Self-management for neural dynamics in brain-like information processing

Benjamin Dittes, Alexander Gepperth, Antonello Ceravola, Jannik Fritsch, Christian Goerick

2009

Preprint:

This is an accepted article published in Proceedings of the 6th International Conference on Autonomic Computing and Communications. The final authenticated version is available online at: https://doi.org/[DOI not available]

Self-management for Neural Dynamics in Brain-like Information Processing

Benjamin Dittes, Alexander Gepperth, Antonello Ceravola, Jannik Fritsch and Christian Goerick
Honda Research Institute GmbH
Carl-Legien-Str. 30
67073, Offenbach a.M., Germany
<firstname>.<lastname>@honda-ri.de

ABSTRACT

Neural dynamics coupled by adaptive synaptic information transmission provide a very powerful tool for biologically inspired visual processing systems[4]. Currently, progress is limited by the computing time needed to evaluate the underlying equations and by the high number of parameters necessary to tune to achieve the desired system performance.

In this contribution we apply Autonomic Computing techniques to overcome these limitations. We approach the computing time problem with an error model of the differential equations allowing for self-optimization of the evaluation step size and the parameter problem with a self-configuration heuristics to keep neural activation in working range.

We show the equivalence of system behavior compared to the case without self-management, the performance gain achieved by the self-optimization and the stability achieved by the self-configuration.

Categories and Subject Descriptors

D.2.9 [**Software Engineering**]: Management;
G.1.7 [**Numerical Analysis**]: Ordinary Differential Equations

General Terms

Performance, Design, Management

1. INTRODUCTION

In the effort of building brain-inspired, real-time capable intelligent systems, coupled neural dynamics provide a powerful tool. We envision these systems consisting of several neural dynamic modules (NDM), responsible for temporal integration and decision making, coupled by synaptic dynamic modules (SDM), responsible for data transmission and learning. [1, 2] Before this work, to achieve desired system behavior, a designer needed to configure approx. 15 parameters per NDM, of which there may be many, and approx. 8 per SDM connecting two NDM. In addition, evaluation of the differential equations was done at one, system-wide step-size, which also had to be set by the designer and required an intuitive understanding to achieve sufficiently precise but computationally affordable calculations.

Autonomic Computing and its predecessors in control theory and cybernetics have developed a number of techniques for similar problems, among them policy-based control[5], agent-oriented solutions[6], adaptive heuristics[7] and generative models[3], although they typically focus more on technical management problems.

In this paper we introduce an error model for the differential equation evaluation allowing for self-optimization of the update interval in Sec. 2 and a heuristic self-configuration approach for neural activation in Sec 3. We then move to a short evaluation in Sec. 4 and conclude with Sec. 5.

2. UPDATE INTERVAL OPTIMIZATION

A dynamic neural field is a two-dimensional array of model neurons where each neuron's activation is guided by the differential equation

$$\dot{u}_t^{\vec{x}} = \frac{1}{\tau_\mathrm{N}}(-u_t^{\vec{x}} + I_t^{\vec{x}} + L_t^{\vec{x}} + h_t) \tag{1}$$

where τ_N is a time constant, $I_t^{\vec{x}}$ is the external input to the neuron, h_t is the resting potential and $L_t^{\vec{x}} = \sum_{\vec{x}'} \omega^{\vec{x}\vec{x}'} f[u_t^{\vec{x}'}]$ is the lateral input from other neurons of the field, computed by folding the other neurons' output $f[u_t^{\vec{x}'}]$ with an interaction kernel $\omega^{\vec{x}\vec{x}'}$.

Following the approach taken by the adaptive Runge-Kutta evaluation, but reducing to first order to minimize evaluations, we define an evaluation error between one full update step $u_{t+\Delta t}^{\vec{x}}$ and two half steps $u_{t+2*\frac{1}{2}\Delta t}^{\vec{x}}$ as $\mu_\mathrm{N}^{\vec{x}} = |u_{t+2*\frac{1}{2}\Delta t}^{\vec{x}} - u_{t+\Delta t}^{\vec{x}}|$.

