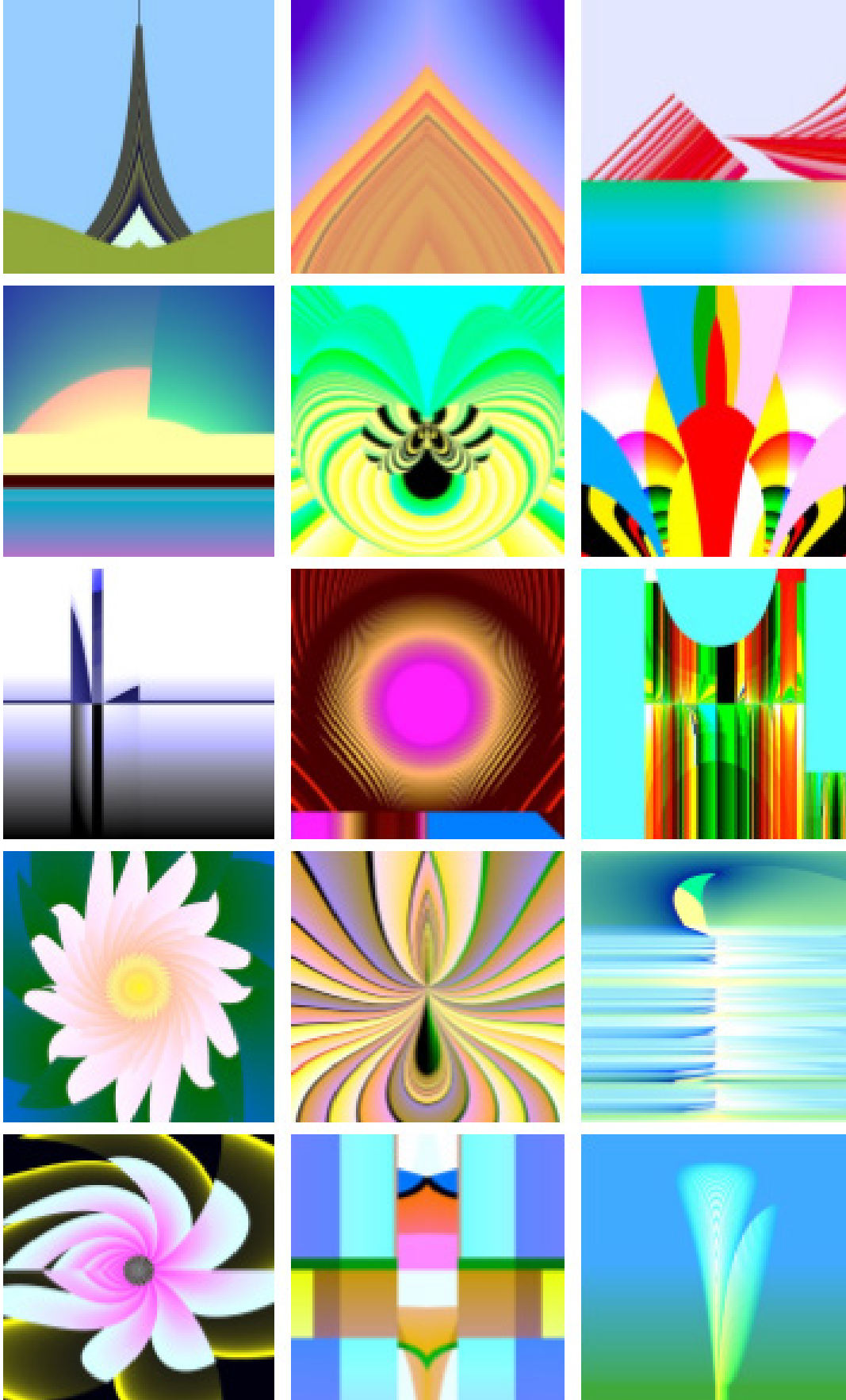


SIGEVolution

newsletter of the ACM Special Interest Group on Genetic and Evolutionary Computation

Volume 12
Issue 2



in this issue

EuroGP
2019 Panel
Discussion:
What is
the Killer
Application of
GP?

