Towards Neuromorphic Evolutionary Computation

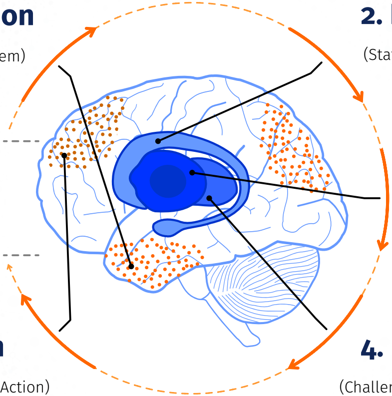
Jorge M. Cruz-Duarte • El-Ghazali Talbi

1. Motivation

(Context & Problem)

2. Landscape

(State of the Art & Gaps)



3. Position

(Our Proposed Approach)

4. Discussion

(Challenges & Open Questions)

5. Vision

(Roadmap & Call to Action)

Motivation I

The von Neumann Bottleneck

AI progress has intensified energy demands on data centres globally. Computation and memory remain physically separated, making energy dominated by **data movement**, not arithmetic.

EC is particularly exposed:

- Fitness eval. $\rightarrow$ repeated memory accesses
- State manipulation $\rightarrow$ sync. overhead
- Pop. updates $\rightarrow$ amplified energy dissipation

Implication

Improving algorithms alone is insufficient. The **computational substrate** must change.

Motivation II

Neuromorphic Computing is Already Mature

NC integrates memory and computation, communicates via sparse asynchronous spikes, and operates in an **event-driven regime**. Yet optimisation remains almost entirely unexplored within this class of processors.[1]

What NC brings to optimisation:

- Memory-compute co-location→reduced data movement
- Intrinsic parallelism via distributed spiking ensembles
- Online adaptability via synaptic plasticity
- Edge and embedded targets with tight energy budgets

Opportunity

The convergence of EC and NC is timely and technically plausible. Both domains gain from it.

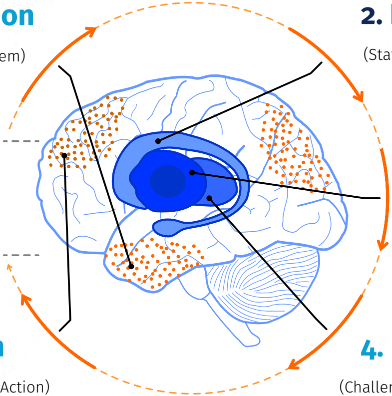
[1] M. Davies et al., "Advancing Neuromorphic Computing With Loihi: A Survey of Results and Outlook", Proceedings of the IEEE **109**, 911–934 (2021)

1. Motivation

(Context & Problem)

2. Landscape

(State of the Art & Gaps)



3. Position

(Our Proposed Approach)

4. Discussion

(Challenges & Open Questions)

5. Vision

(Roadmap & Call to Action)

State of the Art & Gaps

Research at the intersection of NC and optimisation remains fragmented.

- **Problem-specific solvers:** Efficient, but tightly coupled to a problem class or device primitive. Extending to new objectives often requires redesign.

State of the Art & Gaps

Research at the intersection of NC and optimisation remains fragmented.

- **Problem-specific solvers:** Efficient, but tightly coupled to a problem class or device primitive. Extending to new objectives often requires redesign.
  - Ferroelectric memristor networks for the Travelling Salesperson Problem[1].
  - Organic-memristor Max-Cut solvers[2].

[1] Z. Lin and Z. Fan, "A ferroelectric memristor-based transient chaotic neural network for solving combinatorial optimization problems", Symmetry **15**, 59 (2022).

[2] H. Kim et al., "Organic memristor-based flexible neural networks with bio-realistic synaptic plasticity for complex combinatorial optimization", Advanced Science **10**, 2300659 (2023).

State of the Art & Gaps

Research at the intersection of NC and optimisation remains fragmented.

- **Problem-specific solvers:** Efficient, but tightly coupled to a problem class or device primitive. Extending to new objectives often requires redesign.
- **Neuroevolution of SNNs:** Search runs on CPUs. Neuromorphic hardware is used for inference or evaluation. Energy gains occur after deployment.

