CS458: Intro. to Dynamic Analysis of Software Source Code

https://swtv.kaist.ac.kr/courses/cs458-fall2022

Moonzoo Kim

KAIST

Software Testing Verification (SWTV) Group

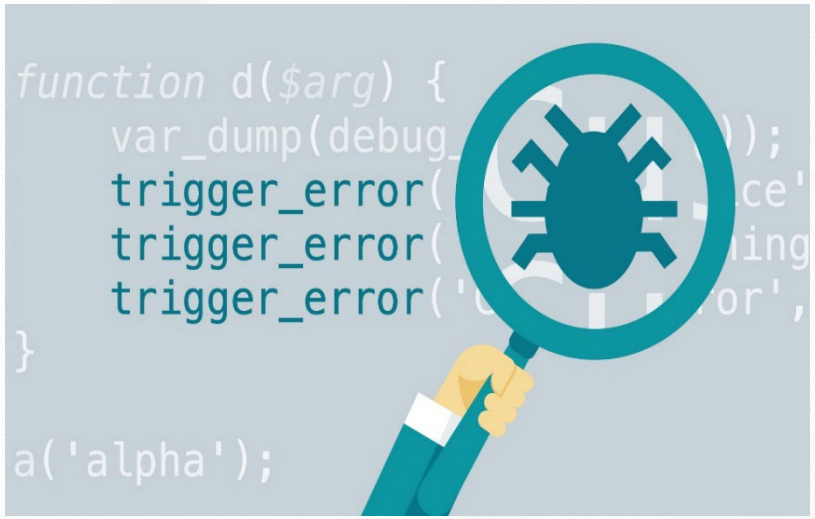
Prof. Moonzoo Kim

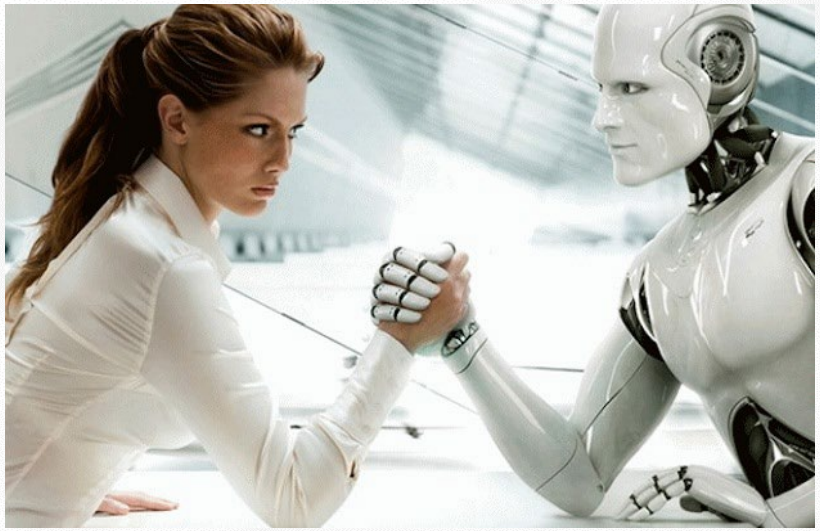
KAIST

E-mail:
moonzoo.kim@gmail.com

https://swtv.kaist.ac.kr/

Necessity of Automated SW Testing



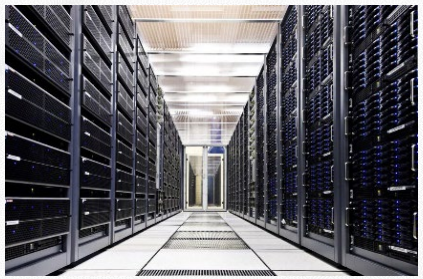


- Safety of SW becomes unreliable due to the high complexity of SW

89 people died caused by Toyota SUA (sudden unintended acceleration)

346 people died due to Boeing 737 MAX crash

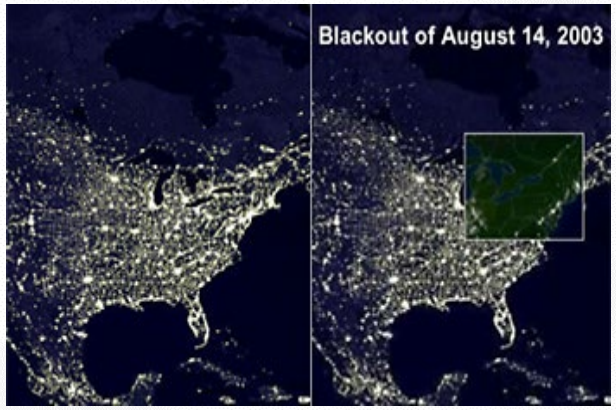
- Modern SW is too large and complex for human engineers to manually test
- Decreased product quality due to the low effectiveness and efficiency of SW bug detection



Scientific automated SW testing is the solution to the problem

Social and Economic Loss due to High Complexity of SW

Although most areas of modern society depend on SW,
reliability of SW is not improved much due to its high complexity



(2003) US & Canada blackout
- 7 states in US and 1 state in Canada suffered 3 days electricity blackout
- Caused by the failures of MISO monitoring SW
- 50 million people suffered and economic loss of 6 billion USD

(2010s) Toyota sudden unintended acceleration
- 89 people died since 2002
- SW bugs detected in 2012
- Fined 1.2 billion USD in 2014

TOYOTA

(2018-2019) Boeing 737 MAX accidents
- 346 people died in 2 accidents
- SW bugs detected in 2019
- Boeing 737 MAX is banned all over the world

BOEING

Accidents of Autonomous Vehicle

주요 자율주행차 사고

일시	관련 기업	사고 내용
2016년 2월 14일	구글	자율주행차의 과실로 시험 주행 중 버스와 접촉사고
2016년 5월 7일	테슬라	대형 트레일러와 충돌, 첫 운전자 사망사고
2017년 11월 8일	나브야	자율주행셔틀, 실주행에서 트럭과 접촉사고
2018년 3월 18일	우버	교차로 건너던 행인 치어 첫 보행자 사망사고
2018년 3월 23일	테슬라	고속도로에서 중앙분리대를 들이받아 운전자 사망

[출처: 중앙일보] "테슬라 사고, 태양 역광 탓" … 자율주행차 또 날씨 오작동

Bugs of Autopilot SW

연합뉴스

도로에 점 3개 찍었더니..테슬라 자율주행차, 中실험서 역주행

입력 2019.04.02. 10:31 댓글 338개

도로 위 작은 점 칠해 자율주행시스템 속여..테슬라 "실제 일어날 문제아냐"

(서울=연합뉴스) 정성호 기자 = 중국의 사이버보안 연구소가 전기차 테슬라의 자율주행 시스템을 속여 이 차가 반대편 차선을 역주행하도록 한 것으로 나타났다.