By several transformations, including the Taylor approximation of the update error caused by the lateral terms, introducing a constraint $\Delta t \ll \tau_\mathrm{N}$, we arrive at an error measure for each neuron

$$\begin{aligned} e_{\max} \equiv \max_{\vec{x}} |u_t^{\vec{x}} - I_t^{\vec{x}} - L_t^{\vec{x}} - h_t \\ - m_f \sum_{\vec{x}'} \omega^{\vec{x}\vec{x}'} (u_t^{\vec{x}'} - I_t^{\vec{x}} - L_t^{\vec{x}} - h_t)| \end{aligned}$$

where m_f is the maximal gradient of the transfer function $f[\cdot]$. By providing the maximally acceptable error we get an optimal update rate:

$$\tilde{\mu}_N = \max_{\vec{x}} \mu_\mathrm{N}^{\vec{x}} = \frac{\Delta t^2}{4\tau_\mathrm{N}} e_{\max}$$

$$\Delta t_{ND} \approx 2\tau_\mathrm{N} \sqrt{\frac{\tilde{\mu}_N}{e_{\max}}} \tag{2}$$

ICAC'09, June 15–19, 2009, Barcelona, Spain.
ACM 978-1-60558-564-2/09/06.

In the same way, based on the differential equation for the update of synaptic weight

$$\dot{w}_t^{\vec{x}\vec{y}} = \frac{1}{\tau_{\mathrm{L}}}\overline{a_t}\,\overline{b_t} - \frac{1}{\tau_{\mathrm{FL}}}(\overline{a_t}+\overline{b_t})w_t^{\vec{x}\vec{y}} - \frac{1}{\tau_{\mathrm{FG}}}w_t^{\vec{x}\vec{y}} \tag{3}$$

with time constants τ_{L}, τ_{FL} and τ_{FG} and the averaged neural output $\overline{a_t}$ and $\overline{b_t}$ at both sides of the synapse we can formulate an error estimate to obtain an update rate. In this case, this is done by analytically maximizing the error measure $\mu_{\mathrm{S}}^{\vec{x}\vec{y}} = w_{t+2*\frac{1}{2}\Delta t}^{\vec{x}\vec{y}} - w_{t+\Delta t}^{\vec{x}\vec{y}}$ over $w_t^{\vec{x}\vec{y}}, \overline{a_t}$ and $\overline{b_t}$ as

$$\Delta t_{SD} = 2\sqrt{\frac{\tilde{\mu}_S\,\tau_{\mathrm{FL}}\,\tau_{\mathrm{FG}}\,\tau_{\mathrm{L}}}{\tau_{\mathrm{FL}}+2\tau_{\mathrm{FG}}}} \tag{4}$$

by providing again the maximally acceptable error $\tilde{\mu}_S$.

3. NEURAL ACTIVATION HEURISTICS

Especially caused by the lateral interaction within the field, a stable system state attractor is always present at full activation. Since no processing can be done in this state and the field is completely static, designers usually try to avoid it. This is typically achieved by hand-tuning scalar factors to the terms in Eqn. 1, adding another four parameters. However, even with a good setting an unexpectedly high input may still bring the field into the undesired state.

To overcome this we propose a heuristics modulating the resting potential h_t in form of a "global inhibition":

$$h_t = h_{\mathrm{base}} - \nu_{\mathrm{ND}}\langle f[u_t^{\vec{x}}]\rangle_{\vec{x}} \tag{5}$$

In addition to the neurobiologically motivated parameter h_{base}, this requires only one parameter ν_{ND} which regulates the requested maximal activation independent of the input.

4. EVALUATION

Fig. 1 demonstrates the presented properties in a small simulation scenario consisting of two NDM connected by one SDM. The simulation runs in real-time. External input data to the NDM is applied at $t = 3.25s$, $5.25s$ and $8.25s$. All other activations result from information transmitted through synaptic connections.

