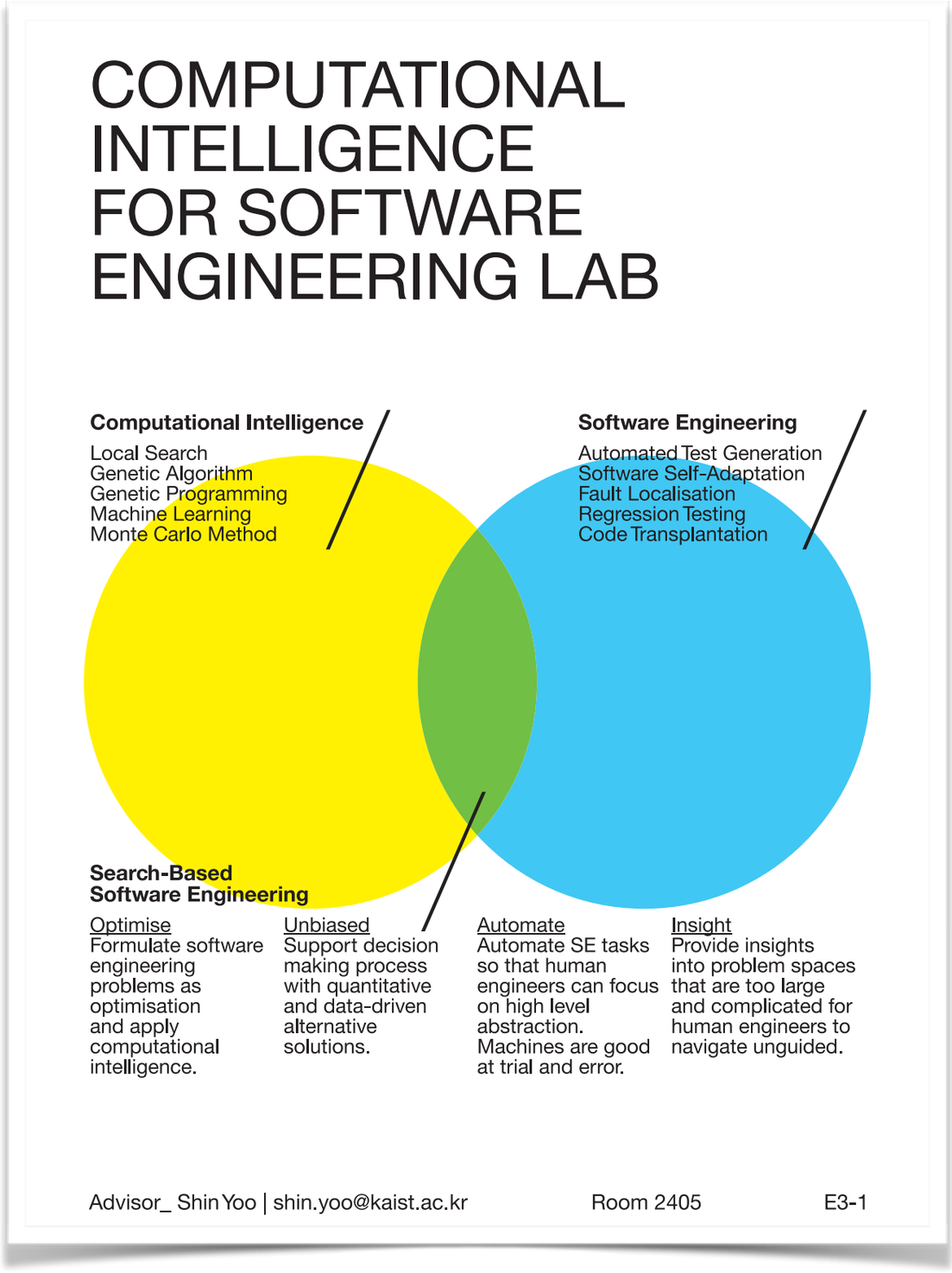
Introduction to Search Based Software Engineering