중국의 게임·인터넷 기업 텐센트 산하 '킨 시큐리티 랩'은 테슬라 전기차의 오토파일럿 시스템에 대한 실험에서 이처럼 차가 역주행하도록 했다고 밝혔다고 블룸버그 통신과 경제 매체 비즈니스인사이더가 1일(현지시간) 보도했다.

이 연구소는 실제 도로 위에 세 개의 작은 점을 칠했고, 그 결과 테슬라의 전기차가 왼편에 있던 반대편 도로로 옮겨 역주행하도록 하는 데 성공했다고 밝혔다.

도로의 교차로 지점에 작은 점을 표시하자 테슬라의 차가 이를 오른쪽 차선으로 인식하고 왼쪽으로 방향을 틀어 주행했다는 것이다.

테슬라의 전기차. [AFP=연합뉴스 자료사진]

The weakness of Data Driven AI/Machine Learning SW (adversarial example)

input image
(street sign)

Anomaly
(bird nest)

Boieng 737 MAX Crash due to SW Bugs

참화 부른 보잉의 '늑장대응'…이제 와서 "열흘내 업그레이드"(종합)

송고시간 | 2019-03-16 06:54

NYT "보잉, 작년말까지 업그레이드 약속"…셧다운發 업무지연 연관성도 주목

(뉴욕=연합뉴스) 이준서 특파원 = 미국 항공기 제작업체 보잉이 전 세계적으로 운항중단 조처가 내려진 '보잉 737맥스(Max)' 기종에 대해 10일 이내 '소프트웨어 업그레이드'에 들어갈 예정이라고 AFP통신이 15일(현지시간) 보도했다.

문제로 지목된 소프트웨어는 '조종특성 향상시스템'(MCAS·Maneuvering Characteristics Augmentation System)이다. 난기류 상황에서 항공기의 급하강을 막아주는 일종의 운항정지 방지 시스템이다.

구체적인 원인 분석은 이뤄지지 않았지만, 4개월여 사이에 재발한 '737맥스 8' 기종의 추락 참사는 MCAD와 무관치 않은 것으로 분석된다.

"보잉, 737 맥스 조종제어 소프트웨어 대폭 수정 중"

SBS이혜미 기자

입력 : 2019.03.13 12:54 | 수정 : 2019.03.13 12:54

미국 보잉사가 안전성 우려가 불거진 737 맥스 기종 전반에 대해 조종제어 소프트웨어를 대폭 수정하고 있다고 월스트리트저널이 보도했습니다.

소프트웨어 수정은 지난 주말 에티오피아 여객기 추락 사고가 발생하기 전부터 진행해 온 것으로 지난해 10월 같은 기종인 인도네시아 라이언에어 여객기가 추락한 데 따른 것입니다.

미 항공당국은 다음 달 말까지 수정작업이 마무리될 것으로 예상하고 있다고 신문은 전했습니다.

보잉의 최신기종을 둘러싼 안전성 논란이 커지면서 보잉의 주가도 이틀째 추락했습니다.

SW Causes of Toyota SUA

SOFTWARE CAUSES OF MEMORY CORRUPTION

Type of Software Defect	Causes Memory Corruption?	Defect in 2005 Camry L4?
Buffer Overflow	Yes	Yes
Invalid Pointer Dereference/Arithmetic	Yes	Yes
Race Condition (a.k.a., "Task Interference")	Yes	Yes
Nested Scheduler Unlock	Yes	Yes
Unsafe Casting	Yes	Yes
Stack Overflow	Yes	Yes

21

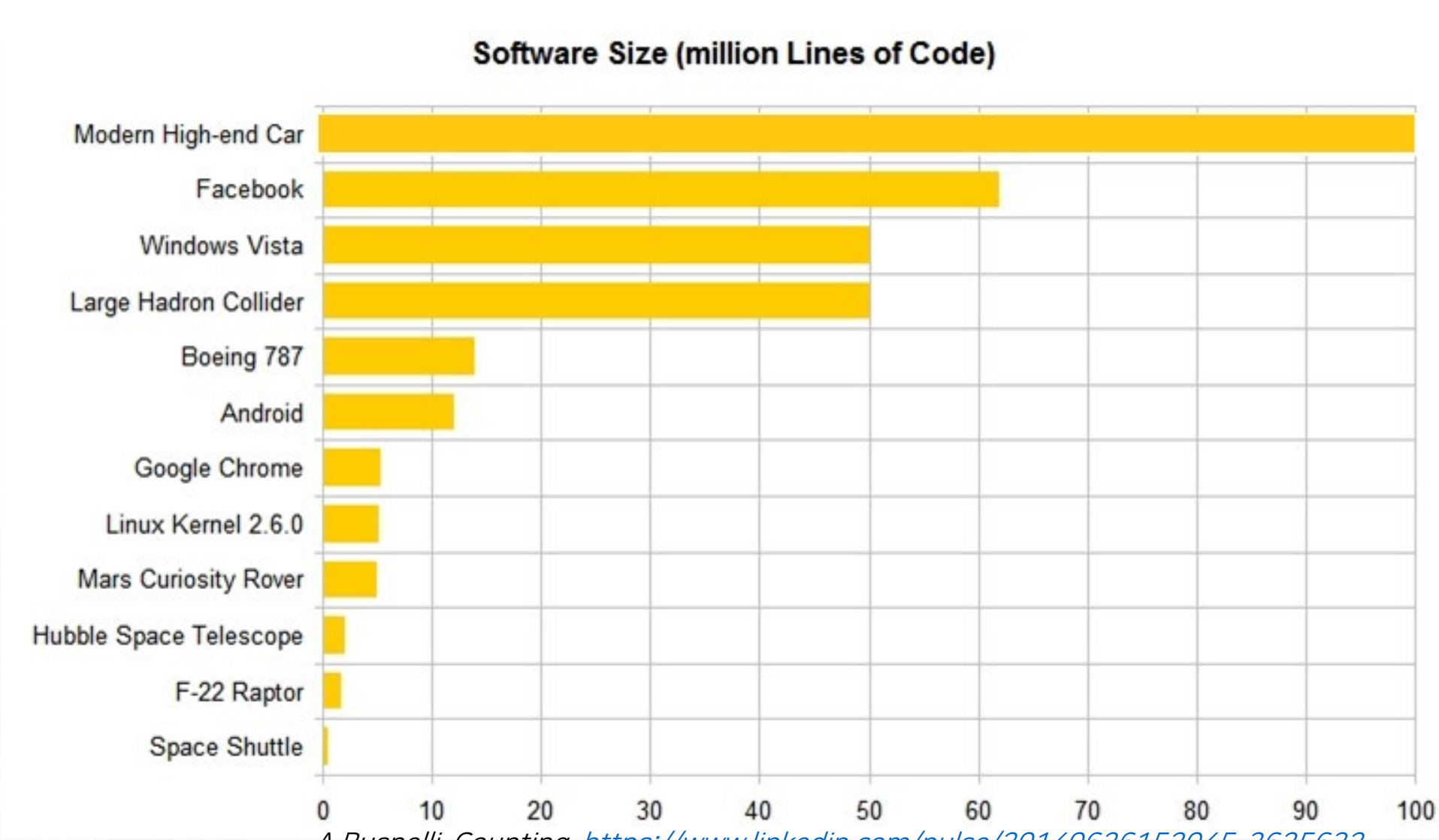
Barr Chapter Regarding Toyota's Software Bugs

