AUTONOMOUS DEVELOPMENT OF SINGING-LIKE INTONATIONS BY INTERACTING BABBLING ROBOTS

Eduardo R Miranda
Interdisciplinary Centre for Computer Music Research (ICCMR)
University of Plymouth, UK - http://cmr.soc.plymouth.ac.uk
eduardo.miranda@plymouth.ac.uk

ABSTRACT

A group of interactive autonomous singing robots were programmed to develop a shared repertoire of vocal singing-like intonations from scratch, after a period of spontaneous creations, adjustments and memory reinforcements. The robots are furnished with a physical model of the vocal tract, which synthesises vocal singing-like intonations, and a listening mechanism, which extracts pitch sequences from audio signals. The robots learn to imitate each other by babbling heard intonation patterns in order to evolve vectors of motor control parameters to synthesise the imitations.

1. INTRODUCTION

For the purpose of this work, we assume that (a) music consists of units of sound organized in specific ways and (b) the ways in which these sounds are organised evolves in a social context and is culturally transmitted. Models of the mechanisms underlying the dynamics of such organization, evolution and cultural transmission are bound to provide new insights into building interactive intelligent music systems. In this paper we introduce an experimental AI system whereby a group of interactive robots programmed with appropriate motor (vocal), auditory and cognitive skills can develop a shared repertoire of short intonation patterns from scratch, after a period of spontaneous creation, adjustment and memory reinforcement. The robots develop vectors of motor control parameters to produce imitations of heard intonation patterns. A larger paper describing this research in more detail will appear in [8].

2. THE MODEL

The robots in a group are expected to form a common repertoire of intonation patterns from scratch (i.e., all by themselves, with no previous exposure to music) by "babbling" imitations to each other. A robot must develop a repertoire that is similar to the repertoires of its peers. Metaphorically speaking we could say that the intonation patterns create some form of "social identity" for the robots, which can be assessed in terms of the similarity of their repertoires. The importance of imitation for the acquisition of behaviour has gained much attention after the discovery of mirror neurons in the frontal lobes of macaque monkeys [10].

2.1. The Architecture

The robots (Fig. 1) are equipped with a voice synthesiser, a hearing apparatus and a memory device. The voice synthesiser is implemented as a physical model of the vocal tract, which is able to synthesise formants and a number of vocal-like sounds. The robots need to compute three vectors of parameters for a synthesiser in order to produce vocal-like intonations: a) lung pressure, b) the width of the glottis (interarytenoid), and c) the length and tension of the vocal chords (cricothyroid) [1, 6]. As for the hearing apparatus, it employs short-term autocorrelation-based analysis to extract the pitch contour of a vocal sound [7].

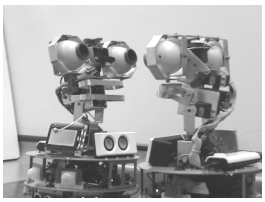

Figure 1 The model uses commercially available robots (Dr Robot DRK8080), which were adapted at ICCMR for high-quality voice synthesis and analysis with sampling rate at 22,050 Hz.

Essentially, the memory of a robot stores its repertoire of intonations, but it also stores other information such as probabilities, thresholds and reinforcement parameters. The robots have two distinct modules to store intonations in their memories: a *perceptual map* and a *motor map*.

The perceptual map stores information in terms of pitch contour (Fig. 2, Top), where: SH = start intonation in the higher register; SM = start intonation in the middle register; SL = start intonation in the lower register. Then, VLSU = very large step up; LSU = large step up; MSU = medium step up; SSU = small step up; RSB = remain at the same ban; SSD = small step down; MSD = medium step down; LSD = large step down; VLSD = very large step down. It is assumed that an intonation can start at three different voice registers: SL, SM and SH. Then, from this initial point $\{t(n), n=0\}$ the next pitch at $t(n+1)$ might jump, or step up or down, and so forth. It is important to note that pitch frequency values or labels for musical notes are not relevant here because the objective is to represent abstract melodic contours rather than a sequence of pitches (or musical notes) drawn from a specific tuning system. This is very important because one should not assume that the robots must sing in any pre-established musical scale. Rather, they should be

given the ability to establish their own tuning system collectively.

