Estimation of Distribution Algorithms for Permutation-based Optimization Problems

Josu Ceberio

Intelligent Systems Group
Department of Computer Science and Artificial Intelligence
University of the Basque Country UPV/EHU

NICaiA: Nature Inspired Computation and its Applications Meeting
Donostia, 22-23 April 2013

Outline

Outline

1. Estimation of Distribution Algorithms
2. Permutation-based Optimization Problems
3. Previous approaches
4. Probability Models on Rankings
5. Some Experiments
6. Conclusions

Evolutionary Algorithms

Similar to Genetic Algorithms

Given set of candidate individuals

Estimation of Distribution Algorithms

Definition

Given set of candidate individuals

Learn a probability distribution

Estimation of Distribution Algorithms

Definition

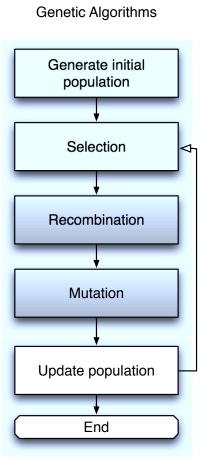
Given set of candidate individuals

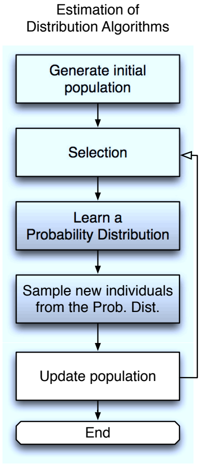
Learn a probability distribution

Sample the probability distribution to obtain the new population

Estimation of Distribution Algorithms

Scheme





Extensively used for a wide variety of problems.

Extensively used for a wide variety of problems.

But not so much on **permutation-based optimization problems**.

Outline

Permutation-based Optimization Problems

Definition

A specific subset of NP-Hard optimization problems.

Permutation-based Optimization Problems

Definition

Problems whose solution can be naturally represented as a **permutation**.

Permutation-based Optimization Problems

What is a permutation?

A permutation π is...

Permutation-based Optimization Problems

What is a permutation?

A permutation π is...

...a bijection of the set $\{1, \ldots, n\}$ into itself, $\pi = (\pi(1), \ldots, \pi(n))$.

Permutation-based Optimization Problems

What is a permutation?

$$\pi = (1, 4, 5, 8, 2, 6, 3, 7) \qquad n = 8$$

Permutation-based Optimization Problems

Examples

Travelling Salesman Problem

Permutation-based Optimization Problems

Examples

Travelling Salesman Problem

Quadratic Assignment Problem

Permutation-based Optimization Problems

Examples

Travelling Salesman Problem

Quadratic Assignment Problem

Linear Ordering Problem

Permutation-based Optimization Problems

Examples

Travelling Salesman Problem

Quadratic Assignment Problem

Linear Ordering Problem

Rank Aggregation Problem

Permutation-based Optimization Problems

Examples

Travelling Salesman Problem

Quadratic Assignment Problem

Linear Ordering Problem

Rank Aggregation Problem

Flowshop Scheduling Problem

Permutation-based Optimization Problems

Examples

Travelling Salesman Problem

Quadratic Assignment Problem

Linear Ordering Problem

Rank Aggregation Problem

Flowshop Scheduling Problem

Let an $n \times n$ matrix $B = [b_{ij}]$ be given.

Permutation-based Optimization Problems

The Linear Ordering Problem (LOP)

Let an $n \times n$ matrix $B = [b_{ij}]$ be given.

Determine a simultaneous permutation of rows and columns of H such that the sum of superdiagonal entries is maximized.

$$f(\pi) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} b_{\pi(i)\pi(j)}$$

Permutation-based Optimization Problems

The Linear Ordering Problem (LOP)

0	16	11	15	7
21	0	14	15	9
26	23	0	26	12
22	22	11	0	13
30	28	25	24	0

(1, 2, 3, 4, 5) f= 138

Permutation-based Optimization Problems

The Linear Ordering Problem (LOP)

0	16	11	15	7
21	0	14	15	9
26	23	0	26	12
22	22	11	0	13
30	28	25	24	0

(1, 2, 3, 4, 5) f= 138

→

0	14	21	15	9
23	0	26	26	12
16	11	0	15	7
22	11	22	0	13
28	25	30	24	0

(2, 3, 1, 4, 5) f= 158

Permutation-based Optimization Problems

The Linear Ordering Problem (LOP)

