

# Um Estudo Empírico de Hiper-Heurísticas

Igor Ribeiro Sucupira

Flávio Soares Corrêa da Silva (Orientador)

Instituto de Matemática e Estatística  
Universidade de São Paulo

Julho de 2007

## Definição

- ▶ Uma hiper-heurística é uma heurística que pode ser utilizada para resolver qualquer problema de otimização, sem que alterações em seu código fonte sejam necessárias.
- ▶ Portanto, a hiper-heurística não pode conter informações específicas sobre um problema e deve receber esse tipo de informação como parâmetro.
- ▶ Em geral, os principais parâmetros recebidos pelas hiper-heurísticas são funções chamadas *heurísticas de baixo nível*. Cada uma dessas funções é capaz de receber uma solução a partir de outra.
- ▶ Por isso, é comum as hiper-heurísticas serem definidas como “heurísticas que gerenciam heurísticas”.

## Por Que Desenvolver Hiper-Heurísticas?

- ▶ Resultados já existentes sugerem que, com o uso de hiper-heurísticas, podem-se desenvolver rapidamente algoritmos de qualidade razoável para lidar com problemas de otimização pouco conhecidos.

# Uma Hiper-Heurística de Busca por Entornos (*HHES*)

- ▶ Hiper-heurística de E. Soubeiga, desenvolvida em 2000.
- ▶ Recebe como parâmetros:
  - ▶ Uma solução qualquer para o problema considerado.
  - ▶ Uma função capaz de avaliar soluções.
  - ▶ Um conjunto de heurísticas de baixo nível.
- ▶ Baseia-se em uma *função de decisão*: em cada iteração, escolhe uma heurística e a aplica à solução corrente, produzindo a nova solução corrente.

# Uma Hiper-Heurística de Busca por Entornos (*HHES*)

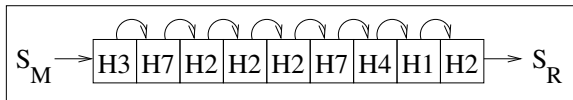
- ▶ Escolha baseada em:
  - ▶ Desempenho de cada heurística desde o início da execução da hiper-heurística: capacidade de melhorar soluções e tempo consumido.
  - ▶ Desempenho de cada seqüência de duas heurísticas.
  - ▶ Tempo decorrido desde a última execução de cada heurística.
- ▶ (Função de decisão) Para cada heurística  $H_i$ :

$$f(H_i) = \alpha f_1[H_i] + \beta f_2[H_i, H_j] + \gamma f_t[H_i]$$

- ▶  $\alpha$ ,  $\beta$  e  $\gamma$  são ajustados dinamicamente.

## Um Algoritmo Genético Hiper-Heurístico (*HHAG*)

- ▶ Desenvolvido em 2002, por P. Cowling et al.
- ▶ Recebe os mesmos parâmetros que *HHES*.
- ▶ Cada cromossomo é uma seqüência de heurísticas.
- ▶ Aptidão do cromossomo é medida aplicando-se sua seqüência de heurísticas à melhor solução já encontrada.
- ▶ Na aptidão, também se leva em conta o comprimento do cromossomo e o tempo necessário para avaliá-lo.



$$A = \frac{c(S_M) - c(S_R)}{9T}$$

# Um Algoritmo Genético Hiper-Heurístico (*HHAG*)

- ▶ Operadores de cruzamento (probabilidade  $\frac{1}{2}$  para cada):
  - ▶ Cruzamento de ponto único.
  - ▶ Selecionar a melhor subsequência de heurísticas adjacentes de cada cromossomo e trocar essas subsequências.
- ▶ Operadores de mutação:
  - ▶ (Mutação clássica) Alterar cada gene, com baixa probabilidade, substituindo-o por uma nova heurística, aleatoriamente escolhida.
  - ▶ Remover do cromossomo a pior subsequência de heurísticas adjacentes (efetuada se o cromossomo for maior que a média).
  - ▶ Selecionar a melhor subsequência de outro cromossomo e inseri-la em uma posição aleatória no cromossomo que sofre mutação (efetuada se o cromossomo for menor que a média).

