

Requisite Social Influence in Self-Regulated Systems

Asimina Mertzani^a and Jeremy Pitt^b

Electrical and Electronic Engineering Dept., Imperial College London, London, U.K.

Keywords: Cybernetics, Self-Regulated Systems, Requisite Influence, Reinforcement Learning, Social Psychology.

Abstract: This paper specifies, implements and experiments with a new psychologically-inspired *4voices algorithm* to be used by the units of a self-regulated system, whereby each unit learns to identify which of several “voices” to pay attention to, depending on a collective desired outcome (e.g., establishing the ground truth, a community truth, or their own “truth”). In addition, a regulator uses a standard Q-learning algorithm to pay attention to the regulated units and respond accordingly. The algorithm is applied to a problem of continuous policy-based monitoring and control, and simulation experiments determine which initial conditions produce systemic stability and what kind of “truth” is expressed by the regulated units. We conclude that this synthesis of Q-learning in the regulator and *4voices* in the regulated system establishes *requisite social influence*. This maintains quasi-stability (i.e. periodic stability) and points the way towards *ethical regulators*.

1 INTRODUCTION

A self-regulated system in cybernetics comprises a designated agent (or agency) acting as regulator by operating on some control variables, and a regulated system which applies changes in those variables (Ashby, 2020). Moreover, the units of regulated system, as communicative agents themselves, can report the effects of applying the changes amongst themselves, being inter-connected by a social network.


From this perspective, establishing the *pathways to requisite influence* (Ashby, 2020) that would ensure timely feedback, responsive control and overall systemic stability, has two requirements: firstly, expression by the regulator to the regulated system and attention of the regulated system to the regulator (and vice versa), and secondly, social influence within the social network of the regulated system, with respect to some overall goal (Nowak et al., 2019).


In this paper, we examine these requirements in the context of a self-regulated system addressing a problem of continuous policy-based monitoring and control, where the regulator specifies a policy and the regulated system applies the policy, and experiences its effect. The regulator uses a standard Q-learning algorithm to learn to pay attention to the regulated units and responds (expresses a new policy) accordingly. However, we also specify and implement a

new psychologically-inspired *4voices algorithm* for the units of the regulated system, whereby in determining their expression to the regulator, each unit of the regulated system learns to identify which of several “voices” to pay attention to, depending on a desired collective outcome.

The psychological motivation is threefold. First, we start from Nowak’s Regulatory Theory of Social Influence (Nowak et al., 2019) which proposes that, for reasons of coherence and cognitive efficiency, in social networks targets seek sources by whom to be influenced, as well as sources seeking targets. Secondly, these targets actually have a polyphony of *inner voices* (Ferryhough, 2017), which, thirdly, are more or less activated by selective auditory attention (e.g. the cocktail party effect (Driver, 2001)).

To test the effectiveness of the new *4voices algorithm* with respect to systemic stability, a set of simulation experiments are designed and executed, with the regulator using a Q-learning algorithm and the units of the regulated system using the *4voices algorithm*. Essentially this uses a set of coefficients to learn how to distinguish between the inner voices depending on the desired outcome (i.e., whether it is to express the community truth, the ground truth or their inner truth, cf. (Deutsch and Gerard, 1955)). We conclude that this synthesis of Q-learning in the regulator and *4voices* in the regulated system establishes *requisite social influence*. This maintains stability and points the way towards *ethical regulators*.

^a  <https://orcid.org/0000-0002-6084-9212>

^b  <https://orcid.org/0000-0003-4312-8904>

2 PROBLEM SPECIFICATION

In this section, we identify the pathways to requisite influence in a self-regulated system as the expression of the regulator to the regulated units, and their attention to the regulator; and the expression of the regulated units to the regulator, and its attention to their feedback; and finally social influence in the social network of the regulated units. This is the ‘generic problem’, and we apply it to a specific problem of continuous monitoring and control. Note this is not necessarily an ‘optimal’ solution to the specific problem: instead we are establishing a test-bed for stability and *requisite social influence* in self-regulated systems.

2.1 The Generic Problem

We refine Ashby’s (Ashby, 2020) exposition of a self-regulated system by adding the *4voices* model of attention and expression to the regulated units, pathways to requisite influence, individual learning and collective learning, as illustrated in Figure 1.