State of the Art & Gaps

Research at the intersection of NC and optimisation remains fragmented.

- **Problem-specific solvers:** Efficient, but tightly coupled to a problem class or device primitive. Extending to new objectives often requires redesign.
- **Neuroevolution of SNNs:** Search runs on CPUs. Neuromorphic hardware is used for inference or evaluation. Energy gains occur after deployment.
    - Evolutionary Optimisation for NC Systems framework[1].
    - Evolutionary SNNs and biologically plausible learning rules[2].

[1] S. Shen et al., "Evolutionary spiking neural networks: a survey", Journal of Membrane Computing **6**, 335–346 (2024).

[2] C. D. Schuman et al., "Real-time evolution and deployment of neuromorphic computing at the edge", in 2021 12th international green and sustainable computing conference (igsc) (IEEE, 2021), pp. 1–8.

State of the Art & Gaps

Research at the intersection of NC and optimisation remains fragmented.

- **Problem-specific solvers:** Efficient, but tightly coupled to a problem class or device primitive. Extending to new objectives often requires redesign.
- **Neuroevolution of SNNs:** Search runs on CPUs. Neuromorphic hardware is used for inference or evaluation. Energy gains occur after deployment.
- **Spiking swarm models:** Often simulation-only. Hardware constraints and energy are rarely measured. Portability across platforms is unclear.

State of the Art & Gaps

Research at the intersection of NC and optimisation remains fragmented.

- **Problem-specific solvers:** Efficient, but tightly coupled to a problem class or device primitive. Extending to new objectives often requires redesign.
- **Neuroevolution of SNNs:** Search runs on CPUs. Neuromorphic hardware is used for inference or evaluation. Energy gains occur after deployment.
- **Spiking swarm models:** Often simulation-only. Hardware constraints and energy are rarely measured. Portability across platforms is unclear.
  - Spiking adaptations of swarm mechanisms for continuous and combinatorial settings[1].
  - Oscillator-based variants of Particle Swarm Optimisation[2].

[1] Y. Fang et al., "A swarm optimization solver based on ferroelectric spiking neural networks", Frontiers in Neuroscience **13**, 855 (2019).

[2] T. Sasaki and H. Nakano, "Swarm Intell. Alg. Spiking Neural-Osc. Netw., Coupling Interact. & Search Perform.", Nonlinear Theory Appl., IEICE **14**, 267–291 (2023).

State of the Art & Gaps

Research at the intersection of NC and optimisation remains fragmented.

- **Problem-specific solvers:** Efficient, but tightly coupled to a problem class or device primitive. Extending to new objectives often requires redesign.
- **Neuroevolution of SNNs:** Search runs on CPUs. Neuromorphic hardware is used for inference or evaluation. Energy gains occur after deployment.
- **Spiking swarm models:** Often simulation-only. Hardware constraints and energy are rarely measured. Portability across platforms is unclear.
- **Hybrid frameworks:** Control may be spiking, but key steps remain conventional. Candidate generation or fitness evaluations often stays off-chip.

State of the Art & Gaps

Research at the intersection of NC and optimisation remains fragmented.

- **Problem-specific solvers:** Efficient, but tightly coupled to a problem class or device primitive. Extending to new objectives often requires redesign.
- **Neuroevolution of SNNs:** Search runs on CPUs. Neuromorphic hardware is used for inference or evaluation. Energy gains occur after deployment.
- **Spiking swarm models:** Often simulation-only. Hardware constraints and energy are rarely measured. Portability across platforms is unclear.
- **Hybrid frameworks:** Control may be spiking, but key steps remain conventional. Candidate generation or fitness evaluations often stays off-chip.
  - LavaBO and its asynchronous variant demonstrate that NC platforms can coordinate Bayesian optimisation pipelines, but the core components remain CPU-based[1].

[1] S. Snyder et al., "Neuromorphic bayesian optimization in Lava", in Proc. international conference on neuromorphic systems (icons) (2023).

State of the Art & Gaps

Research at the intersection of NC and optimisation remains fragmented.