BARR group

Static analysis falls short of detecting such complex bugs accurately
- High false negatives
- High false positives

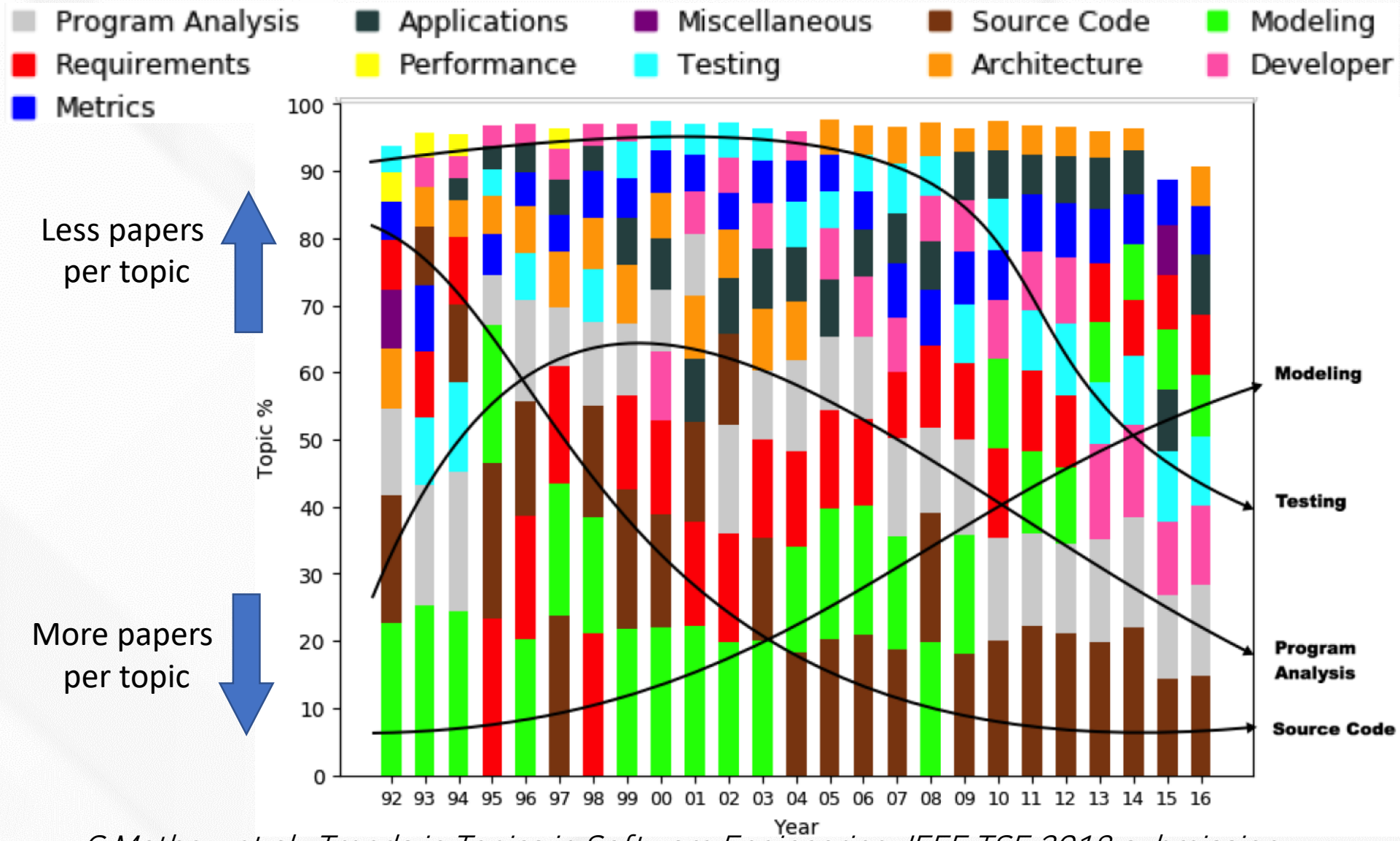
⇒ Systematic and dynamic analysis (i.e. automated sw testing) is MUST for high quality SW

Size and Complexity of Modern SW



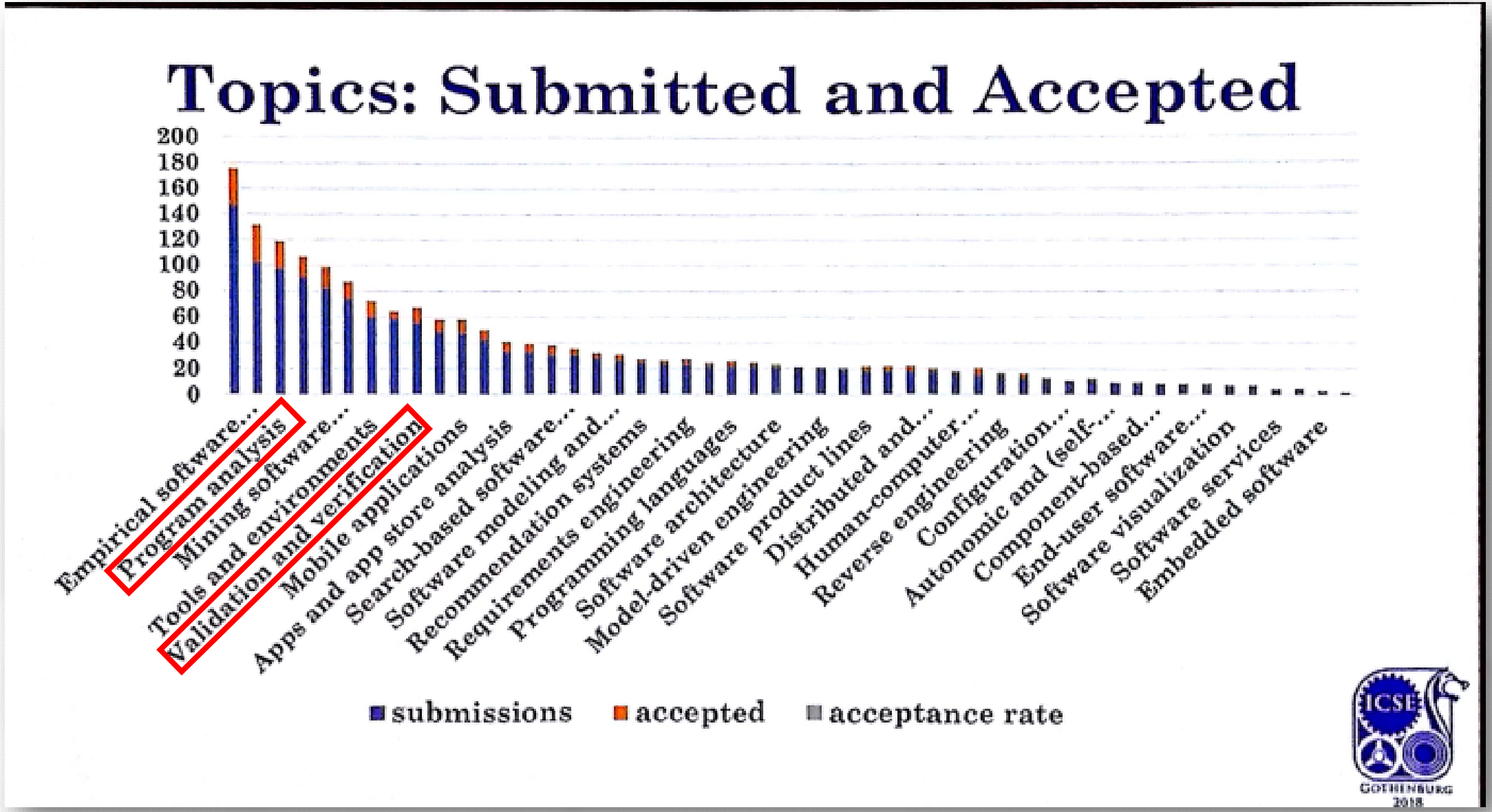
A.Busnelli, Counting, https://www.linkedin.com/pulse/20140626152045-3625632-car-software-100m-lines-of-code-and-counting
http://www.informationisbeautiful.net/visualizations/million-lines-of-code/

