

CAPÍTULO 7

CONCLUSÃO

A aplicação conjunta de técnicas de computação evolutiva, busca local e otimização baseada em restrições tem por objetivo produzir uma solução de compromisso para problemas de otimização caracterizados por apresentarem um grande número de variáveis, múltiplos objetivos e múltiplas restrições. Os dois casos tratados neste trabalho correspondem a problemas reais de escalonamento: geração de turnos completos em torneios e definição de grade horária de professores em instituições de ensino.

Nesta dissertação, a estrutura de apresentação de conceitos e resultados foi definida com o propósito de focalizar inicialmente cada uma das três técnicas envolvidas (capítulos 2, 3 e 4), seguida da apresentação da proposta de aplicação conjunta (ao final do capítulo 4) e dos resultados de aplicação (capítulos 5 e 6). Os dois anexos abordam aspectos importantes deste trabalho: a necessidade do uso de um gerador de números pseudo-aleatórios com a propriedade de repetitividade e, especificamente para o caso de alocação de carga didática, a formalização de uma função-objetivo real com base em dados obtidos a partir de uma realimentação dos professores por intermédio de um questionário a ser usado para mapear as preferências e disponibilidades do corpo docente em cada período letivo.

Como principais contribuições deste trabalho, é possível mencionar:

- a própria iniciativa de aplicação conjunta de técnicas de computação evolutiva, busca local e otimização baseada em restrições, com a distribuição adequada de “papéis” entre cada técnica, procurando explorar o melhor que cada uma poderia oferecer ao longo da busca pela solução ótima;
- a formalização da proposta em termos de conceitos bem definidos no campo da computação evolutiva, como decodificadores, algoritmos de reparação, algoritmos meméticos e efeito Baldwin;
- elaboração e implementação computacional de um algoritmo de expansão de código para o problema da geração de turnos completos em torneios;

- elaboração e implementação computacional de um algoritmo de expansão de código para o problema da definição de grade horária de professores em instituições de ensino. Este algoritmo apresenta uma inovação importante: como a expansão de código pode ser caracterizada como uma caminhada em uma árvore de decisão, a existência de objetivos a serem otimizados foi levada em conta de modo a polarizar o processo de tomada de decisão, sempre no sentido de caminhar por ramos da árvore que indicassem um aumento na qualidade da solução.
- comparação de desempenho entre codificação expandida e codificação compacta.

Os resultados obtidos ao aplicar a técnica proposta aos dois problemas de escalonamento mencionados acima permitem concluir que é possível se beneficiar com o uso de uma representação compacta em conjunto com um algoritmo de expansão de código, quando comparado com o emprego direto de codificação expandida. As razões levantadas para explicar o sucesso desta abordagem são:

- o processo evolutivo somente tem lugar no espaço compacto, onde não há infactibilidade ou onde a factibilidade pode ser facilmente conquistada. Com isso, a existência de restrições praticamente não afeta a taxa de evolução no espaço compacto.
- o algoritmo de expansão de código, responsável pelo atendimento das restrições, é um método de otimização baseado em restrições, de modo que a representação expandida resultante será sempre factível;
- as duas possíveis limitações do algoritmo de expansão de código, ou seja, a impossibilidade de garantir que qualquer representação expandida factível tenha uma representação compacta associada e a existência de ótimos locais como resultado do processo de expansão, são amenizadas respectivamente pela presença de busca local e pela evolução de uma população de representações compactas.

A abordagem proposta, como qualquer outra estratégia de busca implementada em computador, também apresenta suas desvantagens:

- necessidade de definição de uma codificação compacta para cada problema de aplicação, mais especificamente de um espaço de busca compacto, sendo que sua dimensão mostrou-se decisiva na eficiência do processo evolutivo;

- necessidade de implementação de um algoritmo de expansão de código para cada problema de aplicação, levando em conta todas as particularidades do problema;
- custo computacional vinculado ao processo de busca. O algoritmo pode ter reduzida sua capacidade de busca no caso de problemas de tamanho muito elevado.

Para o caso de geração automática de tabelas em torneios, além de superar a abordagem baseada no código expandido, a comparação de desempenho junto a soluções propostas por especialistas, adotadas para problemas do mundo real, permite concluir que a adoção de uma codificação compacta, seguida da expansão de código com reparação e busca local, é muito eficaz na procura de soluções de alta qualidade, superando de forma evidente soluções que vem sendo adotadas na prática.

Como perspectivas futuras para este trabalho são sugeridos os seguintes tópicos:

- verificar a elaboração de novos operadores genéticos que possam adaptar-se melhor a cada contexto de aplicação;
- verificar a possibilidade de incorporação de punições na função de *fitness* para o caso do processo de expansão, juntamente com a busca local e os algoritmos de reparação, requerer recursos computacionais excessivos (por exemplo, número elevado de ocorrências de *backtracking*);
- permitir que um mesmo processo de expansão de código possa operar com diferentes geradores de números aleatórios, todos apresentando a propriedade de repetitividade;
- no caso da aplicação da estratégia de solução proposta neste trabalho a problemas que admitem o emprego de mais de um algoritmo de otimização baseada em restrições, criar condições para que o algoritmo de expansão de código possa optar por um ou outro dentre os algoritmos disponíveis para expansão de código;
- verificar a possibilidade de aplicação do conceito de sistemas auto-adaptativos, conforme discutido no capítulo 2;
- verificar a possibilidade de incorporação das técnicas utilizadas nos sistemas co-evolutivos (capítulo 2), procurando criar funções de *fitness* distintas para as restrições globais e locais;
- extensão da estratégia de solução proposta a outros problemas de escalonamento e possivelmente a outros problemas de otimização com restrições.

Os algoritmos evolutivos em sua composição tradicional encontram dificuldades no tratamento de problemas de otimização com restrições. Também é fato, que técnicas de otimização baseada em restrições quando aplicadas isoladamente podem encontrar sérias dificuldades em virtude da existência de ótimos locais de baixa qualidade. Dessa maneira, este trabalho, por intermédio da aplicação conjunta de ambas as metodologias, mostra que resultados promissores podem ser obtidos, desde que haja um equilíbrio no papel a ser desempenhado por essas técnicas.