EvoStar
2019, Leipzig,
Germany

2019 Awards
for Prof. Mark
Harman

Welcome to the second 2019 newsletter! Our cover illustrates co-evolutionary artwork by **Alan Blair**; starting from images of landmark buildings and natural objects, a virtual evolutionary artist competes against a deep convolutional neural network art-critic producing intriguing results. Our first article documents the inspiring and at times heated EuroGP 2019 panel discussion organized by **Ting Hu** and **Lukas Sekanina**. The discussion was kicked off by the following questions: What is the killer application that will make GP become a part of our everyday lives? What are the application problems that GP is particularly suitable for? What can set GP apart from other learning algorithms, and allow it to stand out in the next waves of AI research? **Gusz Eiben**'s stated his inspiring and sound research motto: "Given the fact that evolution can produce intelligence, it is plausible that Artificial Evolution can produce Artificial Intelligence." **James Foster**'s commented that he was "amused to see that we panelists not only disagree about what a 'killer app' is, we disagree about what 'GP' is! We continue with a lively recount by **Sarah Thomson** of the EvoStar 2019 highlights in the beautiful city of Leipzig, Germany this Spring. We also report on the outstanding achievements of Prof. **Mark Harman**, the founder of Search-based Software Engineering, who obtained both ACM and IEEE awards in 2019 (a contribution from **Bill Langdon**).

We take the opportunity to let our SIGEVO members know the results of the 2019 ACM SIGEVO election (for the term of 1 July 2019 – 30 June 2025), which can be found at <https://www.acm.org/elections/sigs/sigevo-2019-results>. Our warmest welcome to the newly elected members of the executive committee (in alphabetical order): **Hernán Aguirre**, **Enrique Alba**, **Peter A.N. Bosman**, **Anna Isabel Esparcia-Alcázar**, **Manuel López-Ibáñez** and **Markus Wagner**.

As ever, please get in touch if you would like to contribute an article for a future issue or have suggestions for the newsletter.

Gabriela Ochoa, Editor.

About the Cover

The artworks on the front cover are created through artist-critic coevolution between a genetic program (HERCL) artist and a convolutional neural network (LeNet) critic. Hierarchical Evolutionary Re-Combination Language (HERCL) is a general purpose GP paradigm which keeps the number of competing individuals small, while still preserving diversity, and also does a good job of encouraging shorter programs and preventing bloat. In the present work, the HERCL program acts as a function from pixel values x, y to intensity values R, G, B . In each round, a new critic is trained by gradient descent to accept real photographs of a famous landmark and reject images generated by the artist in all previous rounds; then, a new artist is evolved to produce an image that is acceptable to that critic. Code for previous images is available for recombination, allowing the artist to revisit and further develop earlier themes. The resulting AI artist, whimsically named Hercule LeNet, produces images which have low algorithmic complexity but are nevertheless realistic enough to fool the adversarially trained deep learning critic. After several hundred rounds, rich and varied artworks are produced, exhibiting such diverse artistic styles as minimalism, impressionism, fauvism, pointillism and suprematism as well as psychedelic and fractal art.

This work was presented at EvoMusArt 2019 in the paper "Adversarial evolution and deep learning - how does an artist play with our visual system?"

Further details and images can be found at <https://PickArtSo.com>

Please contact Alan Blair (<http://www.cse.unsw.edu.au/~blair>) if you would like to discuss this work.

EuroGP 2019 Panel Discussion: What is the Killer Application of GP?

Ting Hu and Lukas Sekanina
EuroGP 2019 Program Chairs

Introduction:

While the Internet existed as far back as the 1960s, it was not until the advent of the World Wide Web in the early 90s, that this technology began impacting our everyday lives as an information dissemination tool. This application is so dominant that many would treat the Web as the synonym of the Internet. A technique only takes off when it finds a vehicle to communicate with the wide audience, i.e., through a killer application.

What is the killer application that will make GP become a part of our everyday lives? What are the application problems that GP is particularly suitable for? What can set GP apart from other learning algorithms, and allow it to stand out in the next waves of AI research?



Panelists:

Gusz Eiben, Vrije Universiteit Amsterdam, the Netherlands

Gabriela Ochoa, University of Stirling, Scotland

James Foster, University of Idaho, USA

Risto Miikkulainen, University of Texas at Austin and Cognizant, USA



Moderator:
Ting Hu, Queen's University, Canada

Gusz Eiben

My research motto that connects Artificial Intelligence and Evolutionary Computing is the following:

Given the fact that evolution can produce intelligence, it is plausible that Artificial Evolution can produce Artificial Intelligence.

