Julian Francis Miller, 1955–2022

Susan Stepney
University of York, UK
Department of Computer Science

Alan Dorin
Monash University, Australia
Department of Data Science and AI

Julian Francis Miller[1]

It is with great sadness that we report the death of our colleague and friend, Julian Miller.

Julian's work is well known throughout the Artificial Life community: His Cartesian genetic programming (CGP) and *in materio* computing are foundational concepts. He also made contributions in morphological computing and neurocomputing, all based on his fascination with evolution as a means of attacking and solving problems. Like many in the ALife community, he had an interdisciplinary career, commencing with a first degree in Physics and a PhD in Mathematics, followed by research in Natural Computing and material computing at the universities of Napier, Birmingham, and York in the UK.

Julian invented CGP (Miller, 1999), a way of encoding graph programs (functional nodes connected by edges) in a string of integers, allowing the string to be evolved in the standard way, with the graph (located on a Cartesian grid, hence its name) produced as the result of a genotype to phenotype mapping. From this simple beginning, Julian and his students continued to develop the approach, and other researchers joined in. Ten years later, the field had grown significantly, with many researchers both using CGP in their own work and extending the original concept. Indeed, the field had grown enough that Julian could edit an entire book on the topic (Miller, 2011).

1 Photograph © 2006 Gabriele Miller; used with permission.

Ten years later still, the field shows no signs of abating, and Julian wrote a 40-page review for *Genetic Programming and Evolvable Machines* on CGP's status, its many variants, and its future prospects (Miller, 2020).

Julian was also a pioneer in the field of *in materio* computing (Miller & Downing, 2002), which exploits the physical properties of unconventional materials, such as liquid crystals (Harding & Miller, 2004) and carbon nanotubes (Miller et al., 2014), to perform computation intrinsically, in what he dubbed a "Field Programmable Matter Array." His original work used evolutionary algorithms directly to configure the materials. Later, he also used Reservoir Computing as a more abstract model for getting these materials to compute (Dale et al., 2017). This is another field with explosive growth, so much so that some authors have even published on the name of the domain itself (Ricciardi & Milano, 2022). Julian was there from the start, contributing his insights and ideas throughout.

CGP and *in materio* computing may be what Julian is best known for, but these contributions were embedded in a deeper research program of understanding development as a fundamental component of evolving embodied computation. From growing a self-repairing "French-flag" organism (Miller, 2004), to assembling complex structures through Artificial Chemistries (Faulconbridge et al., 2011), to the idea of the "software garden" (Miller, 2018), Julian felt that both growth and evolution are essential concepts in complex systems.

His Festschrift *Inspired by Nature* (Stepney & Adamatzky, 2018) was a (slightly late) 60th birthday present from his many academic colleagues. It includes chapters contributed by a wide range of authors who have built on and been inspired by his many research interests. The text covers evolution and hardware, CGP applications, chemistry, and development. Julian retired in 2016, but he did not stop his research. He used the freedom from the quotidian constraints of an academic job to pursue a new interest. He was bringing together his discoveries in evolution, development, networks, and computation to develop a new neural model to evolve programs that build, or grow, neural networks. His most recent publication on that topic has only just appeared (Miller, 2022).

Julian's retirement also allowed him to spend time with his recently acquired beloved new family. His wife Gabi remembers him thus: "He was a loving and generous-hearted husband, a wise step-father to my three grown adults and much loved Grandpa to our four grand-children. Jules will be sadly missed, but also lovingly remembered by all whose life he touched." Many further tributes to Julian from his colleagues can be found in the latest SIGEVO newsletter (Ochoa, 2022).

References

Dale, M., Miller, J. F., & Stepney, S. (2017). Reservoir computing as a model for *in-materio* computing. In A. Adamatzky (Ed.), *Advances in Unconventional Computing* (pp. 533–571). Springer. https://doi.org/10.1007/978-3-319-33924-5_22

Faulconbridge, A., Stepney, S., Miller, J. F., & Caves, L. S. D. (2011). RBN-World: A subsymbolic artificial chemistry. *ECAL 2009*, *5777*, 377–384. https://doi.org/10.1007/978-3-642-21283-3_47

Harding, S., & Miller, J. F. (2004). Evolution *in materio*: Initial experiments with liquid crystal. In *Proceedings, 2004 NASA/DoD Conference on Evolvable Hardware*. https://doi.org/10.1109/EH.2004.1310844

Miller, J. F. (1999). An empirical study of the efficiency of learning boolean functions using a Cartesian Genetic Programming approach. In *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation—Volume 2*, 1135–1142.

Miller, J. F. (2004). Evolving a self-repairing, self-regulating, french flag organism. In *GECCO 2004*, 129–139. https://doi.org/10.1007/978-3-540-24854-5_12

Miller, J. F. (Ed.). (2011). *Cartesian genetic programming*. Springer. https://doi.org/10.1007/978-3-642-17310-3

Miller, J. F. (2018). The software garden. In R. Walsh & S. Stepney (Eds.), *Narrating complexity* (pp. 201–212). Springer. https://doi.org/10.1007/978-3-319-64714-2_15

Miller, J. F. (2020). Cartesian genetic programming: Its status and future. *Genetic Programming and Evolvable Machines*, *21*(1), 129–168. https://doi.org/10.1007/s10710-019-09360-6

Miller, J. F. (2022). IMPROBED: Multiple problem-solving brain via evolved developmental programs. *Artificial Life*, *27*(3–4), 300–335. https://doi.org/10.1162/artl_a_00346

Miller, J. F., & Downing, K. (2002). Evolution *in materio*: Looking beyond the silicon box. In *Proceedings, 2002 NASA/DoD Conference on Evolvable Hardware*, 167–176. https://doi.org/10.1109/EH.2002.1029882

Miller, J. F., Harding, S. L., & Tufte, G. (2014). Evolution-in-materio: Evolving computation in materials. *Evolutionary Intelligence*, *7*(1), 49–67. https://doi.org/10.1007/s12065-014-0106-6

Ochoa, G. (Ed.). (2022). Tributes to Julian F. Miller (1955–2022). *SIGEVOlution: Newsletter of the ACM Special Interest Group on Genetic and Evolutionary Computation*, *15*(1). https://evolution.sigevo.org/issues/HTML/sigevolution-15-1/home.html#h.6uzth8kptc5x, https://doi.org/10.1145/3532942.3532943

Ricciardi, C., & Milano, G. (2022). In materia should be used instead of *in materio*. *Frontiers in Nanotechnology*, *4*. https://doi.org/10.3389/fnano.2022.850561

Stepney, S., & Adamatzky, A. (Eds.). (2018). *Inspired by nature: Essays presented to Julian F. Miller on the occasion of his 60th birthday*. Springer.