SE Research Topic Trends among 11 Major Topics (1992-2016)

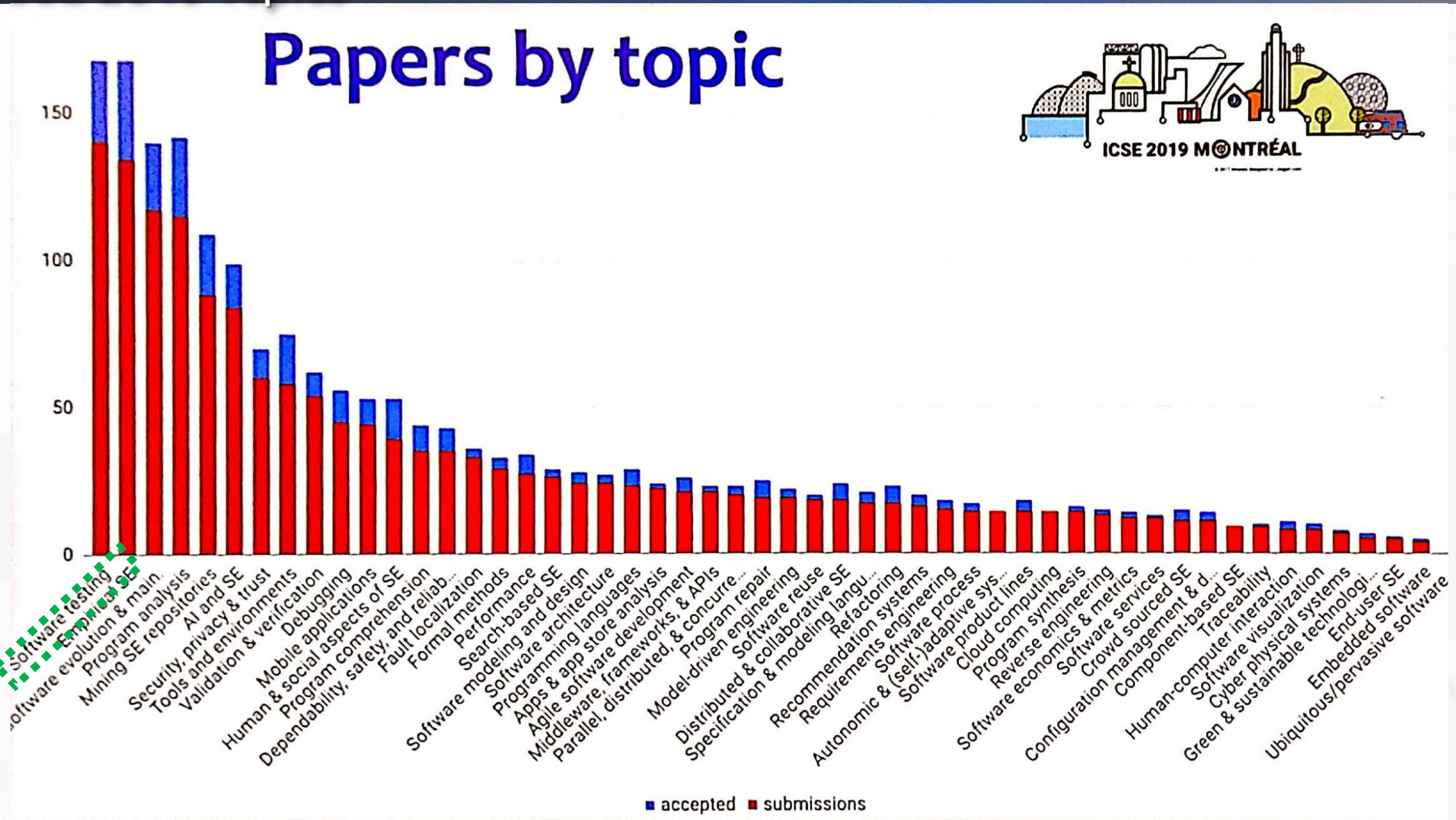


G.Mathew et al., Trends in Topics in Software Engineering, IEEE TSE 2018 submission

