بسم الله الرحمن الرحيم

الآيـــة

قال تعالى :

﴿ قُلْ إِنَّ صَلَاتِي وَنُسُكِي وَمَحْيَايَ وَمَمَاتِي لِلَّهِ رَبِّ الْعَالَمِينَ ﴾

صدق الله العظيم

(الانعام ١٦٢)

Acknowledgements

Thank Allahfirst and foremost, which His grace is righteousandPraise be to Allah...

I would like to express my appreciation to my supervisor

Prof.Izzeldin Mohamed Osman

Whoprefer tosupervisethisresearchand gavea lot ofhis time and effortin advanceadvice, guidanceand mentoring.I thank him for his endless patience and encouragement, for his exigency and rigor and for his insights and suggestions that helped to shape my research skills.I could hardly express my gratitude, and Godhas apay and reward. I am very grateful to myVenerableteachers,for supporting me through all these years, also I am greatly indebted to my friends, for all the good moments they provided but, specially, for listening to me when I needed to talk, and for making me laugh when I most needed it... suchgood memories I cannot capture in words!

Last but not least, I owe my deepest gratitude to my mother, who always encouraged me towork hard and always believed in me, and also to my father and sisters , for always being therewhen I needed them most.

With abundant thanks for all

Researcher

Abstract

Combinatorial testing has attracted a lot of attention from both industry and academia. A number of reports suggest that combinatorial testing can be effective for practical applications. However, there are few systematic, controlled studies on the effectiveness of combinatorial testing.

The behavior of a software application may be affected by many factors e.g input parameters, environment configurations and state variables .It is impractical to test all possible combinations of values of all those factors. Instead of testing all possible combinations, a subset of combinations is generated to satisfy some well-defined combination strategies. A key observation is that not every factor contributes to every faultand it is often the case that a fault is caused by interactions among a few factors. Combinatorial design can dramatically reduce the number of combinations to be covered.Thisrequires indepth knowledge of the business being modeled by the software application.

This research reports an experiment that applied combinatorial testing to the ElectronicRegistration program of the Omdurman IslamicUniversity usingPictMaster tool. This tool can first accept a predefined test suite, then generate additional test cases to achieve the required coverage while minimizing redundant coverage of interactions.

The experimental results showed that combinatorial testing detected hardto find software faults with a small number of tests.

المستخلص

اجتذب الاختبار الاندماجى (combinatorial testing)الكثير من الاهتمام فى الاوساط الصناعية والاكاديمية فى الآونه الاخيرة. حيث اشارت التقارير الى فعاليتة للتطبيقات العملية بالرغم من قلة الدراسات المنهجية التى توضح ذلك على وجه الخصوص عملية الاختبار الاندماجى على المدخلات (input parameter) والتى تعتبر فى منتهى الاهمية . ولكن معظم الدراسات لا توثق تفاصيل عملية النمذجه .

يتأثر سلوك التطبيقات بعديد من العوامل على سبيل المثال معاملات الادخال (input parameters) تكوينات البيئه (environment configurations) والمتغيرات (state variables).

تعتبر عملية اختبار كل مزيج ممكن من قيم كل تلك العوامل غير عملية وبدلا من ذلك يتم انشاء مجموعه فرعية ليتم اختبارها باستخدام استراتيجيات محدده .

الملاحظة الرئيسية هى ان ليس كل عامل يساهم فى كل خطأ . بل فى كثير من الاحيان ينتج الخطأ من التفاعلات بين عدد قليل من العوامل . التصميم للاختبار الاندماجى يقلل بشكل كبير من المجموعات المراد تغطيتها ولكن لا يزال فعال جدا من حيث الكشف عن الاخطاء . على وجه الخصوص تعتبر نمذجة معاملات الادخال (input parameter) هى الخطوة الاساسية فى عملية الاختبار الاندماجى وتعتمد فعالية وكفاءة الاختبار على النموذج المستخدم . ولكن معظم الدراسات لاتوثق تفاصيل عملية النمذجه .