Specifically, following Ashby’s description of a self-regulated system, the regulated units need to form a reliable expression to make the regulator aware of their state, and the regulator needs to find the appropriate way to attract the attention of the units to communicate the effects of the change of policy. This becomes then an iterative process starting from the side of the regulator who is selecting a policy, communicates it to the regulated units and influences them while it continues to the side of the regulated units which have to attend to that change, distinguish between the voices of social influence, and form an expression that will make the regulator aware of the actual effect of the change.

Accordingly, the aim of this paper is to design a model in which the group (the regulated system) learns where to pay attention and how to express itself and the regulator learns to attend to that expression, change policy, and communicate this change.

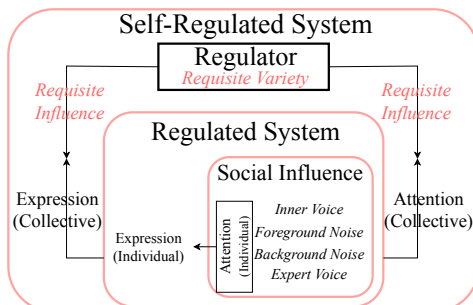


Figure 1: The Self-Regulated System.

2.2 The Specific Problem

The effectiveness of our *4voices* model is tested in a specific scenario of continuous monitoring and control: job scheduling in cloud computing. In more detail, the system operates in *epochs*, in each epoch, each agent i delegates a job to the cloud, and receives a feedback whether the job will be processed in the next epoch t . This specific problem was selected because it satisfies the necessary requirements of being dynamic and non-deterministic, since it comprises heterogeneous networked units; the preferable policy is a matter of context, opinion and self-adaptation; while it is relatively easy to define some metrics that describe its operation.

Specifically, the self-regulated system comprises N agents connected over a social network which participate in a cloud system. In every epoch t , each agent i delegates a job with size $j_i \in \{1, J\}$ and urgency $u_i \in \{1, U\}$ to the cloud, where J is the maximum job size and U is the maximum urgency. Based on its policy, the regulator determines the number of jobs to be processed m in the next epoch. Then, the regulator orders all jobs based on urgency (descending) and size (ascending), selects the m first jobs to process and informs the agents.

Next, the regulator processes those jobs based on the described order, which results in a total cost C equal to the cost J of each job of size j_i aggregated with a fixed cost F . C is equally distributed among the m agents, resulting in the cost c_i , and the average cost \bar{C} is equal to the total cost divided by the number of jobs m . Also, jobs are processed with a delay q_i . The average delay \bar{Q} equals the sum of the delays divided by m , $\bar{Q} = \frac{\sum_{k \in \{1, \dots, m\}} q_k}{m}$. The maximum delay Q_M equals the sum of the delays of N jobs (assuming that they were all accepted and processed according to the described order), divided by N , $Q_M = \frac{\sum_{k \in \{1, \dots, N\}} q_k}{N}$.

3 THE 4VOICES ALGORITHM

This section provides the formal specification of the *4voices*, the process of selection between those voices as well as the learning of the regulated units and the regulator. This section concludes with a specification of the formal *4voices* algorithm, which comprises three components: processes for the specific problem, processes of attendance and selection between the voices, and processes of learning.

3.1 Voices of the Regulated System

Drawing inspiration from psychological insights suggesting that people facing decision-making problems often activate an internal monologue used as a distinct type of auditory thinking called “inner speech” or “inner voice” (Fernyhough, 2017), we include this polyphony in inner speech in our formal model. Following the identification in ergonomics of intimate, personal, social and public distances (Osborne and heat, 1979), we define four voices corresponding to individual’s ‘own voice’ (Fernyhough, 2017), based on direct personal experience; the ‘expert voice’, emanating from education or ‘trusted’ expertise (Horne et al., 2016); the ‘foreground noise’, from the individual’s social network (Nowak et al., 2019); and the ‘background noise’, coming from the community or culture in which the individual is embedded (Deutsch and Gerard, 1955).

Applying the *4voices* algorithm to the problem of job scheduling, depending on the delay, the cost, the urgency and whether the job is accepted to be processed or not, each agent i forms an opinion which is reflected in its own voice n_i^o . As such, if the job of agent i is accepted, then the absolute value of the own voice n_i^o reflects the quality of service divided by the cost, while the actual value is negated to reflect the intuition that a higher noise corresponds to higher dissatisfaction while a lower noise reflects satisfaction, and is given by Equation 1:

$$n_i^o = -\frac{Q_M - q_i}{c_i} \quad (1)$$

If the job of the agent is not accepted then, the own voice n_i^o remains the same as in the previous epoch.