Of course, there are many forms of AI and one way to classify them is to distinguish digital and embodied AI. Think, for instance of Siri versus the Nao.

My personal favorite is embodied intelligence. This brings us to using Artificial Evolution / Evolutionary Computing to make robots intelligent. This is not new per se, as the field of evolutionary robotics is already well established with the seminal book dating back to the year 2000. What is new(ish) in my statement here is the firm belief that GP can be a key technology to this end. As it happens the huge majority of studies that address evolving good robot controllers employs neural networks. However, since **Koza's** artificial ants example we know that GP is applicable for developing / designing controllers for virtual creatures. GP is then also applicable for developing / designing controllers virtual for physical robots.

One may ask why we should bother about using GP if neural networks can do the job. The answer is: *explainability*. As we all know explainability of AI systems is becoming more and more important. Some even say that it is essential for acceptance and for creating a legal framework. And this is where GP provides an advantage over NNs. On a more generic level, symbolic AI approaches are more amenable in this respect and I see GP as a combination of symbolic and non-symbolic, a.k.a. top-down and bottom-up, approaches. The best of both worlds, indeed. To realize this potential we need more research and a specific research agenda

into this application area.

To summarize, my answers to the kick-off questions are:

1. What is the killer application that may make GP become a part of our everyday lives? Evolving controllers for digital agents and physical robots.

2. What are the applications problems that GP is particularly suitable for?

Evolving controllers for digital agents and physical robots.

3. What can set GP apart from other learning algorithms and allow it to stand out in the next waves of AI research?

The eligibility / readability of the solutions it provides. These make the solutions more safe and explainable than the current options.

Gabriela Ochoa

GP has already proved to be a powerful problem-solving and design tool in many domains, so there are already several killer-applications that are part of our everyday lives! The strength of Evolutionary Computation is that it facilitates the emergence of novelty and unexpected results, while allowing the incorporation of expert knowledge in the solution designs. Another strength of EC that sets it apart from other computational intelligence techniques is the ability of incorporating multiple and often conflicting objectives.

An application domain with many recent success stories is *search-based software engineering*. EAs and GP have been used for the automatic detection and repair of software defects (bugs), improving the efficiency of software, and even automatically transplanting software artifacts across different devices and systems. With automatic software test-generation the testing process can be made faster, more comprehensive, and more effective. The programs can automatically find crashes, pinpointing the line responsible for the crash. An automated debugging tool, named *SapFix*, has been already deployed at Facebook by **Mark Harman** and his team. *SapFix* mines from a pool of software fixes most often used by engineers in the past and generates and proposes the most effective fixes for the issue at hand. Thus, it can significantly reduce the amount of time engineers spend on debugging. These GP/EC empowered tools accelerate the process of shipping robust, stable code updates to millions of devices, and liberate programmers for more creative tasks.

GP has shown innovative applications in healthcare. An example is the home monitoring device for Parkinson's dyskinesia, a project led by **Stephen L. Smith** and **Michael Lones** that won the Hummies Gold award in 2018. Dyskinesia is defined by involuntary jerking and spasms of the muscles that can affect the whole body. The monitoring device is based on a predictive model that identifies episodes of dyskinesia from accelerometer time-series data. Cartesian GP was used to evolve symbolic mathematical expressions to characterise movements in the time domain. The model is able to discriminate movements based on their overall shape, rather than on frequencies, which sets it apart from alternative models. This home monitoring device has been approved for European clinical use and is already in routine use internationally including three large UK hospitals, Leeds, Harrogate, and Scarborough, and the Ruijin Hospital in Shanghai, China.

James Foster

In fact, there are several different meanings to the term "killer app" (KA). In a marketing sense, a KA is a program whose functionality is so important that consumers will buy the hardware just to be able to run that app. The first KA was VisiCalc, and the term KA was coined in court when Microsoft was sued for having monopoly power from its KA. I suspect GP will never be a KA in this sense. GP is much more likely to be used "under the hood", and so consumers are unlikely to be aware of it.

A KA could also be hardware or software that makes a ubiquitous technology possible. In this sense, GP is already a KA. GP is used to design volatile memory in cell phones. I think it has also been used to design airplane parts. There are other examples. Given that we use phones and airplanes, we already have our KA. I suspect this is unsatisfactory because we don't really want a KA. Rather, we want the attention that a KA would bring.

