



Internal Report 2010-13

September 2010

Universiteit Leiden

Opleiding Informatica

Articulatory Speech Synthesis
with Parallel Multi-Objective Genetic Algorithm

Francesco D'Este

MASTER'S THESIS

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Articulatory Speech Synthesis with Parallel Multi-Objective Genetic Algorithms

Francesco D'Este*, Dr. Erwin Bakker†

LIACS Medialab,
Leiden University.

Keywords: Articulatory Speech Synthesis, Vocal Tract Function Estimation, Evolutionary Parallel Algorithms

Abstract

Within the context of Articulatory Speech Synthesis, human speech is representable through a set of time-varying articulatory parameters (eg: constriction of oropharyngeal sections). The extraction of the configuration of articulatory parameters from speech sound is called Vocal Tract Area Function (VTAF) estimation. Vocal Tract Area Function Inversion is a well known problem in the field of speech synthesis and recognition, and it is generally accepted that a stable method to perform acoustical-to-articulatory mapping does not exist yet. Pioneering work by Atal showed that the problem is an inverse ill-posed problem. In this study we present a novel inversion method based on Multi Objective Evolutionary Algorithms: Lacov-NSGA-II. The system is an adaptation of NSGA-II for VTAF extraction. A number of modifications to the original NSGA-II algorithm are proposed, such as a problem-specific evolutionary operator design and a time-varying search space. Good results obtained from a serial prototype encouraged the development of a parallel version of the algorithm. Experimental tests showed good performance with respect to the intelligibility of the re-synthesised speech signal as well as the objective error measurements.

1 Introduction

Human speech is representable through a set of time-varying articulatory parameters (eg: tongue, lips, etc). The extraction of the configuration of articulatory parameters from speech sound is called Vocal Tract Area Function (VTAF) Estimation. Vocal Tract Area Function Estimation is a well known problem in the field of speech synthesis and recognition, and it is generally accepted that a stable method to perform acoustical-to-articulatory mapping does not exist yet. Moreover, it is often needed in many applications, such as speech recognition, speech synthesis, speech compression, language training for deaf people, etc. Several techniques for VTAF estimation have been suggested in the literature. Early approaches introduced a number of constraints, which were mainly combinations of temporal (dynamic) and morphological (spatial) restrictions. Other approaches tried to relate acoustic information such as formant frequencies and acoustic impedance at the lips to the VTAF. Few studies suggested the use of codebook-based methods, where the estimation is started with a feature

lookup in a pre-built postural database. An important drawback is that the codebook-based approaches need the previous construction of extensive databases of postures and thus can only focus on few particular sound classes. It is also clear that this approach can only be applied to a specific synthesis model under study.

More recent approaches tend to be more general with respect to the synthesis method and often make use of optimisation techniques, such as Genetic Algorithms and Particle Swarm Optimisation. These methods often use a single objective value to be minimised, computed as the weighted sum over a feature vector describing the differences between the target and the actual solution. This approach can lead to good results, however it is often hard to tune the weight of each objective feature. Moreover, experimentation with a single objective value within this study showed a noticeable discontinuous behaviour in the parameter space. For this reason we propose a general multi-objective optimisation method that can be applied to newer and more complex physical vocal tract emulations. Our method consists of an adapted version

*deste.francesco@gmail.com

†erwin@liacs.nl

of the evolutionary meta-heuristic NSGA-II, called Lacov-NSGA-II. VTAF estimation is performed in an evolutionary fashion, having to match a given speech sample by iteratively evolving a population of partial solutions.

The Vocal Tract Model used in our study is the Gnuspeech Tube Resonance Model (TRM), a synthesiser which emulates the resonant behaviour of the vocal tract, physically simulating it as a chain of tubular waveguides. The glottal source is modeled as a sine wave inserted inside the first tube. The model features a cascade of 8 tubes to emulate the oropharyngeal cavity and 5 to model the nasal cavity.

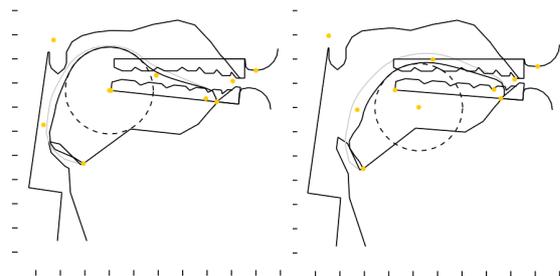


Figure 1: Section of human postures used to pronounce the /a/(left) and /e/(right) phonemes, rendered with VocalTract-Lab (courtesy of Peter Birkholz)[17]

1.1 Outline

This paper is organised in 8 sections. Section 2 is an overview upon Articulatory Speech Synthesis and the Tube Resonance Model. Some pioneering VFAT estimation techniques from the literature are reported in Section 3. Section 4 describes NSGA-II, while sections 5 and 6 report the needed algorithm adaptation and its parallelisation. Experimental results are reported and discussed in Section 7. Conclusions are listed in Section 8.

2 Articulatory Speech Synthesis

Articulatory synthesis consists on a class of techniques for synthesising speech based on physical models of the human vocal tract and the articulation processing occurring there. In the following sections we will describe the Tube Resonance Model, which is the vocal tract emulation software we used in our study.. In order to have a computationally feasible synthesis model, this model features several simplifications. These assumptions include: glottal source approximation to a sine wavetable oscillator, frequency-independent energy loss and the bi-dimensional modelling of the travelling waves system, decoupling of glottal and pharyngeal dynamics. These approximations rule out the possibility to model few phenomena happening in the vocal tract, such as the air vor-

texes, the non-linear coupling of the pressure at the lips and the glottal pulse and the effects of the jaw tilt.