The motor map stores information in terms of three vectors of motor (vocal) parameters: lung pressure, width of the glottis (interarytenoid), and length and tension of the vocal chords (cricothyroid) (Fig. 2, Bottom).

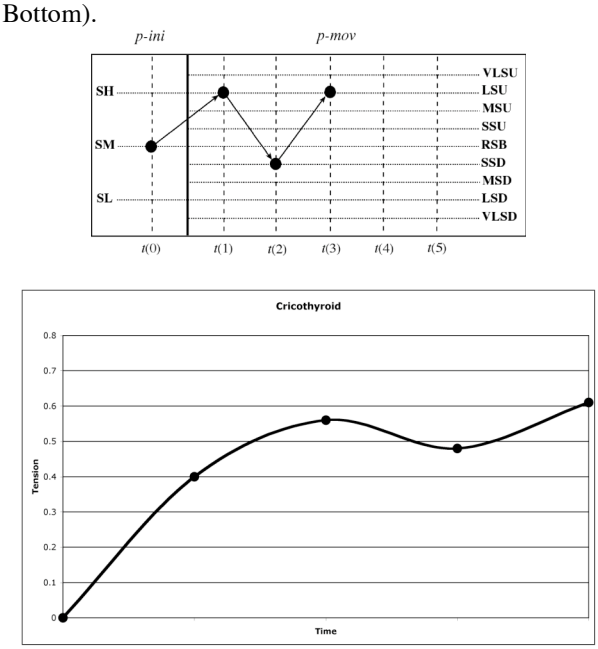

Figure 2 (Top) The representation of an intonation, where $t(n)$ indicates an ordered sequence of n pitches. (Bottom) The cricothyroid vector of the motor control map that produced this intonation. The other two vectors were omitted for the sake of clarity; see also Fig. 8.

2.2. The Interactions

The interaction algorithms were largely inspired by the work of Steels [11] and de Boer [2] on "evolutionary" language games. All robots have identical synthesis and listening apparatus. At each round, each of the robots in a pair plays one of two different roles: the *robot-player* and the *robot-imitator*. The main algorithms are given in detail in [8]. Glimpses at the functioning of these algorithms are given in Figs. 3, 4 and 5. The robots do not sing all at the same time; they interact in pairs. The robot-player starts the interaction by producing an intonation α, randomly chosen from its repertoire. The robot-imitator then analyses the intonation α, searches for a similar intonation Δ in its repertoire and produces it. Fig. 3 shows an example where the robot-player and the robot-imitator hold in their memories two intonations each. The robot-player picks the intonation α from its motor-repertoire and produces it. The robot-imitator hears the intonation α and builds a perceptual representation β of it. Then it picks from its own perceptual repertoire the intonation Δ that is most perceptually similar to the heard intonation β and produces it as an imitation. Next, the robot-player hears the imitation Δ and builds a perceptual representation ψ of it. Then it picks from its own perceptual repertoire the intonation ϕ that is most perceptually similar to the imitation ψ. If the robot-player finds another intonation ϕ that is closer to Δ than α is, then the imitation is seen as unsatisfactory, otherwise it is satisfactory. In Fig. 3, the robot-player babbles the original intonation α to itself and concludes that α and ϕ are different. Then, it sends a negative feedback to the robot-imitator.

When an imitation is unsatisfactory the robot-imitator has to choose between two potential courses of action. If it finds out that Δ is a weak intonation in its memory (because it has not received enough reinforcement in the past) then it will slightly move it away from α (by means of a deviation coefficient), as a measure to avoid repeating this mistake again. But if Δ is a strong intonation (due to a good past success rate), then the robot will leave Δ untouched (because it has been successfully used in previous imitations and a few other robots in the community also probably consider this intonation as being strong) and will create a new intonation λ similar to Δ to include it in its repertoire; that is, the robot produces a number of random intonations and then it picks the one that is perceptually most similar to Δ. Let us assume that in Fig. 3 the intonation Δ has a good past success rate. In this case, the robot-imitator leaves it untouched and creates a new intonation λ to include in its repertoire.