## Avaliações das Hiper-Heurísticas

- ▶ Diversas hiper-heurísticas já foram implementadas, desde 2000, com resultados aparentemente bons.
- ▶ Porém, avaliações foram efetuadas de forma bem simples:
  - ▶ Na maioria dos casos, apenas um problema.
  - ▶ Sempre poucas instâncias.
  - ▶ Não há comparações com resultados externos.
- ▶ Procuramos, neste trabalho, avaliar hiper-heurísticas através de sua aplicação a 2 problemas distintos, para os quais existem conjuntos de instâncias já experimentadas por bons algoritmos.

## Implementações Realizadas

- ▶ Duas novas hiper-heurísticas baseadas em meta-heurísticas evolutivas (caminho mais claro para exploração).
- ▶ Uma hiper-heurística “minimal”.
- ▶ Implementamos as duas hiper-heurísticas que apresentamos anteriormente e experimentamos algumas alterações.
- ▶ Todas as hiper-heurísticas foram implementadas como bibliotecas e enxergam os parâmetros como “caixas-pretas”.
- ▶ Cada solução é uma estrutura composta por:
  - ▶ Um *double*: o custo da solução.
  - ▶ Um apontador para *void*: a solução propriamente dita.
- ▶ Com isso, não é necessário receber uma função de avaliação.

## Experimentos com a Hiper-Heurística de Soubeiga (*HHES*)

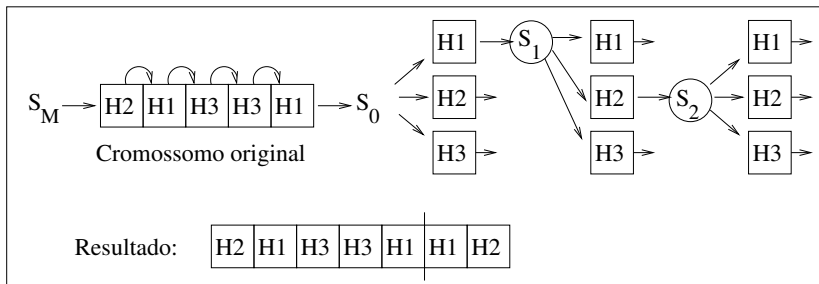
- ▶ *HHESv2* e *HHESv3*: utilizar matrizes de dimensões 3 e 4 para armazenar informações sobre o desempenho de seqüências de heurísticas (sugestão de Soubeiga em sua tese).

$$f(H_i) = \alpha f_1[H_i] + \beta f_2[H_i, H_j] + \gamma f_3[H_i, H_j, H_k] + \delta f_4[H_i, H_j, H_k, H_l] + \psi f_t[H_i]$$

- ▶ *HHESv4*: desconsiderar o tempo de execução ao avaliar as heurísticas de baixo nível (que na prática são operadores simples/rápidos).

## Experimentos com o Alg. Genét. Hiper-Heurístico (*HHAG*)

- ▶ *HHAGv2*: com um novo operador de mutação, baseado em otimização local gulosa:



## Experimentos com o Alg. Genét. Hiper-Heurístico (*HHAG*)

- ▶ *HHAGv3*: no cálculo de aptidão, tratamos de maneira diferente os cromossomos com aptidão negativa:

$$A = -|d| C T$$

(nos outros casos,  $A = \frac{d}{C T}$ )

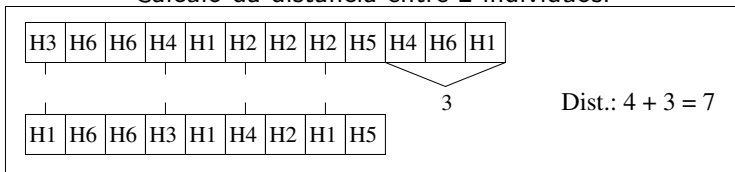
- ▶ Assim, a aptidão diminui à medida em que  $C$  e  $T$  aumentam, em todos os casos.
- ▶ *HHAGv4*: comprimento do cromossomo deixa de ser usado no cálculo da aptidão.
  - ▶ Medida do consumo de tempo é mais importante e já cumpre o papel de evitar o crescimento exagerado dos cromossomos.

# Uma Hiper-Heurística de Reconexão de Caminhos (*HHRC*)

- ▶ Cada indivíduo é uma seqüência de heurísticas.
- ▶ Avaliação de um indivíduo depende tanto do resultado de sua aplicação à melhor solução quanto do consumo de tempo desta aplicação.
- ▶ Conjunto de referência ( $C_R$ ) deve possuir poucos indivíduos, bem avaliados, que sejam bastante distintos entre si.
- ▶ Construção de  $C_R$ , no início do algoritmo, baseada em:
  - ▶ Geração de indivíduos aleatoriamente.
  - ▶ Extensão desses indivíduos com o novo operador de mutação.
  - ▶ Seleção de indivíduos bem avaliados e distintos para composição de  $C_R$ .

