Experimental Study of Automated Offline Parameter Tuning on the Example of irace and the Traveling Salesman Problem

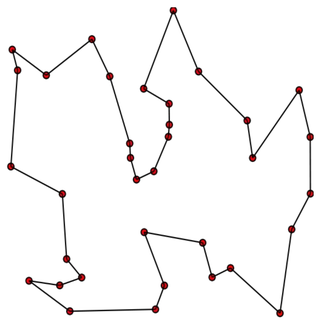
Daniil Chivilikhin
Computer Technologies Lab
ITMO University, St. Petersburg, Russia

ITMO UNIVERSITY

This work was financially supported by the Government of Russian Federation, Grant 074-U01, and also partially by RFBR, research project No. 14-01-00551 a.

Problem: How does algorithm tuning efficiency depend on tuning time?

Traveling Salesman Problem

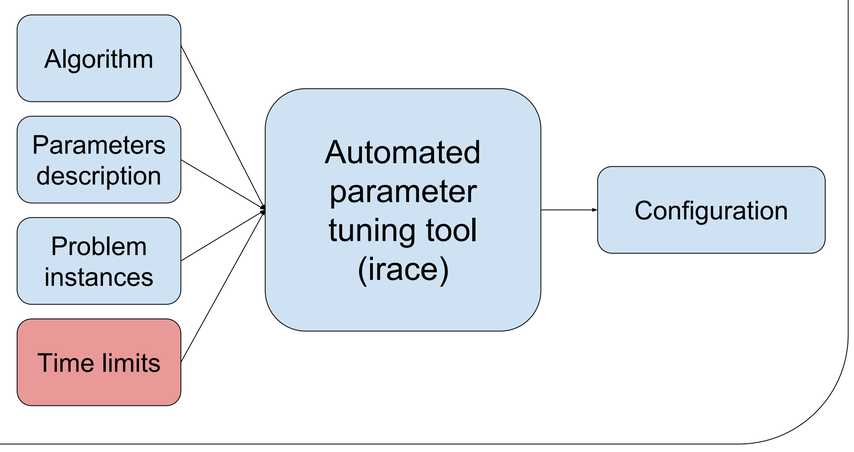


ACOTSP: Ant colony based TSP solver

Table 1: ACOTSP parameters description

Name	Description	Type	Possible values	Condition
algorithm	ACO algorithm	categorical	as, mmas, eas, ras, acs	–
localsearch	local search type	categorical	0, 1, 2, 3	–
alpha	influence of pheromone trails	real	[0, 5]	–
beta	influence of heuristic information	real	[0, 10]	–
rho	pheromone evaporation rate	real	[0.01, 1]	–
ants	number of ants	integer	[1, 100]	–
nnls	number of nearest neighbors in local search	integer	[5, 50]	localsearch $\in \{1, 2, 3\}$
dlb	use don't look bits in local search	categorical	0, 1	localsearch $\in \{1, 2, 3\}$
q0	probability of best choice in tour construction	real	[0, 1]	algorithm $\in \{acs, mmas\}$
rasrank	number of ranks in RAS	integer	[1, 100]	algorithm $\in \{ras\}$
elitistants	number of elitist ants in EAS	integer	[1, 750]	algorithm $\in \{eas\}$

Automated Parameter Tuning



Research Questions

RQ1

Given fixed time limits for both ACOTSP (t) and irace (T), how stable is the performance of irace in terms of the effectiveness of the tuned ACOTSP on the test set? That is, how different across different irace runs will the performance of parameterized ACOTSP be?

RQ2

How does the performance of irace depend on time limits of irace and ACOTSP?