2018 KIISE SE Society Summer School
School of Computing, KAIST
Shin Yoo

Me

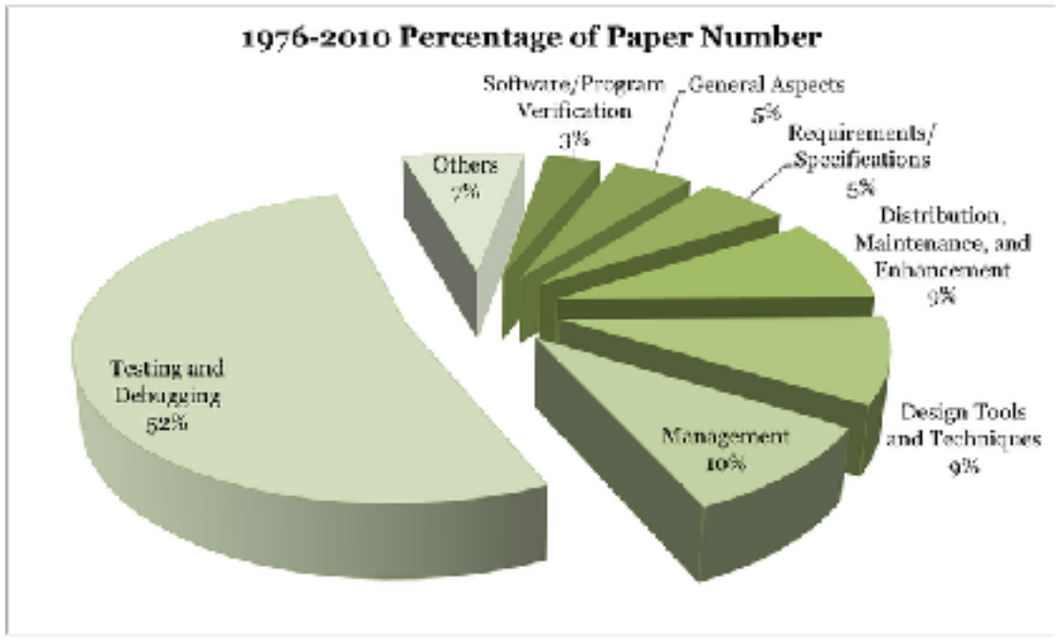
- Shin Yoo, Associate Professor, School of Computing, KAIST
  - KAIST 2015-
  - Assistant Professor at University College London, UK 2012-2105
  - Postdoc 2009-2012
  - PhD at King's College London, UK 2006-2009
- COINSE (Computational Intelligence for Software Engineering) Lab
- Research interest: SBSE, regression testing, automated debugging, evolutionary computation, information theory, program analysis…
- shin.yoo@kaist.ac.kr



What are we doing here?

- SBSE is vast now: it applies to almost the entire SDLC (although testing has been the major target, just like any SE research).

- It is an **approach** rather than a **domain.** I cannot really lecture about a single technique that always works - rather, my goal is to encourage you **to think about your specific problem in terms of meta-heuristic**.