Another possible definition of KA is a software product with sales high enough that we can make a living off it. I suspect this is what most of us have in mind. But if this is the case, then GP is already a KA. **Risto**, for example, is doing very well using GP. There are other examples, too. Why are we asking this question now? We have complained about lack of visibility for as long as I've been in this field. But I think we are jealous of the press, funding, and career prospects of deep learning. Remember, this too shall pass.

Good marketing is the key. It is really the time to think how to make GP more visible, taking advantage of current public hype on AI and machine learning. Why is GP the last resort algorithm? To raise public awareness, having accessible and usable tools is important. But so is an enduring entity tasked with actual marketing. Perhaps BEACON can do this? Or SPECIES? Or perhaps the solution is to stop worrying about being recognized. Let's just keep advancing the science and technology of AI (as **Gusz** and **Risto** argue) and automatic programming, because that's what we like to do. We can make a difference, too, as **Gabriela** reminds us. There is value in being a happy scientist or engineer, even if we are unknown.

I was amused to see that we panelists not only disagree about what a "killer app" is, we disagree about what "GP" is! I think genetic programming is using evolutionary techniques to produce computational artifacts—programs, in hardware, software, or simulation. **Gusz** argued that "GP" requires a specific syntax. Perhaps we should reclaim "automatic programming", as **Steve Gustafson** has argued. I have found the PR value of "I teach computers to program themselves" is much greater than "I do GP".

Risto Miikkulainen

GP/EC is a creative approach to AI. Comparing to other machine learning methods, GP is more flexible and can be tailored to fit specific application problems. This means one also needs to explore the problem structure to fully exploit the power of GP applications.

GP is particularly suitable for design problem when the outcome could be novel and unexpected. Similar to natural evolution, computational evolution is capable of creating unexpected solutions that could be missed using conventional approaches.

Possible GP killer application examples include quantum computing, co-evolution of solutions with hardware architecture, like field programmable gate arrays (FPGA), or complex graph designs.

GP could contribute to the research on general AI, since it can adapt to new environment/problem or has the potential of choosing the best suitable algorithm(s) to solve a particular problem.

Conclusion:

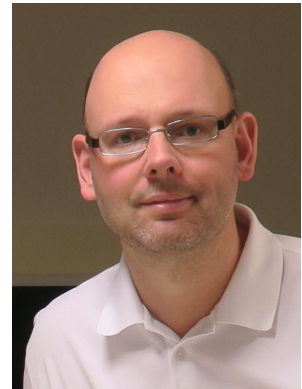
Now might be the best time to be a GP researcher, given the wide awareness in the general public. Computer science, and specifically artificial intelligence and machine learning, is currently the most rapidly growing as a scientific field as well as an industry. For the GP community, our task is to take this opportunity and make the field more visible. It is especially important to ignite the same passion in young students and pass this passion to the next generation.

Dr. Ting Hu is an assistant professor at the School of Computing, Queen's University. Before she joins Queen's in September 2019, she has been an assistant professor at the Department of Computer Science, Memorial University since 2015. She received her PhD in Computer Science from Memorial University in 2010, and postdoctoral training at Geisel School of Medicine, Dartmouth College from 2010 to 2015. Dr. Ting Hu's research focus lies on two inter-related areas, bio-inspired computing and bioinformatics. She is interested in 1) designing robust meta-heuristic evolutionary algorithms, a creative approach to AI, 2) mining large-scale biomedical data using complex networks and machine learning techniques, and 3) using simulated computational evolution to study core mechanisms of natural evolution.

Lukas Sekanina received all his degrees from Brno University of Technology, Czech Republic. He was awarded with the Fulbright scholarship to work with NASA Jet Propulsion Laboratory at

Caltech in 2004. Prof. Sekanina was a visiting professor with CEI UPM Madrid (2012), Pennsylvania State University, Erie (2001) and a visiting researcher with Department of Informatics, University of Oslo (2001). Prof. Sekanina (co) authored over 150 papers mainly on evolutionary circuit design, approximate computing and evolvable hardware and 1 patent. He holds one gold, two silver and one bronze medal from the Humies competition organized at GECCO. He is currently a full professor and Head of the Department of Computers Systems at Faculty of Information Technology, Brno University of Technology.

Lukas Sekanina



EvoStar 2019

By **Sarah Thomson**, University of Stirling

EvoStar 2019 began with a clear, bright day in the clear, bright city of Leipzig, Germany. Attendees flocked to the conference through the organised and clean streets of the city, taking in the intricate architecture on every turn.