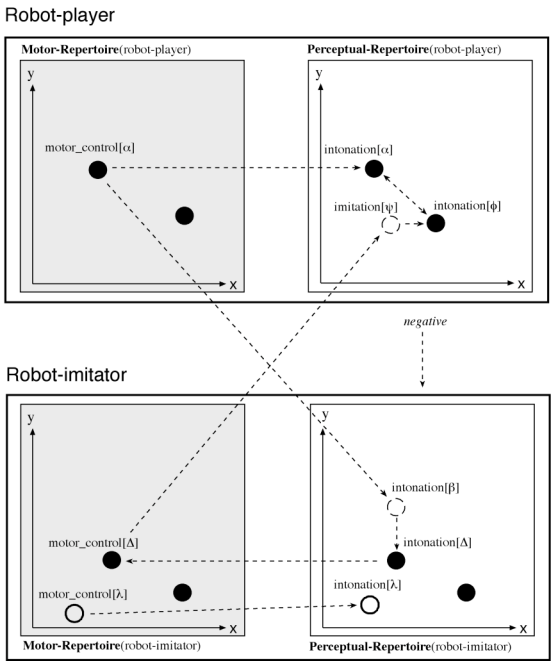

Figure 3 Example of an unsuccessful imitation. (Note: For didactic purposes, the co-ordinates of Figs. 3, 4 and 5 do not correspond to the actual parameters of the model. For the sake of clarity, the plotting is in an idealized two-dimensional representation of the motor and perceptual repertoires.)

Fig. 4 shows what would have happened if the intonation Δ did not have a good past success rate: in this case the robot-imitator would have moved Δ away from β slightly. Finally, Fig. 5 shows what would have happened if the robot-player had concluded that α and ϕ were the same, meaning that the imitation was

successful. In this case, the robot-imitator would have reinforced the existence of the intonation Δ in its memory and would have moved it slightly towards the representation of the heard intonation β. Before terminating the round, both robots perform final updates. Firstly they scan their repertoire and merge those intonations that are considered to be perceptibly close to each other; the merge function removes two intonations and creates a new one by averaging their values. Also, at the end of each round, both robots have a certain probability P_b of undertaking a spring-cleaning to get rid of weak intonations; those intonations that have not been sufficiently reinforced are forgotten. Finally, at the end of each round, the robot-imitator has a certain probability P_a of adding a new randomly created intonation to its repertoire; we refer to this coefficient as the "creativity coefficient". The signal feedback is implemented as follows: if feedback is positive, then the robot makes a couple up-and-down movements of its head. If negative, then it makes a couple of left-to-right movement of its head.

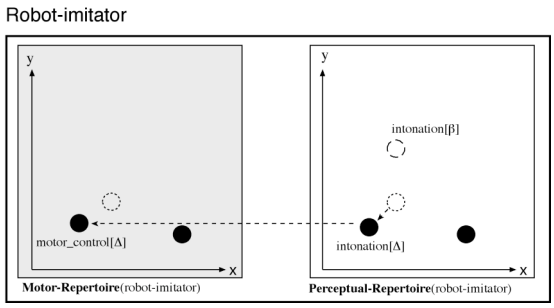

Figure 4 An example where the unsuccessful imitation involved an intonation that has a poor past success rate.

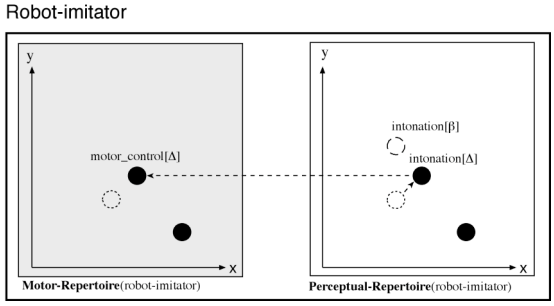

Figure 5 An example of a successful imitation.

3. THE BEHAVIOR OF THE MODEL

The graph in Fig. 6 shows a typical example of the development of the average repertoire of a group of 5 robots, with snapshots taken after every 100 interactions over a total of 5,000 interactions.

