

Structural Accuracy of Probabilistic Models in BOA

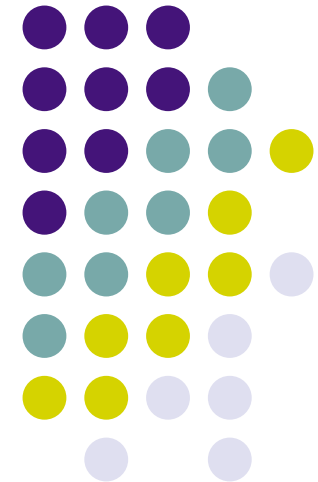
Claudio F. Lima¹

Work in collaboration with Martin Pelikan², David E. Goldberg³,
Fernando G. Lobo¹, Kumara Sastry³, and Mark Hauschild²

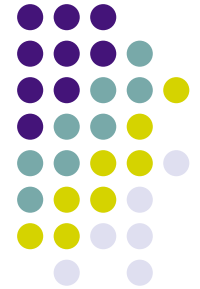
(1) University of Algarve, Portugal

(2) University of Missouri at St. Louis, USA

(3) University of Illinois at Urbana-Champaign, USA

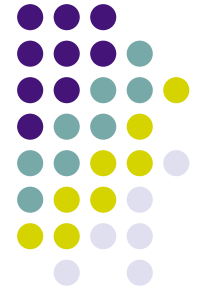


Outline



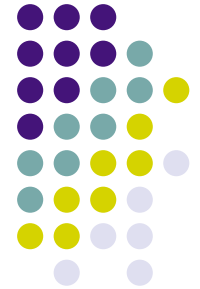
- Motivation
- Background and previous work
 - Bayesian optimization algorithm (BOA)
 - Fitness estimation in BOA
 - Substructural local search in BOA
- Measuring structural accuracy of PMs in BOA
- Influence of BOA's parameters
 - Population Size
 - Selection Strategy
 - Replacement Strategy
- Conclusions

Motivation



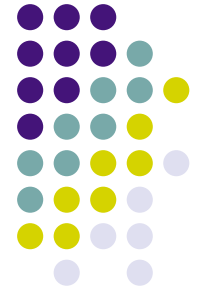
- Probabilistic models of EDAs allow better recombination of subsolutions
- EDAs: Beyond efficient recombination
 - Get we can more from these models? Yes!
- Efficiency enhancement on EDAs
 - Evaluation relaxation by fitness estimation
 - Local search in substructural neighborhoods

Bayesian Optimization Algorithm



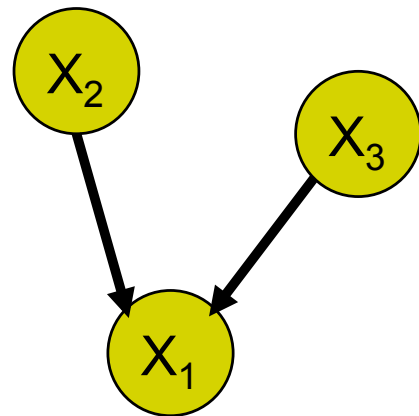
- Pelikan, Goldberg, and Cantú-Paz (1999)
- Use Bayesian networks to model good solutions
- Model structure => acyclic directed graph
 - Nodes represent variables
 - Edges represent conditional dependencies
- Model parameters => conditional probabilities
 - Conditional Probability Tables based on the observed frequencies
 - Local structures: Decision Trees or Graphs

A 3-bit Example



Model Structure

↓
Directed Acyclic Graph

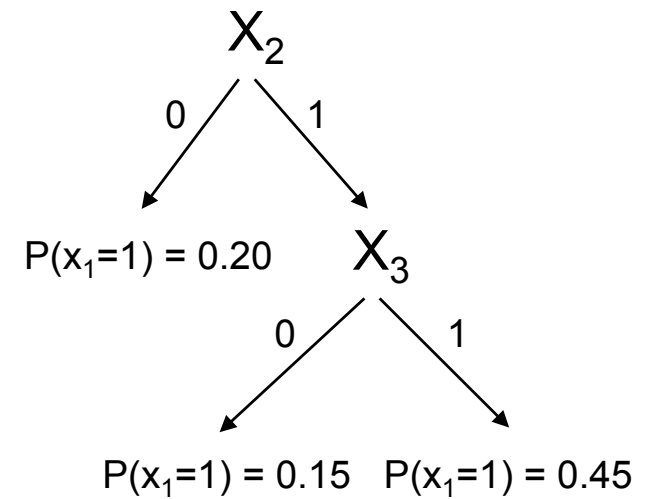


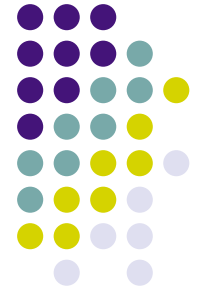
Model Parameters

↙ ↘
Conditional Probability Tables

X_2X_3	$P(X_1=1 X_2X_3)$
00	0.20
01	0.20
10	0.15
11	0.45

Decision Trees





Modeling Fitness in BOA

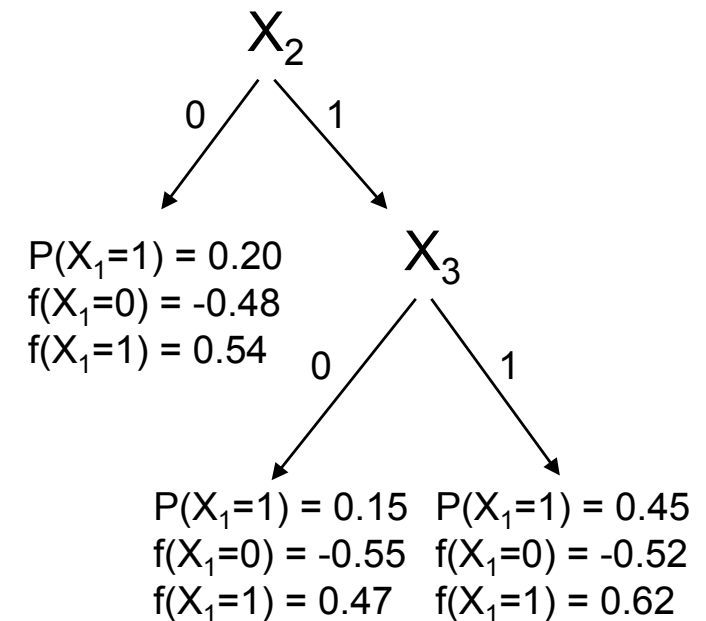
- Bayesian networks extended to store a surrogate fitness model (Pelikan & Sastry, 2004)
- Learned model defines the structure of the approximation
- The coefficients are learned from a proportion of the population (schema averages, least squares, etc)
- Surrogate fitness is used to estimate the fitness of the remaining individuals (therefore reducing evals)

$$f_{est}(X_1, X_2, \dots, X_\ell) = \bar{f} + \sum_{i=1}^{\ell} (\bar{f}(X_i | \Pi_i) - \bar{f}(\Pi_i))$$