0	16	11	15	7
21	0	14	15	9
26	23	0	26	12
22	22	11	0	13
30	28	25	24	0

(1, 2, 3, 4, 5) f= 138

→

0	14	21	15	9
23	0	26	26	12
16	11	0	15	7
22	11	22	0	13
28	25	30	24	0

(2, 3, 1, 4, 5) f= 158

→

0	25	24	28	30
12	0	26	23	26
13	11	0	22	22
9	14	15	0	21
7	11	15	16	0

(5, 3, 4, 2, 1) f= 247

Permutation-based Optimization Problems

The Linear Ordering Problem (LOP)

Applications

Aggregation of Individual Preferences

Triangulation of Input-Output tables in economy

Ranking in Sports Tournaments

Scheduling with Precedences

...

Permutation-based Optimization Problems

The Flowshop Scheduling Problem (FSP)

Given a set of n jobs and m machines
and known processing times p_{ij}

Example n=5 & m=4

Permutation-based Optimization Problems

The Flowshop Scheduling Problem (FSP)

Given a set of n jobs and m machines
and known processing times p_{ij}

Find the sequence for scheduling jobs optimally.

Example n=5 & m=4

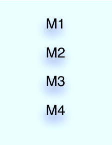
Permutation-based Optimization Problems

The Flowshop Scheduling Problem (FSP)

Given a set of n jobs and m machines
and known processing times p_{ij}

Find the sequence for scheduling jobs optimally.

Example n=5 & m=4



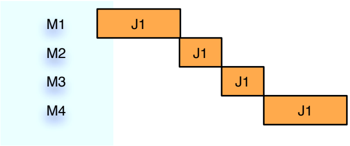
Permutation-based Optimization Problems

The Flowshop Scheduling Problem (FSP)

Given a set of n jobs and m machines
and known processing times p_{ij}

Find the sequence for scheduling jobs optimally.

Example n=5 & m=4



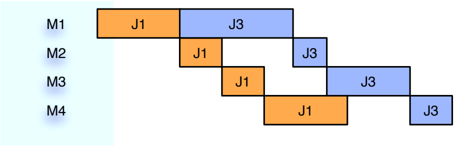
Permutation-based Optimization Problems

The Flowshop Scheduling Problem (FSP)

Given a set of n jobs and m machines
and known processing times p_{ij}

Find the sequence for scheduling jobs optimally.

Example n=5 & m=4



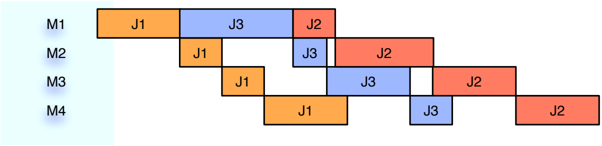
Permutation-based Optimization Problems

The Flowshop Scheduling Problem (FSP)

Given a set of n jobs and m machines
and known processing times p_{ij}

Find the sequence for scheduling jobs optimally.

Example n=5 & m=4



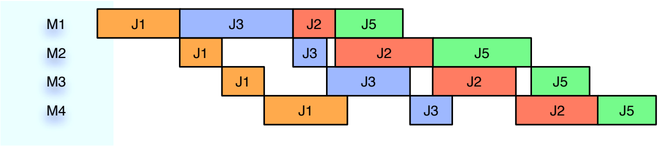
Permutation-based Optimization Problems

The Flowshop Scheduling Problem (FSP)

Given a set of n jobs and m machines
and known processing times p_{ij}

Find the sequence for scheduling jobs optimally.

Example n=5 & m=4



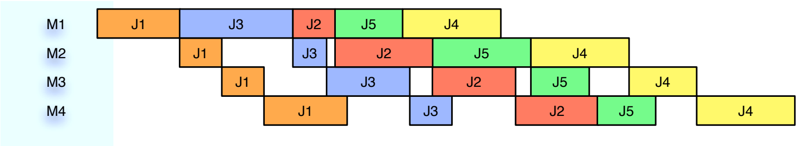
Permutation-based Optimization Problems

The Flowshop Scheduling Problem (FSP)

Given a set of n jobs and m machines
and known processing times p_{ij}

Find the sequence for scheduling jobs optimally.

