

A Probabilistic Graphical Model for Tuning Cochlear Implants

Íñigo Bermejo¹, Francisco Javier Díez¹, Paul Govaerts², and Bart Vaerenberg^{2,3}

¹ ETSI Informática, UNED, Juan del Rosal 16, Madrid, Spain

² The Eargroup, Herentalsebaan 71, B-2100 Antwerp-Deurne, Belgium

³ Laboratory of Biomedical Physics, University of Antwerp, Belgium

Abstract. Severe and profound hearing losses can be treated with cochlear implants (CI). Given that a CI may have up to 150 tunable parameters, adjusting them is a highly complex task. For this reason, we decided to build a decision support system based on a new type of probabilistic graphical model (PGM) that we call tuning networks. Given the results of a set of audiological tests and the current status of the parameter set, the system looks for the set of changes in the parameters of the CI that will lead to the biggest improvement in the user's hearing ability. Because of the high number of variables involved in the problem we have used an object-oriented approach to build the network. The prototype has been informally evaluated comparing its advice with those of the expert and of a previous decision support system based on deterministic rules. Tuning networks can be used to adjust other electrical or mechanical devices, not only in medicine.

1 Introduction

Cochlear implants (CI) are being successfully applied to treat severe and profound hearing losses. A CI consists of a speech processor that analyzes the sound and an array of electrodes placed into the cochlea which pass an electrical signal directly to the auditory nerve.

After implantation, CIs need to be programmed or “fitted” to optimize the user's hearing capability. This is usually a challenging and time-consuming task that is typically performed by highly trained audiologists or medical doctors. CI centers and manufacturers have developed their own heuristics, usually in the form of simple “if-then” rules applied in a very flexible but individual and uncontrollable way. Recipients using incorrectly programmed CIs experience poor performance and outcomes.

One of those applications, called FOX [1], was developed by Otoconsult, an audiological clinic in Antwerp, Belgium. It is being used in several centers across Europe. FOX is based on parameterized deterministic rules, which entails some limitations, such as the difficulty to maintain the knowledge base when the number of rules increases and the inability to learn from data. The Opti-FOX project [2] was conceived to overcome these limitations. In the beginning, an approach

with supervised classification algorithms—such as the k -NN classifier—was attempted, but failed to progress due to the complexity of the problem and the small number of records available to learn from. In order to improve the results of FOX, the most promising approach seemed to build a probabilistic graphical model (PGM) because this type of model can combine expert knowledge, the power of probabilistic reasoning, and the ability to learn from data.

This paper describes briefly a new type of PGM especially tailored for tuning programmable devices and how it has been used to build a decision support system for fitting CIs.

2 Tuning Networks

A tuning network consists of an acyclic directed graph (ADG) containing chance, decision and utility nodes, and a probability distribution. As in other types of PGMs, a decision node represents a variable that is under the direct control of the decision maker, while chance nodes represent features of the system over which the decision maker has no direct control, and utility nodes represent the decision maker’s preferences, measured on a numerical scale. In tuning networks, each property of the system is modeled by a relative-value node that represents a change in its value and, optionally, by an absolute-value node. In the case of a tunable parameter (for example, the sensitivity of the microphone), the relative-value node is a decision node because the programmer of the CI can increase, decrease, or keep the value of the parameter; the absolute-value node is represented as a chance node for which we have evidence, because the value of a tunable parameter is always known. We may also have evidence about the absolute-value nodes that represent measurements, such as the result of a test. Utility nodes are always relative-value nodes, as they represent the increase or decrease in the user’s performance as a consequence of tuning some parameters.

An important component of tuning networks is the *tuning model*, a new canonical model based on the property of *independence of causal interaction* (ICI) [3–5]. Canonical models represent how a variable is probabilistically influenced by a set of parent variables [6], in general assuming a pattern of causal interaction. Their main advantage is that the number of parameters (conditional probabilities) is proportional to the number of parents, while in the general case it grows exponentially. ICI models assume that each parent produces the effect with a certain probability, independently of the values of the other parents, and the global effect is determined by a function, specific of each type of ICI model, that combines the individual effects; for example, in the noisy OR the effect is present when at least one of the causes has produced the effect.

A unique feature of the tuning model is that it assumes that every variable involved has exactly three values: increased, decreased, and not-changed, while other ICI models, such as the noisy OR and the noisy AND, assume that all variables are boolean, and other models, such as the noisy MIN and the noisy MAX impose no restriction about the number of values of each variable [5]. The tuning model assumes that a change in one of the parents causes a change in the

child variable with a certain probability. When some of the parents induce an increase and others cause a decrease, the global effect depends on whether there are more increases than decreases, or vice versa, or there is a tie. It is therefore similar to a majority voting function.