- **Problem-specific solvers:** Efficient, but tightly coupled to a problem class or device primitive. Extending to new objectives often requires redesign.
- **Neuroevolution of SNNs:** Search runs on CPUs. Neuromorphic hardware is used for inference or evaluation. Energy gains occur after deployment.
- **Spiking swarm models:** Often simulation-only. Hardware constraints and energy are rarely measured. Portability across platforms is unclear.
- **Hybrid frameworks:** Control may be spiking, but key steps remain conventional. Candidate generation or fitness evaluations often stays off-chip.

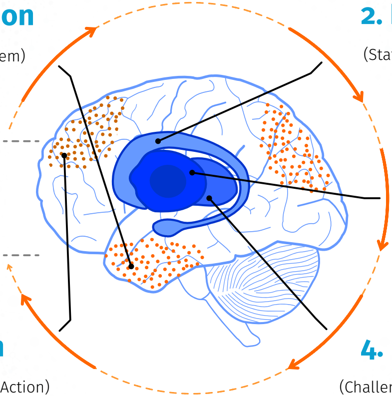
Problem-specific neuromorphic solvers lack generality, whereas more flexible frameworks remain hybrid and therefore cannot realise the full benefits of event-driven execution.

1. Motivation

(Context & Problem)

2. Landscape

(State of the Art & Gaps)



3. Position

(Our Proposed Approach)

4. Discussion

(Challenges & Open Questions)

5. Vision

(Roadmap & Call to Action)

Approach I: Towards Fully Neuromorphic Optimisation

Assumption: Coordination and operator sel. can be expressed as spike-driven dynamics. EC offers NC rigorous BBO benchmarks; NC offers EC sub-watt, event-driven execution.

Five on-chip requirements:

1. Operator Embodiment
2. Memory Architecture
3. Representation & Learning
4. Hardware Deployment
5. Adaptive & Embedded Systems

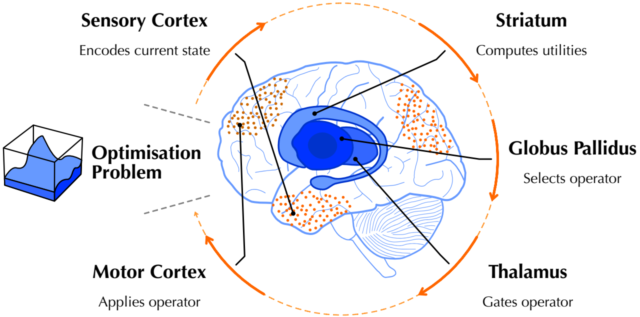


Figure: Conceptual reference design for Neuromorphic Evolutionary Computation (NEC).

A Cortico-Basal-Thalamic Loop (CBTL)-inspired spike-driven loop selects and reinforces search operators based on an observed optimisation state (conceptual schematic).

Approach II

Operator Embodiment

Existing prototypes implement operators as procedural routines. A fully NEC optimiser requires each operator to be **embodied as a spiking circuit** mapping encoded memory states to new candidates.

Operator families & spiking mechanisms:

- Lévy flights → stochastic spike generation
- Differential mutation → synaptic superposition
- Particle swarm → recurrent reservoirs
- Spiral search → approx. trigonometric projections

Goal

Identify operator formulations that naturally arise from spiking dynamics with tractable neuron & synapse budgets.

Approach III

Memory Architecture

Current systems maintain solution memory in conventional RAM. A fully NEC optimiser requires an **associative structure** that stores, retrieves, and updates solution vectors through spike-based operations.

Viable substrates:

- Recurrent attractor networks[1]
- Vector Symbolic Architectures (VSA)[2]

Solutions $\equiv$ stable activity patterns; retrieval via nearest-attractor settling.

Bottleneck

Fitness-based replacement requires **Winner-Takes-All (WTA)** circuits to keep neuron counts tractable.