The beauty of Leipzig, the conference city for 2019.

The conference began with a warm welcome from **Marc Schoenauer** and **Gesine Grande**, and kicked off with a truly inspiring and idea-inducing keynote by **Risto Miikkulainen**. A central point of **Risto's** talk was the paradigm of using evolutionary computation and deep learning together. He acknowledged that deep learning might be the 'hot topic' in AI, but that evolutionary computing brings a unique 'creative' aspect to AI that deep learning lacks. EC, critically, can come up with new designs — ones that human experts would never have thought of.

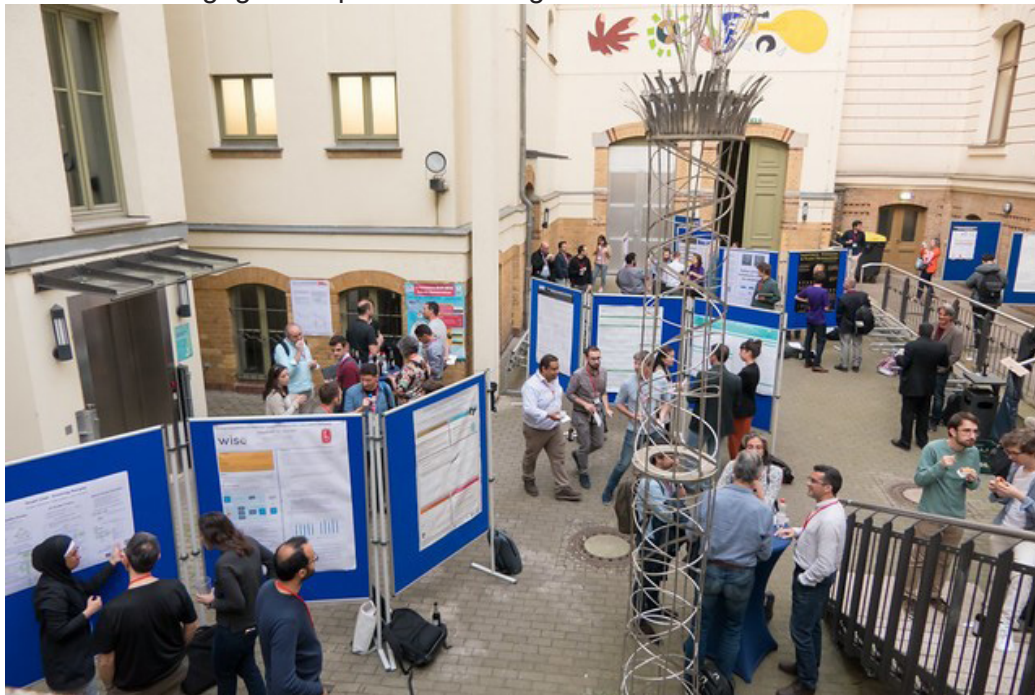


Prof **Risto Miikkulainen** on using EC to creatively optimise the growth of plants

Among his fascinating results he described to us his work in evolving (with computation, of course) the settings for variables for how best to grow certain crops. They found that, contrary to what any human biologist would suggest, the EC designed the solution such that the plants had 24 hours of sunlight exposure!

The sessions that day were concurrent EuroGP, EvoMusArt, and EvoApplications sessions. The GP popularity was as strong as ever, with an overflowing room! As ever, some incredible evolutionary art and design was presented at EvoMusArt.

In the evening, the skies were still blue and the poster session began. Posters were stationed in a beautiful outdoor courtyard, and attendees were treated to authentic German bratwurst sausages. The atmosphere was electric and intellectual discussion was bouncing everywhere, with poster-authors engaged in spirited and delighted debate.



Posters are presented in a pretty courtyard and intellectual debate fills the space.

The next day, EuroGP and EvoApplications sessions continued and EvoCOP began. We had our first best paper session of the conference — EvoMusArt, followed by EvoApplications and then EuroGP.

That evening, we had the opportunity to indulge in some culture with a guided tour of a famous Leipzig landmark building. We stood and took in the ethereal view before us. Then, to eat! Attendees flocked to the Ratskeller, a classy establishment that provided us with a plentiful and varied buffet.