2.1 The Tube Resonance Model

In this section we will describe the Tube Resonance Model (TRM)[8], an articulatory speech synthesiser distributed within the TTS system Gnuspeech. The resonant behaviour of the oropharyngeal and nasal tract is emulated using digital waveguides. The vocal tract is divided into 8 regions (tubes) of unequal length, where the particular regions correspond to the human articulations of tongue, teeth and mouth, The cross-sectional area of each region can be varied independently over time. The difference between the cross-sectional areas of subsequent tubes gives raise to differences in acoustic impedance of the medium. This phenomena is modelled using two-way scattering junctions. The nasal cavity is formed by 5 equal-length tubes, and it is connected to the vocal tract through a particular tube (velum). This connection is modeled using a single three-way scattering junction. In the next few sections we will cover the basics of acoustic tubes physic dynamics.

2.1.1 Simulating an Uniform Tube

Fluid motion in a rigid tube with uniform cross-sectional area can be approximated as primarily parallel to the waveguide parallel axis. This means that sound pressure waves are assumed to travel through the medium as a one-dimensional longitudinal plane waves (as the model in the previous section). A delay line like the one in Figure 2¹ can be used to simulate travelling waves in the digital domain, being a system which samples both in time and space: each delay unit stores the instantaneous pressure for the corresponding section of the tube. At each sample increment, the wave values are shifted to the right into the next delay unit.

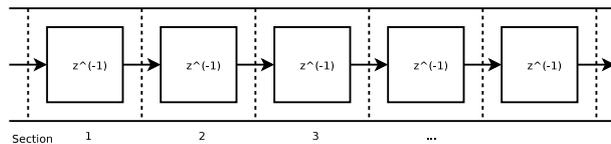


Figure 2: Simple Delay Line

Sound waves can travel through the tube in both directions at the same time. The instantaneous pressure at any delay unit is then the sum of the left-to-right and the right-to-left traveling waves. This superposition of waves results in constructive and destructive interference through all the delay units, yielding particular resonance in the spectra to be enhanced or suppressed. Superposition can be simulated using a

¹In the context of Digital Signal Processing, z^{-n} denotes a Delay line with a delay of L samples. A unit delay (a delay of one sample) is therefore expressed as z^{-1} .

bi-directional delay line (called also *waveguide*), as shown in Figure 3. In this structure there are two lines of delay units, one simulating the left-to-right and the second the right-to-left traveling wave. The instantaneous sound pressure at each tube section can be easily derived as the sum of the bottom and the below delay units.

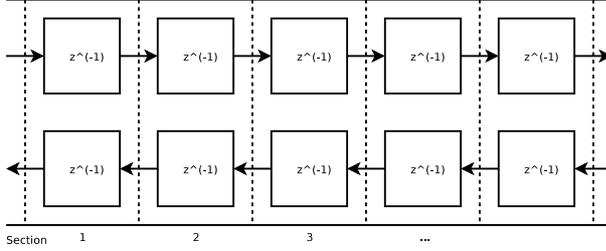


Figure 3: Bi-directional Delay Line

When the left-to-right traveling wave reaches the end of the tube, it is reflected back inverted into the right-to-left delay line. In the digital domain, this is performed simply multiplying the value in the rightmost delay unit by -1 and feeding this value into the right-to-left delay line. A traveling wave reaching a closed end of a tube is reflected back into the tube in phase, without inversion. This simulates the reflection of the traveling wave at the glottis site, as shown in Figure 4.

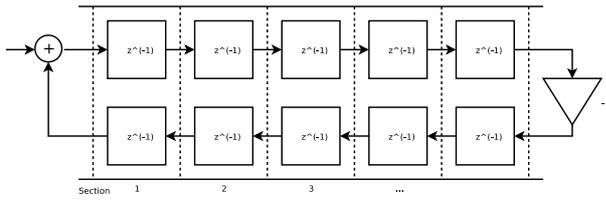


Figure 4: Bi-Directional Delay Line with reflections occurring at boundaries

In the real vocal tract energy loss takes place as viscosity and thermal conduction of the air. Approximating this behaviour, ignoring any frequency-dependent effect of the phenomenon, can be modeled multiplying the pressure waves in each delay unit by a loss factor before shifting it to the next unit. Normally this factor is a value less than 1. The addition of the energy loss to the tube model is shown in Figure 5.

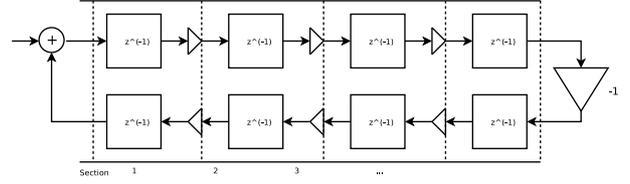


Figure 5: Bi-Directional Delay Line with reflections at boundaries and energy loss on junctions

2.1.2 Modeling Non-Uniform Tubes

To approximate the behaviour of wave traveling through a smooth-varying non-uniform tube, we can simulate it as a series of equal-length cylindrical sections, each being the average cross-sectional area of the area of the corresponding real vocal tract part. Each tube in the non-uniform tube gives rise to a particular acoustic impedance. If two subsequent tubes have different impedance, part of the pressure of the traveling wave is reflected and part is transmitted at the junction of the two sections. To model this behaviour, we can make use of the two-way *scattering junctions*, as shown in Figure x. The scattering coefficient k_m is calculated with the formula:

$$k_m = \frac{Z_{m+1} - Z_m}{Z_{m+1} + Z_m} \quad (1)$$

where Z_m is the impedance of section m . Note that if two subsequent tubes have the same impedance Z_k , then no reflection occurs in the scattering junction. Since $Z_m = \rho v / S_m$, where S_m is the cross-sectional area of the tube section m , and since air density ρ and speed of sound V are the same in both directions, we can recast the above formula as:

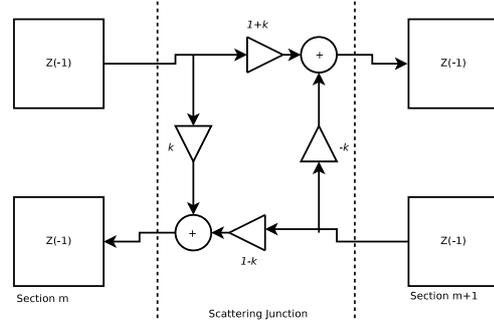


Figure 6: 2-way Scattering Junction

$$k_m = \frac{S_{m+1} - S_m}{S_{m+1} + S_m} \quad (2)$$

Which can be further re-casted as a function of section radii, since $S = \pi r^2$

$$k_m = \frac{r_{m+1}^2 - r_m^2}{r_{m+1} + r_m} \quad (3)$$

Note that if section $m + 1$ is closed, $k_{m+1} = 1$ which means that the traveling wave is completely reflected back.

2.1.3 N-Way Junctions

The nasal and oropharyngeal tract are connected through the velum. This nasal cavity is modeled with a separate tube resonance model, which remains fixed into a steady-state posture. The velum is modeled through a three-way *scattering junction*.

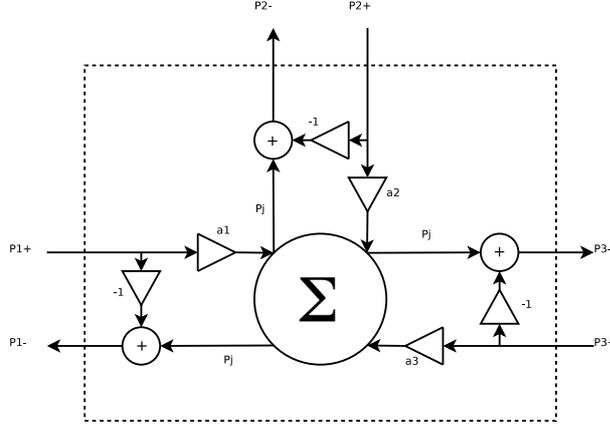


Figure 7: 3-way Scattering Junction

N-Way junctions are a generalisation of the two-way scattering junction. As before, the relative impedance of the connecting tubes determine the reflection characteristics of the junction which are a function of the relative cross-sectional areas of the sections of the tubes adjacent to the junction. The scattering coefficient a_i for the tube i can be computed with the following formula:

$$a_i = 2 \frac{\Gamma_i}{\sum_{j=1}^N \Gamma_j} \quad (4)$$

where we define $\Gamma_i = 1/Z_i$ as the admittance of tube i . We can express the formula as a function of the cross-sectional areas S_i as:

$$a_i = 2 \frac{S_i}{\sum_{j=1}^N S_j} \quad (5)$$

or in term of tube radii r_i as:

$$a_i = 2 \frac{r_i^2}{\sum_{j=1}^N r_j^2} \quad (6)$$

We can then derive the total pressure P_j of the junction with

$$P_j = \sum_{i=1}^N a_i P_i^+ \quad (7)$$

where P_i^+ is the incoming pressure for tube i , while each outgoing pressure wave P_i^- can be computed by subtracting the incoming pressure from the junction pressure:

$$P_i^- = P_j - P_i^+ \quad (8)$$

Equation 8 can be used to derive the air pressure at the lips.

3 VTAF Estimation

In this section we will review existing approaches for VTAF estimation and we will introduce our novel approach. We have partitioned all existing methods in 3 classes of algorithms: Codebook-based, Analytical and Optimization-based.

3.1 Codebook - based approaches

Codebook-based approaches require the pre-computation of acoustic features of a subset of the postural space, as well as appropriate routines to uniformly spread and match the probes. These methods are usually bound to a particular subset of sounds, and the voice synthesis model must be well defined before the codebook generation. From the literature it emerges that among all the approaches, these methods seem to be advantageous with respect to accuracy. However, it is clear that these methods lack in flexibility, because the vocal tract shapes that can be recovered are strictly related to the codebook. This data is usually limited to few speakers or to a particular vocal tract model, thus adequate mapping exists only for few classes of voice sounds, like vowels or simple consonant-vowel transitions.