The same 3-bit Example



X_2X_3	$P(X_1=1 X_2X_3)$	$f(X_1=0 X_2X_3)$	$f(X_1=1 X_2X_3)$
00	0.20	-0.49	0.53
01	0.20	-0.38	0.51
10	0.15	-0.55	0.47
11	0.45	-0.52	0.62



Estimated fitness:

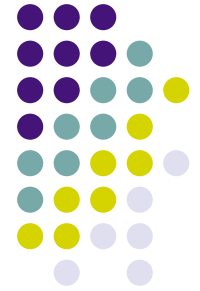
$$f_{est}(X_1, X_2, X_3) = \bar{f} + \bar{f}(X_1|X_2X_3) - \bar{f}(X_2X_3) + \bar{f}(X_2) + \bar{f}(X_3)$$

Why Substructural Neighborhoods?



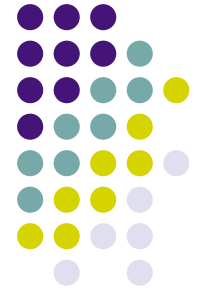
- An efficient mutation operator should search in the correct neighborhood
- Oftentimes this is done by incorporating domain- or problem-specific knowledge
- However, efficiency typically does not generalize beyond a small number of applications
- Bitwise local search have more general applicability but with inferior results

Substructural Neighborhoods

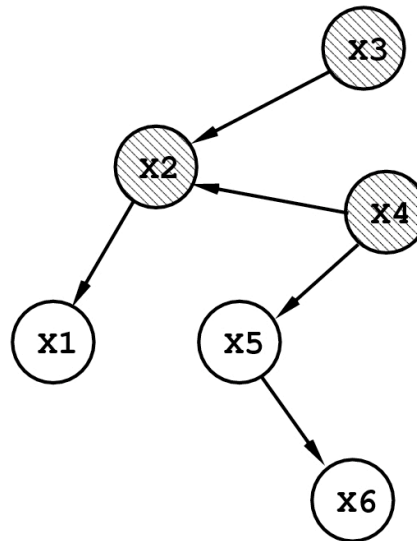


- Neighborhoods defined by the probabilistic model of EDAs
- Exploits the underlying problem structure while not losing generality of application
- Exploration of neighborhoods respect dependencies between variables
 - If $[X_1 X_2 X_3]$ form a linkage group, the neighborhood considered will be 000, 001, 010, ..., 111

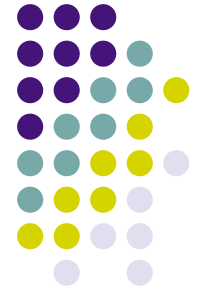
Substructural Neighborhoods in BOA



- Each individual contains l neighborhoods
- Each neighborhood considers x_i and the set of parent variables

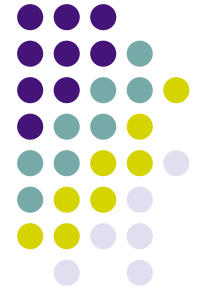


BOA + Substructural Hillclimbing



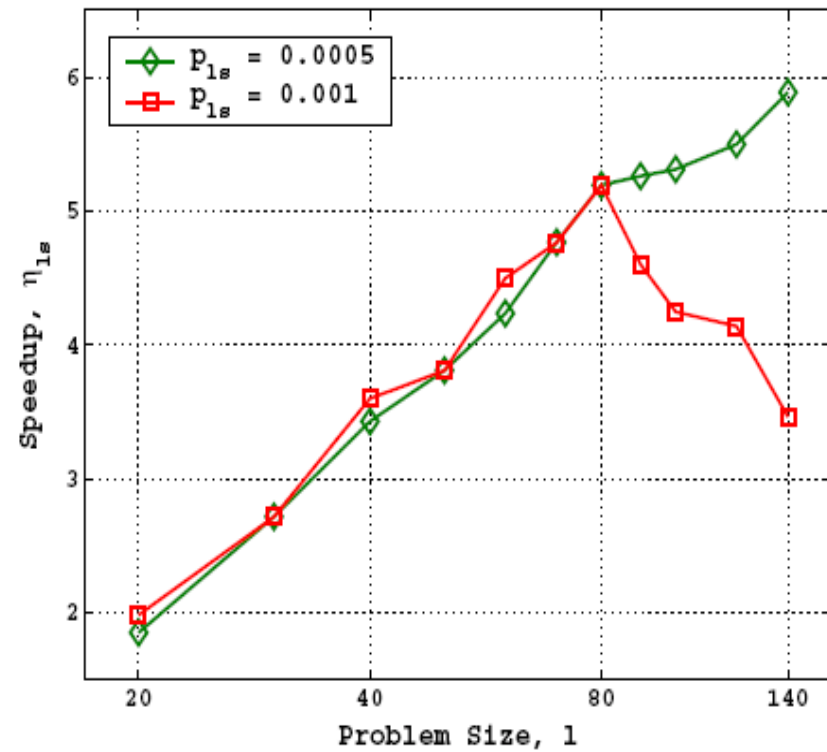
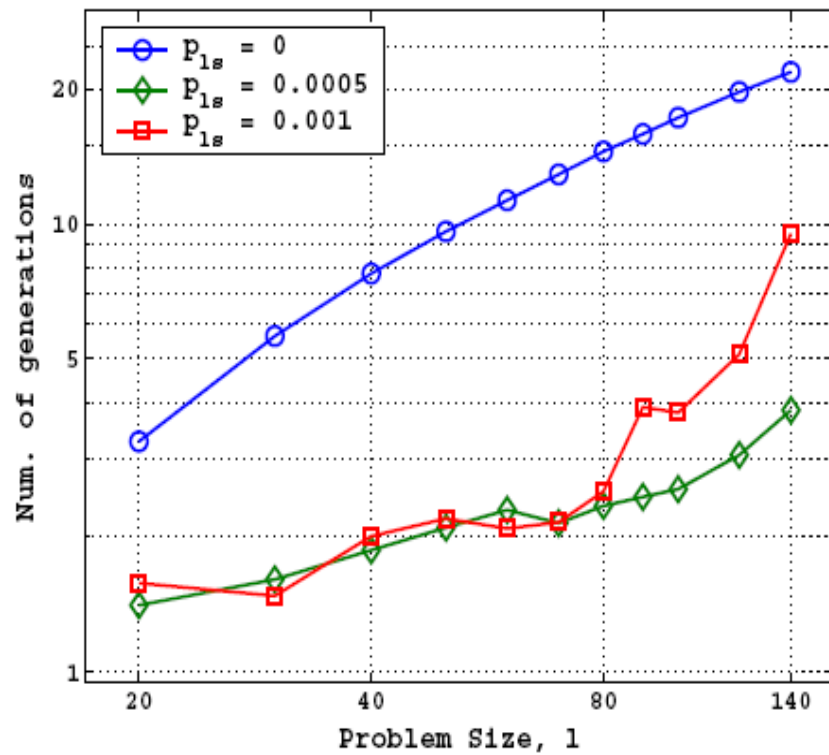
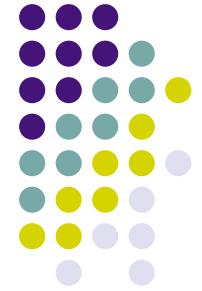
- After model sampling each offspring undergoes local search with a certain probability p_{ls}
- Current model is used to define the neighborhoods
- Choice of best subsolutions => surrogate fitness model
- Cost of performing local search is then minimal