[1] D. Krotov and J. J. Hopfield, "Dense associative memory for pattern recognition", Advances in neural information processing systems **29** (2016)

[2] A. R. Voelker and C. Eliasmith, "Improv. spiking dynamical networks: Accurate delays, higher-order synapses, and time cells", Neural Comput. **30**, 569–609 (2018)

Approach IV

Representation and Learning

Search state variables must be **produced within neural ensembles**, not computed outside the spiking substrate.

Candidate mechanisms:

- Diversity → homeostatic plasticity & divisive normalisation
- Convergence → temp. integration via LIF dynamics
- Learned repr. → reward-modulated plasticity + spike-based dim. reduction

Direction

Train encoder networks directly from stored solutions and fitness values[1].

[1] B. Nessler et al., "Bayesian Computation Emerges in Generic Cortical Microcircuits through Spike-Timing-Dependent Plasticity", PLOS Comput. Bio. **9**, 1–30 (2013)

Approach V

Hardware Deployment

Neuromorphic processors operate **several orders of magnitude** below CPU-based simulation. Event-driven execution yields further gains when neural activity is sparse.

Target platforms:

- Loihi 2[1] $\rightarrow$ direct energy measurement & operator complexity limits
- TrueNorth[2], SpiNNaker[3] $\rightarrow$ portability across NC platforms
- BrainScaleS[4] $\rightarrow$ distinct comm. & timing properties

Goal

Clarify hardware limits on operator complexity, memory size, and decision latency on physical NC hardware.

[1] M. Davies et al., "Advancing Neuromorphic Computing With Loihi: A Survey of Results and Outlook", Proceedings of the IEEE **109**, 911–934 (2021)

[2] F. Akopyan et al., "Truenorth: Design and tool flow of a 65 mW 1 million neuron programmable neurosynaptic chip", IEEE TCAD **34**, 1537–1557 (2015)

[3] S. B. Furber et al., "The spinnaker project", Proceedings of the IEEE **102**, 652–665 (2014)

[4] C. Pehle et al., "The brainscales-2 accelerated neuromorphic system with hybrid plasticity", Frontiers in Neuroscience **16**, 795876 (2022)

Approach VI

Adaptive and Embedded Systems

A fully NEC optimiser extends beyond energy efficiency. STDP enables tracking **time-varying objectives without external control**, relevant for embedded platforms with strict resource and communication constraints.

Properties and targets:

- Sub-watt real-time adaptation (drone, sensor platform)
- BrainChip Akida[1], SynSense Xylo[2] → milliwatt to sub-mW on-device inference
- Parallel async. search → multiple operator ensembles via sparse spike-based signals[3]

Vision

A unified NC agent that adapts continuously within a single computational substrate.

[1] E. Bråtman and L. Dow, *Neuromorphic Medical Image Analysis at the Edge: On-Edge training with the Akida Brainchip*, 2023

[2] H. Bos and D. Muir, *Sub-mW Neuromorphic SNN audio processing applications with Rockpool and Xylo*, 2022

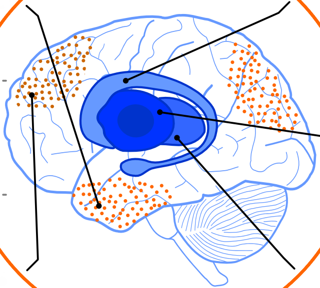
[3] K. Gurney et al., "A computational model of action selection in the basal ganglia. i. a new functional anatomy", Biological cybernetics **84**, 401–410 (2001)

1. Motivation

(Context & Problem)

2. Landscape

(State of the Art & Gaps)



3. Position

(Our Proposed Approach)

4. Discussion

(Challenges & Open Questions)

5. Vision

(Roadmap & Call to Action)

Challenges & Open Questions I

❄ Numerical Precision and Stability

Challenge: Spiking representations offer limited effective precision, typically 8–12 bits[1], which affects the reliability of fitness comparisons, diversity estimates, & weight updates.

Open Questions:

- What numerical accuracy is required for consistent convergence?
- Can adaptive precision be applied without distorting spike-timing correlations on which Spike-Timing-Dependent Plasticity (STDP) depends?