Being part of a group, the agents are also exposed to the voices of other agents. Thus, apart from their own voice n_i^o , they audit the noise generated by the group n_i^b (background noise), the noise from their social network n_i^f (foreground noise) and the noise generated by the experts n_i^e (experts voice).

This way, the expert voice n_i^e which agent i might audit reflects the actual state of the system, and is equal to the average quality of service with respect to the average individual cost aggregated with the urgency factor \mathcal{U}_f . \mathcal{U}_f represents the trade-off between trying to partially satisfying everyone (accepting all the jobs regardless of the urgency resulting in high average delay), and fully satisfying few (prioritising the few the jobs of which are urgent resulting in small average delay). Specifically, the urgency factor \mathcal{U}_f equals the sum of the urgencies of the m accepted jobs minus the sum of the urgencies of the $N - m$ rejected

jobs, divided by the sum of urgencies of all N jobs.

$$\mathcal{U}_f = \frac{\sum_{k \in \{1, \dots, m\}} u_k - \sum_{k \in \{m+1, \dots, N\}} u_k}{\sum_{k \in \{1, \dots, N\}} u_k}$$

Overall, the expert voice is given by Equation 2, where the first term corresponds to the ‘objective’ quality of service while the second corresponds to the urgency multiplied by a normalising constant u_c .

$$n_i^e = -\frac{Q_M - \bar{Q}}{C} + (\mathcal{U}_f * u_c) \quad (2)$$

Note that the term “expert” refers to the agents initialised to have an overview of the system, expressing not just an opinion based on their personal experience (as the other agents do), but an objective opinion reflecting the actual state of the regulated system.

Furthermore, interconnected units comprising the regulated system might be influenced by the opinion (i.e. the noise) of their peers or they might seek for the opinion of others if they are not confident about their own opinion (or, in this case, voice), as depicted in Nowak’s RTSI theory (Nowak et al., 2019). To reflect the influence from the social network, we include the foreground noise in the voices of our model. Specifically, every agent i is connected with at least one other agent and can be influenced by them. The social network SN is a Small-World Scale-Free (Klemm and Eguíluz, 2002) network resembling real life networks, combining the properties of both ‘scale free’ (Watts and Strogatz, 1998) and ‘small world’ (Barabási et al., 1999) networks, namely distribution of degree following the power law, high clustering coefficient, small average shortest path. The network comprises N nodes (agents), initialised to have m_{SN} number of fully connected nodes, and μ probability of adding an edge to one of the fully connected nodes.

Following the formal specification of social influence in (Mertzani et al., 2022), each agent i has a credence $cr_{i,j}$ to each of its agents j in their social network SN_i . Consequently, each agent i orders its neighbours with a descending order of credence. The foreground noise corresponds to the own voice produced by the agent j , where j is selected according to the following process: agent i selects the first agent from the list CR_i and with probability $p - \epsilon_{net}$ uses its own voice to produce the foreground noise. Otherwise, the agent i selects a random agent. This stochasticity in the selection of agent is used to restrict agent from being always influenced by the same agent.

Accordingly, the foreground noise attended by agent i is the own voice of the agent j from i ’s social network and is given by Equation 3:

$$\begin{aligned} n_i^f &= n_j^o | \max_{j \in SN_i} (cr_{i,j}), & w.p. (p - \epsilon_{net}) \quad (3) \\ &= n_k^o | k \in \text{random}(SN_i), & w.p. \epsilon_{net} \end{aligned}$$

The background noise corresponds to the noise produced by the community and, thus, is defined as the average of the own voices of a random sample of the population, the size of which is $v = \frac{N}{5}$ agents, as presented in Equation 4:

$$n_i^b = \frac{\sum_{i \in \{1, \dots, v\}} n_i^o}{v} \quad (4)$$

3.2 Voice Selection for Individual & Collective Expression

Each agent i has to select between the four voices and form its individual expression e_i . This decision is made upon comparison of the values of attention to the corresponding voices. Specifically, i has an attention to each of the four voices, with att_i^o being the attention to the own voice, att_i^e the attention to the expert voice, and att_i^f and att_i^b the attention to the foreground and background noise respectively. Therefore, in every epoch, the agent compares the attentions and selects the greatest. For instance, if $\max(att_i^o, att_i^e, att_i^f, att_i^b)$ is att_i^o then i selects the own voice and forms its individual expression ($e_i = n_i^o$).

