DOI: http://dx.doi.org/10.7551/978-0-262-32621-6-ch117

Optimizing the crossregulation model for scalable abnormality detection

Danesh Tarapore[1,2,3], Pedro U. Lima[2], Jorge Carneiro[3] and Anders Lyhne Christensen[4]

[1] ISIR, Université Pierre et Marie Curie 6, CNRS UMR 7222, F-75252, Paris Cedex 05, France.
[2] Instituto de Sistemas e Robótica, Instituto Superior Técnico, 1049-001 Lisbon, Portugal.
[3] Instituto Gulbenkian de Ciência, 2780-156 Oeiras, Portugal.
[4] Instituto de Telecomunicações, Instituto Universitário de Lisboa (ISCTE-IUL), 1649-026 Lisbon, Portugal.
daneshtarapore@gmail.com

The engineering of fault-detection systems for multirobot systems (MRS) is a well-studied problem (e.g, Christensen et al. (2009)). Most fault-detection models are built on the assumption that normal behavior is known, and can be characterized in advance. The models are trained to recognize predefined normal behaviors, and behaviors not recognized are labeled abnormal. While such an approach does provide some interesting results of robust fault detection and fault tolerance, they may not be applicable when normal behavior can change as a result of unforeseen environmental conditions and online learning for instance. Furthermore, prior information required to characterize normal behaviors, may not always be available.

The adaptive immune system in vertebrates has to allow the body's cells and tissues to function normally, while mounting a response against possible abnormalities (e.g., cancerous cells) (Leon et al., 2003). The characteristics of these abnormalities are in principle open-ended and therefore differ from current approaches to fault detection in robots. The *crossregulation model* (CRM) (Leon et al., 2003) captures this robust maintenance of immunological tolerance, allowing the system to discriminate between antigens based solely on their density and persistence in the environment. In our previous work (Tarapore et al., 2013), the CRM has been used to develop a decentralized abnormality-detection system for a MRS, wherein each robot simulates its own private population of virtual cells of the immune system. The system was able to tolerate normal swarm behaviors, characterized as persistent and abundant, while mounting an immune response against abnormal behaving agents. Despite these salient features, the model incurred a severe computational cost, consequent to the high-dimension space of behavioral features in which the the normal and abnormal behaviors were classified. Here, we propose optimizations to the CRM, that allow for an improved scalability in terms of the number of features used for behavior classification.

The CRM describes the population dynamics of cells of the adaptive immune system, consisting of three cell types: (i) antigen presenting cells (APCs) that present the antigen on their surface. Individual APCs have a fixed number of conjugation sites (s) on which T-cells can conjugate; (ii) effector T-cells ($\mathtt{T_E}$) that can potentially mount immune responses which, depending on receptor specificity, may be directed to abnormal foreign pathogens or to normal body-antigens; and (iii) regulatory T-cells ($\mathtt{T_R}$) that suppress proliferation of $\mathtt{T_E}$ cells with similar specificities. A mathematical formulation of T-cell–APC interaction dynamics is detailed in Tarapore et al. (2012). Below, we highlight these interactions, and then describe the optimizations to the model.

The CRM provides a system of differential equations governing the density of each of the clonal types (i) of $\mathtt{T_E}$ (E_i), and $\mathtt{T_R}$ (R_i) T-cells. The subpopulations of each of these clonal types is subject to the following: (i) growth by proliferation (division of parent cells into two daughter cells) of their individual activated cells; and (ii) shrinkage consequent to cell death (see parameters in Table 1).

The density of activated $\mathtt{E_i}$ and $\mathtt{R_i}$ cells of each clonal type i, is dependent on their interactions with APCs A_j of each subpopulations j. The resulting T-cell–APC conjugates C_{ij} is described by the following equation:

$$\dot{C_{ij}} = \gamma_c \theta_{ij} (T_i - \textstyle\sum_{j=1}^{M} C_{ij})(A_j s - \sum_{i=1}^{N} C_{ij}) - \gamma_d C_{ij} \quad (1)$$

where $T_i = E_i + R_i$, and γ_c and γ_d involve the conjugation and deconjugation rates between APCs and T-cells, respectively (parameters in Table 1), and θ_{ij} is the affinity of the interactions between T_i and A_j. We integrate at each time step, the steady state values of the conjugates.

The density of activated $\mathtt{T_E}$ and $\mathtt{T_R}$ cells is computed from the quasi-steady state densities of the conjugates. The conjugated $\mathtt{T_E}$ cells are activated in the absence of $\mathtt{T_R}$ cells on the same APC. In contrast, conjugated $\mathtt{T_R}$ cells can only be activated if at at least one $\mathtt{T_E}$ cell is simultaneously conjugated to the same APC.

The implementation of the CRM is optimised by simplifying the dynamics of the conjugates, assuming that the total T-cell density is in excess of the density of conjugated cells.

$$\dot{C_j} = \gamma_c (\textstyle\sum_{i=1}^{N} \theta_{ij} T_i)(A_j s - C_j) - \gamma_d C_j \quad (2)$$

From eq. 2, the quasi-steady state density of the conjugated cells is calculated as the following function, for each

Table 1: Parameters of the crossregulation model.