[1] M. Boerlin et al., "Predictive coding of dynamical variables in balanced spiking networks", PLoS computational biology **9**, e1003258 (2013)

Challenges & Open Questions II

Scalability

Challenge: Population-coded representations scale as $\mathcal{O}(M \cdot D \cdot N_{\text{ens}})$ neurons, becoming prohibitive for realistic dimensionality D and exhausting resources for operators and control circuits.

Open Questions:

- Which dimensionality-reduction schemes (projections, decompositions) are compatible with neuromorphic substrates?
- Can encodings remain stable enough for STDP eligibility traces to accumulate meaningful correlations?

Challenges & Open Questions III

Operator Design

Challenge: Modern operators (e.g., CMA-ES[1], SHADE[2]) rely on covariance estimates and global synchronisation that are difficult to realise in spiking circuits.

Open Questions:

- Which operator classes are naturally expressible via local interactions and stochastic spiking dynamics?
- Can a principled composition framework (analogous to algebraic EC descriptions[3]) be defined for neuromorphic operators?

[1] N. Hansen, "The CMA evolution strategy: A tutorial", arXiv preprint arXiv:1604.00772 (2016)
[2] R. Tanabe and A. S. Fukunaga, "Improving the search performance of SHADE using linear population size reduction", in 2014 IEEE CEC (2014), pp. 1658–1665
[3] J. M. Cruz-Duarte et al., "Towards a generalised metaheuristic model for continuous optimisation problems", Mathematics **8**, 2046 (2020)

Challenges & Open Questions IV

Learning Mechanisms

Challenge: STDP and reward-modulated rules operate on short spike-timing windows, whereas operator selection evolves over much longer evaluation horizons. Bridging this temporal gap requires eligibility traces with non-trivial circuit cost.

Open Questions:

- Can utility mappings for operator reinforcement be learned across tasks via meta-learning?
- How can three-factor learning rules[1] be translated to SNNs for operator selection?

[1] N. Frémaux and W. Gerstner, "Neuromodulated Spike-Timing-Dependent Plasticity, and Theory of Three-Factor Learning Rules", Front. Neural Circuits **9**, 85 (2016)

Challenges & Open Questions V

Benchmarking

Challenge: Existing suites (e.g., COCO[1], NeuroBench[2]) do not jointly cover solution quality, energy, and latency trade-offs central to NEC.

Open Questions:

- How should NEC evaluations be structured as multi-objective (quality + energy + latency) comparisons?
- What minimal reporting checklist enables fair comparison across neuromorphic platforms?

[1] N. Hansen et al., "COCO: a platform for comparing continuous optimizers in a black-box setting", Optimization Methods and Software **36**, 114–144 (2021)

[2] J. Yik et al., *Neurobench: a framework for benchmarking neuromorphic computing algorithms and systems*, 2025

Challenges & Open Questions VI

Theory

Challenge: Spike-driven optimisers cannot always be framed as regular EAs with well-defined generations and update operators, leaving convergence guarantees from classical EC[1] without clear counterparts.

Open Questions:

- Under what conditions does the best-so-far stochastic process converge (in probability or a.s.)?
- How do No-Free-Lunch results[2] and computational complexity extend to spike-driven operators?

[1] D. Delahaye et al., "Simulated annealing: from basics to applications", in *Handbook of metaheuristics* (Springer, 2018), pp. 1–35

[2] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization", IEEE Transactions on Evolutionary Computation **1**, 67–82 (2002)

Challenges & Open Questions VII

Tooling and Accessibility

Challenge: Existing frameworks (Nengo, Brian, BindsNET, NEST) target computational neuroscience, not optimisation. They lack fitness-based replacement, tournament selection, and operator composition primitives.

Open Questions:

- What interface abstractions would allow EC practitioners to express spiking operators as naturally as in classical libraries (e.g., ParadisEO[1], CUSTOMHyS[2])?
- Which hardware backends (Loihi, SpiNNaker, CPU emu) should a minimal NEC library target first?