Additionally, to encourage exploration and avoid biasing the agents towards the voice selected during the first epochs, we introduce a stochastic parameter (following the approach used for the computation of the foreground noise). Specifically, with probability $p - \epsilon_{voice}$, the voice with the greatest attention is selected, while with probability ϵ_{voice} the voice is selected randomly. Note that, the four attentions are initialised to be equal, so that each agent is neutral in the beginning, and are updated throughout the epochs based on the experience of each agent.

Having access to its own voice, the foreground noise, the background noise and the expert voice, each agent i can select between those four voices and form its individual expression e_i . The individual expressions of the agents are aggregated and form the collective expression \mathcal{V} that is presented in Equation 5:

$$\mathcal{V} = \sum_{i \in \{1, \dots, N\}} e_i \quad (5)$$

3.3 Learning to Attend to Voices

For the update of the attention, we use the reinforcement coefficient c which determines the amount of positive or negative update occurring in every epoch. Thus, depending on the value of the experimental pa-

rameter *update* either the agent learns to pay attention to the voice that deviates less from the experts (*update* = ‘Exp’), either learns to pay attention to the voice that resembles the average expression of the group (*update* = ‘Col’) or learns to pay attention to the voice that seems to have the best immediate effect (*update* = ‘Ind’) resulting in a lower own voice (agent more satisfied) in the next epoch.

Specifically, in the first case, after attending all the voices, the agent reinforces its attention to the voice that deviates less from the expert voice, by increasing the attention to this voice by c , and penalises the attention to the voice that deviates the most, by decreasing it by c . If an expert can be accessed by that agent, the voice of which the attention is increased is simply the expert voice, but if an expert cannot be accessed, then the attention to one of the other voices will be increased. In the second case (*update* = ‘Col’), the attention to the voice that deviates less from the average expression of the agents (corresponding to $\frac{\mathcal{V}}{N}$) is increased and the one that deviates the most is decreased. Finally, when *update* = ‘Ind’ the update in the attention is based on the comparison of the value of each voice agent in epoch t with the own voice of i in $t + 1$. So, the agent computes the values of $(n_i^o)^t - (n_i^o)^{t+1}$, $(n_i^f)^t - (n_i^f)^{t+1}$, $(n_i^b)^t - (n_i^b)^{t+1}$, and $(n_i^e)^t - (n_i^e)^{t+1}$ and the attention to the lower is increased by c , while the one that is greater is decreased.

The process of updating the agents’ attention constitutes a simple form of learning, where agents are reinforcing their attention to the voices based on the value of the information that they offer in evaluating the actions of the regulator. This process enables agents to learn to which voice attention should be paid, depending on what they consider more reliable to form their expression, based on their goal, i.e. defined by the parameter *update*. Therefore, if the agents consider that the regulator should be aware of what the experts think, then they will reinforce attention to the expert voice in order to form their individual expression based on that voice. This reflects the confirmation bias observed in human societies, since often individuals tend to listen to the opinions that are aligned with their beliefs, rather than the opinions that are more appropriate.

3.4 Learning to Attend to Expressions

Moving from individual to regulator’s learning, here we aim to design a regulator that has the required rational complexity to react effectively to the expressions of the regulated system and produce systemic stability. Therefore, instead of using reinforcement coefficients, we use Reinforcement Learning (RL).

In RL, such a learning process can be modeled as a Markov Decision Process (MDP). The MDP is defined as $(S, \mathcal{A}, P, R, \gamma)$ where \mathcal{A} is a set of actions, S is a set of states, and P is the state transition probability function, \mathcal{R} the reward function and γ is the discount factor. Thus, s is a state from the set S , a is an action selected from the set \mathcal{A} , $P(s, a)$ defines the probability of a transition from state s_t to state s_{t+1} when an agent executes action a and $r \in R$ is the immediate reward received when agents performs an action.

