

Figura 1: Desvio relativo médio da busca tabu em função do tenure. (a) Sem randomização. (b) Com randomização.

Soluções 2

Exercício 1 (Algoritmos gulosos, 2pt)

Quais dos seguintes sistemas de conjuntos é independente? Quais satisfazem a propriedade de troca? Prova as afirmações.

- a) Caso um conjunto de intervalos é livre de interseções, então qualquer subconjunto também é, logo é sistema é independente. O exemplo $S = \{[0, 2], [3, 5]\}$ e $T = \{[1, 4]\}$ mostra que a propriedade de troca não é satisfeita.
- b) O sistema é trivialmente independente, porque contém todos subconjuntos de V . Ele também satisfaz a propriedade de troca, por conter todos subconjuntos: caso $|S| > |T|$ existe um elemento em $x \in S \setminus T$ e $T \cup \{x\} \in \mathcal{V}$ para qualquer x .

Exercício 2 (Busca Tabu para QBP, 3pt)

- a) Um exemplo de uma implementação é <http://www.inf.ufrgs.br/~mrpritt/msc/bqp.{hpp, cpp}>.
- b) A figura 1(a) mostra os desvios relativos médias em função do tabu tenure. Aparentemente o desempenho da busca tabu cresce com o tenure, com um tenure de $n/25$ sendo melhor.
- c) A randomização melhora o desempenho da busca tabu. A (leve) tendência de produzir resultados melhores com uma tenure base maior se mantém. melhor

Exercício 3 (Heurística construtiva para QBP, 2pt)

- a) Um exemplo de uma implementação é <http://www.inf.ufrgs.br/~mrpritt/msc/bqp.{hpp, cpp}>.

- b) A figura 2 mostra os histogramas dos desvios relativos de valor ótimo das seis instâncias. Com exceção da instância bgp50, temos uma clara tendência para melhores soluções com menos variação com α crescente. Para $\alpha = 1$, obtemos poucas soluções gulosas fixas, pior que os melhores valores encontrados com $\alpha = 0.8$. Para bqp50 uma construção gulosa reduz o número soluções para três bandas finas.

Exercício 4 (Busca dispersa para QBP, 3pt)

Um exemplo de uma implementação é <http://www.inf.ufrgs.br/~mrpritt/msc/scatterserch.tgz>.

- a) Na Tabela 1 vemos os resultados da busca dispersa para os diferentes parâmetros b_1 e b_2 . Mostramos a média do tempo de execução em segundos ($t(s)$), do número de iterações($iters$) e do desvio percentual da melhor solução conhecida ($dev\%$) das duas instâncias de cada tamanho T ; $bqpT-x$, com $x \in \{1, 7\}$. A busca dispersa encontra o melhor valor conhecido em todos casos para a maior parte dos parâmetros, portanto utilizamos o tempo de execução para escolher o melhor conjunto de parâmetros. Podemos ver que a configuração $b_1 = 18$ e $b_2 = 2$ sempre encontrou a melhor solução conhecida e foi a mais rápida para todos os casos exceto para instâncias de tamanho 50, onde ela foi, em média, dez milissegundos mais lenta. Notamos a influência dos parâmetros no número de iterações somente nas instâncias de tamanho 500, onde vemos que um número maior de iterações nas configurações com b_1 muito pequeno.

Para cada instância e configuração de parâmetros calculamos o desvio percentual do tempo de execução em relação ao tempo de execução do conjunto de parâmetros mais rápido para a dada instância. Na Tabela 2 vemos a média, sobre todas as instâncias, desse desvio chamado de $dev\%$, que também indica que a configuração $b_1 = 18$ e $b_2 = 2$ é a melhor opção segundo os experimentos.

- b) Utilizando os parâmetros $b_1 = 18$ e $b_2 = 2$ comparamos a busca dispersa com a etapa da busca local ativada e desativada; a Tabela 3 mostra os valores $t(s)$, $iter$ e $dev\%$ para os dois casos, para cada instância $bqpT-x$. Podemos ver a importância da busca local na qualidade da solução encontrada, pois para os casos de tamanho maior que 100 não conseguimos mais encontrar a melhor solução conhecida sem a busca local. Sem a busca local precisamos de mais tempo e de mais iterações para terminar a busca mesmo encontrando resultados de pior qualidade, portanto devemos sim manter a etapa de busca local.