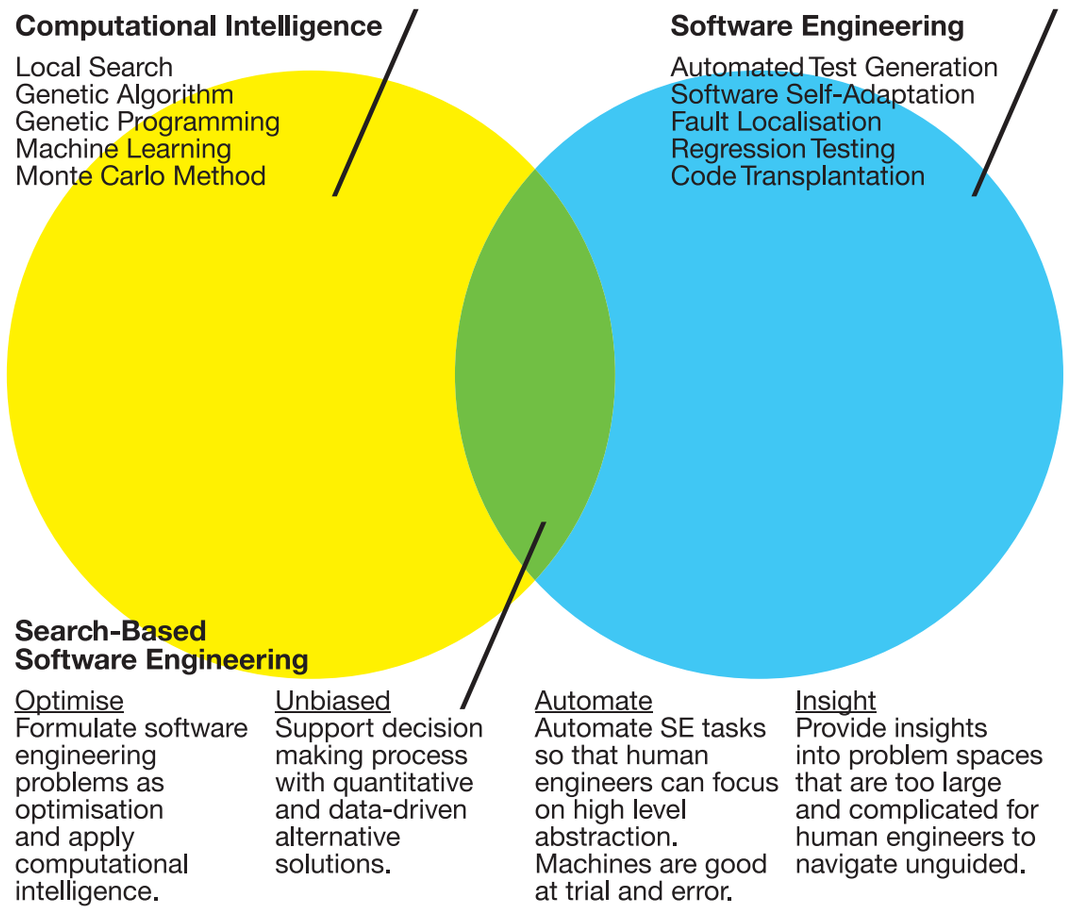


Algorithms vs. Problem Instances

- What I CAN teach you is algorithms: a very small subset of what we call computational intelligence.

- For historical reasons, SBSE has been closely related to evolutionary methods.

- But the way of thinking applies to any other machine learner in general.