Param.	Description	Value (a.u.)
l	Length of binary feature vector	l
N	Maximum number of T-cell clones	2^l
M	Maximum number of APC subpopulations	2^l
A_j	Density of APCs of population j	$-$
s	Maximum number of conjugation sites on APC	3
E_i	Density of effector cells of clone i	$-$
R_i	Density of regulatory cells of clone i	$-$
T_i	Density of T-cells of clone i	$E_i + R_i$
C_{ij}	Density of conjugates between T_i and A_j	$-$
γ_c	Conjugation rate of T-cells to APCs	10^{-1}
γ_d	Deconjugation rate of T-cells from APCs	10^{-1}
π_E	Proliferation rate of effector cells	10^{-3}
π_R	Proliferation rate of regulatory cells	0.7×10^{-3}
δ	Death rate of effector and regulatory cells	10^{-6}

existing APC subpopulation j:

$$C_j = (\gamma_c A_j s \Sigma_{i=1}^{N} \theta_{ij} T_i)/(\gamma_d + \gamma_c \Sigma_{i=1}^{N} \theta_{ij} T_i)$$

The total number of conjugated effector and regulatory cells on the APC subpopulation j is then calculated, proportional to the relative frequency of $\mathtt{T_E}$ and $\mathtt{T_R}$ cells, and weighted by their affinity to the APC subpopulation j. For the conjugated effector Ec_j and regulatory Rc_j cells at APC subpopulation j, we have:

$$Ec_j = C_j \frac{\Sigma_{i=1}^{N} \theta_{ij} E_i}{\Sigma_{i=1}^{N} \theta_{ij} T_i} \quad \text{and} \quad Rc_j = C_j \frac{\Sigma_{i=1}^{N} \theta_{ij} R_i}{\Sigma_{i=1}^{N} \theta_{ij} T_i}$$

Finally, the density of activated effector regulatory cells is computed (as in the unoptimized model (Tarapore et al., 2012)), and the population of $\mathtt{T_E}$ and $\mathtt{T_R}$ cells is updated.

The unoptimized and optimized CRM are implemented on a distributed embodied MRS of 20 e-puck-like robots, so as to detect fault-simulating agents, while maintaining tolerance towards normal swarm behavior. Behaviors that are abundant (performed by most agents) are to be tolerated, and rare behaviors (exhibited by fewer agents) are to be detected as abnormal.

Individual features of an agent's behavior are encoded in Boolean form, and then concatenated to form a binary string, the *feature vector* (FV). A FV comprises 3, 6, 9, 12 and 15 features (Tarapore et al., 2014). At every control cycle, each agent senses the FVs of its ten nearest neighbors, and computes the number of agents assigned to each FV. In the agent's internal CRM instance, APCs are generated corresponding to each of the feature vectors perceived, and the CRM is then executed to determine their status.

Normal behaviors exhibited by the swarm are aggregation, dispersion, flocking and homing towards a moving landmark. The fault-simulating behaviors performed are, (i) move continually in a straightly line; (ii) perform a random walk; (iii) circle around a fixed point; or (iv) stop completely. These behaviors mimic faults such as, software bugs, sensor faults, motor malfunctions and a broken battery. The CRM was able to detect abnormalities in almost all

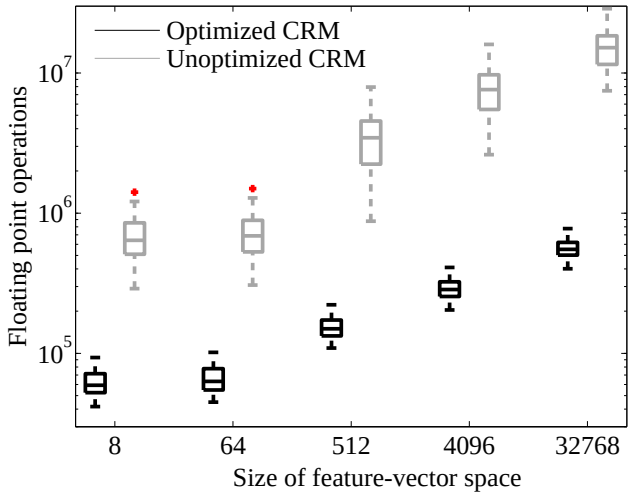

Figure 1: Computational costs of unoptimized (light) and optimized (dark) abnormality detection using the CRM with FV space for 3, 6, 9, 12 and 15-bits FV length, across 20 replicates. Each box corresponds to the average number of FLOs per agent, per control cycle.

normal/fault-simulating behavior combinations (Tarapore et al., 2014). In one such experiment one fault-simulating agent performed random walk, and the other agents of the swarm performed aggregation. We show the number of floating points operations (FLOs) executed by the CRM (Fig. 1). Results indicate that the computational cost increases with an increase in FV length, for both the unoptimized and the optimized models. However, optimizations to the model yield a reduction in execution cost, particularly for large FVs (up to an order of magnitude reduction in FLOs for 15-bit FVs). The results encourage us to use the optimized version of the model in more complex scenarios.

References

Christensen, A. L., O'Grady, R., and Dorigo, M. (2009). From fireflies to fault tolerant swarms of robots. *IEEE Trans. Evol. Comput.*, 13(4):1–12.

Leon, K., Lage, A., and Carneiro, J. (2003). Tolerance and immunity in a mathematical model of T-cell mediated suppression. *J. Theor. Biol.*, 225:107–126.

Tarapore, D., Christensen, A. L., Lima, P. U., and Carneiro, J. (2012). Environment classification in multiagent systems inspired by the adaptive immune system. In *Proc. of ALIFE*, pages 275–282. MIT Press, Cambridge, MA.

Tarapore, D., Christensen, A. L., Lima, P. U., and Carneiro, J. (2013). Abnormality detection in multiagent systems inspired by the adaptive immune system. In *Proc. of AAMAS*, pages 23–30. IFAAMAS, Red Hook, NY.

Tarapore, D., Lima, P. U., Carneiro, J., and Christensen, A. L. (Submitted, 2014). Scalable abnormality detection in multiagent systems.