# Uma Hiper-Heurística de Reconexão de Caminhos (*HHRC*)

Cálculo da distância entre 2 indivíduos:



- ▶ Em cada iteração da hiper-heurística, combina-se cada indivíduo de  $C_R$  com cada um dos demais (reconexão), produzindo-se 6 indivíduos para cada par.

## Uma Hiper-Heurística de Reconexão de Caminhos (*HHRC*)

- ▶ Os 6 indivíduos produzidos na reconexão são estendidos.
- ▶ Cada novo indivíduo será incluído em  $C_R$  (no lugar de seu pior indivíduo) caso seja melhor que algum indivíduo de  $C_R$ .
- ▶ Sempre que um indivíduo for incluído em  $C_R$ , escolheremos um dos atuais indivíduos de  $C_R$  para ter seu tamanho reduzido.

## Um Algoritmo Hiper-Heurístico de Estimação de Distribuição (*HHED*)

- ▶ Cada indivíduo é uma seqüência de heurísticas e é avaliado como no *HHRC*.
- ▶ População possui 35 indivíduos.
- ▶ População inicial é gerada aleatoriamente e, em seguida, seus indivíduos são estendidos.
- ▶ Cada iteração consiste em criar uma nova população aleatoriamente e estender seus indivíduos.
- ▶ Probabilidades são redefinidas através da análise dos 11 melhores indivíduos da população.

## Uma Hiper-Heurística Simples (*HHDS*)

- ▶ Cada iteração seleciona uma heurística aleatoriamente e a aplica à solução corrente.
- ▶ Heurísticas com melhor desempenho histórico têm mais chances de serem selecionadas.
- ▶ Atualização da medida de desempenho de cada heurística  $H_i$  sempre que ela for aplicada a uma solução  $S_A$ , produzindo uma solução  $S_B$ :

$$A[H_i] = (1 - \alpha) A[H_i] + \alpha (c(S_A) - c(S_B)).$$

## Aplicação ao Problema do Desafio ROADEF 2005

- ▶ Competição aberta ao público.
- ▶ Problema de escalonamento proposto pela Renault.
- ▶ Participantes tiveram um ano para desenvolver algoritmos.
- ▶ Nos experimentos, empregamos as 16 instâncias de conjunto A, disponível desde o início da competição.
- ▶ Como referência para os resultados, adotamos:
  - ▶ Os resultados obtidos, na etapa final, pelo algoritmo vencedor.
  - ▶ Os resultados obtidos, na etapa final, pelo algoritmo que ficou em último (18º) lugar nessa etapa.  
(A etapa inicial da competição contou com 55 times)
- ▶ 4 heurísticas de baixo nível para cada um dos dois tipos de função objetivo (portanto, 8 heurísticas implementadas).

## Aplicação ao CVRP

- ▶ Problema do Roteamento de Veículos com Restrições de Capacidade.
- ▶ Problema NP-difícil estudado há mais de 40 anos.
- ▶ Utilizamos as 20 instâncias criadas por de Golden et al. em 1998.
- ▶ Solução inicial gerada com o método Savings.
- ▶ Como referência para os resultados, adotamos as melhores soluções conhecidas, segundo Cordeau et al. (2004).
- ▶ Implementamos 6 heurísticas de baixo nível, sendo 2 delas para fazer modificações aleatórias.

## Detalhes dos Experimentos

- ▶ 9 minutos por execução.
- ▶ 5 execuções de cada hiper-heurística para cada instância.
- ▶ Resultado de cada hiper-heurística em cada instância é a média dos 5 resultados obtidos em suas execuções.
- ▶ “Nota” de cada hiper-heurística  $hh$  em cada instância  $I$ :

$$10 \frac{M_I - r_{hhI}}{M_I - m_I}$$

- ▶  $r_{hhI}$  é o custo obtido por  $hh$  em  $I$ .
- ▶  $M_I$  ( $m_I$ ) é o maior (menor) dos custos obtidos em  $I$ .