[1] J. Dreo et al., "Paradiseo: from a modular framework for evo. comput. to the automated design of MHs: 22 years of Paradiseo", in Proc. GECCO (2021), pp. 1522–1530

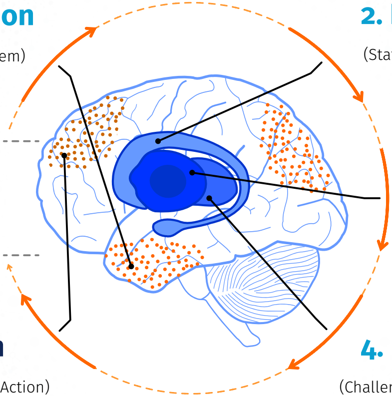
[2] J. Cruz-Duarte et al., "Hyper-heuristics to customise metaheuristics for continuous optimisation", Swarm Evol. Comput. **66**, 100935 (2021)

1. Motivation

(Context & Problem)

2. Landscape

(State of the Art & Gaps)



3. Position

(Our Proposed Approach)

4. Discussion

(Challenges & Open Questions)

5. Vision

(Roadmap & Call to Action)

NEC Design Principles

Event-driven First

Use sparse, async. updates
Avoid global synchronisation

On-device state

Encode search states in spikes.
Minimise host-side control.

Measured trade-offs

Report solution quality, energy, and latency together.

Composable operators

Use interfaces that support composition and ablation.

Reproducibility

Release code, seeds, and profiling scripts.

Roadmap

Advancing towards fully NEC requires **shared benchmarks**, **open-source implementations**, and **cross-disciplinary venues** bridging EC and NC.

Short Term (1-2 y)

Expand the spiking operator collection.

Establish benchmarking protocols (quality, energy, & latency).

Prototype associative memories for solution storage.

Deploy reference loops on NC HW for direct measurements.

Mid Term (2-4 y)

Develop fully spiking perturbation operators.

Learn state representations with spike-based encoders.

Scale across multiple chips with efficient communication.

Quantify energy-perf. trade-offs across problem classes.

Long Term (>4 y)

Design NEC combining temp. coding, inhibition, & learning.

Support continual opt. of dynamic obj. on-chip plasticity.

Ground spike-driven search in information-theoretic bounds

Validate end-to-end embedded deployments.

Call to Action

Advancing NEC requires shared practices across communities to lower barriers to entry and facilitate meaningful comparisons of results.

Release

Release reference implementations of coordination and operator selection circuits.

Report

Report solution quality, energy consumption and latency together when benchmarking.

Provide

Provide access to hardware or validated energy models to ensure fair comparison.

NeurOptim: Our research initiative

- NeurOptim[a] is a research initiative exploring how **Neuromorphic Computing** and **Computational Intelligence** can give rise to a new class of **Optimisation Algorithms**.
- Collective progress:
    - Open Science
    - Transparent methodologies
    - Reproducible experiment.

[a] `https://neuroptim.org`

Complementary Information

cec_pos106*

Towards Neuromorphic Evolutionary Computation: A Position Paper

Jorge M. Cruz-Duarte and El-Ghazali Talbi
University of Lille, CNRS, Inria, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France.
E-Mails: {jorge.cruz-duarte, el-ghazali.talbi}@univ-lille.fr

***Abstract*—Neuromorphic Computing (NC) and Evolutionary Computation (EC) are mature, yet largely fragmented. This position paper argues that their convergence is timely and technically plausible. We define Neuromorphic Evolutionary Computation (NEC) as a paradigm that bridges NC and EC. We review the current landscape and its limitations, with an emphasis on the prevalence of hybrid execution and incomplete energy-latency reporting. We then present a probable Cortico-Basal-Thalamic Loop (CBTL)-inspired conceptual architecture for operator selection and outline what must be implemented on-chip to achieve an NEC approach, including operator embodiment, associative memory, and state estimation. We summarise open challenges spanning numerical precision, scalability, learning rules, benchmarking, theory, and tooling, including the need for convergence analysis framed in terms of the best-so-far process. We close with a phased roadmap and concrete reporting requirements towards a reproducible NEC ecosystem.**

***Index Terms*—Neuromorphic Evolutionary Computation, Event-driven Optimisation, Neuromorphic Computing, Spiking Neural Networks, Operator Selection, Energy-Latency Benchmarking, Best-so-far Convergence**

constraints that extend beyond algorithmic design. EC provides effective search behaviour in domains where gradients are inaccessible, or objectives are irregular [2,3]. Still, their theoretical support remains limited, their performance deteriorates in high-dimensional spaces, and their execution incurs synchronisation costs associated with processors optimised for dense linear algebra.