The robots developed repertoires averaging twelve intonations each. (Note that some may have developed more or less than twelve intonations.) After a drastic increase of the repertoire at about 800 interactions, the robots settled to an average of seven intonations each until about 2,200 interactions, when another slight increase took place. Then they settled to an average of nine intonations until about 3,800 interactions. From 3,800 interactions onwards the robots steadily increased their repertoires. The pressure to increase the repertoire is mostly due to the probability P_a of creating a new random intonation, combined with the rate of new inclusions due to unsatisfactory imitations. The effect of running the simulation with a larger group and for longer has been discussed in [8]. Essentially, the size of the repertoire tends to stop; it does not increase indefinitely.

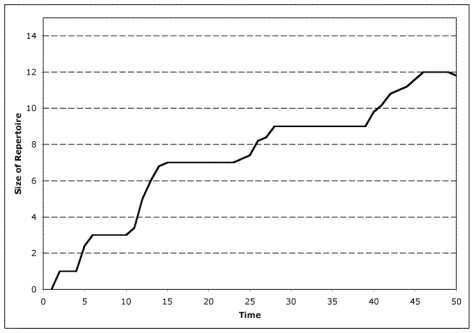

Figure 6 The evolution of the average size of the repertoire of intonations of the whole group of robots. In this case the group developed an average repertoire of 12 intonations. (The time axis is in terms number of interactions multiplied by 100.)

Fig. 7 portrays the perceptual memory of a robot randomly selected from the group after 5,000 interactions. In this case, the length of the intonations varied from three to six pitches. (The minimum and maximum length of the intonation to be evolved is fixed beforehand.) This robot developed eleven intonations; one below the average of the group.

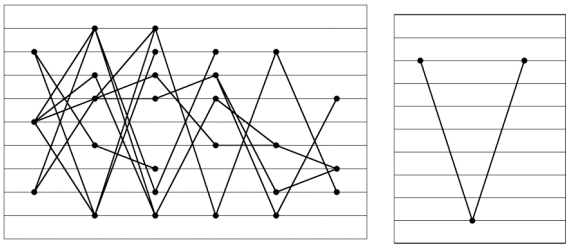

Figure 7 (Left) The perceptual memory of one robot and (Right) one of the perceptual patterns from, in this case lasting for 3 pitches. For the sake of clarity, the background metrics and labels of the perceptual representation are not shown. Please refer to Fig. 2(Top) for an explanation of the perceptual representation.

Although the repertoire size tends to increase with time, the imitation success rate tends to stay consistently high. However, this is highly dependent upon the number of robots in the group: the higher the number of robots, the deeper the fall of the success rate and the longer it takes to re-gain the 100% success rate stability [8].

An interesting feature of this model is that the robots do not necessarily have to develop the same motor representations for what is considered to be perceptibly identical. As an example, Fig. 8 shows the cricothyroid control vector developed by three different robots (Robot

1, Robot 2 and Robot 3) to represent what is essentially the same intonation shown in Fig. 7(Right).

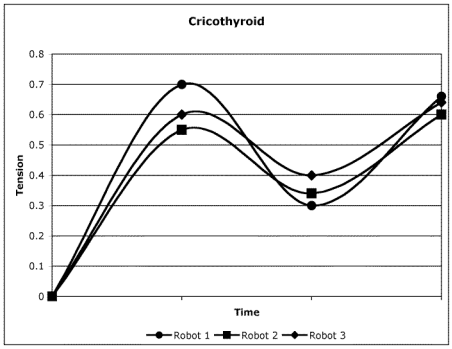

Figure 8. The corresponding motor control vectors developed by three different robots for the 3-pitch intonation shown in Fig. 7(Right): (Left) the lung pressure vector and (Right) the cricothyroid vector.

The imitation of an intonation pattern requires the activation of the right motor parameters in order to reproduce it. The robot-imitators assume that they always can recognise everything they hear because in order to produce an imitation a robot will use the motor vectors that best match its perception of the intonation in question. It is the robot-player who will assess the imitation.

