Extended Searching Process Analysis

Seminar of machine learning and modeling,
Faculty of Mathematics and Physics, Charles University in Prague

Matej Mojzeš

Supervisor: Jaromír Kukal
Faculty of Nuclear Sciences and Physical Engineering
Czech Technical University in Prague
Department of Software Engineering in Economics
Trojanova 13, 120 00 Prague, Czech Republic
mojzemat@fjfi.cvut.cz

December 20 2012

Searching Problem

- Having an objective function (typically multi-modal one)
- We search for
  - Optimal solution
  - Sub-optimal solution
  - Feasible solution
- In $\mathbf{D} \subset \mathbb{Z}^n, \mathbb{R}^n$
- E.g.:
  - Quadratic Assignment Problem
  - Scheduling Problem
  - Artificial Neural Network learning

Searching Techniques

- Sophisticated – long runs
  - Genetic Optimization,
  - Fast Simulated Annealing,
  - Cuckoo Search, etc.
- Unsophisticated – independent attempts
  - Random shooting
- Slightly sophisticated – short runs
  - Steepest Descent
  - Sophisticated search, but restarted prematurely ... **when**?

Point of Knowing the Restarting Time

- Could be considered useless when not knowing the optimal objective function value of an unknown problem and/or its complexity, but...
- Very useful for tuning of heuristics on
  - Benchmarking tasks
  - Testing tasks
  - Smaller complexity of the optimized problem
- Generally useful when optimizing parameters as function of problem complexity
- Subsequently, by generalization of gained experience, we can run the heuristic on full-complexity problem instance with the best possible configuration

Terminology of Searching Process

- $\mathbf{U}$: non-empty *set of states*
- $\mathbf{G} \subset \mathbf{U}$: non-empty *set of goals*
- $N \in \mathbb{N}$: maximum number of searching steps
- *Searching process* (SP): any algorithm generating the sequence of $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N) \in \mathbf{U}^N$
- *Number of searching steps* (time complexity of SP): $n = \min\{k \in \mathbb{N} | \mathrm{x}_k \in \mathbf{G}\}$, should the search end with a failure $n = +\infty$

Stochastic Search

- $n \sim \{1, 2, \dots, N, +\infty\}$
- $p_n \geq 0$ for $n \leq N$: the probability of finding the solution in n-th step of the SP
- $p_{\mathrm{succ}} = \sum_{n=1}^{N} p_n$ as the probability of *success*
- $p_\infty = 1 - p_{\mathrm{succ}}$ as the probability of *failure*
- We will be studying SP with $p_{\mathrm{succ}} > 0$ only

Time Complexity Measures

- $\mathrm{E}\, n = p_{\mathrm{succ}}^{-1} \sum_{n=1}^{N} n\, p_n$ as *mean number of searching steps* in the case of successful search
- $\sqrt{\mathrm{D}\, n} = p_{\mathrm{succ}}^{-1/2} \left(\sum_{n=1}^{N} (n - \mathrm{E}\, n)^2\, p_n\right)^{1/2}$ as *standard deviation of the searching step number* in the case of successful search
- $FEO = \mathrm{E}\, n / p_{\mathrm{succ}}$ (Feoktistov 2006)[1]
- Also, we could use
  - Logarithmic measures $\mathrm{E} \ln n$, $\sqrt{\mathrm{D} \ln n}$
  - Aggregated measures $F = \mathrm{E} \ln n + \frac{C \cdot \sqrt{6}}{\pi} \cdot \sqrt{\mathrm{D} \ln n} - \ln p_{\mathrm{succ}}$ where $\frac{C \cdot \sqrt{6}}{\pi} \cong 0.4501$ (Mojzeš et al. 2011)[2]

[1] Feoktistov, V.: Differential Evolution: In Search of Solutions. Springer (2006)

[2] Mojzeš, M., Kukal, J., Tran, V.Q., Jablonský, J.: Performance Comparison of Heuristic Algorithms via Multi-criteria Decision Analysis. In: Proc. of Mendel 2011 Soft Computing Conference, pp. 244–251, Brno Univ Technology Press (2011)

Extended Searching Process (XSP)

- If the SP is successful in the first run, then the searching task is done. Otherwise, should the process end with a failure, we continue to repeat new runs until succeeding.
- $(\mathbf{x}'_1, \ldots, \mathbf{x}'_N, \mathbf{x}''_1, \ldots, \mathbf{x}''_N, \ldots)$
- **Axiom 1**: The only one possibility of how to guarantee $p_{\text{succ}} = 1$ is by substituting SP with unconstrained XSP.
- **Axiom 2**: If $p_{\text{succ}} = 1$ the mean value of number of steps is the only one acceptable criterion of SP quality.