Substructural Hillclimbing in BOA



1. Consider the first variable X_i according with the ancestral reverse ordering of variables in the Bayesian network.
2. Choose the values (x_i, π_i) associated with the maximal substructural fitness $\bar{f}(X_i|I_i)$.
3. Set variables (X_i, I_i) of the considered individual to values (x_i, π_i) if the overall fitness of the individual is improved by doing so, otherwise leave the individual unchanged.
4. Repeat steps 2-3 for all remaining variables following the ancestral reverse order of variables.

Scalability Results (5-bit trap)

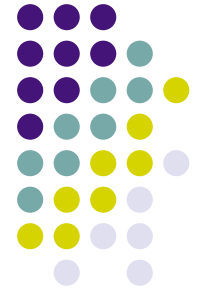


Scalability Results (5-bit trap)



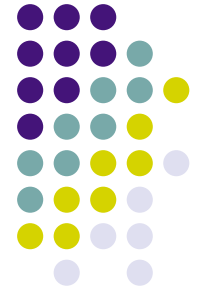
- Substantial speedups are obtained ($\eta=6$ for $l=140$)
- Speedup scales as $O(l^{0.45})$ for $l < 80$
- For bigger problem sizes the speedup is more moderate
- $p_{ls} = 5 \times 10^{-4}$ adequate for range of problems tested, but *optimal* proportion should decrease for higher problem sizes

Scalability Issues



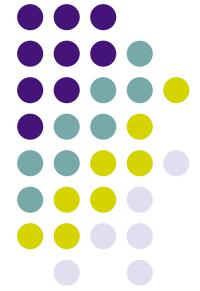
- Optimal proportion of local search slowly decreases with problem size
- Exploration of substructural neighborhoods is sensitive to the accuracy of model structure
- Spurious linkage size grows with problem size
- BOA's sampling ability is not affected because conditional probabilities nearly express independence between spurious and linked variables

Overfitting in EDAs



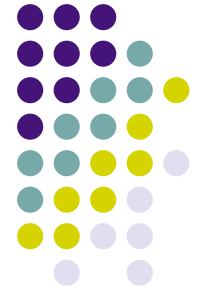
- EDAs can solve efficiently challenging optimization problems
 - Where exploiting problem structure is a plus
- But oftentimes their probabilistic models do not exactly reflect the problem structure
- Why?
 - PMs are learned from a sample of limited size (population)
 - Particular features of that specific sample are also encoded
 - Well-known problem in ML → **model overfitting**

What can affect model accuracy in BOA?

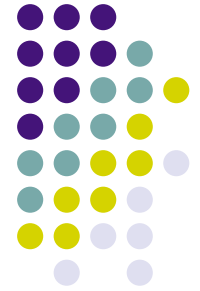


- Bayesian Network Learning
 - Search Procedure (Wu *et al.*, 2006; Echevoyen *et al.*, 2007)
 - Score Metric (Pelikan, 2005; Correa *et al.*, 2006)
- Parameters
 - Population Size (Pelikan, 2005)
 - Selection Strategy (Johnson *et al.*, 2001; Santana *et al.*, 2005)
 - Replacement Strategy
- Problem Structure
 - Subfunction Size
 - Subfunction Overlapping Degree
 - Subfunction Fitness Distribution
 - Hierarchy

Experimental Setup for Measuring Structural Accuracy of PMs

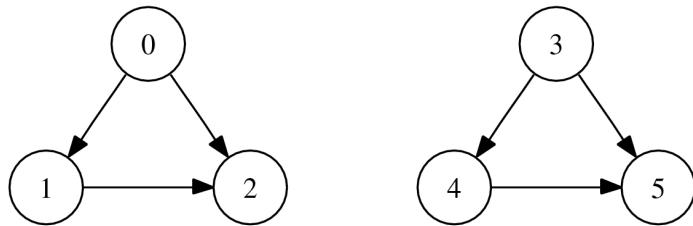


- Test problem(s) structure should be known
- Learning problem structure should be crucial to solve it efficiently
- Easy control of problem size and difficulty in order to test ***scalability***
- *m-k* deceptive trap functions:
 - Concatenated ***m*** trap functions of size ***k*** each
 - Total problem size is $m \cdot k$ (in case of overlapping, $m(k-o)+o$)
 - ***k***-order statistics are required to (efficient) success
 - Lower-order statistics lead the search away from the optimum



A simple example for Trap3

- Ideal BN for Trap3:



- Dependency relations

- $0 \leftarrow \{ \}$
- $1 \leftarrow \{ 0 \}$
- $2 \leftarrow \{ 0, 1 \}$
- $3 \leftarrow \{ \}$
- $4 \leftarrow \{ 3 \}$
- $5 \leftarrow \{ 2, 3 \}$

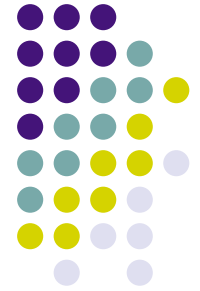
- Ideally, relations should respect a certain chain
- But the dependency that relates all **k** variables is crucial!
- And most difficult to learn
- In this way, **k**-order statistics are maintained
 - $p(x_0, x_1, x_2) = p(x_0) \cdot p(x_1 | x_0) \cdot p(x_2 | x_0, x_1)$

What are we going to measure?