3.2 Analytical and Statistical approaches

Analytical methods usually make use of a mathematical inversion of the physical dynamics of the articulatory model. However, being the problem ill-posed it is often very hard to derive a closed and computationally feasible form of these dynamics. Moreover, these methods are clearly bound to a specific synthesis model. Statistical methods have been suggested in the literature as well. Dusan et al introduced a method to derive the VTAF using the Maeda's Statics Articulatory Model, built by statistical analysis of X-ray films of a French female speaker. To estimate the dynamical model parameter, they used the Expectation-Minimisation (EM) algorithm for Maximum Likelihood estimation of model parameters. Mokhtari et al. suggested a method to derive a good compression of the parameter space using Principal Component Analysis on Japanese MRI and audio data. The first two principal components explained covariation in vocal-tract shape and length accounting for 96% of the total variance. Multiple linear regression models were then evaluated for their accuracy in reconstructing the area functions of the dynamic utterance, using either carefully measured formants or cepstral coefficients defined in various frequency bands.

```

while Termination Condition do
     $R_t = P_t \cup Q_t$ ;
     $F = \text{non-dominated-sort}(R_t)$ ;
     $P_{t+1} = \emptyset$  and  $i = 1$ ;
    repeat
        crowding-distance-assignment( $F_i$ );
         $P_{t+1} = P_{t+1} \cup F_i$ ;
         $i = i + 1$ ;
    until  $|P_{t+1}| + |F_i| \leq N$ ;
    Sort( $P_{t+1}, \prec_n$ );
     $P_{t+1} = P_{t+1} \cup F_i[1 : (N - |P_{t+1}|)]$ ;
     $Q_{t+1} = \text{make-new-pop}(P_{t+1})$ ;
     $t = t + 1$ ;
end

```

Algorithm 1: NSGA-II main loop

3.3 Optimisation - based approaches

Sankaran et al. introduce a VT Inversion by Analysis-by-Synthesis, which was an inversion model based on convex optimisation. They mapped acoustic and geometric continuity constraints to a cost function used during the optimisation. To speed up the process, they used a big database for acoustic-to-articulatory mapping, organised with a bin structure in the formant space. A drawback of this approach is that the cost function might be non-convex and could exhibit and the landscape might show high discontinuity. In this terms the method does not seem to be generalisable with respect to the synthesis method. Other recent approaches using evolutionary computing have been proposed in [7]. Mahmoud suggests the estimation of the VTAF using particle swarm optimisation (PSO) techniques.

4 NSGA-II

We propose a method based on NSGA-II, which is a multi-objective evolutionary optimisation algorithm (MOEA). These class of algorithm showed better results and more robustness compared to single-objective evolutionary algorithms during our early experimentation. Within this approach, partial solutions are evolved through an iterative use of mutation, evaluation and selection routines. A solution is a vector describing the degree of opening of each tube of the Tube Resonant Model, as well as its glottal source frequency. The multi objective cost function is a vector that measures feature differences (eg: MFCC, zero cross, displacement of the formant frequencies, etc.) between the target speech sample and the actual evaluation. It should be noticed that the concept of MOEA induces the existence of a set of solutions rather than a single pseudo-optimal solution. The solution to a MO problem can be indeed defined as a Pareto set. Pareto optimality is a concept that formalises the trade-off between a given set of mutually contradicting objec-

tives: a solution is Pareto optimal when it is not possible to improve one objective without deteriorating at least one of the other. A set of Pareto optimal solutions constitute the Pareto front.

NSGA-II iteratively approximates the extraction of the first Pareto front using a fast non-domination sorting function. The function has complexity $O(MN^2)$, where M is the number of objectives while N is the population cardinality. The non-dominance sorting induces a partitioning of the population into subsets called **fronts**. Each solution in front F_i dominates all the solutions in the subsequent fronts $F_{i-1} \dots F_{i-n}$. We say that if solution $\mathbf{x} \in F_i$, then it has rank i . NSGA-II defines a Crowded Comparison operator \prec_n which guides the selection process at various stages of the algorithm towards a uniformly spread out Pareto-optimal front. The operator uses the rank and a density estimate to define a partial order between the solutions. Between two solutions with differing non-domination ranks, it selects the solution with lower rank. Otherwise, if both solutions belong to the same front, it chooses the solution which is located in a region with lesser number of solutions. Initially, a random parent population P_0 is generated. Binary tournament, selection, recombination and mutation operators are used to generate a new population Q_0 of size N . From the first generation onward, the procedure is different. Algorithm 1 shows the NSGA-II main routine. Initially, the algorithm performs the union of parent P_t and child Q_t populations into a mixed population R_t of size $2N$. Then the population R_t is sorted according to non-domination. A new parent population P_{t+1} is formed by adding the first front of R_t until the size exceeds N . After that, the solutions from the last accepted front are sorted according to \prec_n and the first N points are picked. This results in the new population P_{t+1} of size N . Solutions in P_{t+1} are then used to generate a new child population Q_{t+1} of size N , through selection, crossover and mutation.

The Termination Condition can be set as needed, i.e. a maximum number of iterations.

4.1 Mathematical Formulation

The problem of estimating the VTAF can mathematically defined as a multi-objective optimisation problem with linear constraints.

$$\min_x [\mu_1(x), \mu_2(x), \dots, \mu_n(x)]^T$$

subject to:

- $g(x) \leq 0$,
- $h(x) = 0$
- $x_l < x < x_u$

where u_i is the i^{th} objective cost function, g and h are respectively the inequality and equality constraints, and \mathbf{x} is the vector of optimisation variables. This is the vector describing the configuration of the vocal tract, by means of 8 real values that regulate the degree of opening of each tube.

The equality constraints consist of parameters that are maintained fixed in our research. These include fundamental frequency, frication parameters, parameters modelling the nasal cavity, etc. Articulatory constraints (minimal and maximal degree of opening of a single tube) are included by means of fixed inequality constraints. The extraction of the best VTAF from the first Pareto front is performed finding the solution which minimises the normalised sum of all the cost functions.

5 Lacov-NSGA-II

A number of modifications to NSGA-II were needed to obtain a robust and stable method. The resulting algorithm was called Lacov-NSGA-II. In Lacov-NSGA-II, we introduced new features to the search algorithm by using a dynamic search space strategy and specialised Mutation Operators.

