

MAPEAMENTO SISTEMÁTICO SOBRE OTIMIZAÇÃO INTERATIVA EM DYNAMIC ADAPTIVE SEARCH BASED SOFTWARE ENGINEERING

João Miguel de Souza Barros (PIBIC/CNPq/FA/Uem), Aline Maria Malachini Miotto Amaral (Orientador), e-mail: ra105415@uem.br, ammmamaral@uem.br.

Universidade Estadual de Maringá / Centro de Ciências Exatas e da Terra/Maringá, PR.

Ciência da Computação - Metodologia e Técnicas da Computação.

Palavras-chave: engenharia de software baseada em busca. busca adaptativa dinâmica. otimização interativa.

Resumo

Problemas complexos de Engenharia de Software (ES) podem ser modelados como problemas de otimização e serem resolvidos por algoritmos baseados em busca. Neste sentido foi proposto um campo de pesquisa denominado *Search Based Software Engineering* (SBSE). Apesar de favorecer a resolução de problemas de ES, as soluções em SBSE, muitas vezes, desconsideram as preferências do usuário. Para minimizar este problema, foi desenvolvida uma abordagem chamada Otimização Interativa - *Dynamic Adaptive Search Based Software Engineering* (DASBSE) que surgiu com o objetivo de unir a Aprendizagem de Máquina (AM) com a otimização baseada em busca e assim melhor incorporar as expectativas do usuário em problemas resolvidos pela SBSE. Este trabalho teve como objetivo realizar um mapeamento sistemático para identificar pesquisas promissoras relacionadas à DASBSE. A razão para se realizar mapeamentos sistemáticos, é que estes resumem as evidências existentes em relação a um tema de interesse científico. Os resultados coletados pela pesquisa, servirão para auxiliar no desenvolvimento de trabalhos relacionados a área, podendo trazer novas ideias e pontos de vista para pesquisadores e atuantes no mesmo.

Introdução

Os problemas da Engenharia de Software (ES) geralmente envolvem restrições conflitantes, informação ambígua e/ou imprecisa, e grandes conjuntos de escolhas ou decisões. Resolver problemas desse tipo é, em geral, uma tarefa complexa para a qual não há uma solução exata. Nesse sentido, os problemas de ES podem ser modelados como problemas de otimização para serem resolvidos por algoritmos baseados em busca.

Harman e Jones (2001) propuseram um campo de pesquisa denominado *Search Based Software Engineering* (SBSE) que aplica metaheurísticas a fim de obter automaticamente soluções ótimas ou o mais próximo do ótimo possível para problemas complexos de ES.

Os algoritmos evolucionários, entre eles os algoritmos genéticos, estão entre as técnicas mais utilizadas em SBSE (HARMAN et al., 2012b). Apesar de favorecer a resolução de problemas de ES, os algoritmos evolucionários, muitas vezes, desconsideram as preferências do usuário. Há problemas em que é praticamente impossível definir uma função de avaliação que englobe todas as características desejadas, uma vez que, nem todas as características podem ser modeladas matematicamente (TAKAGI, 2001). Para minimizar este problema, foi desenvolvida uma abordagem chamada Otimização Iterativa com objetivo de incluir as perspectivas humanas no processo de otimização (HARMAN et al., 2012b).

Apesar do benefício, a Otimização Iterativa traz um problema para o processo de busca que é a fadiga humana (HARMAN et al., 2012a), uma vez que os algoritmos de otimização realizam inúmeras consultas ao usuário. Uma abordagem chamada *Dynamic Adaptive Search Based Software Engineering* (DASBSE) surgiu com o objetivo de unir a Aprendizagem de Máquina (AM) com a otimização baseada em busca (HARMAN et al., 2012a) para reduzir o problema da fadiga.

Nesse contexto, o objetivo deste trabalho foi realizar um mapeamento sistemático sobre otimização interativa em DASBSE para responder às seguintes questões: *Como acontece a interação ser-humano computador durante o processo de otimização em DASBSE?*; *Em que momento do processo de otimização em DASBSE acontecerá a interação ser-humano computador?*; Quais métodos e técnicas devem ser implementados no algoritmo evolucionário para suportar a interação do usuário na otimização em DASBSE?

Materiais e métodos

O processo de mapeamento sistemático da literatura (MS) tem como objetivo fazer uma pesquisa em largura, e não em profundidade (KITCHENHAM, 2004). Os itens que fazem parte do protocolo de um mapeamento sistemático são: objetivos de busca, palavras-chave, filtros, mecanismos de busca acadêmica, critérios de exclusão/inclusão, e dados a serem levantados.

A ferramenta StArt¹ foi utilizada para a realização do mapeamento sistemático e as bibliotecas digitais pesquisadas foram: ACM, IEEE, Springer e Science Direct. Previamente a execução do mapeamento sistemático, foi definida a *string de busca* e os *critérios de seleção de trabalhos* (Tabela 1). A *string de busca* utilizada foi: ("*dynamic adaptive*" OR "*dynamical adaptive*") AND (("search-based" OR "search based" OR "based on search") OR (*interactive optimization*)) AND ("*software engineering*") AND ("*user preference*" OR "*user interaction*" OR "*user engagement*" OR "*human interaction*" OR "*human preference*" OR "*human engagement*").

As etapas realizadas para a condução do mapeamento são apresentadas na Tabela 2. Tais etapas foram essenciais para a organização

¹ http://lapes.dc.ufscar.br/tools/start_tool.

e acompanhamento do progresso das atividades do mapeamento sistemático realizado.

Tabela 1 – Critérios de seleção de trabalho.