Experimental setup

- ACLIB is used (http://www.aclib.net)
- tsp-rue-1000-3000 symmetrical TSP dataset
  - Training set: 150 instances
  - Test set: 150 instances
  - Used Concorde for optimal solutions
- Measure how close ACOTSP gets to optimal solution

$$r_{tT}^{ijk} = \frac{l_{\text{Concorde}}^{j}}{l_{\text{ACOTSP}(C_i^{tT})}^{jk}}$$

$$R_{tT}^{ik} = \frac{1}{|I_{\text{test}}|} \sum_{j=0}^{|I_{\text{test}}|-1} r_{tT}^{ijk}$$

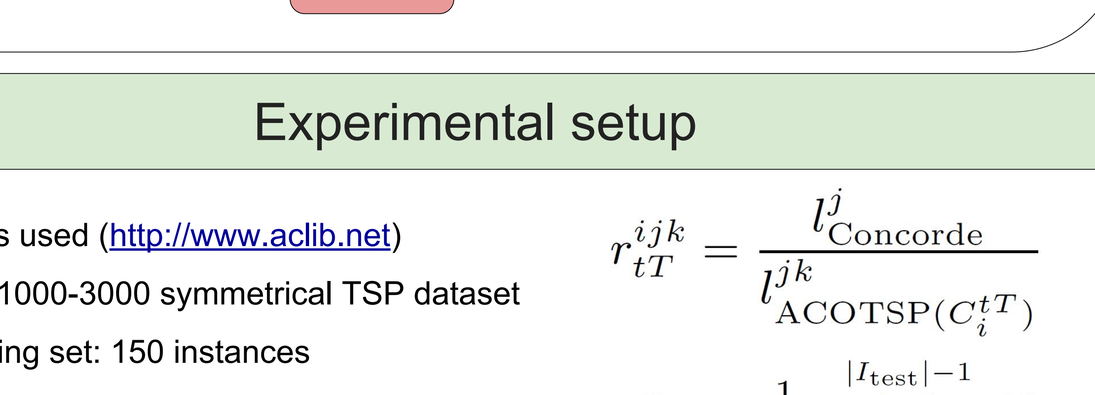
- i – configuration run
- j – instance
- k – test tun

RQ1

- irace limit T = 36 hours
- ACOTSP limit t = 30 seconds
- 20 configurations for each t and T
- 20 test runs for each configuration

Table 2: Statistical data on RQ1 configuration phase

Parameter	min	max	mean	std	median
alpha	1.24	4.75	3.38	1.01	3.53
beta	0.65	9.92	5.22	2.65	5.25
rho	0.08	0.98	0.52	0.26	0.55
ants	25	71	57	10	58
nnls	8	18	13	3	13
q0	0.08	0.93	0.53	0.24	0.55