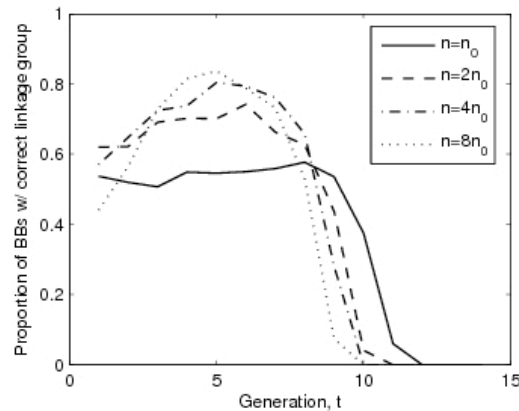
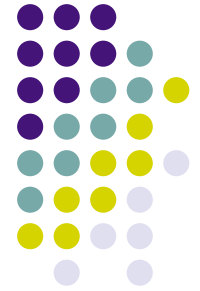
- Proportion of subfunctions with **correct** linkage group
 - Example: $2 \leftarrow \{ 0,1 \}$
- Proportion of subfunctions with **spurious** linkage group
 - Example: $2 \leftarrow \{ 0,1,3 \}$
- Proportion of subfunctions with any of the above
 - Sum of the two previous proportions
- Average size of spurious linkage
 - Number of spurious variables

Influence of BOA's parameters

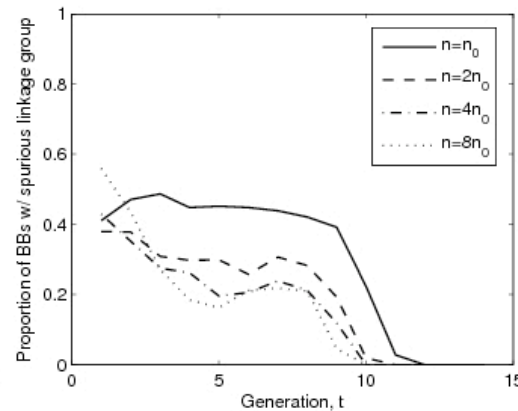


- Population Size
- Selection Strategy
 - Selection method
 - Selection intensity
- Replacement Strategy
 - Replacement method
 - Elitist intensity, diversity preservation

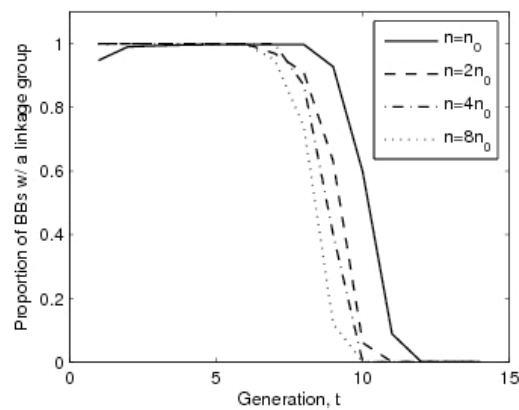
Influence of Population Size



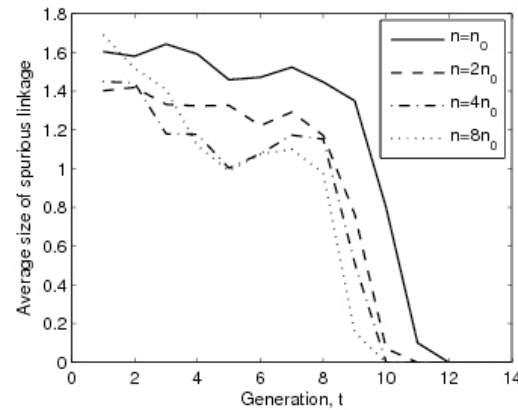
(a) Correct linkage groups



(b) Spurious linkage groups



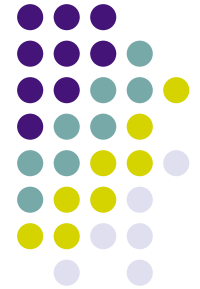
(c) All linkage groups



(d) Average size of spurious linkage

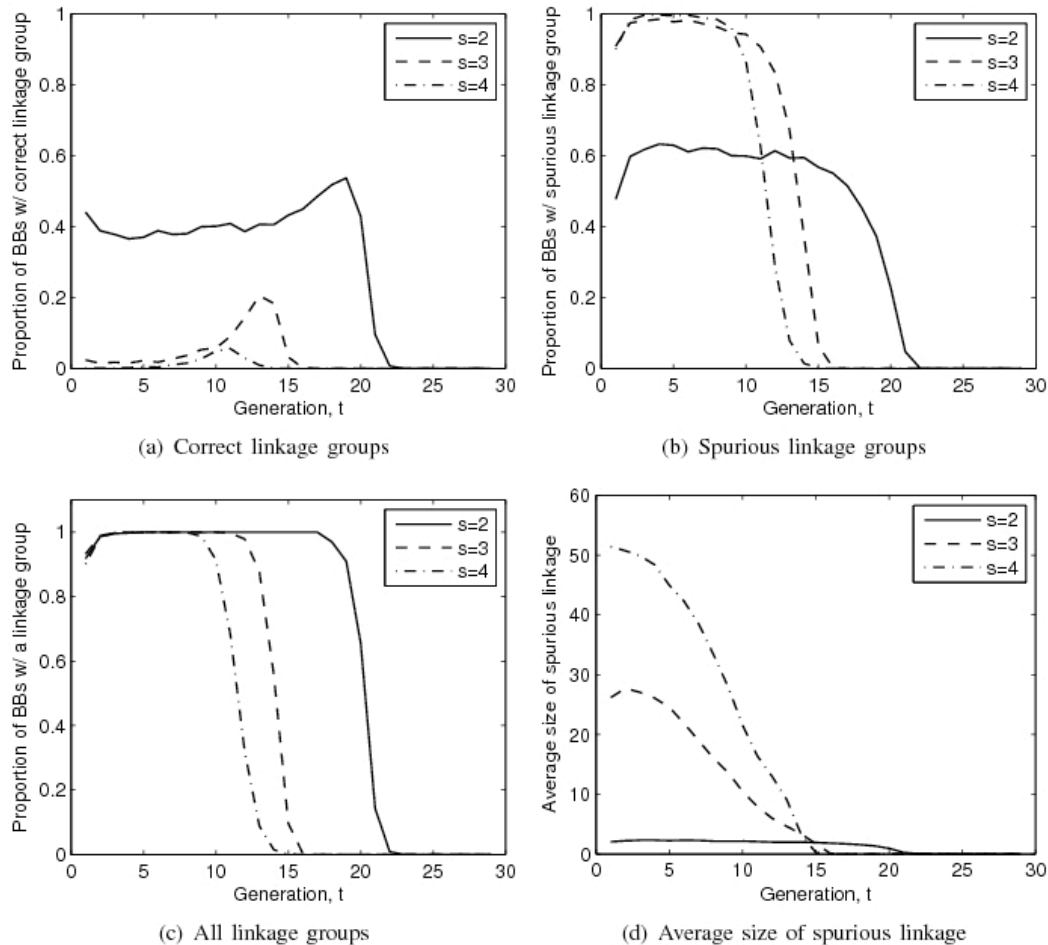
- $m = 10$ and $k = 5$
- n_0 is the minimal population size required to solve problem
- Exponentially increasing population sizes: n_0 , $2n_0$, $4n_0$, and $8n_0$
- Increasing population size slightly improves model accuracy