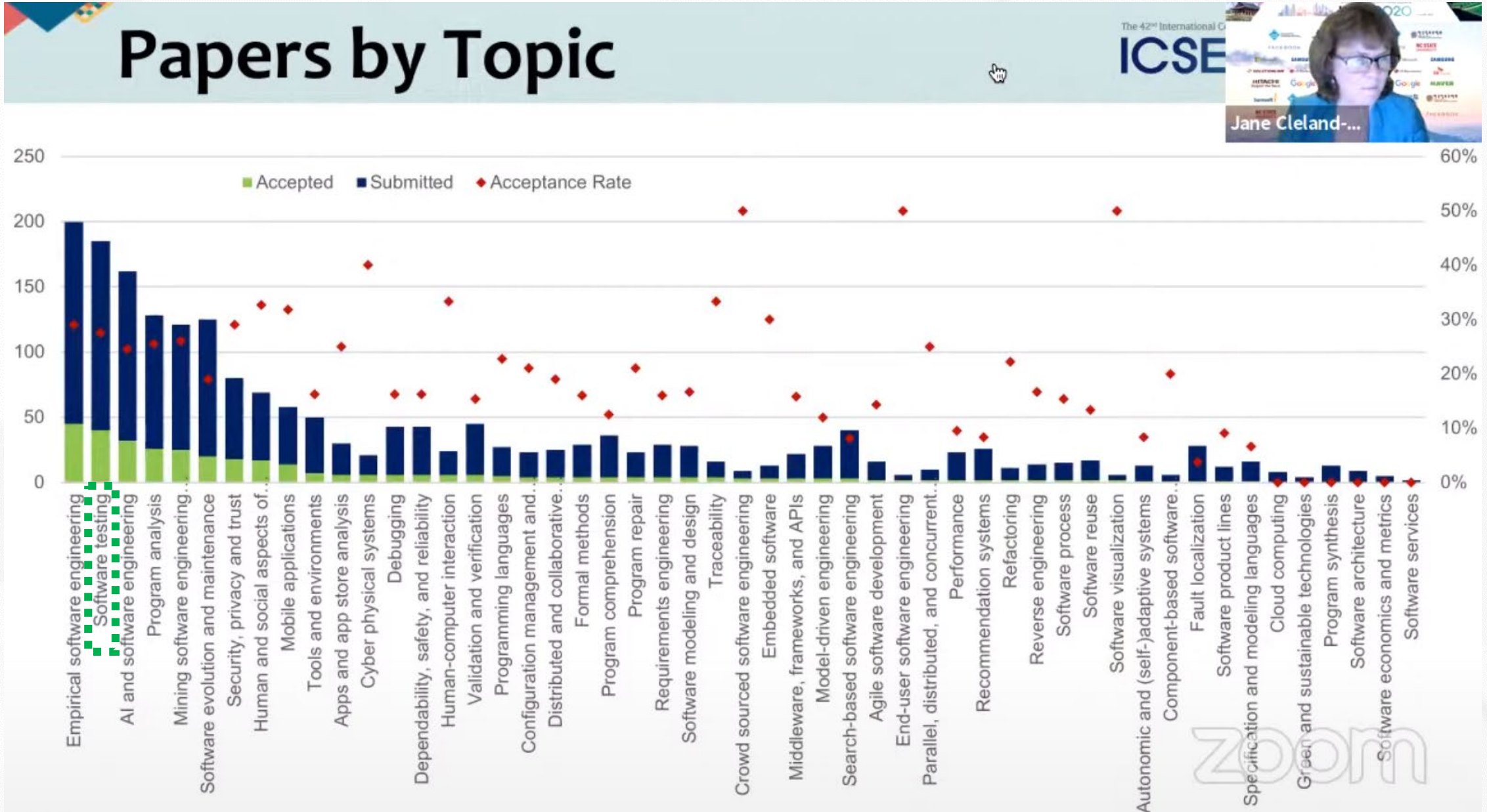
ICSE 2018 Topics (Top SE conf. w/ accept. rate: 20%)



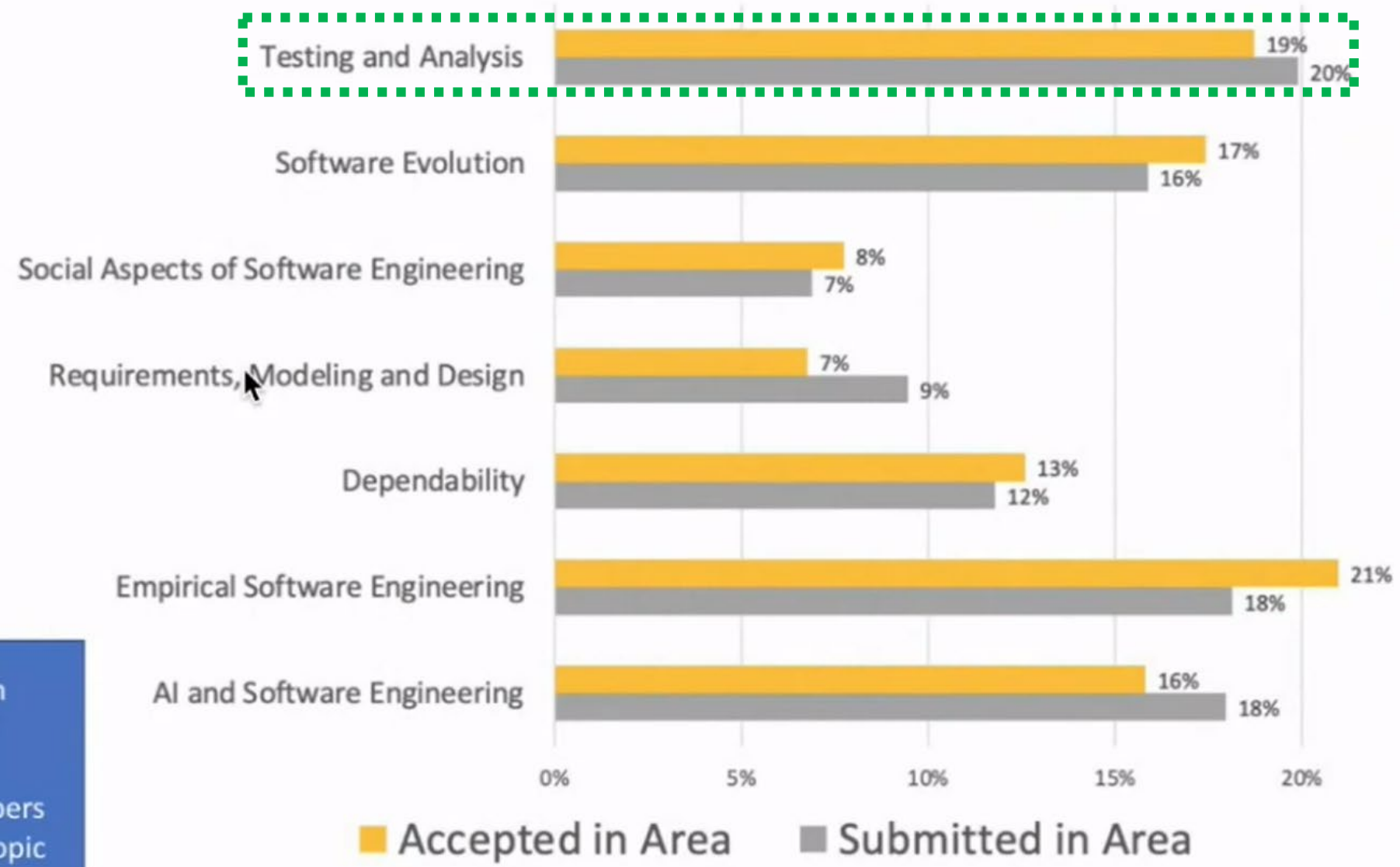
ICSE 2019 Topics



ICSE 2020 Topics



ICSE 2021 Topics



Papers can be in multiple areas

Percentages of papers with at least one topic in given area.