Example n=5 & m=4



$$\pi = (1, 3, 2, 5, 4)$$

Outline

EDAs for discrete domains:
UMDA, MIMIC, Dependency Trees, EBNA

EDAs for discrete domains:
UMDA, MIMIC, Dependency Trees, EBNA

Learn a probability distribution over a set $\Omega = \Omega_1 \times \Omega_2 \times \cdots \times \Omega_n$ where $\Omega_i = \{1, \ldots, r_i\}$ and $r_i \in \mathbb{N}, i = 1, \ldots, n$.

EDAs for discrete domains:
UMDA, MIMIC, Dependency Trees, EBNA

Learn a probability distribution over a set $\Omega = \Omega_1 \times \Omega_2 \times \cdots \times \Omega_n$ where $\Omega_i = \{1, \ldots, r_i\}$ and $r_i \in \mathbb{N}, i = 1, \ldots, n$.

The sampling does not provide a permutation individual, but an individual in Ω.

EDAs for discrete domains:
UMDA, MIMIC, Dependency Trees, EBNA

Learn a probability distribution over a set $\Omega = \Omega_1 \times \Omega_2 \times \cdots \times \Omega_n$ where $\Omega_i = \{1, \ldots, r_i\}$ and $r_i \in \mathbb{N}, i = 1, \ldots, n$.

Modifying the sampling step (PLS), the learnt relation between variables is denaturalized.

Existing approaches

Adaptations of existing EDAs

Standard representations are not effective due to the **mutual exclusivity constraints** associated with permutations.

EDAs for the real domain:
UMDA$_c$, MIMIC$_c$, EGNA

EDAs for the real domain:
UMDA$_c$, MIMIC$_c$, EGNA

The sampled real vectors are transformed into permutations with the Random Keys strategy.

$$\begin{array}{ccc} (0.30, 0.10, 0.40, 0.20) & & \\ (0.25, 0.14, 0.35, 0.16) & & \\ (0.60, 0.20, 0.80, 0.40) & \rightarrow & (3, 1, 4, 2) \\ (0.33, 0.05, 0.35, 0.29) & & \\ (0.27, 0.15, 0.31, 0.20) & & \\ \ldots & & \end{array}$$

EDAs for the real domain:
UMDA$_c$, MIMIC$_c$, EGNA

The sampled real vectors are transformed into permutations with the Random Keys strategy.

$$\begin{array}{ccc} (0.30, 0.10, 0.40, 0.20) & & \\ (0.25, 0.14, 0.35, 0.16) & & \\ (0.60, 0.20, 0.80, 0.40) & \rightarrow & (3, 1, 4, 2) \\ (0.33, 0.05, 0.35, 0.29) & & \\ (0.27, 0.15, 0.31, 0.20) & & \\ \ldots & & \end{array}$$

However, the high redundancy between the sampled real vectors and the permutations make this approaches very ineffective.

Existing approaches

Models based on low-order marginals

Maintain the information relative of first order marginals, i.e. the probability of item i being at position j.

Maintain the information relative of first order marginals, i.e. the probability of item i being at position j.

Node Histogram-Based Sampling Algorithm (NHBSA)

Edge Histogram-Based Sampling Algorithm (EHBSA)

EDAs	TSP	QAP	LOP	FSP
UMDA	0.507	0.029	0.151	0.053
MIMIC	0.676	0.039	0.149	0.035
EBNA_{BIC}	0.505	0.031	0.150	0.054
TREE	1.255	0.052	0.176	0.060
UMDA_c	1.279	0.211	0.330	0.153
EGNA_{ee}	1.183	0.165	0.311	0.142
IDEA-ICE	1.209	0.080	0.174	0.073
EHBSA_{wt}	**0.003**	0.025	0.137	**0.027**
NHBSA_{wt}	1.068	**0.011**	**0.136**	0.027

Models based on low-order marginals

A Natural Extension - Store higher order marginals

Calculate the probability of a specific set of items $(i_1, \ldots, i_k)$ being at specific positions $(j_1, \ldots, j_k)$.

Models based on low-order marginals

A Natural Extension - Store higher order marginals

Calculate the probability of a specific set of items $(i_1, \ldots, i_k)$ being at specific positions $(j_1, \ldots, j_k)$.

By increasing the order of the marginals, better performance is obtained,

...but the computation is not longer affordable.

Outline

A distance-based exponential model.

A distance-based exponential model.

Analogous over permutations to the Gaussian distribution.

Probability Models on Rankings

The Mallows model - Definition

A distance-based exponential model.

Analogous over permutations to the Gaussian distribution.

Two parameters:

Consensus ranking σ_0

Spread parameter θ

A distance-based exponential model.

Analogous over permutations to the Gaussian distribution.