فى هذا البحث تم تطبيق الاختبار الاندماجى (combinatorial testing) على برنامج التسجيل الالكترونى لجامعة امدرمان الاسلامية باستخدام اداة ال Pict Master , حيث يمكن لهذة الاداه ان تقبل مجموعة اختبار (test suite) محدده مسبقا ثم تولد حالات اختبار اضافية لتحقيق التغطية المطلوبة مع التقليل من التكرار فى التفاعلات .

يتكون المنهج الذى اتبع فى هذا البحث من ثلاث عمليات . اولا انشاء نموذج عن طريق تنفيذ مهمتين رئيسيتين هما تعريف المعاملات والقيم ,تعريف العلاقات والقيود .

ثانيا توليد مجموعه من الاختبارات (Test Cases set). واخيرا الحصول على النتائج المتوقعه . ومن ثم تحليل النتائج بطريقة ال (black box testing) باستخدام (User Testing Dominant Style) .

وقد اظهرت النتائج التجريبية ان الاختبار الاندماجى (combinatorial testing) استطاع العثور والكشف عن الاخطاء فى البرنامج بعدد قليل من الاختبارات (Test Cases) وانه اكثر كفاءة من الاختبار المعروف .وتم قياس فعالية الاختبار الاندماجى بحجم الاخطاء التى تم الكشف عنها .

بالاضافة الى ذلك تم توثيق تفاصيل عملية النمذجه وتحليل النتائج وبالتالى تم القاء الضوء على هذه الخطوات والتفاصيل الهامه التى غالبا ما تم تجاهلها فى الدراسات السابقة .

List of Tables

Table		Page number
2-1	Show test suite size for configuration with 10 V-valued parameters …….	14
2-2	Show test suite size for a configuration with p 2-value parameters …….	15
3-1	Parameters and Values Conversion ……...	20
3-2	Base Data Values ……	21
3-3	Application Constrains ……..	22
3-4	Summary……	46
4-1	Result …….	52

List of Figure

Figure		Page number
3-1	Electronic Registration Application (1)…….	18
3-2	Electronic Registration Application (2)…….	19
3-3	Parameters and value hierarchy columns ……..	23
3-4	Setting window ……..	24
3-5	Test Cases……..	24
3-6	Statistical information ……...	25
3-7	Notepad format …….	26
3-8	Constrains Table setting window ……..	27
3-9	Adding Constrains ……	28
3-10	New test cases ……...	29
3-11	Statistical information window …….	29
3-12	Notepad format ……..	30
3-13	Sub-models setting window ……..	31
3-14	Sub-models table ……...	31
3-15	New test cases ……...	32
3-16	Progress Bar while minimizing the number of test cases …..	32
3-17	Statistical information window …….	33
3-18	Notepad format ……..	33
3-19	Extended sub-models setting window ……...	34

3-20	Extended sub-models table ……	35
3-21	Progress Bar while minimizing the number of test cases …..	35
3-22	New test cases (24) ……	36
3-23	Statistical information window …….	36
3-24	Desired coverage (3-way) setting window ……	37
3-25	Progress bar during generation with specified coverage...…	38
3-26	New test cases (27)……	38
3-27	Statistical information window……..	39
3-28	Desired coverage (4-way) setting window ……	39
3-29	Progress bar during generation with specified coverage …...	40
3-30	New test cases (79) ……	40
3-31	Statistical information window …….	41
3-32	Desired coverage (5-way) setting window ……	41
3-33	Progress bar during generation with specified coverage …...	42
3-34	New test cases (160) ……..	42
3-35	Statistical information window …….	43
3-36	Desired coverage (6-way) setting window ……	44
3-37	Progress bar during generation with specified coverage …...	44
3-38	New test cases (283) ….....	45
3-39	Statistical information window …….	46
3-40	Expected results setting window ……...	47
3-41	Expected results table ……	48
3-42	Expected results …….	48
3-43	Statistical Information window ……	49
4-1	Test Report ……	51