

#161823: How does Adaptsuite try to ensure the same quality by running fewer tests?

Authors - [Andre Pestana](/jems2/index.php?r=person/view&id=190289) (</jems2/index.php?r=person/view&id=190289>) (University of Sao Paulo)
- [Alfredo Goldman](/jems2/index.php?r=person/view&id=467) (</jems2/index.php?r=person/view&id=467>) (IME - USP)
- [Renan de Melo Oliveira](/jems2/index.php?r=person/view&id=115402) (</jems2/index.php?r=person/view&id=115402>) (University of Sao Paulo)

Abstract Systems testing, while crucial, is recognized as an expensive activity. Large systems have many tests and testing them in their entirety, besides being expensive, is time consuming. The proposal of the Adaptsuite library is to choose the best set of tests to run in a time limit, based on parameters.

Topics

Software Quality Assurance

Measurement and Analysis

Value in Software Quality

Verification, Validation and Testing

conference SBQS 2016 - Track of Technical Works

track Technical Works Trail

Category

Status rejected

files

Description	file name	Type	Size	Created
Paper manuscript	161823.pdf (/jems2/index.php?r=paper/download&p=161823&f=0)	pdf	284.64 KB	Jul 04, 2016 - 11:03 PM (BRT)

Paper Options

Reviews

Review 1 - [Marllos Paiva Prado](/jems2/index.php?r=person/view&id=92524) (</jems2/index.php?r=person/view&id=92524>)



1 - Originality and novelty author : 3) Improves an existing technique to some extent	2 - Technical soundness and contribution author : 3) Solid work but marginal contribution	3 - Relevance to the call for papers author : 4) Definitely relevant	4 - Readability, organization and presentation author : 2) Poor, needs considerable rework	5 - Reference to previous work: Does it give adequate credit to previous work? Does it contrast the results with previous work? Are the bibliography references correct? author : 3) Average	6 - Recommendation for this paper author : 2) Probably reject (low quality)	7 - Reviewer familiarity with the subject author : 3) Moderately familiar (knowledgeable in the area, though not an expert)	Best paper award tpc : 1) It definitely does not deserve	References to SBQS author : 1) No.
---	---	--	--	--	---	---	--	--

1: Summary: A brief description of the paper.:

The article describes a solution for TCP (Test Case Prioritization) called Adaptsuite. The solution differs from the others by taking into account a time limit in the prioritization of test cases.

2: Paper strengths: What are the positive aspects of this paper.:

The tool developed by the authors seems relevant from a technical point of view. It takes into account variables that are really important in prioritizing test cases and demonstrates potential to solve practice problems.

3: Paper weakness: What are the negative aspects of this paper.:

Unfortunately, the writing of the article lacks significant improvements in order to be truly effective in communicating the relevance and contributions of the proposal.

4: Comments to authors:Comments and suggestions to authors to improve the paper quality.:

At some points in the text, the reader has to work hard to understand what is being said. In others, the information seems secondary to the understanding of the proposal.

On the second page, for example, the authors dedicate an entire paragraph to talking about Fault Localization, only to say that it will not be evaluated at the end. This consideration could be removed or reallocated to the conclusion of the article. The methodology session should be dedicated to what was actually done and how it was done.

Another note on this page is that citation [10] does not contain the name of the work in the references.

On the third page, the authors present the definition of failure and error, assigning it to the JUNIT tool. However, this seems to be a definition of the author Massol, and not of the tool or the creators of JUNIT. If this is the case, credits must be correctly assigned.

Still on p. 3, section 3.2 the authors dedicate the last 3 paragraphs to explain how to download and use the Eclemma tool. This is information that should be on github, along with a ReadMe file from the tool. It is unnecessary in a research paper and diverts the attention of the reader who is trying to form an idea of how the variables together relate to the prioritization of test cases.

On page 4 I highlight two points that the authors need to explain better to the reader:

(i) é dito logo no início da página que "Caso um teste não possua um relatório de cobertura, a AdaptSuite sempre irá atribuir valor de cobertura igual a 100%. Por esse motivo recomenda-se que todos os testes possuam um relatório de cobertura antes de usar a AdaptSuite" Essa limitação é técnica (atribui-se 100% porque isso simplifica a implementação da ferramenta) ou teórica (100% é a porcentagem mais adequada para fazer uma priorização justa em caso de ausência de cobertura)? De toda forma, isso não afetaria a aplicabilidade da proposta em um cenário real, sendo portanto um requisito que a ferramenta deveria detectar e cobrar do usuário? Se sim, considerações deveriam ser feitas ao final do trabalho. Se não, deve-se esclarecer o motivo.