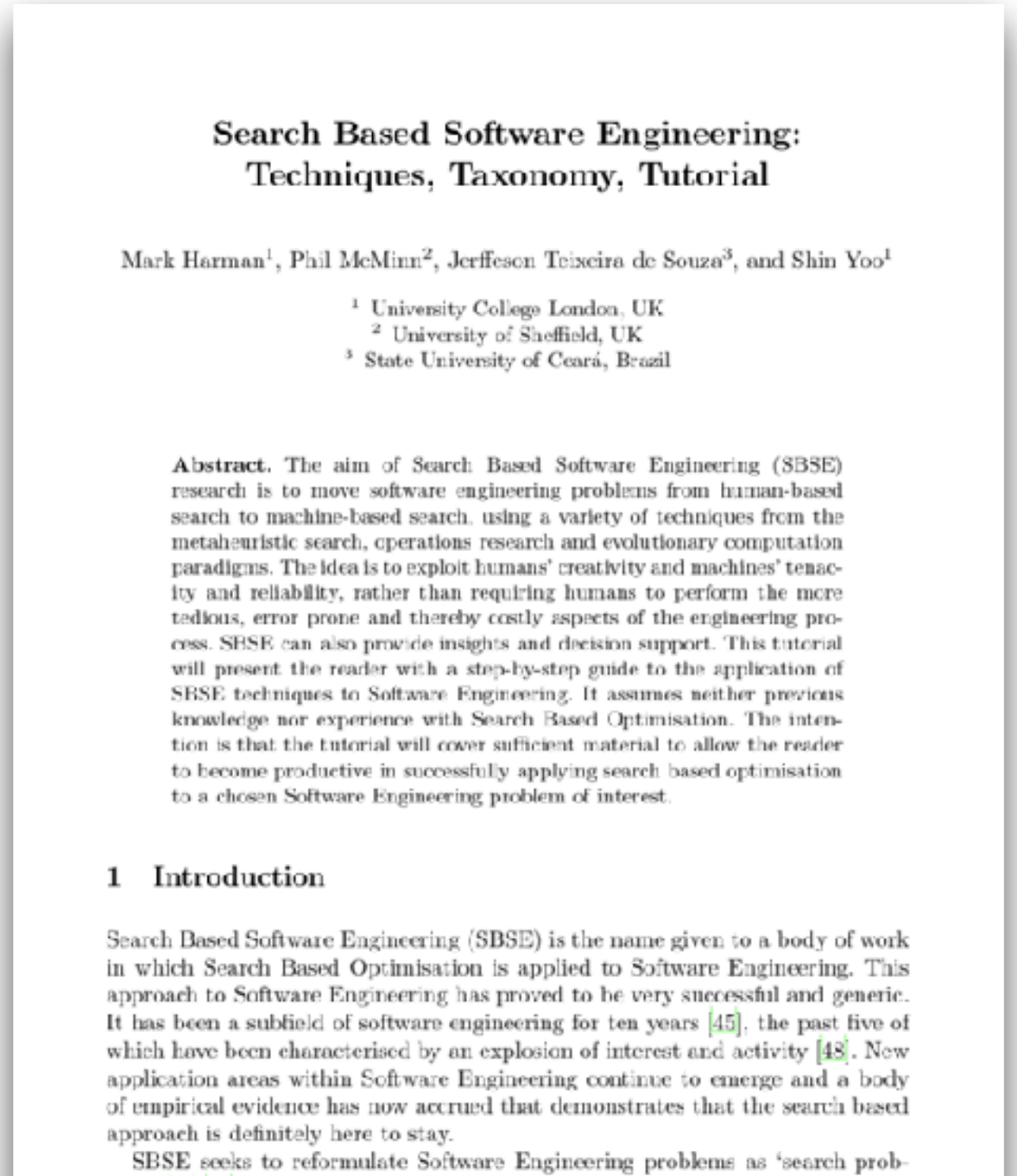
Course Outline: Day 1

- 13:00-14:00 Introduction to SBSE

- 14:00-16:00 Fitness Landscape, Random Search, Local Search

- 16:00-18:00 Evolutionary Computation

M. Harman, P. McMinn, J. T. de Souza, and S. Yoo. Empirical Software Engineering and Verification, volume 7007 of Lecture Notes in Computer Science, chapter Search Based Software Engineering: Techniques, Taxonomy, Tutorial. Springer–Verlag, 2012.



Search Based Software Engineering: Techniques, Taxonomy, Tutorial

Mark Harman[1], Phil McMinn[2], Jerffeson Teixeira de Souza[3], and Shin Yoo[1]

[1] University College London, UK
[2] University of Sheffield, UK
[3] State University of Ceará, Brazil

Abstract. The aim of Search Based Software Engineering (SBSE) research is to move software engineering problems from human-based search to machine-based search, using a variety of techniques from the metaheuristic search, operations research and evolutionary computation paradigms. The idea is to exploit humans' creativity and machines' tenacity and reliability, rather than requiring humans to perform the more tedious, error prone and thereby costly aspects of the engineering process. SBSE can also provide insights and decision support. This tutorial will present the reader with a step-by-step guide to the application of SBSE techniques to Software Engineering. It assumes neither previous knowledge nor experience with Search Based Optimisation. The intention is that the tutorial will cover sufficient material to allow the reader to become productive in successfully applying search based optimisation to a chosen Software Engineering problem of interest.