5.1 Dynamic Search Space

GA algorithms are usually used to optimise parameters of a model which is static during the search process, which means that the cardinality of the parameters that the heuristic can tune does not change in time. We introduce a dynamic modelling of the problem space: the heuristic starts with almost all tubes grouped together as they were a single one. A special operator is designed to 'split' the tubes, enabling independent tuning. We define the *model topology*, which is a boolean vector \mathbf{V} with length n , where $v_i = 1$ if the i^{th} tube cross-sectional area is controllable and n is the tube cardinality. Otherwise, $v_i = 0$. This means that only the tubes marked as 1 can be adjusted by the search algorithm. It is important to notice that each individual inherits its own topological information, letting independent 'topological species' being present

in the population. From this perspective, the evolutionary process imposes a competition between individuals but also a competition between topologies.

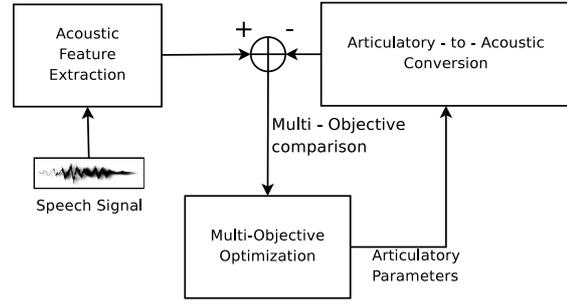


Figure 8: Inversion System Scheme

5.2 Operators

The Mutation Operators had to be modified in order to work with the parameter grouping topology feature. Also, some problem-specific knowledge was inserted into the algorithm, by means of new mutation operators. The following sections describe these Operators.

5.2.1 Mutation Operator

The mutation Operator performs a perturbation to the actual solution. Magnitude and direction of the perturbation are extracted from a gaussian distribution. Magnitude is then multiplied with the inverse generation number to achieve a cooling down-effect, similar to the one used in Simulated Annealing techniques. The mutation operator takes the model topology into account. First, it generates a random position i and checks if v_i is tunable. If v_i is tunable, it performs the same mutation to all the parameters that are bound to the i^{th} tube, otherwise the routine is restarted until a usable tube is found.

5.2.2 Splitting Operator

The Splitting Operator induces the growth of the search space for each solution. The Splitting operator randomly selects a candidate, generates a random position between tubes 1-8 and performs the splitting marking as 1 the correspondent array cell in the topology information. With respect with the previous definitions, this simply means: (a) Choose a random tube i and (b) Set $v_i = 1$.

5.2.3 Horizontal/Vertical Shifts

We experimentally ascertained that similar acoustic feature values could be obtained within vertically or horizontally shifted isomorphic postures. The Vertical Shift operator performs a random shifting of the whole posture, by adding to each of the TRM tube cross-sectional radius a value extracted from the Mutation Operator. The Horizontal Shift operator is similar to the Vertical Shift Operator, but the shift operation is performed horizontally rather than vertically. The first and the last tube are swapped.

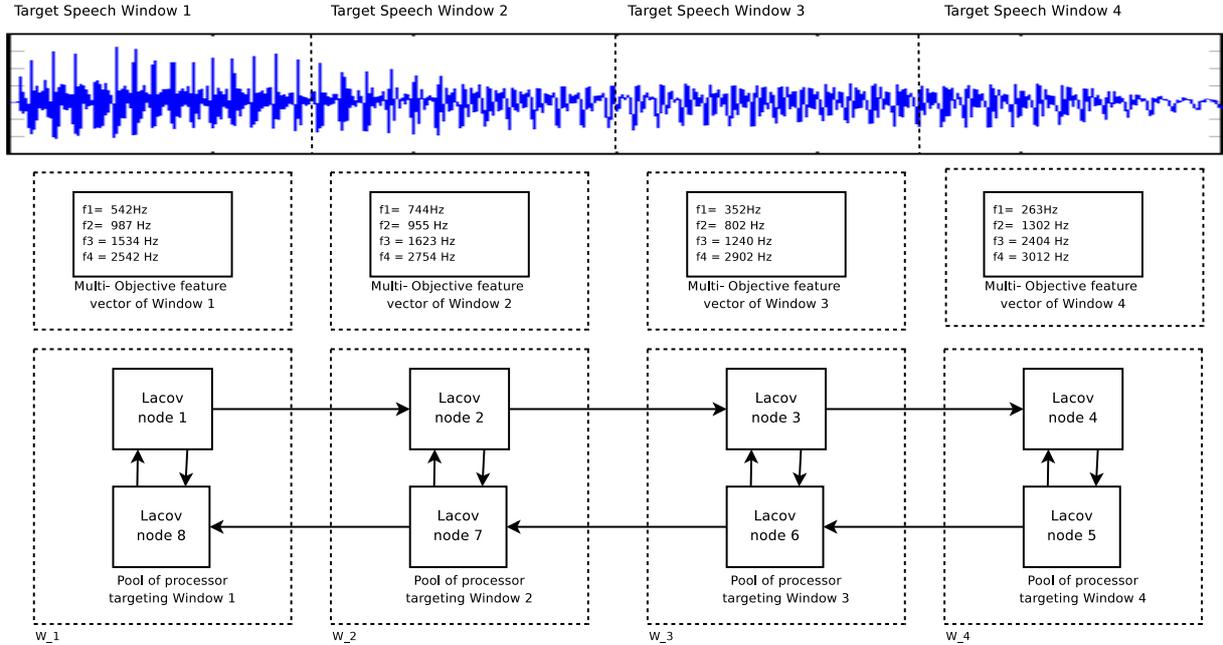


Figure 9: Parallel version of Lacov instance with 8 workers and 1 master node (not shown). Arrows denote Deme migration.