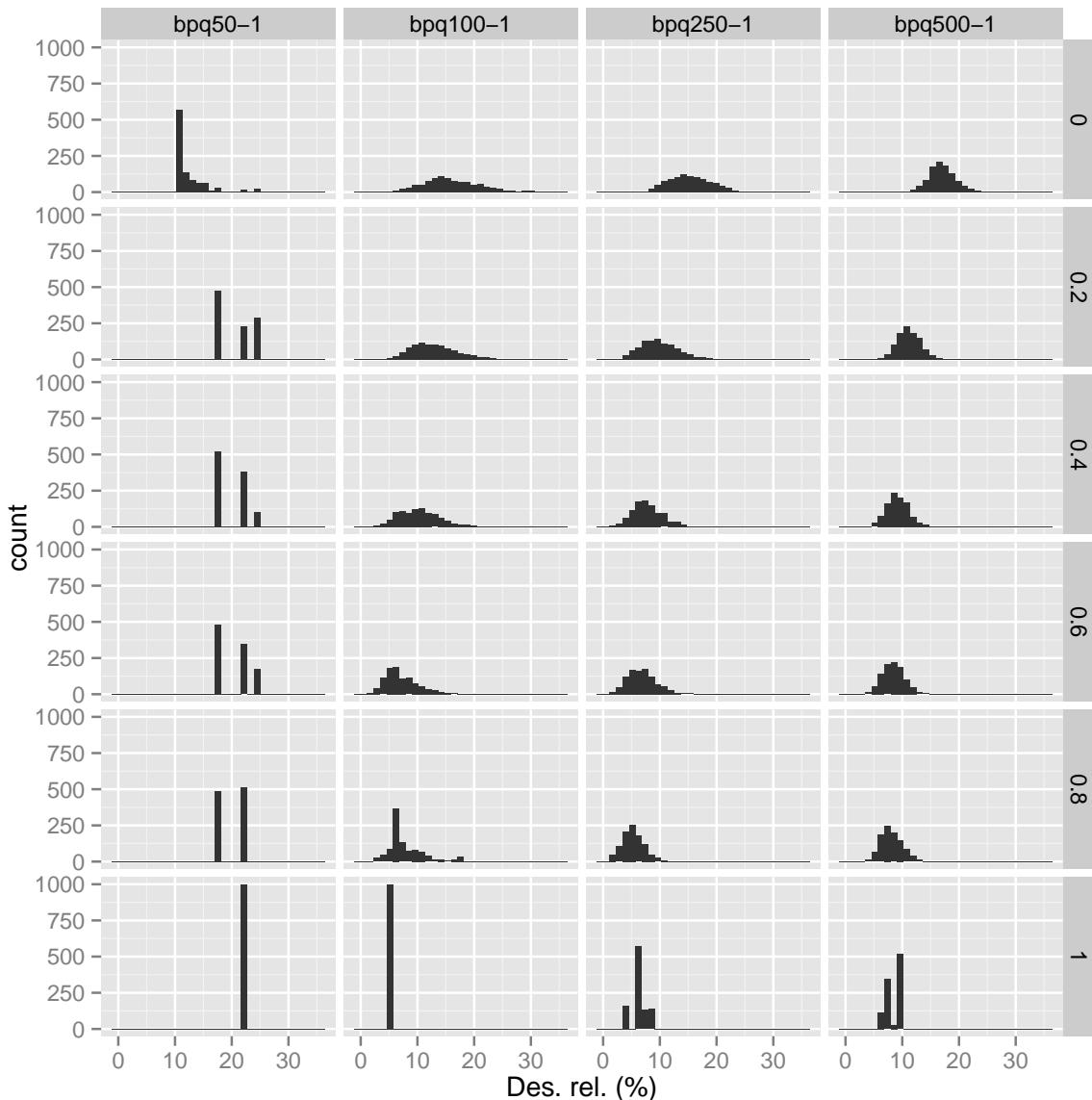


Figura 2: Histograma do desvio relativo da construção semi-gulosa em função da instância e do valor de α .

Tabela 1: Resultados da calibração de b_1 e b_2 .

T		50			100			250			500		
b_1	b_2	t(s)	iters	dev%									
2	18	0.03	1.0	0.00	0.12	2.0	0.00	0.82	1.5	0.00	20.32	39.0	0.07
4	16	0.03	1.0	0.00	0.10	1.5	0.00	0.75	1.0	0.00	11.80	15.5	0.01
6	14	0.03	1.0	0.00	0.11	2.0	0.00	0.78	1.5	0.00	8.14	3.0	0.00
8	12	0.03	1.0	0.00	0.10	1.5	0.00	0.79	1.5	0.00	7.11	2.0	0.00
10	10	0.03	1.0	0.00	0.11	2.0	0.00	0.79	1.5	0.00	7.97	3.5	0.00
12	8	0.03	1.0	0.00	0.10	1.5	0.00	0.76	1.5	0.00	9.04	12.0	0.00
14	6	0.02	1.0	0.00	0.10	2.0	0.00	0.74	1.5	0.00	7.51	3.5	0.00
16	4	0.02	1.0	0.00	0.09	1.5	0.00	0.76	2.0	0.00	7.71	12.0	0.00
18	2	0.03	1.0	0.00	0.08	1.5	0.00	0.71	1.5	0.00	6.67	2.5	0.00

t(s) é o tempo de execução médio em segundos.

iters é o número médio de iterações do algoritmo.

dev% é o desvio percentual médio da melhor solução conhecida.

Tabela 2: Desvio percentual do tempo de execução.

b1	b2	dev t%
2	18	81.01
4	16	38.40
6	14	29.34
8	12	20.58
10	10	26.54
12	8	24.84
14	6	13.93
16	4	13.36
18	2	8.09

Tabela 3: Avaliação da busca local.

T	x	Ativada			Desativada		
		t(s)	iter	dev%	t(s)	iter	dev%
50	1	0.03	1	0.00	0.02	2	0.00
50	7	0.02	1	0.00	0.02	2	0.00
100	1	0.09	2	0.00	0.11	5	0.00
100	7	0.08	1	0.00	0.09	4	0.00
250	1	0.71	1	0.00	0.87	13	0.34
250	7	0.77	2	0.00	0.89	13	0.05
500	1	6.62	2	0.00	8.99	12	0.85
500	7	6.68	3	0.00	8.02	15	0.67