Q_∞ Measure

- n^*: length of XSP
- $p^*_n = p^*_{N(k-1)+j} = (1 - p_{\mathrm{succ}})^{k-1} p_j$
- $\mathrm{E}\, n^* = \sum_{n=1}^{\infty} n\, p^*_n = \sum_{k=1}^{\infty} (1 - p_{\mathrm{succ}})^{k-1} \sum_{j=1}^{N} (N(k-1) + j) p_j =$

$$\frac{N\, p_{\mathrm{succ}}\, (1 - p_{\mathrm{succ}})}{p^2_{\mathrm{succ}}} + \frac{p_{\mathrm{succ}}\, \mathrm{E}\, n}{p_{\mathrm{succ}}} = \mathrm{E}\, n + N \cdot \frac{1 - p_{\mathrm{succ}}}{p_{\mathrm{succ}}}$$

- $Q_\infty = \mathrm{E}\, n + N \cdot (p^{-1}_{\mathrm{succ}} - 1)$
- $Q_\infty \geq FEO$

Applications of Q_∞

- Quality measure
- Comparison of heuristics
- Premature termination
- Search for optimal N, motivation:
  - N too low – we could not find solution yet
  - N too high – we should have started new search already

Bayesian Estimation of Q_∞

- M independent runs of SP yield time complexities $n_1, \ldots, n_M \in \{1, \ldots, N, +\infty\}$ of individual runs
- $M^* = \mathrm{card}\{k \mid n_k < +\infty\}$: the number of successful runs
- $\mathrm{E}^* = \frac{1}{M^*} \sum_{n_k < +\infty} n_k$ estimates $\mathrm{E}\, n$
- $\mathrm{D}^* = \frac{1}{M^*-1} \sum_{n_k < +\infty} (n_k - E^*)^2$ estimates $\mathrm{D}\, n$
- Naive approach: $p_{\mathrm{succ}} \approx M^*/M$
- Bayesian approach:
  - $Q^*_\infty = E^* + N \cdot \frac{M-M^*+1}{M^*}$
  - $s^*_\infty = \sqrt{\frac{\mathrm{D}^*}{\mathrm{M}^*} + \frac{N^2}{M^*} \cdot \frac{(M+1)(M-M^*+1)}{(M^*)^2(M^*-1)}}$

Example 1: Optimal N

- Hilbert matrix inversion
- $f(\mathbf{x}) = \left|\left|H^{-1}\mathbf{x}\right|\right|_1$
- $H^{-1} =$

$$\begin{pmatrix} 36 & -630 & 3360 & -7560 & 7560 & -2772 \\ -630 & 14700 & -88200 & 211680 & -220500 & 83160 \\ 3360 & -88200 & 564480 & -1411200 & 1512000 & -582120 \\ -7560 & 211680 & -1411200 & 3628800 & -3969000 & 1552320 \\ 7560 & -220500 & 1512000 & -3969000 & 4410000 & -1746360 \\ -2772 & 83160 & -582120 & 1552320 & -1746360 & 698544 \end{pmatrix}$$

- $\mathbf{x} \in \{-1, 0, 1\}^6$
- Steepest Descent (slightly sophisticated approach)

Results for $M = 1000$

Table : Estimation of Q_∞

N	Q_∞	Q^*_∞	s^*_∞
20	259.621	247.333	3.238
21	258.419	243.554	3.007
22	257.751	**240.148**	**2.808**
23	**257.560**	244.956	2.806
24	257.796	234.255	2.486
25	258.415	236.238	2.440
26	259.381	238.077	2.396
27	260.661	239.785	2.355
28	262.229	237.063	2.241
29	264.059	238.609	2.206
30	266.130	242.103	2.205

Comparison of Two Heuristics on a Single Task

- z-score technique
- $z = \frac{|Q^*_{\infty,\mathrm{A}} - Q^*_{\infty,\mathrm{B}}|}{\sqrt{(s^*_{\infty,\mathrm{A}})^2 + (s^*_{\infty,\mathrm{B}})^2}}$
- $p_{\mathrm{value}} = 2 - 2\Phi(z)$
- For more than two heuristics
  - Multiple testing
  - False Discovery Rate
  - H heuristic instances $\Rightarrow H \cdot (H-1)/2$ pair tests

Additivity Principle for More Tasks

- We suppose battery of B tasks
- $Q_{\infty,\mathrm{T}} = \sum_{k=1}^{B} Q_{\infty,k}$
- $Q^*_{\infty,\mathrm{T}} = \sum_{k=1}^{B} Q^*_{\infty,k}$
- $s^*_{\infty,\mathrm{T}} = \sqrt{\sum_{k=1}^{B} (s^*_{\infty,k})^2}$
- Applications
  - Pair comparison of heuristics
  - Multiple comparison of heuristics