4. CONCLUSION AND FURTHER WORK

At the core of our the system introduced in this paper is a selective mechanism inspired by Neo-Darwinian evolutionary theory [3, 4, 5], whereby some form of "mutation" takes place (e.g., intonations move closer to or away from other intonations in memory) and patterns are "born" (e.g., with random additions through the "creativity coefficient") and "die" (e.g., the spring-cleaning mechanism).

At the introduction we suggested that models such as the one presented in this paper have the potential to shed new insights into building interactive music systems. What sort of systems can be built informed by such models? We are aiming at the development of technology for implementing intelligent systems that can improvise music in real-time with human musicians. However, instead of manually programming these machines with prescribed rules for generating music, we aim at programming them with the ability of developing these rules autonomously, dynamically and interactively. This paper demonstrated one of the various intertwined algorithms that may lead to such capability.

A limiting aspect of the present system is that the robots only deal with short pitch sequences. The natural progression in this research is to furnish the robots with the ability to deal with longer pitch sequences, rhythm and other musical attributes. Although the symbolic sensory-motor-like memory mechanism developed for storing intonations served well the present model, it is not efficient for storing longer pitch sequences, let alone other musical attributes. In order to increase the complexity of the model, it is necessary to improve the memory mechanism, which would probably be more efficient by storing information about generating the sequences rather than the sequences themselves. We are looking into the possibility of doing this by means of algorithms for the evolution of grammars [9] and neural networks that mimic the behaviour of mirror neurons [12]. Also, we are currently considering ways in which to embed the robots with more sophisticated physiological and cognitive abilities.

At present, this model could run entirely as software. Indeed, we have run scaled up experiment by means of a simulation using software agents rather than robots (e.g., 20 agents for 40,000 interactions). Considering that each robotic interaction takes an average of 30 secs for intonations ranging from three to six pitches, a run with 40,000 interactions would take approximately 2 weeks to complete. Software simulation is desirable in such circumstances to fine-tune the model prior to running the robotic simulation. Eventually, we will have to move to more morphologically apt robotic platforms (e.g., with a mechanical vocal tract or robots with arms and hands to play instruments, etc.) so as to consider the role of embodiment and morphology in the development of behavior. But then, we will inevitably meet the problem of time for running scaled up experiments – the more embodiment we have the more difficult it is to simulate with software agents.

5. REFERENCES

[1] Boersma, P. (1993). "Articulatory synthesizers for the simulations of consonants", *Proc of Eurospeech'93* 1907-1910.

[2] de Boer, B. (2001). *The Origins of Vowel Systems*. Oxford: Oxford University Press.

[3] Fisher, R. A. (1930). *The Genetical Theory of Natural Selection*. Oxford: Clarendon Press.

[4] Haldane, J. B. S. (1932). *The Causes of Evolution*. London: Longman Green.

[5] Huxley, J. S. (1942). *Evolution: The Modern Synthesis*. London: Allen and Unwin.

[6] Miranda, E. R. (2002). *Computer Sound Design: Synthesis Techniques and Programming*. Oxford: Elsevier/Focal Press.

[7] Miranda, E. R. (2001). "Synthesising Prosody with Variable Resolution", *AES Convention Paper 5332*. New York: Audio Engineering Society.

[8] Miranda, E. R. (in press). "Emergent songs by social robots", *Journal of Experimental & theoretical Artificial Intelligence*. (Accepted for publication)

[9] Miranda, E. R., and Todd, P. (2007). "Computational Evolutionary Musicology", In E. R. Miranda and J. A. Biles (Eds.) *Evolutionary Computer Music*, pp. 218-249. London: Springer.

[10] Rizzolatti, G. and Craighero, L. (2004). "The mirror-neuron system", *Annual Review of Neuroscience* **27** 169-192.

[11] Steels, L. (1997). "The Origins of Syntax in Visually Grounded Robotic Agents", *Proc of International Joint Conference on Artificial Intelligence (IJCAI'97)*.

[12] Westerman, G. and Miranda, E. R. (2003). "Modelling the Development of Mirror Neurons for Auditory-Motor Intergration", *Journal of New Music Research* **31** 367-375.