## Tabelas de Resultados

### *Desafio ROADEF 2005*

<i>HHESv4</i>	4,855
<i>HHESv3</i>	4,846
<i>HHESv1</i>	4,828
<i>HHESv2</i>	4,765
<i>HHDS</i>	4,758
<i>HHRC</i>	3,983
<i>HHED</i>	3,813
<i>HHAGv2</i>	3,737
<i>HHAGv1</i>	3,602
<i>HHAGv3</i>	3,581
<i>HHAGv4</i>	3,518
<i>Klau</i>	0,573

### *CVRP*

<i>HHED</i>	2,701
<i>HHRC</i>	2,463
<i>HHDS</i>	1,910
<i>HHESv1</i>	1,587
<i>HHESv4</i>	1,427
<i>HHESv2</i>	1,279
<i>HHAGv4</i>	1,178
<i>HHESv3</i>	1,170
<i>HHAGv2</i>	1,027
<i>HHAGv3</i>	0,542
<i>HHAGv1</i>	0,516

## Resultados Gerais

- ▶ Resultados no problema do Desafio ROADEF foram bons.
- ▶ No CVRP, as soluções ficam, em média, com custo 7% acima da melhor conhecida.
  - ▶ Parece razoável para uma hiper-heurística, mas falta um referencial adequado.
  - ▶ Um bom algoritmo específico (Ergun et al. – 2003) chega a aproximadamente 3,8% em cerca de 1 hora. Seu critério de parada não é o tempo.

## Avaliações Comparativas

<i>Hiper-heurística</i>	<i>ROADEF</i>	<i>CVRP</i>	<i>Média</i>
<i>HHESv1</i>	7,862	4,933	<b>6,398</b>
<i>HHESv4</i>	8,055	4,498	<b>6,277</b>
<i>HHED</i>	3,707	8,115	<b>5,911</b>
<i>HHESv2</i>	7,284	4,458	<b>5,871</b>
<i>HHDS</i>	7,258	4,063	<b>5,661</b>
<i>HHESv3</i>	7,330	3,910	<b>5,620</b>
<i>HHRC</i>	3,974	6,793	<b>5,384</b>
<i>HHAGv2</i>	3,484	3,447	<b>3,466</b>
<i>HHAGv4</i>	2,231	3,695	<b>2,963</b>
<i>HHAGv3</i>	2,495	2,165	<b>2,330</b>
<i>HHAGv1</i>	2,087	2,292	<b>2,189</b>

## Experimentos com a Hiper-Heurística de Soubeiga (*HHES*)

- ▶ Armazenar e utilizar informações sobre o desempenho de seqüências de heurísticas com tamanhos 3 e 4 (*HHESv2* e *HHESv3*) teve impacto ligeiramente negativo.
- ▶ Desconsiderar o tempo de execução (*HHESv4*) ao avaliar as heurísticas não causou diferença relevante.

## Experimentos com o Alg. Genét. Hiper-Heurístico (*HHAG*)

- ▶ Resultados com o novo operador de mutação (*HHAGv2*) foram superiores aos obtidos pela versão original (*HHAGv1*).
- ▶ Alterar o cálculo da aptidão no caso dos valores negativos (*HHAGv3*) teve efeito ruim (em relação a *HHAGv2*).
- ▶ Alterar o cálculo da aptidão para desconsiderar o comprimento dos cromossomos (*HHAGv4*) teve impacto ligeiramente negativo (em relação a *HHAGv2*).

## Comparações entre Hiper-Heurísticas

- ▶ O algoritmo genético (*HHAG*) teve os piores resultados em ambos os problemas.
- ▶ Melhores resultados:
  - ▶ *HHES*: melhor no problema do Desafio ROADEF e melhor na tabela de avaliações comparativas.
  - ▶ *HHED*: melhor no CVRP.
  - ▶ *HHDS*: método bastante simples, com bons resultados em ambos os problemas.
- ▶ Portanto, as hiper-heurísticas de busca por entornos se destacaram um pouco mais que as hiper-heurísticas evolutivas.

## Conclusão

- ▶ Hiper-heurísticas permitem que haja um bom compromisso entre qualidade e tempo de desenvolvimento.
- ▶ A experimentação com hiper-heurísticas é um caminho que deve ser explorado:
  - ▶ Novos desenvolvimentos e aplicações.
  - ▶ Novos experimentos.

## Conteúdo na Internet

- ▶ A dissertação e esta apresentação estão na minha página:  
<http://www.ime.usp.br/~igorrs>
- ▶ O código fonte está no SourceForge:  
<http://sourceforge.net/projects/hyperheuristics>