How to Improve the Quality of SW

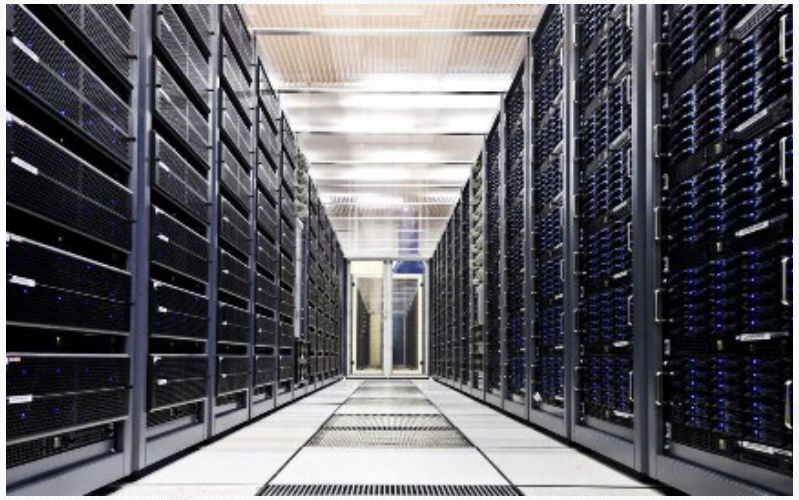
1. Systematic testing (can be still manual)
   - Coverage criteria
   - Mutation analysis
2. Testing through automated analysis tools
   - Scientific treatment of SW with computing power
     - Generate test inputs to detect bugs
     - Localize detected faults
     - Repairing the fault with patches
3. Formal verification
   - Guarantee the absence of bugs

Significance of Automated SW Testing

- Software has become more ubiquitous and more complex at the same time

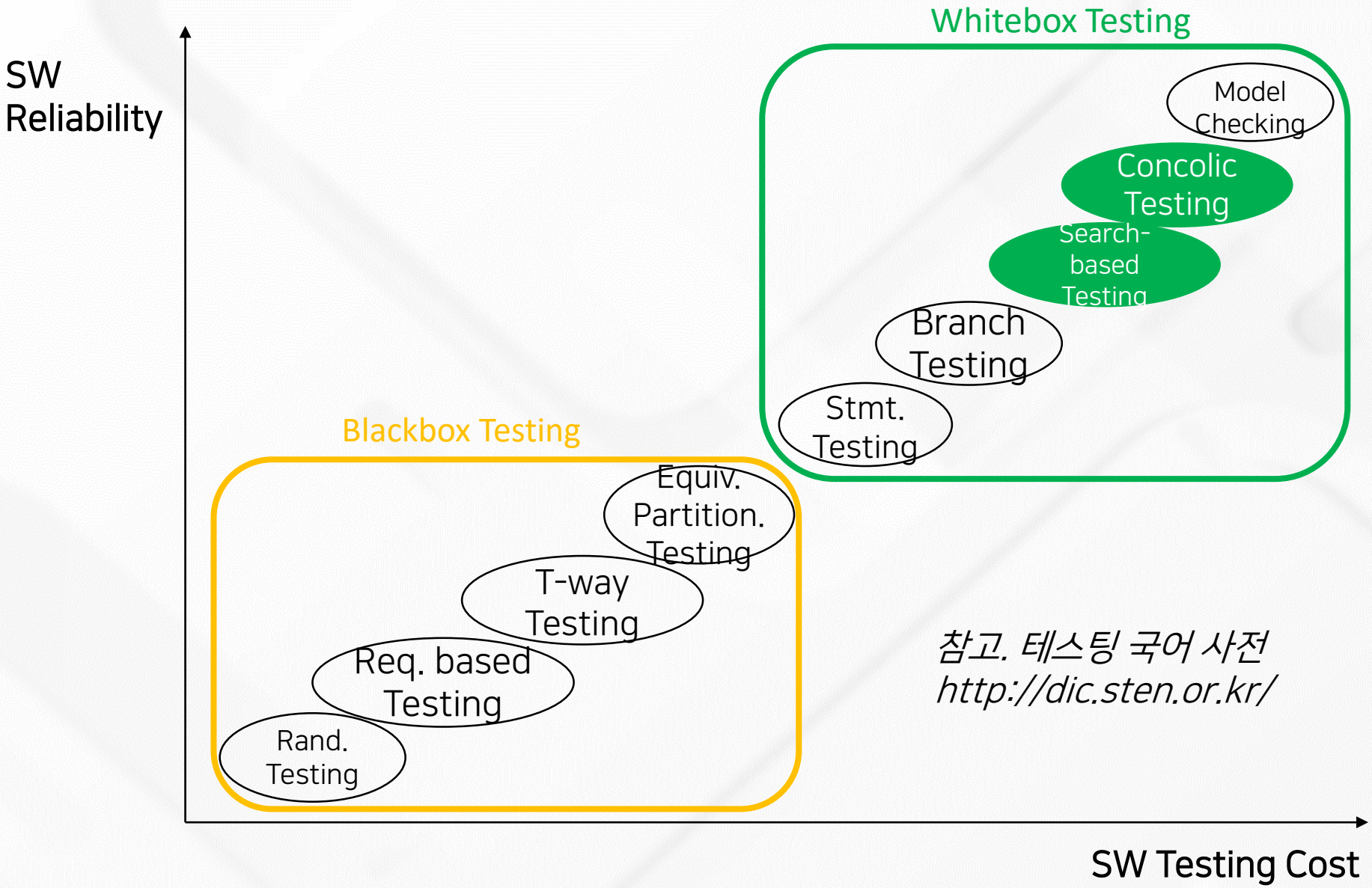
Human resources are becoming **less reliable and more expensive** for highly complex software

Computing resources are becoming **ubiquitous and cheap**
Amazon AWS price: you can use thousands of CPUs @ 0.03$/hr for 2.5Ghz Quad-core CPU



› To-do: Develop **automated and scientific software testing tools** to utilize computing resource effectively and efficiently

Various SW Testing Techniques w/ Different Cost and Effectiveness



참고. 테스팅 국어 사전
http://dic.sten.or.kr/

Strong IT Industry in South Korea

Embedded Software in 2 Different Domains

Conventional Testing

Concolic testing

Model checking

	Consumer Electronics	Safety Critical Systems
Examples	Smartphones, flash memory platforms	Nuclear reactors, avionics, cars
Market competition	High	Low
Life cycle	Short	Long
Development time	Short	Long
Model-based development	None	Yes
Important value	Time-to-market	Safety