Example 2: Heuristics Comparison on a Battery of Tasks

	PSO			FF			CS		
Task	E^*	r	s	E^*	r	s	E^*	r	s
Michalewicz	6922	0.98	537	3752	0.99	725	3221	1.00	519
Rosenbrock	32756	0.98	5325	7792	0.99	2923	5923	1.00	1937
De Jong	17040	1.00	1123	7217	1.00	730	4971	1.00	754
Ackley	23407	0.92	4325	5293	1.00	4920	4936	1.00	903
Rastrigin	79491	0.90	3715	15573	1.00	4399	10354	1.00	3755

- Basic statistics of PSO, FF and CS from (Yang and Deb 2009)[3], (Yang 2009)[4]

[3] Yang, X.-S., Deb, S.: Cuckoo search via Lévy flights. In: Proc. of World Congress on Nature & Biologically Inspired Computing, pp. 210–214, IEEE Publications (2009)

[4] Yang, X.-S.: Firefly algorithms for multimodal optimization. In: Stochastic Algorithms: Foundations and Applications, SAGA, 169–178 (2009)

Example 2: Heuristics Comparison on a Battery of Tasks

Table : Additive quality measures of PSO, FF and CS

	PSO		FF		CS	
Task	Q^*_∞	s^*_∞	Q^*_∞	s^*_∞	Q^*_∞	s^*_∞
Michalewicz	9983.224	190.083	5772.202	162.949	**4221.000**	**113.559**
Rosenbrock	35817.224	567.919	9812.202	327.941	**6923.000**	**218.453**
De Jong	18040.000	151.041	8217.000	**124.624**	**5971.000**	126.044
Ackley	33189.609	575.849	6293.000	502.261	**5936.000**	**135.485**
Rastrigin	91713.222	569.722	16573.000	451.347	**11354.000**	**388.847**
TOTAL	188743.280	1018.657	46667.404	778.209	**34405.000**	**496.047**

$$p_{\text{value}}(\mathrm{PSO},\mathrm{FF}) = 2.95 \times 10^{-2670}$$
$$p_{\text{value}}(\mathrm{PSO},\mathrm{CS}) = 2.95 \times 10^{-4032}$$
$$p_{\text{value}}(\mathrm{FF},\mathrm{CS}) = 2.75 \times 10^{-40}$$

Optimal Restarting of Published Heuristics

- We (may) have E^*, r, s
- Bayesian estimate
  - $p_{\text{succ}} = \frac{M^*+1}{M+2}$
  - Other two parameters
- $N_{\text{opt}} = ?$
- $Q_{\text{opt}} = ?$
- Distribution of n?

Typical Distributions

- Log-normal $\mathrm{F}(n) = \Phi\left(\frac{\ln n - \mu}{\sigma}\right)$
- Gamma $\mathrm{F}(n) = \int_0^n \frac{x^{k-1}\exp(-x/T)}{\Gamma(k)T^k}\mathrm{d}x$
- Weibull $\mathrm{F}(n) = 1 - \exp(-(n/T)^k)$
- Unknown parameters are estimated from E^* and s via moment method

Parametric Interruption of PSO

	N_{opt}			
Task	Log-normal	Gamma	Weibull	Δ_{rel}
Michalewicz	8797	8744	**8174**	7.12%
Rosenbrock	50846	50078	**45817**	10.04%
De Jong	21430	21300	**19785**	7.72%
Ackley	35379	35091	**33105**	6.48%
Rastrigin	90973	90807	**87453**	3.88%

Parametric Interruption of FF

	N_{opt}			
Task	Log-normal	Gamma	Weibull	Δ_{rel}
Michalewicz	6353	6220	**5624**	11.72%
Rosenbrock	17518	17255	**16176**	7.78%
De Jong	10074	9955	**9050**	10.29%
Ackley	**10941**	13703	14322	24.67%
Rastrigin	32975	31809	**28401**	14.38%

Parametric Interruption of CS

	N_{opt}			
Task	Log-normal	Gamma	Weibull	drel
Michalewicz	5269	5153	**4592**	13.14%
Rosenbrock	13515	13039	**11756**	13.49%
De Jong	7943	7779	**6946**	12.82%
Ackley	8509	8291	**7365**	13.80%
Rastrigin	24892	24079	**21928**	12.31%

Conclusions

- Q^*_∞ can examine performance of a given heuristic algorithm on a given task
- Via using own experimental data or results published in papers by other authors
- Knowledge of E^* and reliability – we may compare Q^*_∞, but not in the statistical sense
- Moreover, knowing standard deviation, we can
  - Test Q^*_∞ values statistically
  - Estimate N_{opt} minimizing Q_∞ (being aware of the imminent sensitivity to selection of a parametric model)