Influence of Selection Strategy



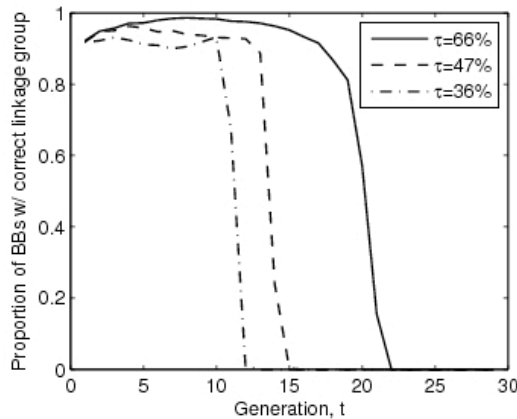
- Tournament Selection
 - Pick best solution from a tournament of s individuals
 - Repeat n times (w/ or wo/ replacement)
- Truncation Selection
 - Choose the best δ % proportion of the population
- Different selection dynamics
 - In terms of selection variance and lost of diversity (Blickle & Thiele, 1997)

Tournament Selection

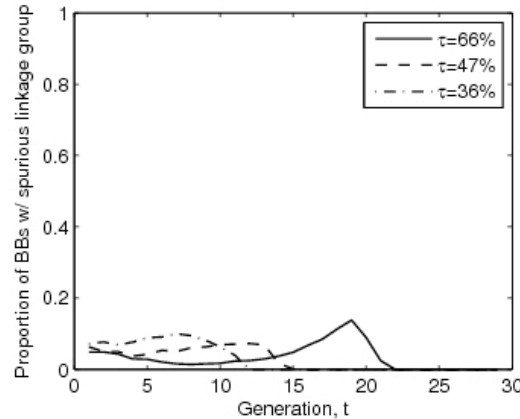


- $m = 24$ and $k = 5$ ($l = 120$)
- Tournament sizes $s = 2, 3, 4$
- Proportion of accurate structures is not significant (in particular for larger s)
- However, w/ or wo/ spurious linkage, all substructures are represented
- Higher selection intensity increases spurious linkage size

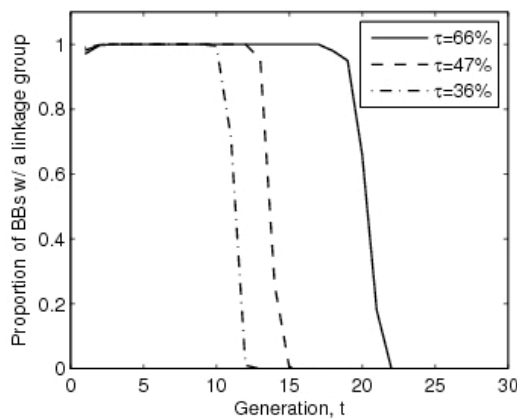
Truncation Selection



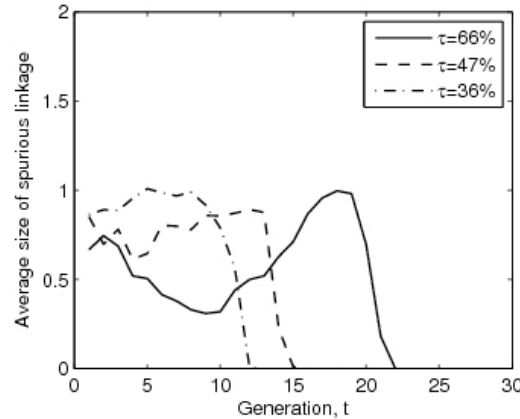
(a) Correct linkage groups



(b) Spurious linkage groups



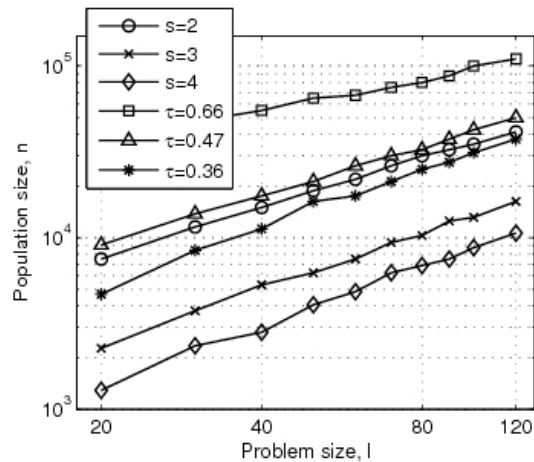
(c) All linkage groups



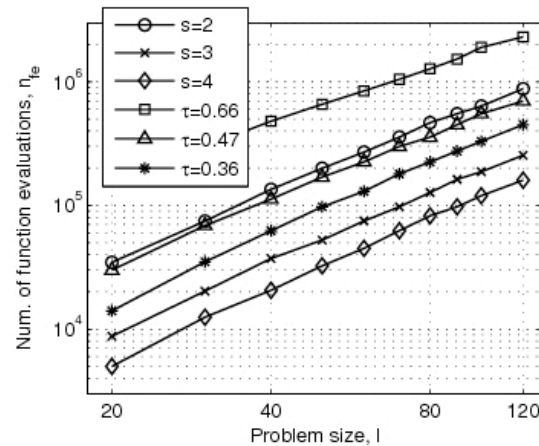
(d) Average size of spurious linkage

- $m = 24$ and $k = 5$ ($l = 120$)
- Truncation thresholds $\delta = 66\%, 47\%, 36\%$
- Same selection intensity as $s = 2, 3, 4$
- Proportion of accurate structures is now close to 100%
- Number of spurious variables is low
- Higher selection intensity has small impact on model structural accuracy