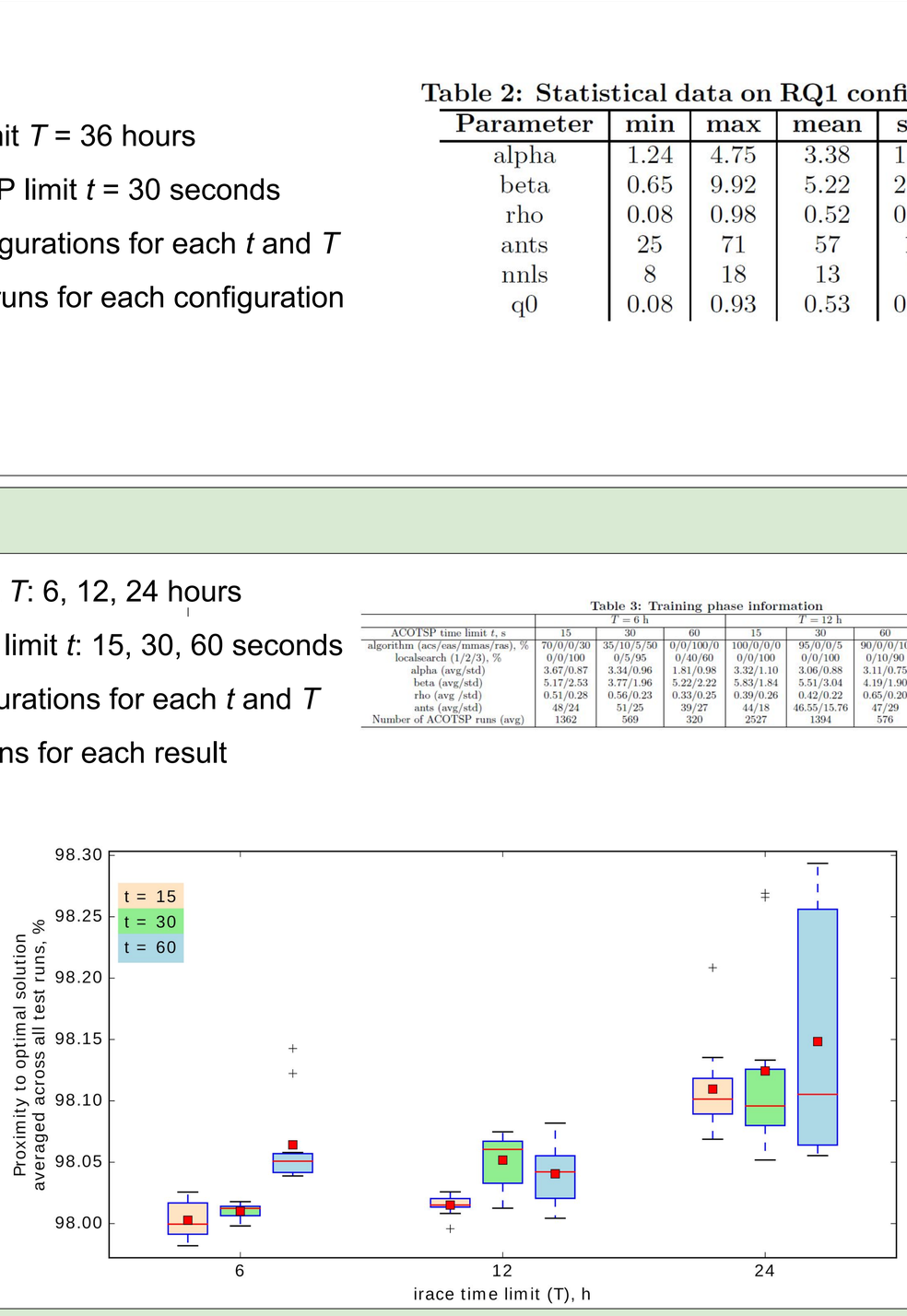
Boxplots of average tour length ratios for t = 30 s and T = 36 h