Tuning networks differ from influence diagrams [7] in that they do not have a total ordering of the decisions because the order in which the parameters are tuned does not affect the result. Additionally, all the evidence is available before making the decisions, as it is in each session when programming a CI, while in influence diagrams some decisions provide evidence that can be used in subsequent decisions. As a consequence, the algorithms for evaluating these two types of models are very different—see Section 3.2.

3 Construction of the Model

3.1 Model Construction

Variables in the Model In our tuning network, the tunable parameters are those of the CI; as mentioned above, each one is represented by an absolute-value chance node and a relative-value decision node. Each electrode has several tunable parameters, e.g., the T level (the softest electrical input level detectable by the user), the M level (the electrical input level perceived as loud but comfortable), etc. Besides, the CI has a set of tunable parameters that are electrode-independent, i.e., global to the implant, such as the volume of the microphone.

The model also represents the results of a battery of different tests, such as audiometries, phoneme discrimination and speech recognition tests. Each measurement of a test is modeled with a chance node representing the current value of the test (this node receives evidence when performing the test), a chance node representing the expected change in the result of the tests given the changes in the tunable parameters, and a utility variable defining the utility function based on the other two.

Other nodes represent internal properties of the device, such as the amount of energy in the auditory nerve, which depend on the tunable parameters and in turn affect the results of the tests.

The global utility of our model is the sum of the results of all tests; therefore maximizing this utility is the same as optimizing the user’s hearing ability.

The resulting model contains 202 nodes and 664 links.

Elicitation of Numerical Parameters The probabilities and utilities have been assessed by the expert: the probabilities are subjective estimates based on his expertise while the utilities have been estimated by roughly assigning monetary value to positive and negative changes in the results of tests.

Object-Oriented Probabilistic Networks The network, containing sets of repeated structures (such as electrodes, frequency bands and tests) was modeled following the object-oriented paradigm for PGMs as proposed by [8, 9]. A

class defines a structure consisting of a set of attributes and their probabilistic relations and is connected with other classes through their input parameters, namely instances of other classes. An OOPN consists of a set of instances and their causal relations.

3.2 Inference

Inference in a tuning network consists in looking for the optimal strategy, i.e., the set of changes in the tunable parameters that maximizes the global expected utility. As an exhaustive search would be computationally unaffordable, we have implemented a greedy *search and score* algorithm that examines myopically the space of possible strategies. The search is initialized by setting all policies for all decision nodes to “no change”. It then iteratively looks for the single change in the strategy, i.e. a change in a decision node’s policy that maximizes the global utility function.

The score for each strategy, namely the global expected utility given the strategy, is computed using an inference algorithm. Given the high number of variables in the model and its high connectivity, the cost of running exact inference algorithms is unaffordable. For that reason, we decided to use an approximate inference algorithm—namely a likelihood weighting method [10] adapted to networks with utility nodes—whose spatial and temporal complexities grow linearly with the number of nodes instead of exponentially. The main drawback of likelihood weighting is that its accuracy decreases with extremely unlikely evidence, but it still fits our needs as the observed nodes usually have no extreme probabilities.

We compared the results of this greedy algorithm—in simplified versions of our model—with those of an exact inference algorithm (variable elimination) and both returned the same optimal strategy under different evidence scenarios. The execution time of the greedy algorithm, which has been implemented to run in parallel taking advantage of multiple core processors, depends on the number of changes proposed by the optimal strategy, but in a regular desktop computer (Intel Core i5-2500 @ 3.30GHz and 8GBs of RAM) is usually under a minute.

3.3 Evaluation of the Model

We have initially built a prototype for the low-frequency electrodes, i.e., those in the range from 250 to 1000 Hz. This model has been tested on a set of cases taken from a database of real CI users. The recommendations output by our model have been compared with those of FOX, the expert system based on deterministic rules, having the expert as a judge. Given that FOX was built by this expert, it is not surprising that in general FOX’s recommendations agreed with his. In many cases, also the recommendations of our model agreed with both FOX and the expert. There were, however, some cases in which our model recommended some interventions that surprised the expert, but he never deemed them non-sensical. On the contrary, he described them as “intelligent”, “smart” and “worth trying”.

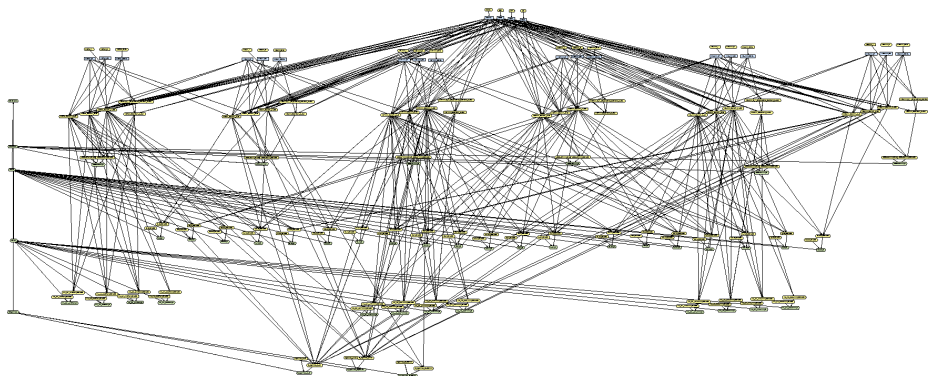


Fig. 1. Screenshot of the prototype network.

