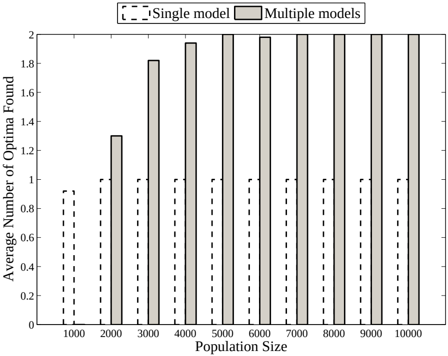

Figure: # of Optima Obtained

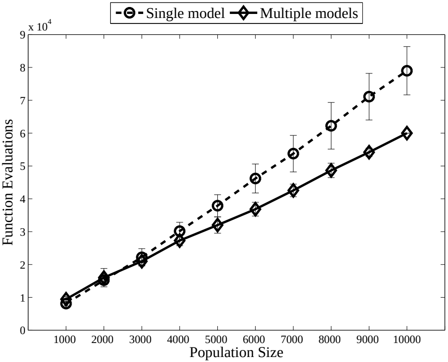

Figure: Function Evaluations

- Single model(ECGA) vs. Multiple models(proposed approach)
- Using tournament selection with tournament size 16
- Each of those experiments are repeated for 50 times

F_2: 4 Optima, Homogeneous Linkage

Using two switch variables

$$F_2(s_1 s_2 \ldots s_{42}) = \begin{cases} G_{00}(s_1 s_2 \ldots s_{40}), & \text{if } s_{41} s_{42} = 00 \\ G_{01}(s_1 s_2 \ldots s_{40}), & \text{if } s_{41} s_{42} = 01 \\ G_{10}(s_1 s_2 \ldots s_{40}), & \text{if } s_{41} s_{42} = 10 \\ G_{11}(s_1 s_2 \ldots s_{40}), & \text{if } s_{41} s_{42} = 11 \end{cases}$$

where the definition of G_{00} to G_{11} are

$$G_{00}(s_1 \ldots s_{40}) = \sum_{i=0}^{4} (\bar{f}_t^{(4)}(s_{8i+1} \ldots s_{8i+4}) + \bar{f}_t^{(4)}(s_{8i+5} \ldots s_{8i+8})) \, ,$$

$$G_{01}(s_1 \ldots s_{40}) = \sum_{i=0}^{4} (\bar{f}_t^{(4)}(s_{8i+1} \ldots s_{8i+4}) + f_t^{(4)}(s_{8i+5} \ldots s_{8i+8})) \, ,$$

$$G_{10}(s_1 \ldots s_{40}) = \sum_{i=0}^{4} (f_t^{(4)}(s_{8i+1} \ldots s_{8i+4}) + \bar{f}_t^{(4)}(s_{8i+5} \ldots s_{8i+8})) \, ,$$

$$G_{11}(s_1 \ldots s_{40}) = \sum_{i=0}^{4} (f_t^{(4)}(s_{8i+1} \ldots s_{8i+4}) + f_t^{(4)}(s_{8i+5} \ldots s_{8i+8})) \, .$$

F_2: Results

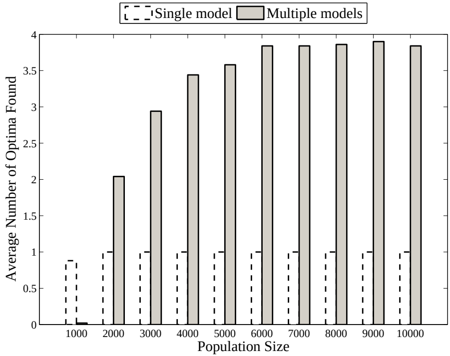

Figure: # of Optima Obtained

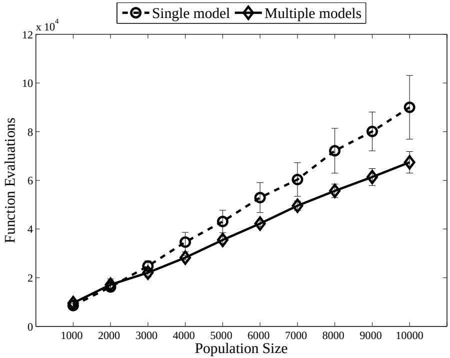

Figure: Function Evaluations

- Single model(ECGA) vs. Multiple models(proposed approach)
- Using tournament selection with tournament size 16
- Each of those experiments are repeated for 50 times

F_3: 2 Optima, Heterogeneous Linkage

Each optimum with different structural decomposition

$$F_3(s_1 s_2 \ldots s_{41}) = \begin{cases} H_0(s_1 s_2 \ldots s_{40}), & \text{if } s_{41} = 0 \\ H_1(s_1 s_2 \ldots s_{40}), & \text{if } s_{41} = 1 \end{cases}$$

where H_0 and H_1 are defined as

$$H_0(s_1 s_2 \ldots s_{40}) = \sum_{i=0}^{9} \bar{f}_t^{(4)}(s_{4i+1} \ldots s_{4i+4}) \ ,$$

$$H_1(s_1 s_2 \ldots s_{40}) = \sum_{i=0}^{8} f_t^{(4)}(s_{4i+3} \ldots s_{4i+6}) + f_t^{(4)}(s_{39} s_{40} s_1 s_2) \ .$$

Note

- Subfunctions are not aligned in variables
- Disruption of good partial solutions is more likely to happen

F_3: Results

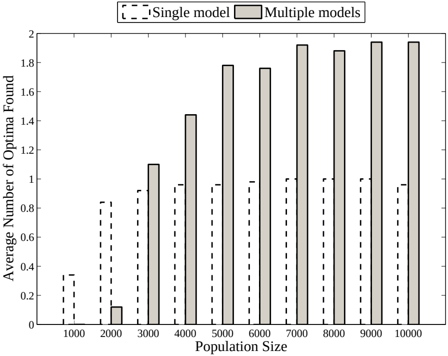

Figure: # of Optima Obtained

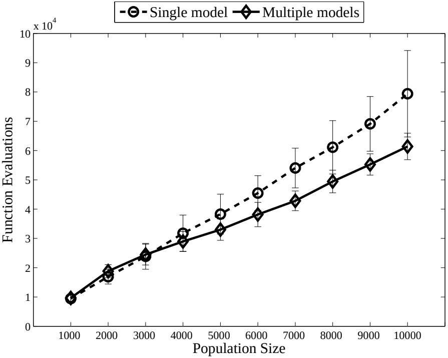

Figure: Function Evaluations

- Single model(ECGA) vs. Multiple models(proposed approach)
- Using tournament selection with tournament size 16
- Each of those experiments are repeated for 50 times

Summary

We have introduced . . .

- an iterative approach for building multiple models
- a heuristics to choose the number of models to use
- an optimization algorithm using the above

Empirical results suggest . . .

- obtaining more global optima per run
- reducing the number of generations spent for convergence