The structural mismatch between EAs and conventional hardware motivates the study of alternative computational substrates. Neuromorphic Computing (NC) processors such as Intel Loihi [4], IBM TrueNorth [5], SpiNNaker [6], BrainScaleS [7], SynSense Xylo [8], and BrainChip Akida [9] integrate memory and computation, communicate via sparse asynchronous spikes, and operate in an event-driven regime rather than through dense synchronous updates [10]. These architectures achieve substantial energy efficiency. For instance, Loihi 2 operates within a power envelope of a few hundred milliwatts for networks of millions of neurons, far below the

cec_pap596

NEVO: A Neuromorphic EVolutionary Optimiser with Spike-Driven Cortico-Basal-Thalamic Coordination

Jorge M. Cruz-Duarte and El-Ghazali Talbi
University of Lille, CNRS, Inria, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France.
E-Mails: {jorge.cruz-duarte, el-ghazali.talbi}@univ-lille.fr

***Abstract*—Neuromorphic computing and evolutionary computation share key properties, including distributed state, event-driven communication, and tolerance to finite precision. Nevertheless, optimisation control is rarely implemented directly in spiking dynamics. This paper introduces the Neuromorphic EVolutionary Optimiser (NEVO), in which evolutionary operator selection is realised as spike-driven action selection within a Cortico-Basal-Thalamic Loop (CBTL), whilst candidate generation and objective evaluation remain algorithmic under a standard black-box interface. NEVO is evaluated on the noiseless BBOB suite across 2D to 40D with 15 instances per function. Results show that spike-driven coordination sustains coherent operator sequences across problem landscapes, achieving successful target attainment in 2D and 3D, whilst exposing scalability limitations beyond 10D. These findings confirm that spike-driven coordination is feasible on neuromorphic substrates and clarify how neuromorphic action selection relates to existing adaptive operator selection strategies. The analysis also identifies concrete design requirements for future implementations.**

***Index Terms*—Neuromorphic Optimisation, Evolutionary Computation, Operator Selection, Spiking Neural Networks, Neural Engineering Framework, Nengo, Basal Ganglia.**

Existing work supports neuromorphic optimisation, but it is commonly problem-bound or hybrid, leaving the optimisation loop outside the scope of spike-driven dynamics. A first strand exploits the device physics or hardwired setup. For instance, ferroelectric and organic memristor networks [9,10], RRAM spiking swarms [11], and Simulated Annealing on Loihi 2 [12] report promising efficiency in tackling combinatorial problems. Their designs are strongly coupled to a formulation or device. Thus, extending them to continuous, constrained, or multi-objective domains typically requires redesigning neural circuitry.

A second strand, often labelled neuroevolution, evolves spiking models but retains a conventional search process. The Evolutionary Optimisation for NC Systems framework [13] and related surveys [14]–[16] show that EC algorithms can configure SNN architectures and learning rules. Nevertheless, NC hardware is primarily used for inference or evaluation. Moreover, swarm-inspired approaches provide spiking interpretations of established Metaheuristics (MHs) [17,18]. These

The convergence of Evolutionary Computation and Neuromorphic Computing is timely, technically plausible, and open for us, as a community, to shape.

Bedankt!
Thanks!
Merci!
Grazie!
¡Gracias!
Obrigado!
Mulţmesc!

NeurOptim

neuroptim.org

Contact information

Jorge M. Cruz-Duarte
`jorge.cruz-duarte@univ-lille.fr`

El-Ghazali Talbi
`el-ghazali.talbi@univ-lille.fr`

FRANCE 2030
RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité
Inria
Université de Lille
cnrs
CRIStAL
Centre de Recherche en Informatique, Signal et Automatique de Lille