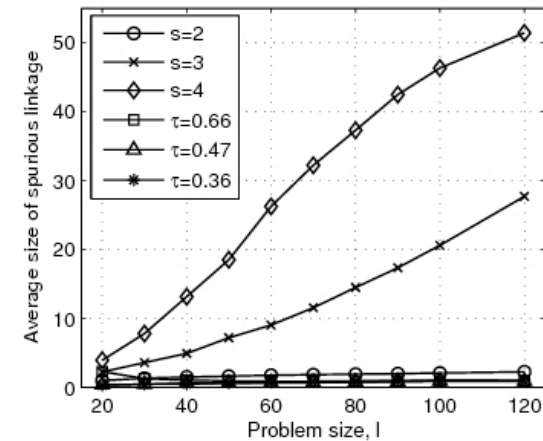
Selection Strategy Scalability



(a) Population size



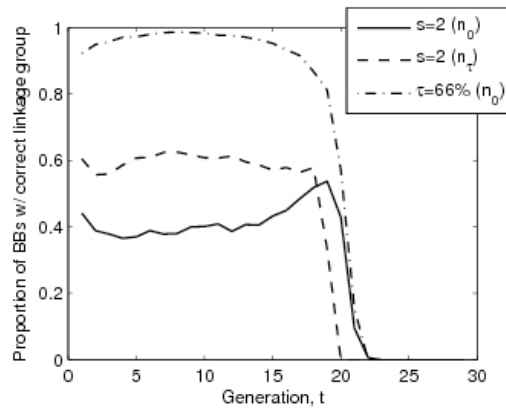
(b) Number of function evaluations



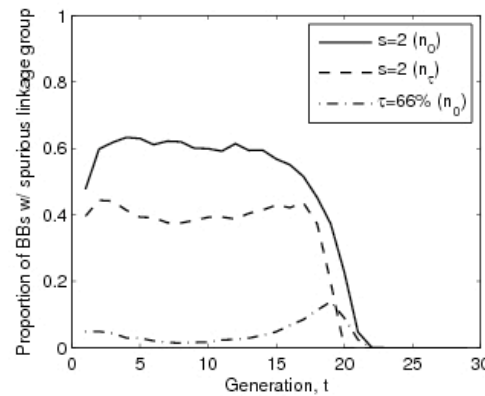
(c) Average size of spurious linkage

- n and n_{fe} required for truncation is higher than for tournament by a significant however constant factor (2-3 times more)

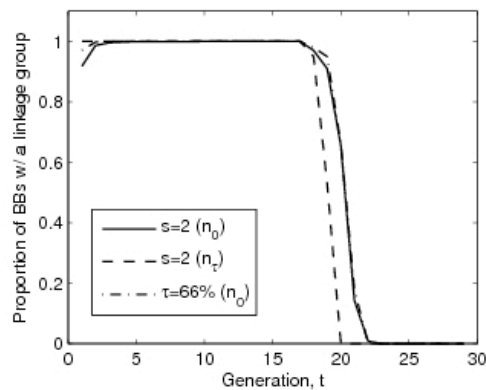
Tournament vs. Truncation: Is it a matter of population size?



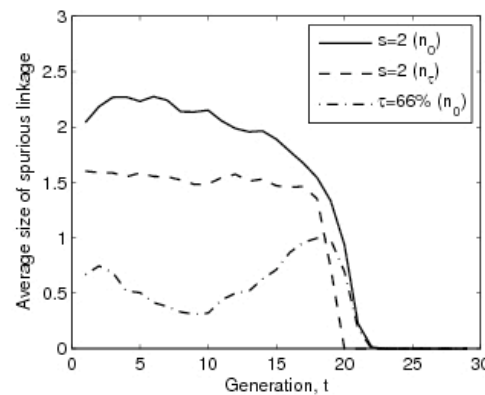
(a) Correct linkage groups



(b) Spurious linkage groups



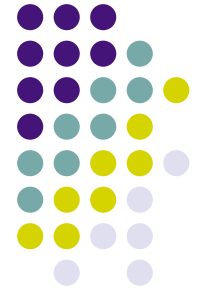
(c) All linkage groups



(d) Average size of spurious linkage

- $m = 24$ and $k = 5$ ($l = 120$)
- Comparing tournament ($s = 2$) and truncation ($\delta = 66\%$) with the same:
 - population size
 - selection intensity
- Tournament selection improves accuracy but is still not close to truncation

What is it then?