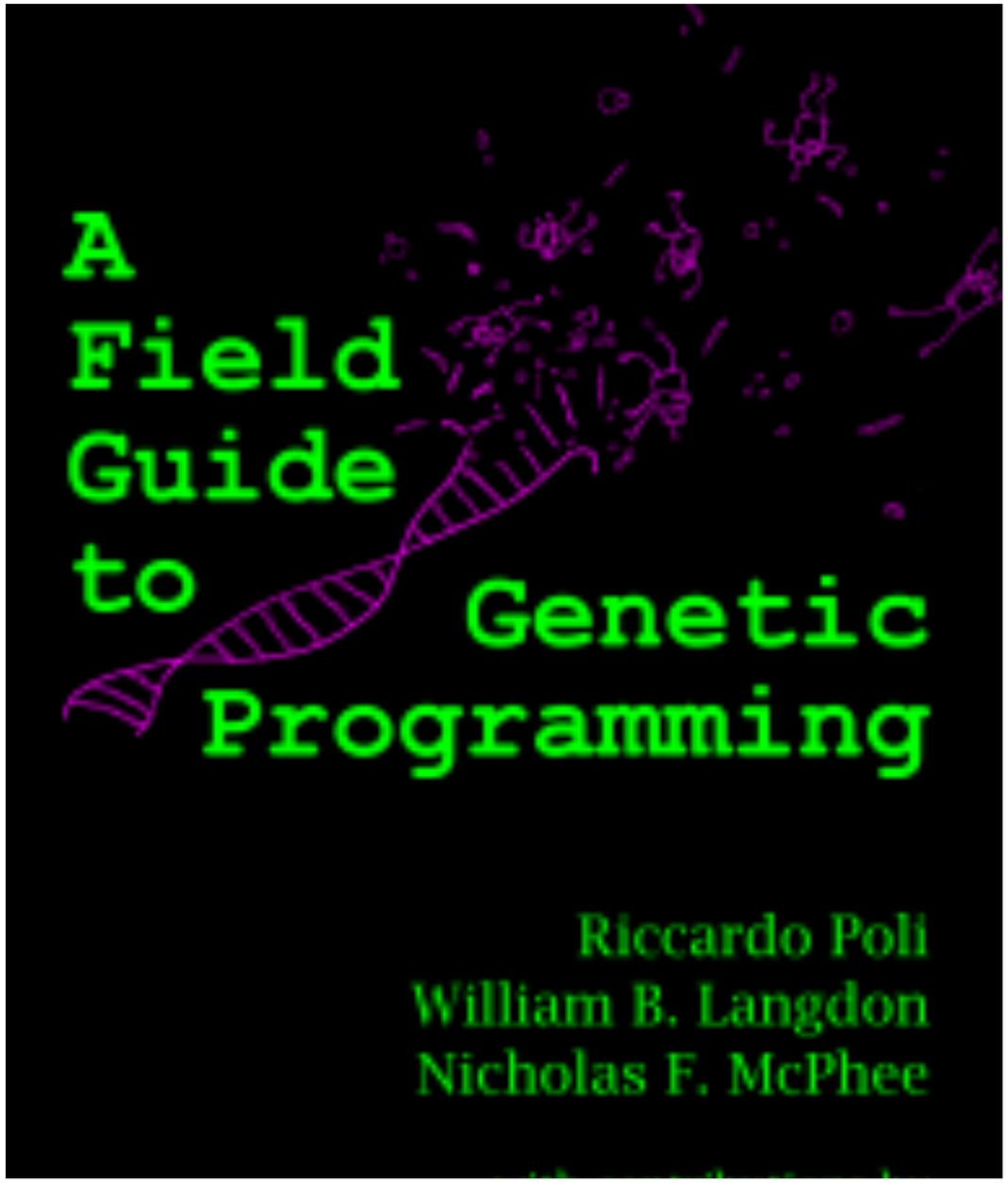
1 Introduction

Search Based Software Engineering (SBSE) is the name given to a body of work in which Search Based Optimisation is applied to Software Engineering. This approach to Software Engineering has proved to be very successful and generic. It has been a subfield of software engineering for ten years [45], the past five of which have been characterised by an explosion of interest and activity [48]. New application areas within Software Engineering continue to emerge and a body of empirical evidence has now accrued that demonstrates that the search based approach is definitely here to stay.