Two parameters:

- Consensus ranking σ_0
- Spread parameter θ

Kendall-τ distance: counts the number of pairwise disagreements.

Probability Models on Rankings

The Mallows model - Definition

A distance-based exponential model.

Analogous over permutations to the Gaussian distribution.

Two parameters:

Consensus ranking σ_0

Spread parameter θ

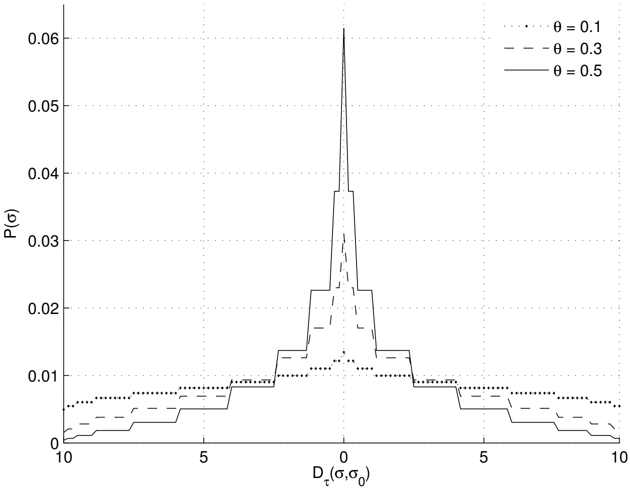
Kendall-τ distance: counts the number of pairwise disagreements.

Probability distribution:

$$P(\sigma) = \frac{1}{\psi(\theta)} e^{-\theta D_\tau(\sigma, \sigma_0)}$$

Probability Models on Rankings

The Mallows model - Definition



A preliminary approach - Flowshop Scheduling Problem

Instance	MaEDA	EHBSA	NHBSA
20×05	0.010	**0.003**	0.006
20×10	0.025	**0.006**	0.007
50×10	0.034	**0.032**	0.033
100×10	**0.008**	0.019	0.015
100×20	0.066	0.067	**0.063**

NHBSA and EHBSA can be outperformed.

Drawbacks:

- Unimodality
- Permutations at the same distance from σ_0 have the same probability

If the distance D can be written as

$$D(\pi, \sigma) = \sum_{j=1}^{n-1} V_j(\pi, \sigma)$$

and the V_j are independent under the uniform distribution

If the distance D can be written as

$$D(\pi, \sigma) = \sum_{j=1}^{n-1} V_j(\pi, \sigma)$$

and the V_j are independent under the uniform distribution, then the Mallows model can be generalized to a n-parameters model

$$P(\sigma) \propto \exp(-\sum_{j=1}^{n-1} \theta_j V_j(\sigma, \sigma_0))$$

If the distance D can be written as

$$D(\pi, \sigma) = \sum_{j=1}^{n-1} V_j(\pi, \sigma)$$

and the V_j are independent under the uniform distribution, then the Mallows model can be generalized to a n-parameters model

$$P(\sigma) \propto \exp(-\sum_{j=1}^{n-1} \theta_j V_j(\sigma, \sigma_0))$$

The Generalized Mallows model.

Outperformed the best performing EDAs.

Probability Models on Rankings

The Generalized Mallows EDA - The Flowshop Scheduling Problem

	AGA	VNS_4	GM-EDA	HGM-EDA
20×05	**13932**	**13932**	13934	**13932**
20×10	**20003**	**20003**	20009	**20003**
20×20	**32911**	**32911**	32920	**32911**
50×05	**66301**	66757	66309	13932
50×10	**85916**	86479	86948	85958
50×20	**121294**	121739	122830	121317
100×05	**240102**	242974	241346	240122
100×10	288988	292425	292472	**288902**
100×20	374974	378402	379691	**374664**
200×10	1039507	1048520	1046146	**1036303**
200×20	1243928	1252165	1252545	**1237959**
250×10	1047364	1055969	1054471	**1043985**
250×20	1249242	1258841	1259104	**1244851**
250×10	1602649	1613663	1610820	**1594830**
250×20	1867750	1879368	1880471	**1859296**
300×10	2248455	2262178	2266665	**2236464**
300×20	2606219	2616542	2618186	**2589509**
350×10	3045116	3060581	3077427	**3026653**
350×20	3472808	3486846	3513912	**3458190**
400×10	3915780	3933989	4000044	**3915542**
400×20	**4435249**	4450237	4584215	4461403
450×10	**4922402**	4943671	5140331	4975776
450×20	**5554795**	5566587	5830506	5618526
500×20	**6754943**	6770472	7225665	6861070

The proposed approach outperformed the state-of-the-art algorithms of the FSP.