Automated Testing Machine (?) vs. Washing Machine

" 6 Interesting Facts About Washing Machines ⋯
Played a Role in Women's Rights

Scholars and many others have long acknowledged the significant role washing machines played in the fight for women's rights. The invention of automatic washing machines reduced the amount of time women spent doing household chores, which allowed them more time to enter the workforce.

https://www.danby.com/blog/6-interesting-facts-washing-machines/

Roadmap for Improving Quality of SW Product and Service

Problem: Huge Economic & Social Cost **due to Software Bugs**

Labor-intensive Manual Testing
Large SW Testing Cost and Time
Low Bug Detection Abilty
Low Product Quality

Solution: AI-based Highly Effective and Low Cost **Automated SW Testing Technique**

movie link https://bit.ly/3NS6RrQ

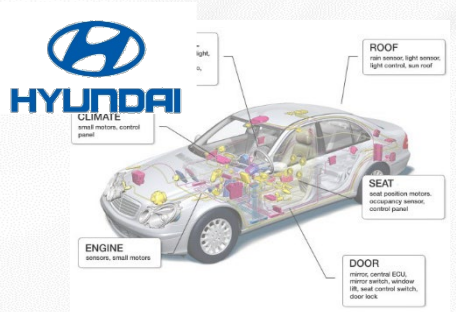
Existing Problems → Developed Solutions

'10-14 Project w/ Samsung Electronics

Detected dozens of crash bugs in the comm. firmware

'18 Project w/ LIGnex1

Detected several SW bugs in the 10 programs in the battleships



'15~20 Project w/ Hyundai/Mobis

Acheived 90% branch cov. and reduced 80% of labor cost by using auto. testing tech.

'20 Project w/ Natl. Security Research Inst.

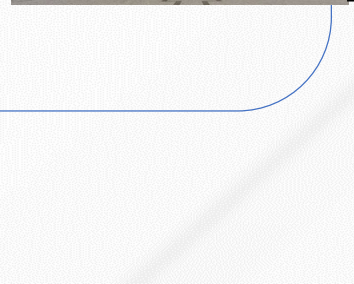
Detected SW bugs in the software in the security equipments

Concolic Unit Testing Project w/ Samsung DMC for 5 years

- Goal: To detect bugs in Samsung smartphones
- Project period: ‘10~’14
- Project funding: 400,000 USD
- Results:
- Developed Concolic unit-testing tool **CONBOL**
- Detected **hundreds crashes** in 4 MLOC smartphone SW

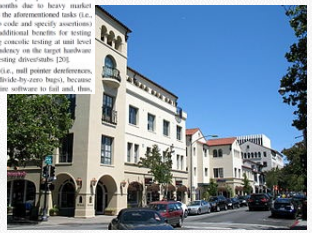
Industrial project progress

2010 2011 2012 2013 2014



- FSE '2011

- ICSE 2012, ICST 2012



- ASE 2013

Successful Industrial Case: AI-based Automated SW Testing System

현대모비스, AI 기반 소프트웨어 검증시스템 도입..."효율 2배로"

2018-07-22 10:00

댓글 ••• 가- 가+

'마이스트' 적용...대화형 검색 로봇 '마이봇'도 도입

(서울=연합뉴스) 윤보람 기자 = 현대모비스[012330]가 인공지능(AI)을 활용해 자율주행, 커넥티비티(연결성) 등 미래 자동차 소프트웨어(SW) 개발에 속도를 낸다.

현대모비스는 AI를 기반으로 하는 소프트웨어 검증시스템 '마이스트'(MAIST: Mobis Artificial Intelligence Software Testing)를 최근 도입했다고 22일 밝혔다.

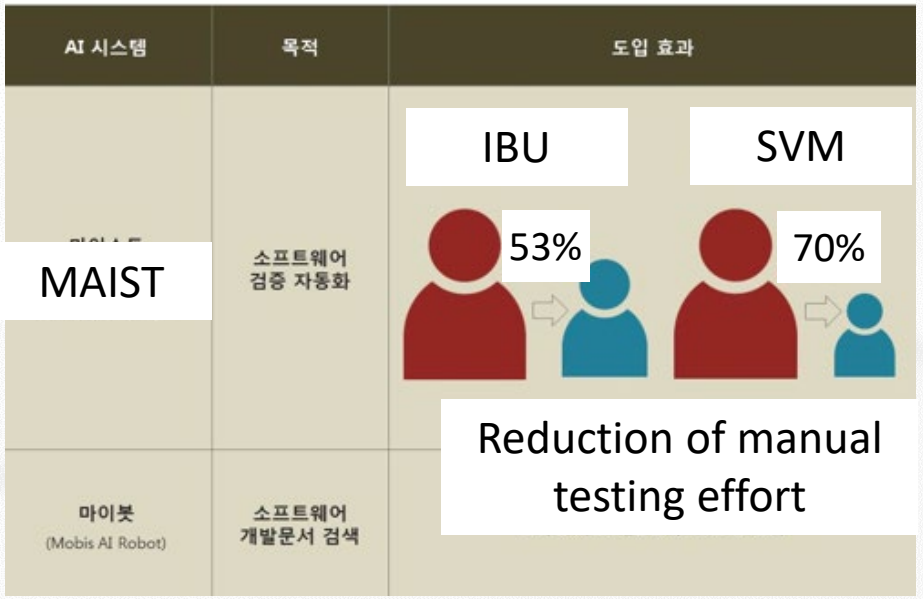
Hyundai Mobis and a research team lead by Prof. Moonzoo Kim at KAIST jointly developed MAIST for automated testing

MAIST automates unit coverage testing performed by human engineers by applying concolic unit testing

MAIST can reduce 53% and 70% of manual testing effort for IBU(Integrated Body Unit) and SVM(Surround View Monitoring)

현대모비스는 하반기부터 소프트웨어가 탑재되는 제동, 조향 등 모든 전장부품으로 마이스트를 확대 적용할 계획이다. 글로벌 소프트웨어 연구기지인 인도연구소에도 적용한다.