5.2.4 Sensitivity Operator

Sensitivity operator is used to relate the movement of the controllable tubes to the objective improvement of each of the first four formant frequency displacements. The operator calculates an approximation of the partial derivative of the objective function, using the following method. First, all the tubes are selected and for each pivot tube, a new posture is generated. Each posture is obtained adding to the i_{th} parameter (and all the associated bound tubes) a constant value δ (0.02 cm). All the new postures that do not violate the postural constraints are evaluated. Finally, the gesture maximising the improvement is maintained in the population.

6 Parallel version

A serial version of Lacov-NSGA-II was implemented using MATLAB, showing promising results. However, the average convergence time of the the prototype was too long to properly evaluate the method. This was mainly caused by the high computational cost of the articulatory synthesis and formant extraction routines. For this reason, we decided to develop a parallel version of the algorithm. The whole code has been ported to C and the Gnspeech TRM source was merged with the project. The implementation makes use of blocking calls from Open MPI 1.2.1 to handle message pass and synchronisation. The peculiarity of this implementation is that it is designed to invert gestures rather than postures. With gesture approximation we mean that the estimation is performed on longer speech samples, comprising a number of steady-state vocal tract configurations and thus describing an articulatory 'scene'. Instead of using a sin-

gle multi-objective vector, representing the average formant frequencies for all the speech signal, here we are using multiple vectors to estimate the target posture in consecutive windows of samples. Our parallelisation method makes use of static subpopulations with migration. This fashion requires the partitioning of the population into some number of demes (subpopulations). Each deme is assigned to a single node (*geographic isolation*) and individuals compete within it. All the nodes are partitioned into n disjoint sets W_1, W_2, \dots, W_n , where n is the window number. Each subset W_i is assigned with a different multi-dimensional objective O_i , which represents the average formant frequencies of the i_{th} window of the target speech sample. Figure 9 shows a possible allocation of 8 nodes into 4 different target windows as an example. Within this implementation, we introduced a new operator called *migrator*. Every n eons, the best m individuals are copied from one deme to another. We have also adopted a *stepping stone model*, which implies that individual can only migrate within neighbouring demes.[3]. Nodes inside the same subset share the best n solutions every p generations, replacing the worst n individuals. Since speech signals (and consequently articulatory informations) are known to show locally stable spectral distributions, subsequent subsets of nodes share the best individuals as well. The system uses a master node to keep track of the result and synchronise the worker pool. The system is scalable with respect to the node number as well as the tube number.

7 Experimental Results

Table 1: Average postural error (in cm.) against pool size. k

k	tube 1	tube 2	tube 3	tube 4	tube 5	tube 6	tube 7	tube 8
2	0.29	0.31	0.37	0.43	0.58	0.52	0.37	0.56
4	0.40	0.43	0.33	0.29	0.49	0.39	0.38	0.48
8	0.32	0.32	0.34	0.29	0.45	0.42	0.37	0.44
16	0.19	0.14	0.13	0.13	0.26	0.24	0.25	0.30
24	0.14	0.09	0.13	0.12	0.20	0.17	0.17	0.22

Table 2: Average error standard deviation (in cm.) against pool size. k

k	tube 1	tube 2	tube 3	tube 4	tube 5	tube 6	tube 7	tube 8
2	0.30	0.27	0.30	0.29	0.29	0.29	0.26	0.37
4	0.31	0.41	0.31	0.30	0.34	0.29	0.28	0.36
8	0.21	0.18	0.28	0.29	0.34	0.31	0.30	0.36
16	0.14	0.11	0.12	0.14	0.37	0.32	0.28	0.29
24	0.10	0.09	0.12	0.09	0.19	0.16	0.12	0.12

Table 3: Average spectral error(left) and average spectral standard deviation(right) of the spectral error against Pool Size in Hertz

k	F1	F2	F3	F4		F1	F2	F3	F4
2	68.3472	71.5417	30.0997	68.7128	-	76.5276	49.9690	25.2459	62.7324
4	36.7394	33.3690	22.7438	29.3055	-	33.6966	35.0231	20.9827	43.8573
8	23.4373	11.7881	15.0395	14.4680	-	24.2506	15.1946	14.4916	21.6443
16	12.9114	17.7670	12.1469	24.5036	-	15.7446	22.9038	13.6048	30.2544
24	6.2474	8.0370	5.7927	10.3057	-	11.3987	7.8609	8.2223	14.03

In this section we discuss the results obtained with the experimental tests. We focused exclusively in vowel sounds, cutting away the possibility to generate plosives, fricatives and sibilants. Consonants are indeed more difficult to model, being often articulated with complete or partial closure of the vocal tract. Articulatory modelling of consonants generally require the addition of a noise source somewhere inside the tube model, and the subsequent need to control its position, bandwidth and band pass frequency. We will address the tuning of these parameters in further studies. The evaluation of the method has been targeted within 2 testing setups. The first consisted in 100 executions targeting randomly-generated target speech sounds. The objective speech was synthesised with the Gnu-speech TRM using an own steady-state-posture generator. The random posture generator randomly chooses 3 independent real values describing the constriction

of tubes 1 - 4 - 8 (corresponding to glottis, palatal cavity and lips). The target VTAF is then built using spline interpolation applied to the above points. With this experiment we measured the objective articulatory and formant-displacement error of each best solution. Since in this test the posture to be inverted was steady-state, we used a single inversion window. Parallel runs of the algorithm were deployed with five different pool sizes (2,4,8,16 and 24 nodes respectively) at DAS-3². The multi-objective function measured the squared absolute displacement of the first 4 average formant frequencies between the target and the evaluated solution. In the second experiment we analysed the behaviour of the method with longer speech samples such as diphones and tri-phones. The target samples were synthesised using the AT&T TTS³, available online. The used speech sounds are listed in Table 5. The difficulty of this