The contribution of the Generalized Mallows EDA was essential for its success.

Based on the Luce's Choice Axiom, this model assumes that the ranking process is composed of a sequence of stages, choosing the most preferred item of the remaining items at each stage.

Based on the Luce's Choice Axiom, this model assumes that the ranking process is composed of a sequence of stages, choosing the most preferred item of the remaining items at each stage.

A parameters vector $\boldsymbol{w} = (w_1, w_2, \ldots, w_n)$ defines the preference of each item to be ranked in top rank.

Based on the Luce's Choice Axiom, this model assumes that the ranking process is composed of a sequence of stages, choosing the most preferred item of the remaining items at each stage.

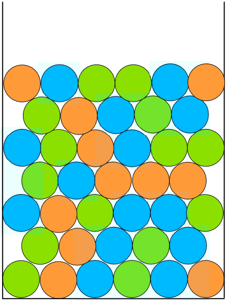
A parameters vector $\boldsymbol{w} = (w_1, w_2, \ldots, w_n)$ defines the preference of each item to be ranked in top rank.

Formally,

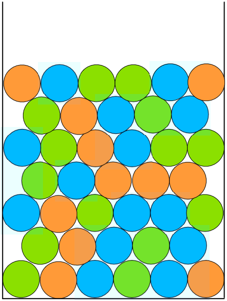
$$P(\pi|\boldsymbol{w}) = \prod_{i=1}^{n-1} \frac{w_{\pi\langle i\rangle}}{\sum_{j=i}^{n} w_{\pi\langle j\rangle}}$$

Probability Models on Rankings

The Plackett-Luce model - Vase model interpretation



A vase with infinite coloured balls.

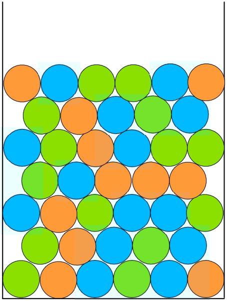


A vase with infinite coloured balls.

With known proportions of each colour.
(w_r, w_g, w_b).

Probability Models on Rankings

The Plackett-Luce model - Vase model interpretation



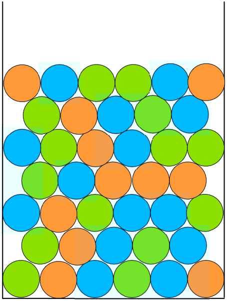
A vase with infinite coloured balls.

With known proportions of each colour.
(w_r, w_g, w_b).

Draw balls from the vase until an ordering of coloured balls is obtained.

Probability Models on Rankings

The Plackett-Luce model - Vase model interpretation



Stage 1

We draw a ball. And it is red.

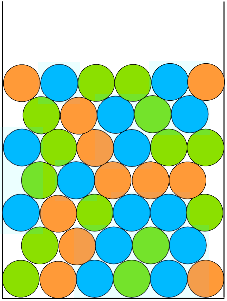
The probability to extract a red ball at this stage is:

$$\frac{w_r}{w_r + w_g + w_b}$$

$\pi =$ 

Probability Models on Rankings

The Plackett-Luce model - Vase model interpretation



Stage 1

We draw a ball. And it is red.

The probability to extract a red ball at this stage is:

$$\frac{w_r}{w_r + w_g + w_b}$$

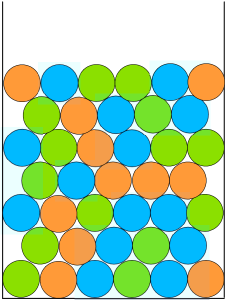
$\pi =$ 

$P(\pi) = \frac{w_r}{w_r+w_g+w_b}$

Probability Models on Rankings

The Plackett-Luce model - Vase model interpretation

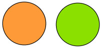
Stage 2



We draw another ball. And it is green.

The probability to extract a green ball from the remaining balls is:

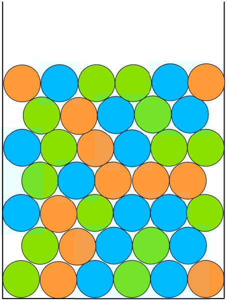
$$\frac{w_g}{w_g + w_b}$$

$\pi =$ 

Probability Models on Rankings

The Plackett-Luce model - Vase model interpretation

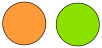
Stage 2



We draw another ball. And it is green.