A policy π maps from states to probability distributions over actions and is denoted by $\pi: S \rightarrow p(\mathcal{A} = a|S)$. An agent's goal is to learn a policy π that maximises its expected return $\mathcal{R}_\pi := \sum_{t=0}^{\infty} \gamma^t R_t$. To learn that policy one can learn an action-value function that is defined as $Q: (S \times \mathcal{A})$, where $Q(s, a)$ represents the value of action a in state s . (Watkins, 1989) introduced the Q-learning algorithm, as a way of learning the optimal state-action value Q based on the Bellman equation. According to the Q-learning algorithm, the function Q is updated at each time step by:

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha(r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t))$$

where α is the step-size and (s_t, a_t, r_t, s_{t+1}) is the transition from state s_t to s_{t+1} with reward r_t and γ is the learning rate.

To enable large scale RL, (Li, 2017) proposed the use of deep neural networks (DNN). (Mnih et al., 2015) introduced Deep Q-Learning Networks (DQN) which are using DNNs for approximating the optimal state-action function Q in Q-learning. At each time step the DQN is using stochastic gradient descent to minimise the loss between the learnt network θ and the target θ^- , described by:

$$\frac{1}{2} (r_t + \gamma \max_{a'} Q_{\theta^-}(s_{t+1}, a') - Q_{\theta}(s_t, a))^2 \quad (6)$$

Here, we use a variant of Q-learning algorithm (Watkins and Dayan, 1992) with Deep Q-Learning Network (DQN) (Mnih et al., 2013), which combines Double Q-Learning (van Hasselt et al., 2015) to prevent overestimating with Dueling Networks (Wang et al., 2016) to generalise learning across different actions without changing the underlying RL algorithm. We use the implementation of DQN from (Raffin et al., 2021), and the model is trained to minimise the loss in Equation 6.

The action space of the regulator has dimension $N \times 1$ and comprises a vector with all possible $m_i \in \{1, N\}$, the state space has dimension $N \times 3$ and comprises 3 vectors describing the jobsizes of each agent in this epoch j_i , the urgency of this job u_i and the signal s_i corresponding to whether the job of the agent was accepted or rejected. The reward equals the negative of the collective expression i.e. $\mathcal{R} = -\mathcal{E}$ to

reward positively low collective expression (satisfaction) and negatively high collective expression (dissatisfaction). We use a Deep Q-Learning Network (DQN) to increase the regulator's complexity compared to the regulated system and enable it to effectively respond to the expressions of the agents. Finally, the process of training lasts 10000 epochs divided into episodes of 2000 epochs, while the learning starts after the 1000 epoch. The exploration constant ϵ is initialised to 1 and, during training, is being progressively reduced to 0.1.

3.5 The 4voices Algorithm

Following to the description above, the algorithm describing our 4voices model is presented in Algorithm 1. The parts in green refer to the specific problem, i.e. job scheduling in cloud, the parts in blue refer to the 4voices, and the parts in purple refer to learning.

```

while  $t < T$  do
    Regulator Specifies Policy;
    Agents Generate Jobs;
    Job Scheduling According to Regulator's Policy;
    for each agent  $i, 1 \leq i \leq N$  do
        Attend Own Voice (Equations 1);
        Attend Expert Voice (Equation 2);
        Attend Foreground Noise (Equation 3);
        Attend Background Noise (Equation 4);
        Voice Selection for Expression (Section 3.2);
    end
    Collective Expression (Equation 5);
    Regulator: Learns to Attend to Expression (Section 3.4);
    for each agent  $i, 1 \leq i \leq N$  do
        Learn to Attend to Voices (Section 3.3);
    end
    inc  $t$ ;
end
    
```

Algorithm 1: The 4voices model.

4 EXPERIMENTAL RESULTS

In this short paper, we describe only two experiments: whether the 4voices establishes the pathways for requisite influence, i.e. attention/expression both ways between regulator and regulated units, and under which initial conditions stability is produced.

We set $\epsilon_{net} = \epsilon_{voice} = 0.3$, $c = 0.01$, $u_c = 10$, $J = U = 10$, $F = 500$ and $N = 100, \mu = 0.5, m_{SN} = \frac{N}{10}$. The experiments were repeated for many runs and the graphs present the outcome after averaging over 10 runs. However, the second experiment presents the results of a single representative run to highlight the emerging behaviour.