(ii) Os autores dizem que quanto menor o valor dessa variável, maior a chance do teste correspondente ser escolhido. Porque esse critério foi adotado? Seria importante esclarecer porque um teste que executa menos vezes deve ter mais chance que um que executa mais vezes.

Em geral, eu senti falta de uma explicação sobre o motivo da escolha dessas variáveis em detrimento de outras possíveis para o objetivo final da priorização.

A seção 5 (Algoritmo) fala sobre encontrar a suíte ótima resolvendo o problema da mochila [12]. Entretanto qual é o algoritmo de fato utilizado pela ferramenta? Os autores aplicam Programação Dinâmica? Isso não está claro.

Na seção 6 também não encontrei a diferença nos dois primeiros códigos exibidos. Eram pra ser o mesmo? Se sim, basta referenciar o primeiro duas vezes.

As cores nos gráficos 1 e 4 não estão fáceis de serem distinguidas quando o artigo é impresso em escala de cinza. Sugiro aumentar a diferença nas tonalidades ou o uso de texturas.

Considerações finais:

Os autores precisam ser mais atenciosos com erros de digitação :

Pag 1: *ser validar*

Pag 3: *Para ser gerar*

Pag 3: *ent ao*

Pag 3: *ent ão*

Pag 6: *pussuindo*

Pag 6: *packages*

Pag 11: *tal cara*

E tornar o texto mais claro e preciso:

Pag 11:

"a última conclusão que pode ser tomada" --> "concluí-se que"

"ela será melhor" --> "a adoção da Adaptsuite será melhor"

ou ainda esclarecer, para quem não é familiarizado com otimização o que significa "considerando cada variável como uma coordenada em R4".

Como princípio geral para melhoria da escrita do trabalho, recomendo aos autores que se coloquem no lugar do leitor e sempre se questionem se o que está sendo escrito está claro para a diversidade do público que irá ler o trabalho.

Questions for rebuttal phase: Provide direct questions that you want answers during the rebuttal phase. Remember that authors will have only 4000 characters to answer all questions from all reviewers, so add here only the most relevant questions.:

Gostaria de saber pelos menos em alto nível, como o algoritmo usado pela Adaptsuite trabalha com as variáveis definidas na resolução do problema da mochila.

5:TPC:Comments for TPC members only.:

Rebuttal

Pergunta 1 revisor 1:

Dado que cada teste tem seu peso a valor definidos, a ferramenta constrói uma matriz com T colunas, onde T é o tempo em milissegundos definido pelo usuário, e n linhas, sendo n o número de testes. Cada elemento na coluna y e linha x dessa tabela representa o valor ótimo da sub mochila de tamanho y considerando os primeiros x itens. Para determinar tal valor, utilizo a seguinte recorrência:

- $Otima[x,y] = Otima[x-1,y]$ se $Peso(X) > y$

- $Otima[x,y] = \text{Max}\{Otima[x-1,y], Otima[x-1, y - \text{Peso}(x)] + \text{Valor}(x)\}$ se $\text{Peso}(x) \leq y$

Essa recorrência basicamente diz que ao tentar adicionar um novo elemento ao conjunto, a nova mochila ótima não mudará se esse novo item não puder ser adicionado à mesma ($\text{Peso}(x) > y$). Caso seja possível adicioná-lo ($\text{Peso}(x) \leq y$), deve-se calcular se vale a pena ou não adicioná-lo. Vale ressaltar que essa matriz começa com todos os seus valores zerados.

Pergunta 2 revisor 2:

Pode-se usar a ferramenta em qualquer IDE. O Eclipse foi mencionado no exemplo por ser a IDE mais popular para programação em Java.

Pergunta 3 revisor 2:

Meu foco foi em mostrar que mesmo usando a solução exata do problema da mochila, que necessita de mais tempo para resolvê-lo em comparação com o algoritmo guloso, ela ainda é a melhor opção usando a metodologia do Problema da Mochila. No começo, não me era claro que encontrar a mochila ótima significava encontrar um teste defeituoso com mais facilidade.

Pergunta 4 revisor 2:

The choice of code coverage as a variable was made so that tests could be prioritized that are more likely to detect an error in the system (the more code executed, the greater the chance of going through a faulty section). The use of the number of failures is intended to prioritize the tests that are more likely to be wrong. Therefore, the tool tries to prioritize these two properties at the same time, unless the user sets a priority manually.

TPC Discussion