Friday was the final day of the conference, and began in style with the EvoCOP best paper nominations session. Next was the second invited keynote of the conference: bioinformatician **Manja Marz**. **Manja** was energetic and articulate, presenting her work on RNA sequencing techniques. She noted that fractal geometry occurs in her field of biological systems — indeed, it is seen in many places in nature, due to the simplicity and ease of replicability -- and drew parallels to ways that code is constructed could perhaps benefit from fractal design, too — she noted that recursion is one such example.



Dr **Manja Marz**, keynote speaker, on how we can borrow ideas from biological systems for designing computer programs.

Then came the closing session, when the awards were presented. Among tough competition, the winners of the main best paper awards were:

- For EuroGP, Can Genetic Programming Do Manifold Learning Too? By **Andrew Lensen**, **Bing Xue**, and **Mengjie Zhang**;
- For EvoApplications, Evolutionary Algorithms for the Design of Quantum Protocols. By **Walter Krawec**, **Stjepan Picek**, and **Domagoj Jakobovic**;
- For EvoCOP, Quasi-Optimal Recombination Operator. By **Francisco Chicano**, **Gabriela Ochoa**, **Darrell Whitley**, and **Renato Tinós**;
- For EvoMusArtI, Swarm-based identification of animation key points from 2D-medialness maps. By **Prashant Aparajeya**, **Frederic Fol Leymarie**, and **Mohammad Majid al-Rifaie**.

Well done to our winners!

ACM-IEEE Awards 2019



Prof. **Mark Harman** being presented with the 2019 IEEE Harlan D. Mills Award by **Lionel Briand**

Mark Harman wins both IEEE and ACM Awards in 2019

By **Bill Langdon**, University College London.

1 Introduction

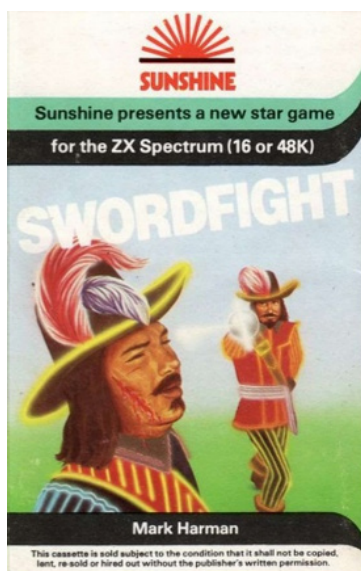
As a result of winning the 2019 IEEE Harlan D. Mills Award, Prof. [Mark Harman](#) gave an invited plenary at this year's International Conference on Software Engineering (ICSE 2109) "For fundamental contributions throughout software engineering, including seminal contributions in establishing [search-based software engineering](#)¹, reigniting research in [slicing](#) and [testing](#), and founding [genetic improvement](#)." ([slides](#)). Annually since 1999 the IEEE Computer Society has given the [Harlan D. Mills Award](#) to one individual in order to "recognize researchers and practitioners who have demonstrated long-standing, sustained, and impactful contributions to software engineering practice and research through the development and application of sound theory".

Each year since 1997 the ACM Special Interest Group on Software Engineering (SIGSOFT) has presented an award for having made a significant and lasting research contributions to the theory or practice of software engineering. It is typically given to an individual or small group. Prof. Harman was also presented with the [SIGSOFT research award](#) at ICSE 2019.

Mark is unique in having been given both prestigious awards in the same year.

¹ Search-based software engineering (SBSE) applies metaheuristic search techniques such as genetic algorithms, simulated annealing and tabu search to software engineering problems. The term SBSE was coined by Harman and Jones in 2001.

[Mark](#)'s keynote lead us through his career starting from 15 year old Z80 machine code games programmer see Figure below, his PhD, finding collaborators at [Dagstuhl](#), founding the Centre for Research on Evolution, Search and Testing ([CREST](#)) and starting the CREST Open Workshops ([COWs](#)). In 2017 he joined Facebook to found the [Sapienz](#) team, so taking SBSE research, including automatic testing and [bug fixing](#), into industry.

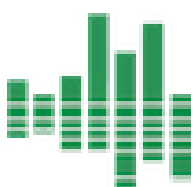


ZX Spectrum Game

Postscript

The Harlan D. Mills Award [nomination](#) deadline for next year is 1 October 2019. Nomination for the ACM SIGSOFT [Outstanding Research Award](#) are due no later than December 15.