SBSE seeks to reformulate Software Engineering problems as 'search prob-

Course Outline: Day 2

- 09:00-11:00: TSP Hands-on
- 11:00-12:00: Genetic Programming
- 13:00-14:00: Search Based Test Data Generation
- 14:00-15:00: Multi-Objective Evolutionary Algorithms
- 15:00-17:00: GP based Fault Localisation Hands-on
- 17:00-18:00: Bio-inspired Algorithms



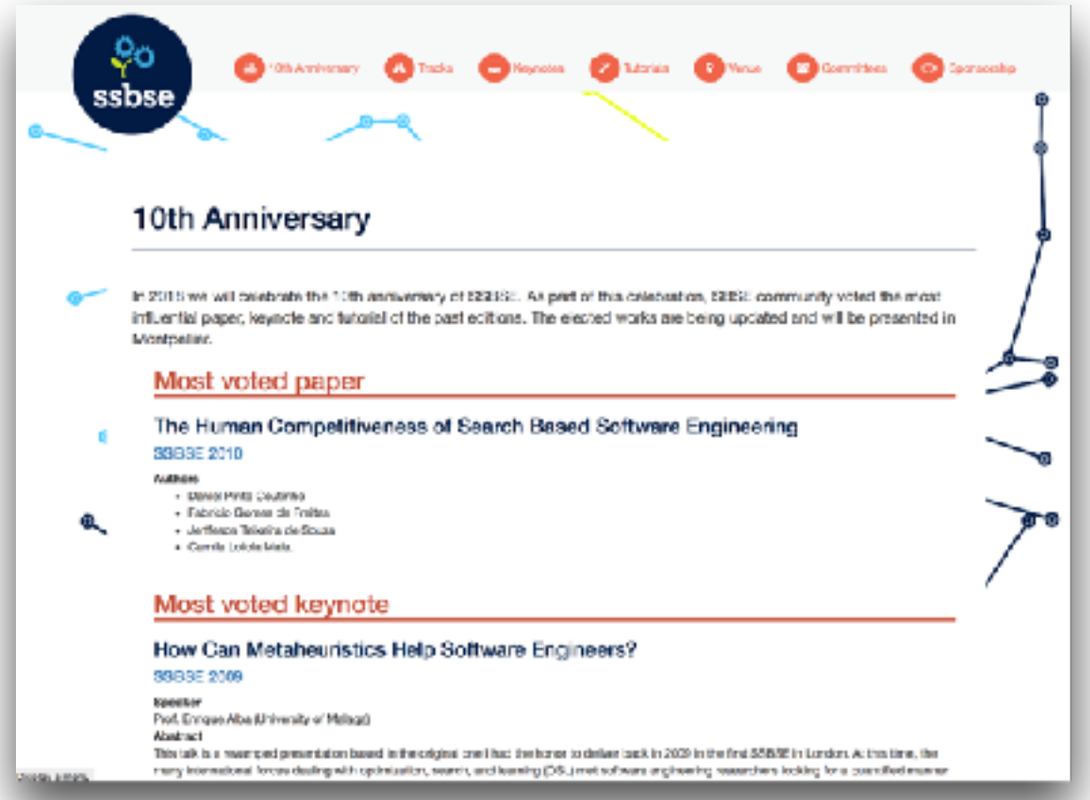
R. Poli, W. B. Langdon, and N. F. McPhee. A field guide to genetic programming. Published via http://lulu.com and freely available at http://www.gp-field-guide.org.uk, 2008. (With contribu- tions by J. R. Koza)

Course Outline: Day 3

- 09:00-10:00: EvoSuite Hands-on

- 10:00-11:00: Search Based Self Adaptation

- 11:00-12:00: GPGPU and SBSE



SSBSE 2018 http://ssbse18.irisa.fr/ (collocated with ASE 2018)

Be Active

- Deliberately designed to:
  - include more interactive hands-on activities
  - somewhat loose schedule ;) so that we can be spontaneous (also will make sure we take sufficient breaks)
  - stop me at any point to ask questions or give comments

Materials

- Clone the repository: https://github.com/ntrolls/sesoc2018

- Install the dependencies this evening for tomorrow's hands-on sessions