The probability to extract a green ball from the remaining balls is:

$$\frac{w_g}{w_g + w_b}$$

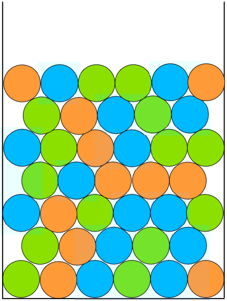
$\pi =$ 

$$P(\pi) = \frac{w_r}{w_r + w_g + w_b} \times \frac{w_g}{w_g + w_b}$$

Probability Models on Rankings

The Plackett-Luce model - Vase model interpretation

Stage 2



We draw the blue ball.

The probability to extract a blue ball is:

$$\frac{w_b}{w_b}$$

$\pi =$ 

$$P(\pi) = \frac{w_r}{w_r + w_g + w_b} \times \frac{w_g}{w_g + w_b} \times \frac{w_b}{w_b}$$

Outline

Some Experiments

The Linear Ordering Problem

Size	Instance	PLEDA	MaEDA	GMEDA	NHBSA	EHBSA
150	t65b11xx	0.056	0.072	0.065	0.064	0.107
	t65d11xx	0.064	0.081	0.074	0.078	0.123
	t65f11xx	0.062	0.081	0.074	0.077	0.121
	t65l11xx	0.041	0.058	0.052	0.041	0.082
	t65n11xx	0.052	0.070	0.065	0.060	0.113
	t70b11xx	0.058	0.074	0.068	0.065	0.109
	t70d11xx	0.067	0.084	0.076	0.076	0.117
	t70f11xx	0.069	0.088	0.078	0.082	0.123
	t70l11xx	0.052	0.074	0.069	0.054	0.102
	t70n11xx	0.056	0.073	0.066	0.061	0.112
	t75d11xx	0.064	0.078	0.073	0.071	0.115
	t75e11xx	0.048	0.069	0.062	0.300	0.300
	t75k11xx	0.055	0.074	0.064	0.292	0.291
	t75n11xx	0.060	0.078	0.071	0.302	0.299
250	t65b11xx	0.074	0.086	0.080	0.088	0.146
	t65d11xx	0.083	0.099	0.090	0.105	0.161
	t65f11xx	0.078	0.094	0.086	0.097	0.159
	t65l11xx	0.061	0.075	0.071	0.068	0.132
	t65n11xx	0.075	0.092	0.084	0.094	0.162
	t70b11xx	0.074	0.088	0.081	0.092	0.157
	t70d11xx	0.080	0.095	0.086	0.097	0.157
	t70f11xx	0.087	0.101	0.094	0.109	0.162
	t70l11xx	0.066	0.082	0.076	0.078	0.150
	t70n11xx	0.074	0.091	0.081	0.094	0.162
	t75d11xx	0.075	0.088	0.080	0.090	0.155
	t75e11xx	0.079	0.094	0.086	0.078	0.143
	t75k11xx	0.079	0.096	0.087	0.082	0.142
	t75n11xx	0.075	0.090	0.082	0.074	0.140

Some Experiments

The Linear Ordering Problem

Size	Instance	PLEDA	MaEDA	GMEDA	NHBSA	EHBSA
150	t65b11xx	**0.056**	0.072	0.065	0.064	0.107
	t65d11xx	**0.064**	0.081	0.074	0.078	0.123
	t65f11xx	**0.062**	0.081	0.074	0.077	0.121
	t65l11xx	**0.041**	0.058	0.052	**0.041**	0.082
	t65n11xx	**0.052**	0.070	0.065	0.060	0.113
	t70b11xx	**0.058**	0.074	0.068	0.065	0.109
	t70d11xx	**0.067**	0.084	0.076	0.076	0.117
	t70f11xx	**0.069**	0.088	0.078	0.082	0.123
	t70l11xx	**0.052**	0.074	0.069	0.054	0.102
	t70n11xx	**0.056**	0.073	0.066	0.061	0.112
	t75d11xx	**0.064**	0.078	0.073	0.071	0.115
	t75e11xx	**0.048**	0.069	0.062	0.300	0.300
	t75k11xx	**0.055**	0.074	0.064	0.292	0.291
	t75n11xx	**0.060**	0.078	0.071	0.302	0.299
250	t65b11xx	**0.074**	0.086	0.080	0.088	0.146
	t65d11xx	**0.083**	0.099	0.090	0.105	0.161
	t65f11xx	**0.078**	0.094	0.086	0.097	0.159
	t65l11xx	**0.061**	0.075	0.071	0.068	0.132
	t65n11xx	**0.075**	0.092	0.084	0.094	0.162
	t70b11xx	**0.074**	0.088	0.081	0.092	0.157
	t70d11xx	**0.080**	0.095	0.086	0.097	0.157
	t70f11xx	**0.087**	0.101	0.094	0.109	0.162
	t70l11xx	**0.066**	0.082	0.076	0.078	0.150
	t70n11xx	**0.074**	0.091	0.081	0.094	0.162
	t75d11xx	**0.075**	0.088	0.080	0.090	0.155
	t75e11xx	0.079	0.094	0.086	**0.078**	0.143
	t75k11xx	**0.079**	0.096	0.087	0.082	0.142
	t75n11xx	0.075	0.090	0.082	**0.074**	0.140