Forthcoming Conferences



TPNC 2019

The 8th International Conference on the Theory and Practice of Natural Computing (TPNC 2019) will take place in Kingston, Ontario, Canada on December 9-11, 2019.

The conference website is at: <http://tpnc2019.irdta.eu/>

Important Dates (all at 23:59 CET)

27th July, 2019	Paper submission
31st August, 2019	Notification of paper acceptance or rejection
7th September, 2019	Final version of the paper for the LNCS proceedings
7th September, 2019	Early registration
25th November, 2019	Late registration
11th March, 2020	Submission to the journal special issue

Organizers

Sara Morales, Brussels. **Manuel Parra-Royón**, University of Granada, **Geoffrey Pond** Kingston, co-chair, **David Silva**, London, co-chair, **Miguel A. Vega-Rodríguez**, Cáceres

2019 Workshop on AI-based Optimisation

AI-OPT 2019

The 2019 Workshop on AI-based Optimisation (AI-OPT 2019)

<https://conference.eng.unimelb.edu.au/ai-opt-2019/> will take place on October, 1-2, 2019 at the University of Melbourne, Australia.

Artificial Intelligence based optimisation techniques such as constraint programming, evolutionary computation, heuristic search, mixed integer programming, and swarm intelligence have found many applications in solving highly complex and challenging optimisation problems. Application domains include important areas such as cybersecurity, economics, engineering, renewable energy, health and supply chain management.

The goal of this informal workshop is to bring together researchers from a wide range of AI-based optimisation areas, discuss current challenges, and foster collaborations. The workshop will consist of technical presentations and allow enough time for discussions and collaborations. PhD students and early career researchers are especially encouraged to participate.

Researchers who are interested in presenting their work are invited to submit a short abstract (at most 1 page) on fundamental aspects and applications of AI-based optimisation.

Important dates:

Monday 15 July 2019	Abstract submission deadline
Tuesday 30 July 2019	Notification
1-2 October 2019	Workshop

Contact and organisers

[Uwe Aickelin](#), The University of Melbourne, Australia

[Frank Neumann](#), The University of Adelaide, Australia

About this newsletter

SIGEVolution is the newsletter of SIGEVO, the ACM Special Interest Group on Genetic and Evolutionary Computation. To join SIGEVO, please follow this link: [\[WWW\]](#)

Contributing to SIGEVolution

We solicit contributions in the following categories:

Art: Are you working with Evolutionary Art? We are always looking for nice evolutionary art for the cover page of the newsletter.

Short surveys and position papers: We invite short surveys and position papers in EC and EC related areas. We are also interested in applications of EC technologies that have solved interesting and important problems.

Software: Are you are a developer of an EC software and you wish to tell us about it? Then, send us a short summary or a short tutorial of your software.

Lost Gems: Did you read an interesting EC paper that, in your opinion, did not receive enough attention or should be rediscovered? Then send us a page about it.

Dissertations: We invite short summaries, around a page, of theses in EC-related areas that have been recently discussed and are available online.

Meetings Reports: Did you participate to an interesting EC-related event? Would you be willing to tell us about it? Then, send us a short summary, around half a page, about the event.

Forthcoming Events: If you have an EC event you wish to announce, this is the place.

News and Announcements: Is there anything you wish to announce, such as an employment vacancy? This is the place.

Letters: If you want to ask or to say something to SIGEVO members, please write us a letter!

Suggestions: If you have a suggestion about how to improve the newsletter, please send us an email.

Contributions will be reviewed by members of the newsletter board.

We accept contributions in LATEX, MS Word, and plain text.

Enquiries about submissions and contributions can be emailed to editor@sigevolution.org

All the issues of SIGEVolution are also available online at: www.sigevolution.org

Notice to Contributing Authors to SIG Newsletters

By submitting your article for distribution in the Special Interest Group publication, you hereby grant to ACM the following non-exclusive, perpetual, worldwide rights:

- to publish in print on condition of acceptance by the editor
- to digitize and post your article in the electronic version of this publication
- to include the article in the ACM Digital Library
- to allow users to copy and distribute the article for noncommercial, educational or research purposes

However, as a contributing author, you retain copyright to your article and ACM will make every effort to refer requests for commercial use directly to you.

Editor: **Gabriela Ochoa**

Associate Editors: **Emma Hart, James McDermott, Una-May O-Reilly and Darrell Whitley**

Design & Layout: **Kate Simpson**