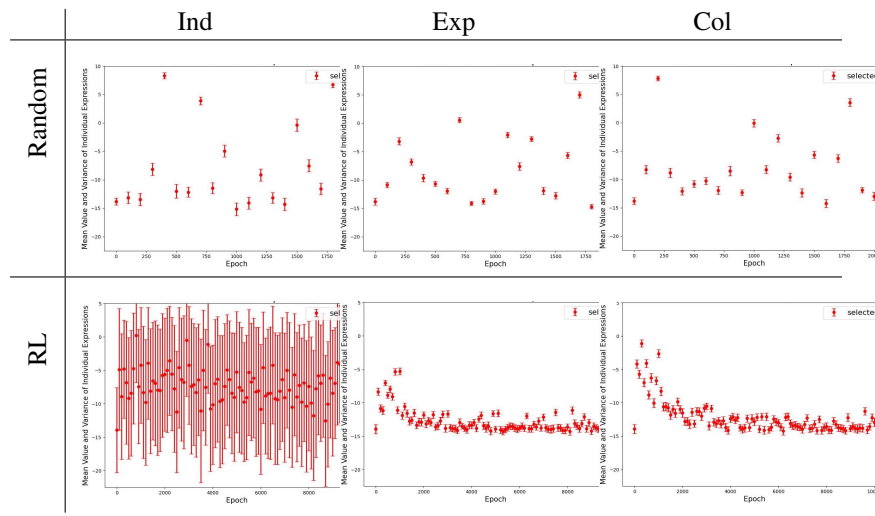


Figure 2: The mean value and the standard deviation of the individual expressions for Random and RL regulator after averaging over 10 runs.

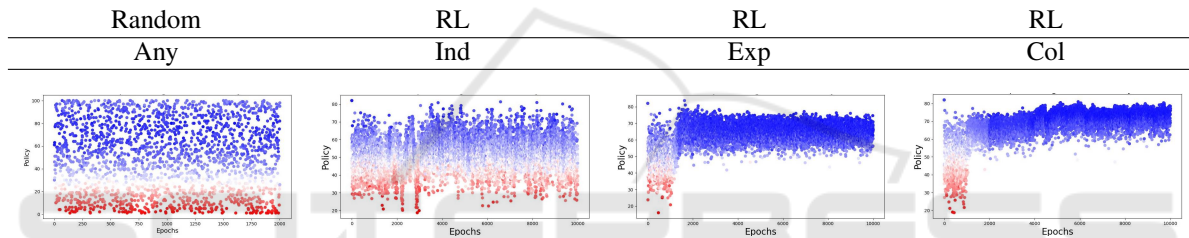


Figure 3: Expression of the Regulated Units when the Regulator is Random and when RL for Different updates of Attention after averaging over 10 runs.

4.1 Pathways for Requisite Influence

This experiment focuses on the observation of whether the *4voices* establishes the pathways for requisite influence. Figure 2 illustrates the mean value and the standard deviation of the individual expressions of the agents (y-axis) for different epochs (x-axis), for different types of update of attentions of the agents’ shown in different columns and for a random and an RL regulator, presented in different rows. Complementing the results of the individual perspective of the agents presented in Figure 2, Figure 3 provides the collective perspective, and specifically shows the policy selected (i.e. number of agents to be included in the next calculation) by the regulator in y-axis at each epoch in x-axis. The different colours of the dots represent the volume of noise, where the blue correspond to comparatively low volume, while the red correspond to comparatively high volume.

From Figure 2, we notice that when the regulator selects a policy randomly, the agents’ noise differs from epoch to epoch, reflecting the agents’ reaction to the policy selected. In contrast, when the regulator decides the policy based on a reinforcement learning

algorithm, the mean value of the agents’ noise is decreased for ‘Exp’ and ‘Col’ types of update of attention. This shows that the agents attend the expression of the regulator and effectively adjust their individual and collective expressions, depending on whether the policy selected is desired or not.

The same observation (i.e. the pathways for requisite influence from the regulator to the regulated units are established) can be seen in Figure 3 that illustrates the collective expression with respect to the selected policy by the regulator. Here, we notice that when the policy includes multiple agents (more than 40) the agents noise becomes lower, while when the regulator includes only a few agents the group reflects the dissatisfaction by increasing the volume of noise.