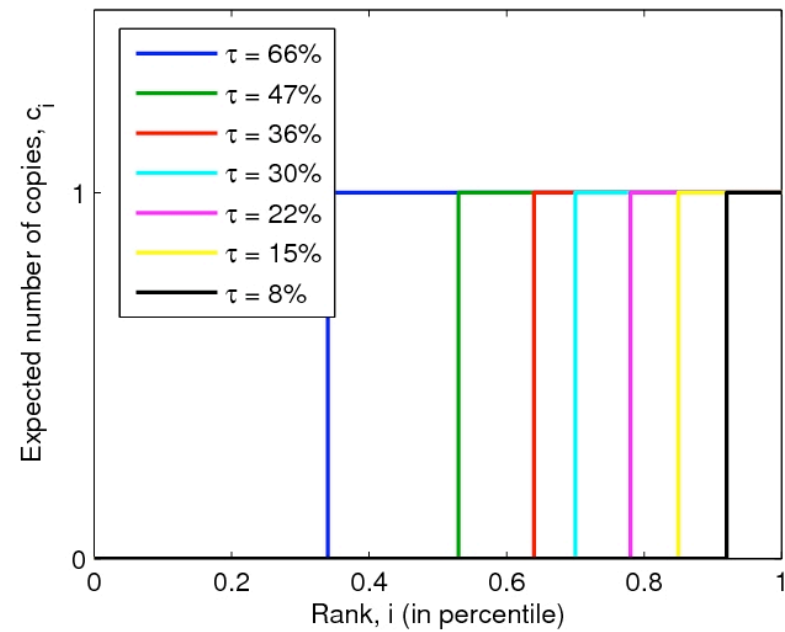
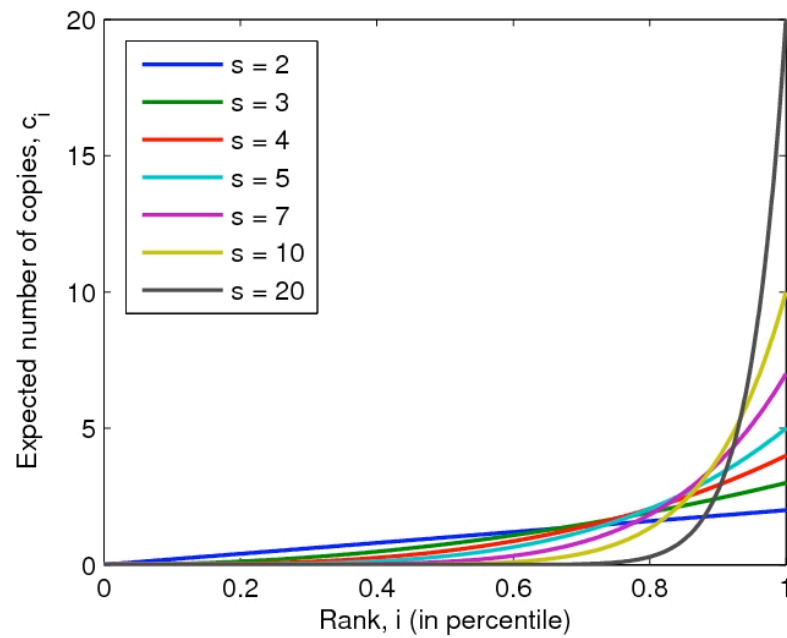


- Distribution of the number of copies in the selected population affects BN learning
 - In tournament the number of copies is somewhat proportional to its rank (best guy gets exactly s copies)
 - In truncation no particular relevance is given to top-ranked solutions
- Model overfitting of top-ranked individuals takes place (in particular for increasing s) for tournament selection

Distributions



- Tournament vs. Truncation

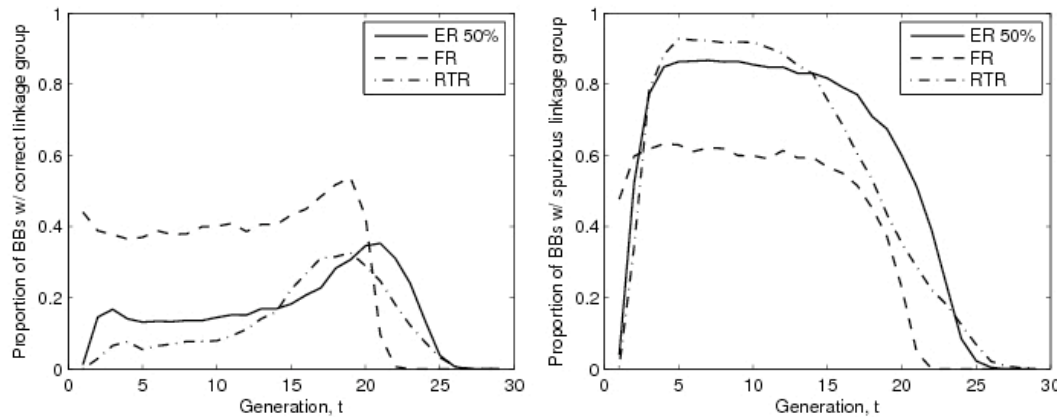


Influence of Replacement Strategy



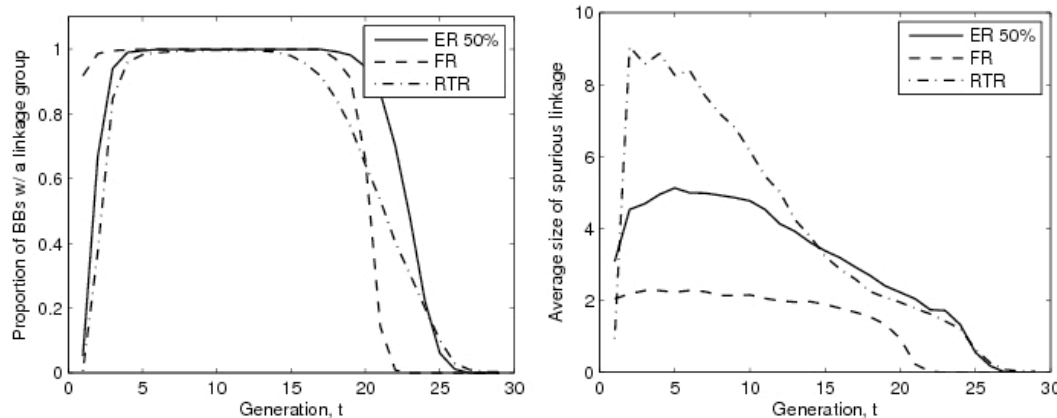
- Full replacement (FR)
 - Offspring population fully replaces the parents
- Elitist replacement (ER)
 - Worst portion of the parents is replaced by offspring
- Restricted tournament replacement (RTR)
 - Offspring directly competes with similar parent for a place in the next population (diversity preservation)

Influence of Replacement (Tournament Selection)