Critério	Descrição
Inclusão	Trabalhos publicados no período de 2008 a 2018
Inclusão	Utilização de interação ser-humano computador em um processo de otimização.
Inclusão	Trabalhos que envolvem otimização em DASBSE.
Inclusão	Trabalhos que envolvem otimização em SBSE.
Inclusão	Utilização de interação ser-humano computador em problemas tratados pela DASBSE.
Exclusão	Trabalhos que não envolvem otimização de engenharia de software.
Exclusão	Trabalhos que não envolvem engenharia de software.
Exclusão	Trabalhos que não abordam DASBSE.
Exclusão	Trabalhos que não têm interação ser-humano computador.
Exclusão	Trabalhos que não estão disponíveis na íntegra.
Exclusão	Trabalhos que não estejam disponíveis em inglês ou português.

Tabela 2– Etapas do mapeamento sistemático.

Etapa	Descrição
Etapa 1	Busca de palavras-chave e sinônimos nas seções título, abstract e palavra-chave, por meio das ferramentas de filtragem oferecidas pelas bibliotecas digitais utilizadas.
Etapa 2	Leitura do título e abstract dos trabalhos encontrados e filtragem dos estudos relevantes de acordo com os critérios de seleção de trabalhos definidos.
Etapa 3	Nos trabalhos selecionados da etapa 02, foi realizada a leitura da introdução e da conclusão para uma nova filtragem.
Etapa 4	Leitura na íntegra dos trabalhos classificados como relevantes, buscando-se informações relacionadas às questões de pesquisa. Um trabalho é considerado relevante quando contém pelo menos um critério de inclusão e nenhum critério de exclusão.

Resultados e Discussão

Na primeira etapa do mapeamento sistemático foi realizada a busca de trabalhos por meio das *strings de busca* e foram encontrados 108 trabalhos (**Etapa 01**). Após a leitura do título e abstract dos trabalhos selecionados na etapa 1 permaneceram no mapeamento 23 trabalhos (**Etapa 02**). Na sequência a Introdução e Conclusão dos trabalhos selecionados na etapa 02 foram lidas (**Etapa 03**) resultando em 11 trabalhos relevantes. Finalmente os 11 trabalhos resultantes da etapa 03 foram lidos na íntegra (**Etapa 04**).

Com base na leitura por completo dos artigos resultantes da Etapa 3 e tendo como foco as questões do MS, pode-se observar que em função do grande excesso de interações realizadas nos algoritmos evolucionários, usados nas abordagens de SBSE, estes apresentam limitações com relação à interação com o usuário uma vez que um número excessivo de interações causa a fadiga do usuário. As estratégias sugeridas são principalmente: intensificar a interação com o usuário no começo do algoritmo, para que este possa refinar o máximo possível o objetivo do algoritmo de acordo com suas preferências, e utilizar a interação com o usuário para melhorar a função de melhora da população, o que envolve utilizar o feedback do usuário sobre uma ou várias soluções no final do algoritmo para melhorá-la. Para a primeira solução, por se tratar de uma estratégia mais simples, não há restrições quanto ao seu uso em algoritmos evolucionários, pois cada

algoritmo deve tratar os objetivos e preferências do usuário de um jeito específico. Na utilização de feedbacks de usuários, o algoritmo evolucionário recomendado é o NSGA-II, por se tratar de um algoritmo eficiente na busca de possíveis soluções com uma baixa complexidade, facilitando assim o seu reuso. Para tratar o feedback do usuário e gerenciar o reuso do algoritmo NSGA-II, é sugerido o uso da técnica de refatoração dinâmica, interativa e multiobjetivo. Além disso, observou-se em muitos trabalhos a menção do uso de algoritmos de AM integrados ao processo de otimização de forma que depois de um determinado número de interações as preferências do usuário possam ser suportadas por um modelo de aprendizagem devidamente treinado a partir das interações reais realizadas pelo usuário.

Conclusões

A otimização interativa em DASBSE se torna cada vez mais uma solução promissora para resolver problemas complexos de engenharia de software. No entanto, muitas lacunas, principalmente com relação a estratégias de integração de algoritmos evolucionários e AM, ainda existem. Este trabalho realizou um mapeamento sistemático para identificar como encontra-se o corpo de conhecimento na área de DASBSE. Pode-se observar que abordagem com resultados interessantes já foram propostas, no entanto um problema que ainda precisa ser melhor trabalhado é como resolver o problema da incorporação da interação com usuário em processos que demandam um número excessivo de interações.

Agradecimentos

Agradecemos à Fundação Araucária pelo suporte financeiro.

Referências

- HARMAN, M.; JONES, B. F. Search-based software engineering. *Information and software Technology*, v. 43, n. 14, p. 833-839, 2001.
- HARMAN, M.; MANSOURI, A. Search based software engineering: Introduction to the special issue of the IEEE transactions on software engineering. *IEEE transactions on Software Engineering*, v. 36, n. 6, p. 737, 2010.
- HARMAN, M.; BURKE, E.; CLARK, J. A.; YAO, X. Dynamic adaptive search based software engineering. In: Empirical Software Engineering and Measurement (ESEM), 2012 ACM-IEEE International Symposium on, IEEE, 2012a, p. 1-8.
- HARMAN, M.; MCMINN, P.; DE SOUZA, J. T.; YOO, S. Search based software engineering: Techniques, taxonomy, tutorial. *Empirical software engineering and verification*, Springer, p. 1-59, 2012b.
- KITCHENHAM, B., (2004). *Procedures for Performing Systematic Reviews*, Joint Technical Report Software Engineering Group, Department of Computer Science Keele University, United King and Empirical Software Engineering, National ICT Australia Ltd, Australia.
- TAKAGI, H. Interactive evolutionary computation: Fusion of the capabilities of ec optimization and human evaluation. *Proceedings of the IEEE*, v. 89, n. 9, p. 1275-1296, 2001.