■ 현대모비스 인공지능 도입 사례



http://m.yna.co.kr/kr/contents/?cid=AKR20180720158800003&mobile

MAIST Paper Presentation @ ICSE 2019 SEIP Track

2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)

Concolic Testing for High Test Coverage and Reduced Human Effort in Automotive Industry

Yunho Kim
School of Computing
KAIST
Daejeon, South Korea
yunho.kim03@gmail.com

Dongju Lee
Software Verification Team
Hyundai Mobis
Yongin, South Korea
dongju.lee@mobis.co.kr

Junki Baek
Software Verification Team
Hyundai Mobis
Yongin, South Korea
jk.baek@mobis.co.kr

Moonzoo Kim
School of Computing
KAIST
Daejeon, South Korea
moonzoo@cs.kaist.ac.kr

***Abstract*—The importance of automotive software has been rapidly increasing because software now controls many components in motor vehicles such as window controller, smart-key system, and tire pressure monitoring system. Consequently, the automotive industry spends a large amount of human effort testing automotive software and is interested in automated software testing techniques that can ensure high-quality automotive software with reduced human effort.**

In this paper, we report our industrial experience applying concolic testing to automotive software developed by Hyundai Mobis. We have developed an automated testing framework MAIST that automatically generates the test driver, stubs, and test inputs to a target task by applying concolic testing. As a result, MAIST has achieved 90.5% branch coverage and 77.8% MC/DC coverage on the integrated body unit (IBU) software. Furthermore, it reduced the cost of IBU coverage testing by reducing the manual testing effort for coverage testing by 53.3%.

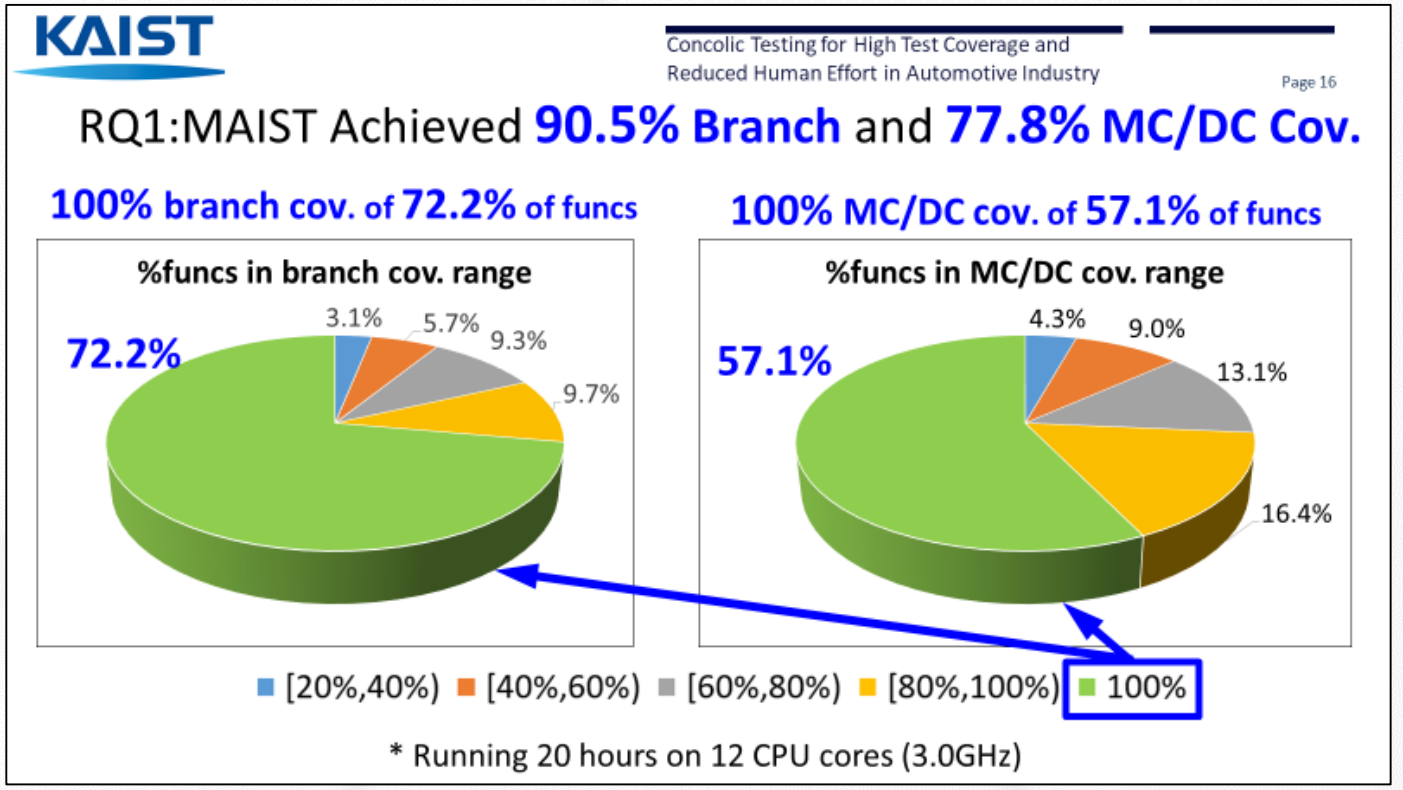
***Keywords*-Automated test generation, concolic testing, automotive software, coverage testing**

I. Introduction

The automotive industry has developed automotive software to control various components in the motor vehicle, for example, the body control module (BCM), smart-key system (SMK), and tire pressure monitoring system (TPMS) [1], [2]. As automotive software becomes larger and more complex with the addition of newly introduced automated features (e.g., advanced driver assistance systems) and more sophisticated functionality (e.g., driving mode systems) [3], [4], the cost of testing automotive software is rapidly increasing. Also, it is difficult for human engineers to develop test inputs that can ensure high-quality automotive software within tight scale embedded software [10]) and has effectively improved the quality of industrial software by increasing test coverage and detecting corner-case bugs with modest human effort.

While we were working to apply concolic testing to automotive software developed by Mobis, we observed the following technical challenges to resolve to successfully apply automated test generation techniques:

1) We need to generate test drivers and stubs carefully to achieve high unit test coverage while avoiding generating test cases corresponding to the executions that are not feasible at the system-level. Otherwise (e.g., generating naive test drivers and stubs that provide unconstrained symbolic inputs to every function in a target program), we will waste human effort to manually filter out infeasible tests that lead to misleading high coverage and false alarms.
2) Current concolic testing tools do not support symbolic bit-fields in C which are frequently used for automotive software.[1] For example, automotive software uses bit-fields in message packets in the controller area network (CAN) bus to save memory and bus bandwidth. However, most concolic testing tools do not support symbolic bit-fields since a bit-field does not have a memory address (Sect. III-D) and most programs running on PCs rarely use bit-fields.
3) Although automotive software uses function pointers to simplify code to dynamically select a function to execute, concolic testing techniques and tools do not support symbolic function pointers due to the limitation of SMT (Satisfiability Modulo Theories) solvers.



ICSE 2019 SEIP Acceptance ratio: 20%

Final Remarks

- For undergraduate students:
  - Highly recommend URP studies or independent studies
    - 이아청 detected several crash bugs in Hyundai Mobis SW during 2018 summer interns
- For graduate students:
  - Welcome research discussions to apply SW analysis techniques
    - Systematically testing/debugging C/C++ programs
- Pre-requisite:
  - Familiar with C/C++ programing
  - Basic understanding of linux utilities and shell scripts
- Expected HW time:
  - >6 hours of analysis/programming per week

25

26