(a) Correct linkage groups

(b) Spurious linkage groups

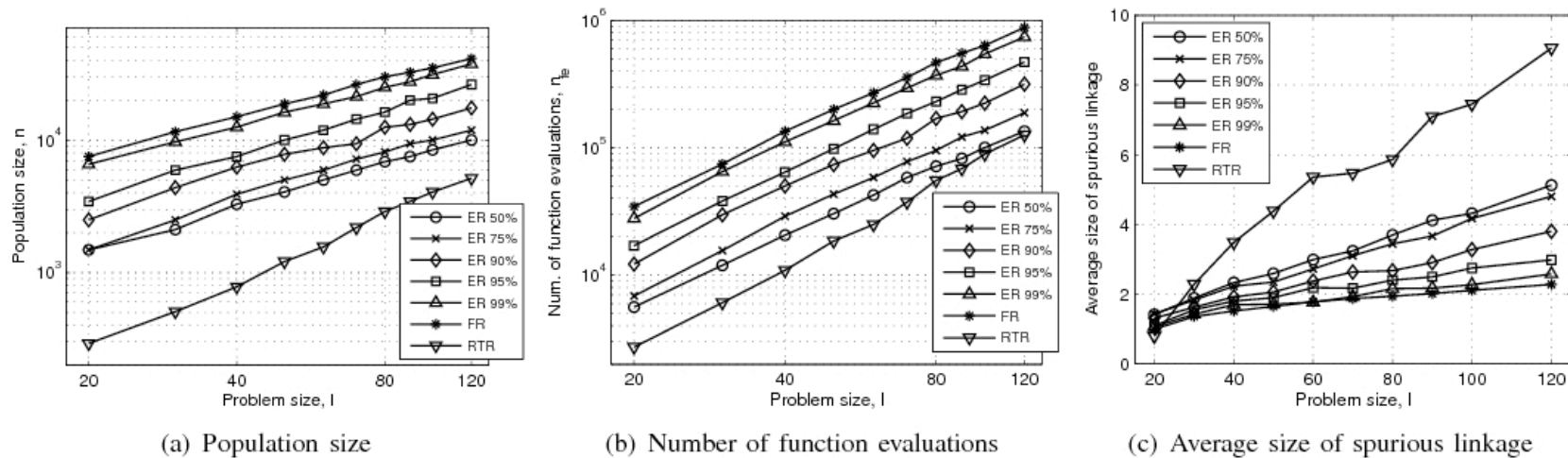


(c) All linkage groups

(d) Average size of spurious linkage

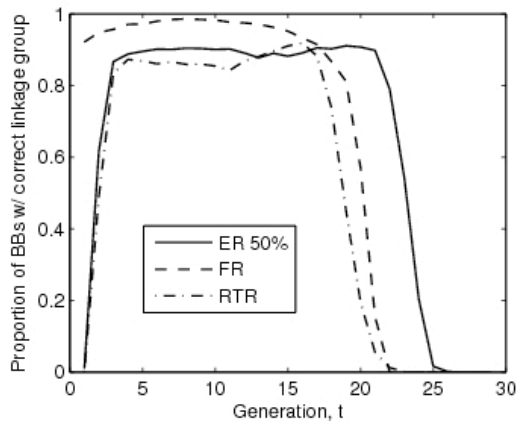
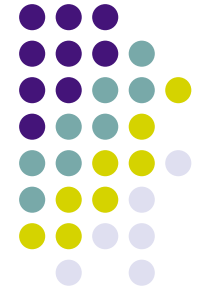
- $m = 24$ and $k = 5$ ($l = 120$)
- Comparing ER50%, FR, and RTR with tournament selection ($s = 2$)
- Structural accuracy is higher with FR
- Replacement strategy does not have the same impact as selection on the spurious linkage size
- Spurious linkage is more frequent with RTR because of the smaller pop. size

Influence of Replacement (Tournament Selection)

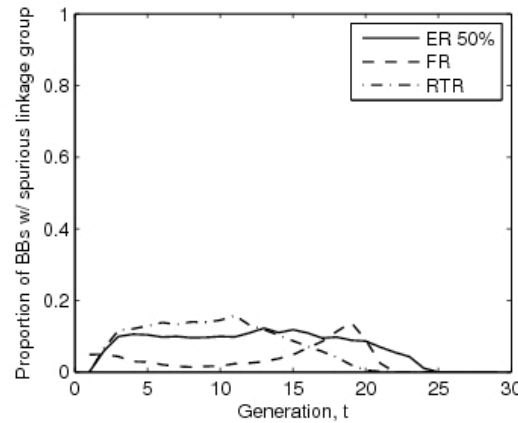


- Additional elitist proportions are shown (1-50%)
- RTR clearly requires less n and n_{fe}
- However, linkage information is not so accurate

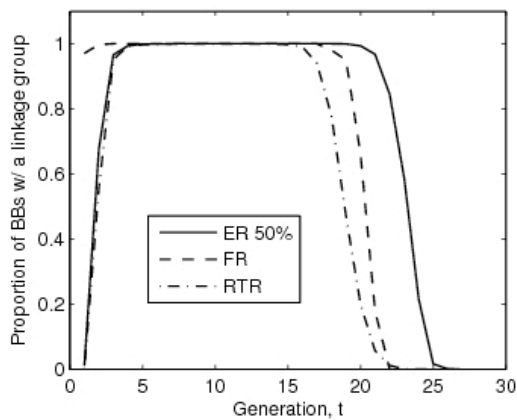
Influence of Replacement (Truncation selection)



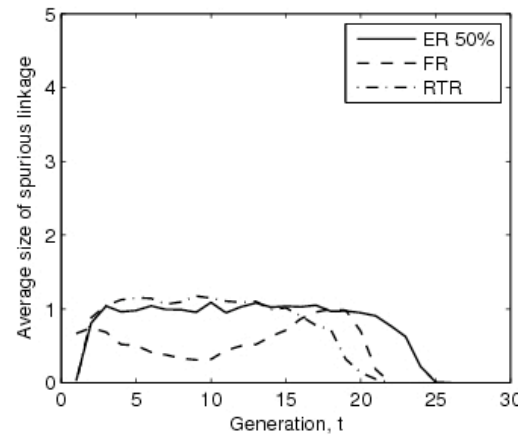
(a) Correct linkage groups



(b) Spurious linkage groups



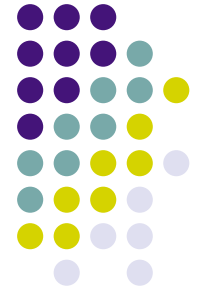
(c) All linkage groups



(d) Average size of spurious linkage

- $m = 24$ and $k = 5$ ($l = 120$)
- Same comparison with truncation selection ($\delta = 66\%$)
- Structural accuracy is close to 100% for all replacement strategies
- FR is still slightly better because it needs larger n
- Size of spurious linkage is quite low

Conclusions



- Trade-off between model accuracy and overall performance in BOA
- Truncation is better than tournament selection for accurate modelling
- For the same purpose, the replacement method is more relevant if tournament selection is used (in which case full replacement is better)
- If overall performance (num. of evals) is our main concern, tournament and RTR are the best options