²<http://www.cs.vu.nl/das3/>

³<http://www2.research.att.com/ttsweb/tts/demo.php>

Table 4: Normalized Average formant displacement (in Hz) error and confusion matrix

	F1	F2	F3	F4	aiθ	ei	iθ	θaie	uau	uθi
aiθ	9.44	4.80	8.63	1.43	100%	0.0%	0.0%	0.0%	0.0%	0.0%
ei	7.59	12.85	1.50	4.50	0.0%	83.3%	0.0%	0.0%	0.0%	16.7%
iθ	7.22	8.20	1.59	1.93	0.0%	0.0%	83.3%	0.0%	0.0%	16.7%
θaie	4.20	5.75	6.75	2.85	0.0%	0.0%	0.0%	100%	0.0%	0.0%
uau	4.08	9.76	3.35	1.15	0.0%	0.0%	0.0%	0.0%	83.3%	16.7%
uθi	8.27	5.95	2.08	1.13	0.0%	0.0%	50.0%	0.0%	0.0%	50.0%

test consists in the fact that the target and the partial solutions are generated using different speech synthesis algorithms (cross-synthesis inversion). Therefore, an exact solution is not likely to exist. In the second experiment, each speech sound has been inverted 5 times and the best objective was picked up. In this test we set the window number to 4 or 6 (depending on the target length). The inversion was performed on each of the 8 tubes of the Vocal Tract model as well as the Glottal Source pitch. A minimum of 2mm and a maximum of 2cm were set as articulatory constraints. These values were set according to the typical human cross-sectional radii of the oropharyngeal cavity. Since in the second test we performed cross-synthesis inversion, we set the glottal pitch to have a wider range of feasible values (min:~120Hz,max:~220Hz), while we used a narrowed one for the second test (min:~180Hz,max:~200Hz). All the other parameters of the model (such as configuration of the nasal cavity, frication parameters, etc) have been maintained fixed. The Mutation Operator generator has been set to match the statistical distribution $\mathcal{N} = (0, 0.2)$. Tube ratios are internally represented as floating point variables. A threshold of maximum 90 generations was set as well. Population size was set to 40. The initial tube topology V was set randomly (having $v_i = 1$ with probability 0.3 and $v_i = v_{i-1}$ otherwise. v_1 is always 1). Operators probabilities have been set as follows. (a) P(Mutation)= 0.3 (b) P(Vertical Shift)= 0.1 (c) P(Horizontal Shift)= 0.1 (d) P(Splitting)= 0.2 (e) P(Sensitivity)= 0.3. The best objective results were mixed in an auditory test file. As a side test, and in order to determine the intelligibility of the re-synthesised speech, we asked 6 people to listen to the test file and choose the order of the perceived phonemes from a list.

7.1 Discussion

The next tables report the results of our tests. Table 1 and 2 respectively report the average postural error for each tube and the relative standard deviation, both grouped by pool size k . Table 3 shows a measure of the spectral error, reporting the average frequency displacement of the first 4 formant, as well as the relative standard deviation.

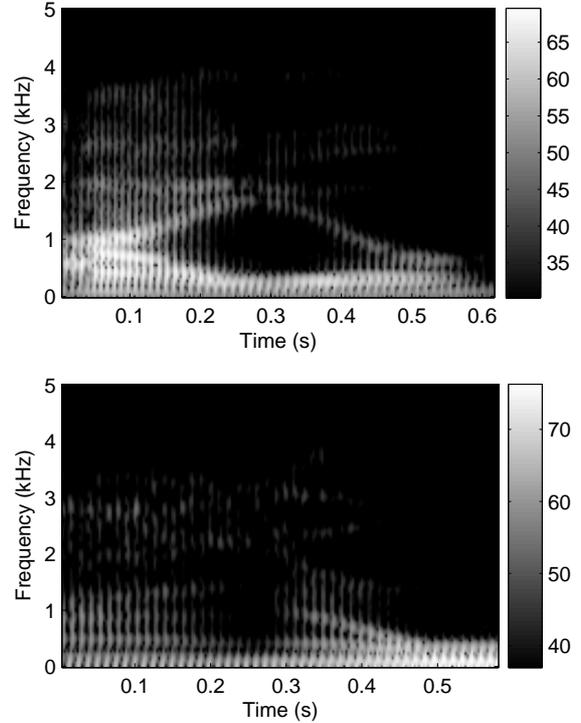


Figure 10: Spectra of the original (above) and the reconstructed speech for 'aio'

We can notice that the system average postural error is around 0.4 with pool size 2, while gets close to 0.1 with higher pool sizes (16 / 24).. As reported in [6], any speech formant configuration can be produced by an infinite number of postures. From this perspective, the little postural mean error is to be considered as a promising result. The observation is reinforced by the results in Table 4, displaying the first 4 formant average displacements and their relative standard deviation. We can observe that the algorithm generally produces VTAF that resembles the target sound from the spectral distribution standpoint.