Experimentation

The Flowshop Scheduling Problem

Config.	Instance	PLEDA	MaEDA	GMEDA	NHBSA	EHBSA
100×20	1	0.074	0.045	0.026	0.060	0.062
	2	0.076	0.044	0.026	0.059	0.057
	3	0.075	0.047	0.026	0.057	0.057
	4	0.075	0.048	0.027	0.061	0.060
	5	0.077	0.048	0.026	0.061	0.060
	6	0.080	0.053	0.029	0.061	0.061
	7	0.076	0.044	0.027	0.061	0.062
	8	0.074	0.047	0.025	0.058	0.057
	9	0.071	0.042	0.025	0.058	0.055
	10	0.078	0.052	0.030	0.061	0.059
200×20	1	0.083	0.070	0.027	0.071	0.076
	2	0.085	0.073	0.023	0.074	0.082
	3	0.085	0.081	0.054	0.070	0.070
	4	0.087	0.077	0.024	0.076	0.082
	5	0.083	0.066	0.022	0.075	0.085
	6	0.079	0.050	0.020	0.073	0.084
	7	0.080	0.051	0.019	0.074	0.085
	8	0.085	0.073	0.029	0.074	0.081
	9	0.085	0.062	0.022	0.078	0.086
	10	0.082	0.074	0.022	0.073	0.077
250×20	1	0.082	0.076	0.041	0.075	0.085
	2	0.085	0.081	0.067	0.075	0.082
	3	0.077	0.065	0.024	0.072	0.082
	4	0.084	0.076	0.037	0.076	0.089
	5	0.090	0.085	0.075	0.079	0.086
	6	0.080	0.074	0.037	0.074	0.086
	7	0.078	0.074	0.041	0.070	0.077
	8	0.088	0.080	0.032	0.078	0.085
	9	0.080	0.074	0.023	0.074	0.083
	10	0.085	0.079	0.029	0.076	0.084

Config.	Instance	PLEDA	MaEDA	GMEDA	NHBSA	EHBSA
100×20	1	0.074	0.045	**0.026**	0.060	0.062
	2	0.076	0.044	**0.026**	0.059	0.057
	3	0.075	0.047	**0.026**	0.057	0.057
	4	0.075	0.048	**0.027**	0.061	0.060
	5	0.077	0.048	**0.026**	0.061	0.060
	6	0.080	0.053	**0.029**	0.061	0.061
	7	0.076	0.044	**0.027**	0.061	0.062
	8	0.074	0.047	**0.025**	0.058	0.057
	9	0.071	0.042	**0.025**	0.058	0.055
	10	0.078	0.052	**0.030**	0.061	0.059
200×20	1	0.083	0.070	**0.027**	0.071	0.076
	2	0.085	0.073	**0.023**	0.074	0.082
	3	0.085	0.081	**0.054**	0.070	0.070
	4	0.087	0.077	**0.024**	0.076	0.082
	5	0.083	0.066	**0.022**	0.075	0.085
	6	0.079	0.050	**0.020**	0.073	0.084
	7	0.080	0.051	**0.019**	0.074	0.085
	8	0.085	0.073	**0.029**	0.074	0.081
	9	0.085	0.062	**0.022**	0.078	0.086
	10	0.082	0.074	**0.022**	0.073	0.077
250×20	1	0.082	0.076	**0.041**	0.075	0.085
	2	0.085	0.081	**0.067**	0.075	0.082
	3	0.077	0.065	**0.024**	0.072	0.082
	4	0.084	0.076	**0.037**	0.076	0.089
	5	0.090	0.085	**0.075**	0.079	0.086
	6	0.080	0.074	**0.037**	0.074	0.086
	7	0.078	0.074	**0.041**	0.070	0.077
	8	0.088	0.080	**0.032**	0.078	0.085
	9	0.080	0.074	**0.023**	0.074	0.083
	10	0.085	0.079	**0.029**	0.076	0.084

The PLEDA performs the best for the LOP,
but is the worst for the FSP.