Additionally, when the agents’ update of attention is based on ‘Exp’ or ‘Col’, their expression reflects their satisfaction and dissatisfaction and establishes the pathways for requisite influence from the regulated units to the regulator. This can be observed in the third and fourth columns of Figure 3, where the regulator identifies the range of policies that is desired by the agents’ and after some iteration selects a policy only from the 50-80 agents interval. *The combi-*

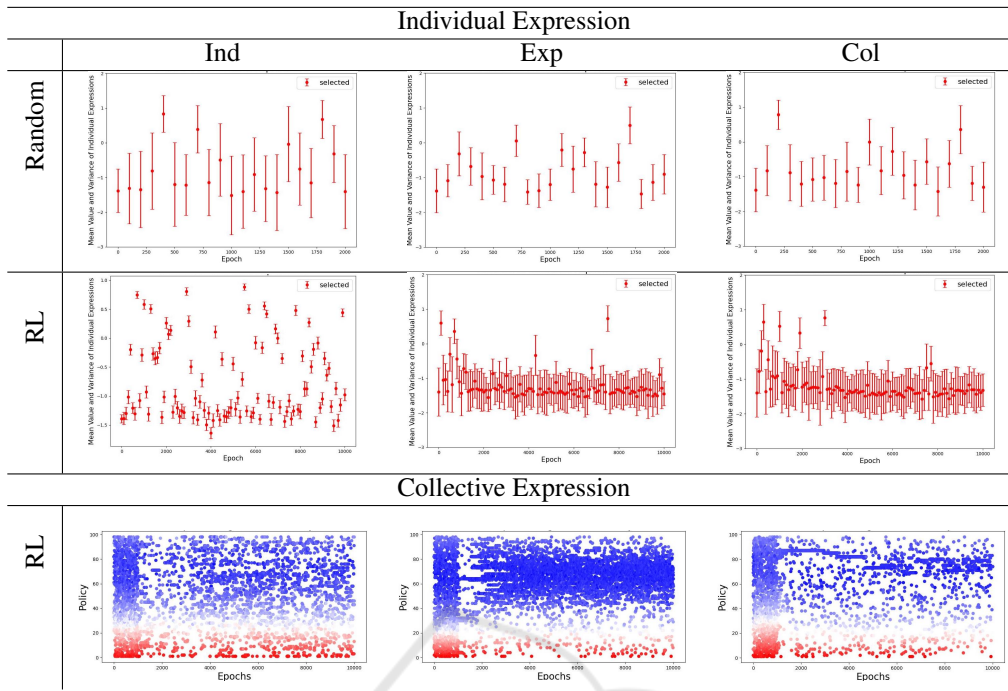


Figure 4: The mean value and the standard deviation of the individual expressions for Random and RL regulator and the policy selection and the corresponding collective expression of an RL regulator of a single run.

nation of the observations made above shows that in any case the pathways for requisite influence from the regulator to the regulated units are established, while the pathways from the regulated units to the regulator are established only for ‘Exp’ and ‘Col’ types of update. This is important since it constitutes a way to enable a collective (comprising the regulated system) to have influence on the learning algorithm of the regulator, by feeding it with the data that forms policy which helps them to achieve their goal (i.e. maximisation of collective satisfaction), similarly to (Hardt et al., 2023).

4.2 Initial Conditions & Stability

Moreover, looking in the graphs resulted from a single run in Figure 4, we notice that there is a trade-off between diversity and congruence. Specifically, moving from the own voice and the foreground noise (selected when the update is ‘Ind’) to the expert voice and background noise (selected when the update of attention is equal to ‘Exp’ and ‘Col’) the standard deviation of the individual expressions is decreased (low diversity). Moreover, the results in the second column of the third row of Figure 4 show that the experts noise remains almost the same if the selected policy remains in the 40-80 agents interval, whereas in the third column we notice that there is a limited interval

of satisfying policies in the ‘Col’ case which is relevant to the network properties and the agents affecting more the others’ expressions.

In this experiment we aimed to observe which initial conditions result in systemic stability, which is crucial when addressing issues of sustainability in socio-technical systems. As such, combining the findings of both experiments, we conclude that *systemic stability is maintained when the regulator selects a policy using RL and the agents’ update their attention based on the divergence from the experts’ voice or collective decision* (update is ‘Exp’ or ‘Col’). This is because, referring back to the third and fourth columns of Figure 3, we notice that the regulator, after enough iterations, identifies the appropriate range of policies which results in comparatively low noise. This produces systemic stability (reflected by the persistent blue coloured volume of noise).