Figure 11 shows the average convergence time per target smoothness and node number. The stop criterion we set was a total formant error below 50 Hz. If the criterion was not reached, we assigned to the experiment the maximum time allowed for each run (900 seconds). The average relative speedup to the 2-node run is always sublinear, being 1.3805 for 4 processors, 2.2850 for 8, 2.560 for 16 and 7.0011 for

24. The low speedup values between 8 and 24 nodes are mainly caused by the frequent communication between the worker pool and the master node. Indeed, in order to keep track of all the data generated in the experiments, we have set frequent updates (once for each generation) between the worker pool and the master node. Better speedup performance could be obtained with rarer updates.

Table 4 (left) shows the formant error for the second test normalised by the window number, as well as the confusion matrix obtained with the subjective auditory test (right). We can notice that the average formant error spans circa from 1 to 9 Hz. From this observation we can conclude that the system was able to give a good approximation of the target sound even if it was synthesised using a different method. Also the results of the intelligibility tests are shown in Table 5 (right). Perfect matches are observable for aiθ and θaie, while almost a perfect match is observable with eθ, iθ and uau. uθi is the sound that was most commonly mis-classified (as iθ). From this perspective, the system showed good intelligibility for the re-synthesised speech.

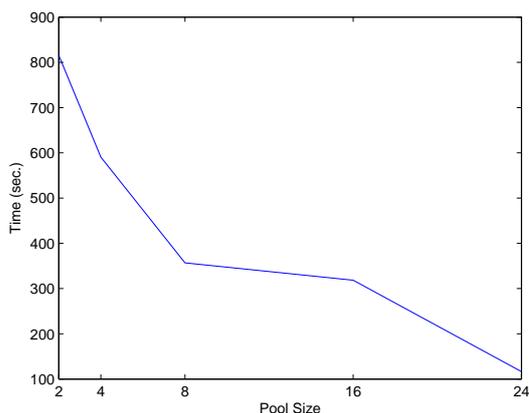


Figure 11: Average Convergence time against Pool Size.

8 Conclusions

We presented a novel approach based on multi-objective evolutionary optimisation to the problem of finding a close-to-natural area function of the vocal tract from the speech signal. Objective and subjective tests showed good results. However, the experiments are preliminary and further studies should be conducted to enhance the performance of the system. This could include: code optimisation and a method for automatic glottal fundamental frequency estimation. The generality of our inversion system was assessed using an own steady-state posture generator and inverting target sounds which were synthesised using a commercial TTS system. Further developments of Lacov-NSGA-II could include: the replace-

ment of the Gnuspeech TRM with newer and more complex articulatory speech synthesisers and the use of a posture database to have better fitting initial populations. The system should be also be extended in order to be able to invert consonant sounds. Finally, further studies should investigate the system behaviour when used as speech signal compressor.

References

- [1] Kalyanmoy Deb et Al *A Fast and Elitist Multi Objective Genetic Algorithm: NSGA-II*. Kanpur Genetic Algorithms Laboratory, 2006
- [2] A.E.Eiben, J.E.Smith *Introduction to Evolutionary Computing* Springer 2008.
- [3] Mariu Nowostawsku, Riccardo Poli *Parallel Genetic Algorithms Taxonomy* Submitted for publication to KES'99.
- [4] David Hill, Leonard Manzara, Craig Schock *Real-time articulatory speech-synthesis-by-rules*, in Proceedings of AVIOS, 1995
- [5] Sorin Dusan, Li Deng *Recovering Vocal Tract Shapes from MFCC Parameters* In Proceedings of the International Conference of Spoken Language Processing, 1998.
- [6] Carre, R. *Distinctive regions in acoustic tubes. Speech production modelling*. Journal d'Acoustique, 1992.
- [7] Mahmoud A. Ismail *Vocal Tract Area Function Estimation Using Particle Swarm* Journal of Computers, 2008.
- [8] Leonard Manzara *The Tube Resonance Model Speech Synthesizer* J. Acoust. Soc. Am. Volume 117, Issue 4, pp. 2541-2541, April 2005
- [9] David Hill et Al. *Real-time articulatory speech-synthesis-by-rules*, Proceedings of AVIOS, 1995
- [10] Lawrence H.Smith, Douglas J.Nelson *The Multiple tube resonance model* Advanced Signal Processing Algorithms, Architectures, and Implementations XII. Edited by Luk, Franklin T. Proceedings of the SPIE, Volume 4791, pp. 33-42, 2002.
- [11] Albert Tarantola, *Inverse Problem Theory*, Society for Industrial and Applied Mathematics, Philadelphia 2005.
- [12] E.Marchetto et al. *Estimation of a Physical Model of The Vocal Folds via Dynamic Programming Technique*, International Workshop on Models and Analysis of Vocal Emissions, 2007
- [13] S.Atal, J.Chang, J.Mathews and W.Tukey. *Inversion of Articulatory-to-Acoustic Transformation in the Vocal Tract by Computer-Sorting-Technique*, Journal of the Acoustical Society of America vol. 63, 1978
- [14] J.Schroeter and M. Sondhi, *Techniques for Estimating Vocal Tract Shapes from Speech Signal*, IEE Trans. on Speech Audio Processing vol.2, 1994

- [15] Biao Luo Jinhua Zheng, *A new methodology for searching robust Pareto optimal solutions with MOEAs*, Evolutionary Computation, 2008.
- [16] Goldberg E., *Genetic Algorithms in Search, Optimization and Machine Learning*, Kluwer Academic Publishers, Boston, 1989
- [17] Peter Birkholz *VocalTractLab Ein neues Software-tool fr die artikulatorische Sprachsynthese in der Lehre.*, 26th Jahrestagung der DGPP, pp. 209211, Leipzig, Germany