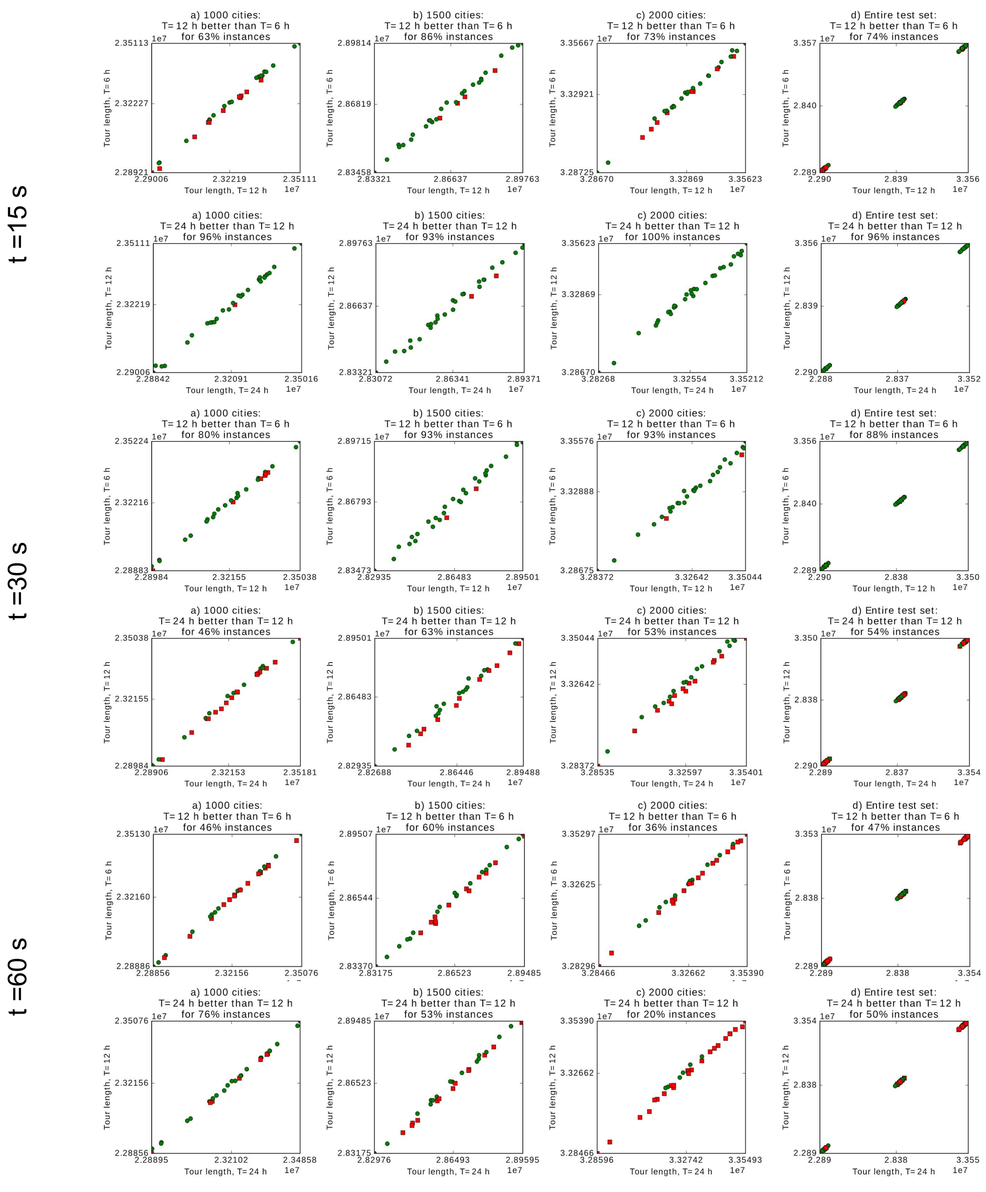
RQ2

- irace limit T: 6, 12, 24 hours
- ACOTSP limit t: 15, 30, 60 seconds
- 10 configurations for each t and T
- 10 test runs for each result

Table 3: Training phase information

ACOTSP time limit t, s	T = 6 h			T = 12 h			T = 24 h		
	15	30	60	15	30	60	15	30	60
algorithm (acs/eas/mmas/ras), %	70/0/0/30	35/10/5/50	0/0/100/0	100/0/0/0	95/0/0/5	90/0/0/10	100/0/0/0	95/0/0/5	60/0/20/20
localsearch (1/2/3), %	0/0/100	0/5/95	0/40/60	0/0/100	0/0/100	0/10/90	0/0/100	0/0/100	0/0/100
alpha (avg/std)	3.67/0.87	3.34/0.96	1.81/0.98	3.32/1.10	3.06/0.88	3.11/0.75	2.8/0.62	2.51/0.77	3.17/1.26
beta (avg/std)	5.17/2.53	3.77/1.96	5.22/2.22	5.83/1.84	5.51/3.04	4.19/1.90	8.31/1.07	5.63/2.77	4.28/3.31
rho (avg /std)	0.51/0.28	0.56/0.23	0.33/0.25	0.39/0.26	0.42/0.22	0.65/0.20	0.32/0.28	0.47/0.28	0.38/0.19
ants (avg/std)	48/24	51/25	39/27	44/18	46.55/15.76	47/29	38/16	41/18	63/26
Number of ACOTSP runs (avg)	1362	569	320	2527	1394	576	5487	2521	1291





Conclusion

- RQ1: irace yields a stable performance (in terms of accuracy on the training set) across several independent tuning runs.
- RQ2: for a fixed ACOTSP time limit t, the test set performance has a non-decreasing behavior with the increase of irace time limit T.