5 SUMMARY & CONCLUSIONS

This paper has essentially addressed two related problems in self-regulated systems. Firstly, ensuring that pathways for requisite influence exist, in the form of awareness (of the regulator) to the expression (of the regulated units), in expression (by the regulator) and attention (by the regulated units) to that expression,

and in the form of social influence in the social network of the regulated units. Secondly, to identify under which initial conditions the self-regulated system can maintain stability, and so establish what could be called *requisite social influence*.

The solution proposed in this paper has been to synthesise ideas from social influence and machine learning to address the dual problems of requisite influence and stability in continuous monitoring and control of a dynamic and non-deterministic system. Specifically, the contributions of this paper are

- based on ideas from opinion formation, dynamic social psychology and psychoacoustics, to introduce the *4voices* model for regulated units, which identifies an own voice, expert voice, foreground noise and background noise as possible sources of social influence;
- to specify the *4voices* algorithm, which combines computation of a signal value for each of the voices, with reinforcement of attention based on the experience from past interactions with the voices for the regulated units, and Deep Q-learning for the regulator to learn the effect of its actions; and
- experimental results which show that for control processes in dynamic and non-deterministic systems, the *4voices* algorithm establishes both the required relational complexity and the pathways for requisite influence, so that the systemic stability is maintained.

Beyond this, according to Ashby (Ashby, 2020), an ethical regulator is required not only to reach reliable decisions from potentially unreliable evidence, but also to evaluate the consequences of its decisions, which raises issues of transparency and accountability in the regulator. Correspondingly, though, an ethically regulated unit should try to provide information to the best of its knowledge and belief. The future challenge lies in ensuring ethical behaviour with respect to *values*, especially in systems with multiple stakeholders with different priorities and preferences with respect to those values.

REFERENCES

- Ashby, M. (2020). Ethical regulators and super-ethical systems. *Systems*, 8(5):3:1–3:36.
- Barabási, A.-L., Albert, R., and Jeong, H. (1999). Mean-field theory for scale-free random networks. *Physica A: Statistical Mechanics and its Applications*, 272(1):173–187.
- Deutsch, M. and Gerard, H. (1955). A study of normative and informational social influences upon individual judgment. *The Journal of Abnormal and Social Psychology*, 51(3):629–636.
- Driver, J. (2001). A selective review of selective attention research from the past century. *British Journal of Psychology*, 92(1):53–78.
- Fernyhough, C. (2017). *The Voices Within*. Wellcome Collection.
- Hardt, M., Mazumdar, E., Mendler-Dünner, C., and Zrnic, T. (2023). Algorithmic collective action in machine learning. ICML'23. JMLR.org.
- Horne, B., Nevo, D., Freitas, J., Ji, H., and Adali, S. (2016). Expertise in social networks: How do experts differ from other users? In *Proceedings Tenth International AAAI Conference on Web and Social Media*, pages 583–586.
- Klemm, K. and Eguíluz, V. M. (2002). Growing scale-free networks with small-world behavior. *Phys. Rev. E*, 65:057102.
- Li, Y. (2017). Deep reinforcement learning: An overview. *arXiv preprint arXiv:1701.07274*.
- Mertzani, A., Pitt, J., Nowak, A., and Michalak, T. (2022). Expertise, social influence, and knowledge aggregation in distributed information processing. *Artificial Life*, 29(1):37–65.
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. A. (2013). Playing atari with deep reinforcement learning. *CoRR*, abs/1312.5602.
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533.
- Nowak, A., Vallacher, R., Rychwalska, A., Roszczynska-Kurasinska, M., Ziembowicz, K., Biesaga, M., and Kacprzyk, M. (2019). *Target in control: Social influence as distributed information processing*. Cham, CH: Springer.
- Osborne, D. and heat, T. (1979). The role of social space requirements in ergonomics. *Applied Ergonomics*, 10(2):99–103.
- Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., and Dormann, N. (2021). Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8.
- van Hasselt, H., Guez, A., and Silver, D. (2015). Deep reinforcement learning with double q-learning.
- Wang, Z., Schaul, T., Hessel, M., van Hasselt, H., Lanctot, M., and de Freitas, N. (2016). Dueling network architectures for deep reinforcement learning.
- Watkins, C. J. and Dayan, P. (1992). Q-learning. *Machine learning*, 8:279–292.
- Watkins, C. J. C. H. (1989). Learning from delayed rewards. *Dissertation, King's College, Cambridge*.
- Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442.