The PLEDA performs the best for the LOP,
but is the worst for the FSP.

The GMEDA performs the best for the FSP,
and is still competitive for LOP.

Outline

1. Estimation of Distribution Algorithms
2. Permutation-based Optimization Problems
3. Previous approaches
4. Probability Models on Rankings
5. Some Experiments
6. Conclusions

Conclusions

Probability Models on Rankings are very powerful over permutation-based optimization problems.

Conclusions

But, their performance in each case varies depending on the problem.

Conclusions

And we haven't got the slightest idea, apart from intuition, of which model is the most appropriate for solving a particular problem.

Study the structure of the problems.

- Which dependencies are relevant in each problem ?
- Can we speak about additively decomposable functions ?
- And the conditional independence ?

Study the structure of the problems.

- Which dependencies are relevant in each problem ?
- Can we speak about additively decomposable functions ?
- And the conditional independence ?

Study the structure of the problems.

- Which dependencies are relevant in each problem ?
- Can we speak about additively decomposable functions ?
- And the conditional independence ?

Study the structure of the problems.

Which dependencies are relevant in each problem ?

Can we speak about additively decomposable functions ?

And the conditional independence ? $\rightarrow$ Riffle Independence?

Study the structure of the problems.

Which dependencies are relevant in each problem ?

Can we speak about additively decomposable functions ?

And the conditional independence ? $\rightarrow$ Riffle Independence?

New properties need to be proposed.

Future work

The Boltzmann Distribution associated to the problem

$$P(\pi) = \frac{e^{-\beta f(\pi)}}{Z_f(\beta)}$$

f: Objective function.

β: Boltzmann constant.

$Z_f(\beta)$: partition function.

Estimation of Distribution Algorithms for Permutation-based Optimization Problems

Josu Ceberio

Intelligent Systems Group
Department of Computer Science and Artificial Intelligence
University of the Basque Country UPV/EHU

NICaiA: Nature Inspired Computation and its Applications Meeting
Donostia, 22-23 April 2013

Parameters Settings

Population size	$10n$
Selection size	n
Offspring size	$10n - 1$
Max. Evaluations	$1000n$

n denotes the size of the problem.

Consensus ranking $\sigma_0 \rightarrow$ Borda algorithm.

Mallows model	Generalized Mallows model
$\sum_{j=1}^{n-1} \bar{V}_j = \frac{n-1}{e^{\theta}-1} - \sum_{j=1}^{n-1} \frac{n-j+1}{e^{(n-j+1)\theta}-1}$	$\bar{V}_j = \frac{n-1}{e^{\theta_j}-1} - \frac{n-j+1}{e^{\theta_j(n-j+1))}-1}$
One spread parameter	$n-1$ spread parameters

Probability of a permutation σ is factorized in $n-1$ independent terms

$$P(\sigma) = \prod_{j=1}^{n-1} P(V_j(\sigma\sigma_0^{-1}) = r_j) = \prod_{j=1}^{n-1} \frac{exp(-\theta_j r_j)}{\psi_j(\theta_j)} \qquad r_j \in \{0, ..., n-j\}$$

The Plackett-Luce model

Learning

Assuming a set $\boldsymbol{\pi} = \{\pi_1, \ldots, \pi_N\}$ and given $\boldsymbol{w} = (w_1, \ldots, w_n)$, the probability to observe the set,

$$P(\boldsymbol{\pi}|\boldsymbol{w}) = \prod_{k=1}^{N} \prod_{i=1}^{n-1} \frac{w_{\pi^k\langle i\rangle}}{\sum_{j=i}^{n} w_{\pi^k\langle j\rangle}}$$

Maximum likelihood estimation of $\boldsymbol{w}$ is given by those parameters that maximize the equation above, or

$$\ell(\boldsymbol{w}) = \sum_{k=1}^{N} \sum_{i=1}^{n-1} \left(\ln w_{\pi^k\langle i\rangle} - \ln \sum_{j=i}^{n} w_{\pi^k\langle j\rangle} \right)$$