On July 31, 2012 a patient at Otoconsult had a poor performance in the speech understanding tests, in spite of having an audiometry in the range of normality. The audiologists using their expertise and FOX's support, were not able to improve her ability to understand spoken words. However, when her implant was fitted using the advice of our prototype, her performance increased to the level of normality. Of course, this isolated result does not prove that our model outperforms FOX or the audiologists in general, but it is a promising result.

4 Conclusions and Future Work

In the context of the European project Opti-FOX, we have built a PGM for programming CIs. The development of tuning networks and our framework for OOPNs has been motivated by the needs encountered in this project, but they can be applied to adjust other electrical or mechanical devices, not only in medicine.

The advantages of our model with respect to FOX, the rule based system, are that our model is capable of complex reasoning whereas FOX only concatenates rules, that FOX is deterministic while our model handles uncertainty, and that our model will be fine-tuned by learning from data. However, FOX is still a more mature project that has been evaluated extensively and includes features that our model still lacks, such as the ability to determine the quantity by which the value of a parameter should be changed.

The most obvious next step in the project is to test the developed prototype on real CI users. Besides, we are currently working on learning the conditional probabilities from a database, in order to fine-tune the probabilities elicited by the expert. Given that our model contains unobservable variables, the usual parametric learning algorithms cannot be applied. Instead, we are using the Expectation Maximization (EM) algorithm, applied to the learning of Bayesian networks as proposed by Lauritzen [11].

Another aspect with room for improvement is the granularity of the variables. Relative-value variables were discretized into three intervals (increase, decrease, no change) to reduce the complexity of the problem. This over simplification prevents our model from accurately predicting the effect of small changes in the parameters of the CI.

Finally, the programming of a CI has a temporal aspect: it usually involves several sessions and the history of each patient is relevant. Unfortunately, the current model only considers the current values of the parameters. Turning our system into a partially-observable Markov decision process (POMDP) would allow us to model that temporal evolution and determine the optimal sequence of tests and parameter adjustments.

References

1. Govaerts, P.J., Vaerenberg, B., Ceulaer, G.D., Daemers, K., Beukelaer, C.D., Schauwers, K.: Development of a software tool using deterministic logic for the optimization of cochlear implant processor programming. *Otology & Neurology* **31** (2010) 908–918
2. Szlavik, Z., Vaerenberg, B., Kowalczyk, W., Govaerts, P.: Opti-fox: towards the automatic tuning of cochlear implants. In: *Proceedings of the 20th Belgian Dutch Conference on Machine Learning*. (2011) 79–80
3. Heckerman, D.: Causal independence for knowledge acquisition and inference. In: *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence (UAI'93)*, Washington, D.C., Morgan Kaufmann, San Mateo, CA (1993) 122–127
4. Heckerman, D., Breese, J.S.: Causal independence for probability assessment and inference using Bayesian networks. *IEEE Transactions on Systems, Man and Cybernetics—Part A: Systems and Humans* **26** (1996) 826–831
5. Díez, F.J., Druzdzel, M.J.: Canonical probabilistic models for knowledge engineering. Technical Report CISIAD-06-01, UNED, Madrid, Spain (2006)
6. Pearl, J.: *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA (1988)
7. Howard, R.A., Matheson, J.E.: Influence diagrams. In Howard, R.A., Matheson, J.E., eds.: *Readings on the Principles and Applications of Decision Analysis*. Strategic Decisions Group, Menlo Park, CA (1984) 719–762
8. Koller, D., Pfeffer, A.: Object-oriented Bayesian networks. In: *Proceedings of the Thirteenth Conference in Artificial Intelligence (UAI-97)*, San Francisco, CA, Morgan Kaufmann (1997) 302–313
9. Bangsø, O., Wullemmin, P.H.: Top-down construction and repetitive structures representation in Bayesian networks. In: *Proceedings of the Thirteenth International Florida Artificial Intelligence Research Society Conference (FLAIRS-2000)*, Orlando, FL (2000) 282–286
10. Shachter, R., Peot, M.: Simulation approaches to general probabilistic inference on belief networks. In Henrion, M., Shachter, R.D., Kanal, L.N., Lemmer, J.F., eds.: *Uncertainty in Artificial Intelligence 5*. Elsevier Science Publishers, Amsterdam, The Netherlands (1990) 221–231
11. Lauritzen, S.L.: The EM algorithm for graphical association models with missing data. *Comput. Stat. Data Anal.* **19**(2) (February 1995) 191–201