The upper graph shows a comparison between the original and the self-managed system, together with the development of h_t. At each full second, h_t is automatically set to -5.0 to reset the field before the next input is processed.

The lower graph shows the update rate, as described in Sec. 2. It can be seen that most of the benefit gained from the self-optimization stems from evaluating NDM and SDM at different rate, where synaptic update is much slower. We can see that the number of NDM/SDM evaluations is reduced to only 38% compared to the original system without self-management.

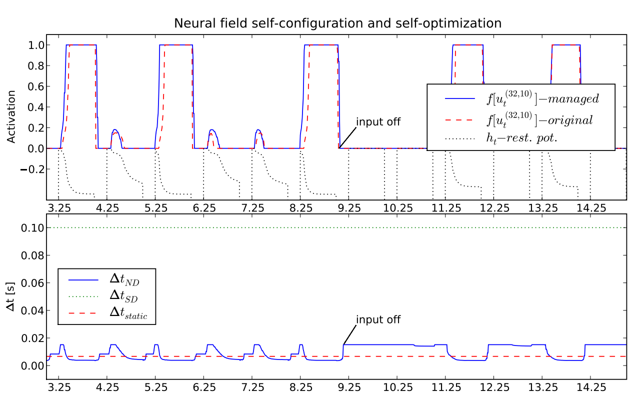

Figure 1: Plot of system behavior – top: comparison of original and self-managed neural activation for one prominent position in the neural field, together with the self-configured resting potential h_t over time; bottom: comparison of the self-optimized update intervals Δt_{ND} and Δt_{SD} with the global update rate of 150Hz of the original system.

5. CONCLUSION

In this contribution we have briefly introduced a self-optimization technique for correctly and autonomously adapting the update rate of differential equation evaluations needed in brain-like system simulation. In addition we introduced a simple heuristics for keeping the neural activation in a working range. We have shown these two extensions to work properly, in addition to keeping a qualitative equivalence with the original system, in a short experiment.

Future work will concentrate on two areas. On the one hand, the substitution of the application specific parameters Δt_{ND} and Δt_{SD} by the generic maximum error boundaries $\tilde{\mu}_N$ and $\tilde{\mu}_S$ is surely a step towards parameter reduction, but we aim to go further, reducing the set to the ones which are really essential to the task the designer wants to solve.

On the other hand, we aim at using the presented techniques to build efficient large-scale neural systems for flexible and adaptive control of autonomous robotic agents that exhibit the necessary robustness in real-world situations. In this respect, the reduced design complexity and increased performance allows for constructing much more complex and realistic embodied systems than before.

6. REFERENCES

[1] S.-I. Amari. Mathematical foundations of neurocomputing. *Proceedings of the IEEE*, 78(9):1441–1463, 1990.

[2] S. Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall, 1999.

[3] T. Heinis and C. Pautasso. Automatic configuration of an autonomic controller: An experimental study with zero-configuration policies. *5th Intl Conf on Aut Comp*, pages 67–76, 2008.

[4] A. Gepperth, J. Fritsch, and C. Goerick. Cross-module learning as a first step towards a cognitive system concept. *1st Intl Conf on Cognitive Systems*, 2008.

[5] J. Kephart and D. Chess. The vision of autonomic computing. *COMPUTER*, pages 41–50, 2003.

[6] G. Tesauro, D. Chess, W. Walsh, R. Das, A. Segal, I. Whalley, J. Kephart, and S. White. A multi-agent systems approach to autonomic computing. *Autonomous Agents and Multiagent Systems, 2004. AAMAS 2004. Proceedings of the Third International Joint Conference on*, pages 464–471, 2004.

[7] G. Tesauro, N. Jong, R. Das, and M. Bennani. A hybrid reinforcement learning approach to autonomic resource allocation. *3rd Intl Conf on Aut Comp*, pages 65–73, 2006.