THE CLOSED LOOP OPTIMIZATION OF DEEP BRAIN STIMULATION PROGRAMMING

A Undergraduate Thesis
Presented to
The Academic Faculty

by

Ravinderjit Singh

Georgia Institute of Technology
May 2016

THE CLOSED LOOP OPTIMIZATION OF DEEP BRAIN STIMULATION PROGRAMMING

Approved by:

Dr. Stephen DeWeerth, Advisor
School of Biomedical Engineering
Georgia Institute of Technology

Dr. Garett Stanley
School of Biomedical Engineering
Georgia Institute of Technology

ACKNOWLEDGEMENTS

I wish to thank Jason White, Jeonghee Kim, and the DeWeerth research group for advising me while I did this work.

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

SUMMARY

Deep brain stimulation (DBS) is a procedure used to treat movement disorders such as Parkinson's disease. The current procedure for programming the parameters for DBS is time consuming and prone to error. The DBS programming procedure can be significantly improved using a closed-loop optimization approach. Due to recent advances in quantitative assessment metrics, the capability to translate a closed-loop optimization procedure for DBS programming from simulation to clinic has become more possible. Previous literature has presented closed-loop approaches that utilize evolutionary algorithms. It is very difficult to implement an evolutionary algorithm in the clinic because they typically require a large number of parameter evaluations. A parameter evaluation is testing how well a certain set of DBS parameters work. It is difficult to do a large number of parameter evaluations due to time constraints and patient fatigue. A response surface based closed-loop optimization approach for DBS programming is presented that has higher potential to be translated to the clinic because it requires much less parameter evaluations.

CHAPTER 1

INTRODUCTION

Parkinson's is a neurodegenerative disease where dopamine producing cells in the substantia nigra are lost. The causes for this disease are not well understood, and there is currently no cure for the disease. Treatments focus on alleviating symptoms such as tremor. The two main types of treatment are drugs like Levodopa, a chemical that can enter the brain where it is converted to dopamine, and Deep Brain Stimulation (DBS), a procedure where electrodes are implanted inside the brain and send electrical impulses to certain nuclei of the brain, like the subthalamic nucleus (STN) or the globus pallidus pars interna (GPi). This thesis will focus on DBS.

Deep Brain stimulation has been shown to alleviate symptoms associated with Parkinson's and was first used in 1997 to replace thalamotomy in treating the tremor associated with Parkinson's disease [1, 2]. The drug that is used to treat Parkinson's, Levodopa, can have long term complications such has dyskinesia, so an alternative, DBS, is also used to treat Parkinson's [2]. The current procedure for programming DBS parameters is very inefficient and time consuming. One study found that the mean time for programming and assessment of a patient with Parkinson's disease from the preoperative period to one year post operation was between 27 and 36.2 hours [3]. The current programming procedure is shown in Figure 1. An initial set of parameters is chosen, and then one parameter is varied until the side effect limit is reached [4]. Once that side effect limit is reached, then a different parameter is varied. This procedure is inefficient and only attempts to find a satisfactory set of parameters, not an optimal set. In addition, there are cases when the programming is so poor that dramatic improvement can be found with

reprogramming [5]. A study that looked at 41 patients who complained of suboptimal results from DBS found that 15 of those patients (37%) were poorly programmed, and these patients experienced significant improvement upon reprogramming [5].

Having an automated standardized approach to DBS programming would also be very beneficial. Currently, a specially trained individual is required to program stimulators. This can require individuals to travel quite far to find a center that can program stimulators. Therefore an automated computer guided approach would also be very beneficial. To perform such an approach, an automated method for assessing disease symptoms is also necessary, and it was found that motion sensors could be useful in assessing parkinsonian symptoms [6]. Using a motion sensor strategy, a recent study has been able to do open-loop parameter tests automatically and search for an optimum within those open loop tests[7]. A closed-loop optimization approach could further improve upon that approach.

This computational study will investigate the application of closed-loop optimization to automate the DBS programming procedure in a computational model (Fig. 2). In a closed-loop optimization scheme, a set of parameters is determined by an optimization algorithm, and then those parameters are tested on the system. The efficacy of those set of parameters is then fed back into the algorithm to determine the next set of parameters to test. This loop is continued until a stopping criterion is met. A good stopping criterion, in this application, ends the procedure when continuing only provides negligible or no extra relief to symptoms. Previous work on closed loop optimization for DBS programming has focused on using evolutionary algorithms like the genetic algorithm [8, 9]. However, it is difficult to implement an evolutionary algorithm in practice because they require a large number of parameter tests. A parameter tests in DBS

programming involves choosing a set of parameters and then evaluating the efficacy of those parameters. Performing a parameter test consumes time and fatigues the patient; therefore, a large number of parameter tests cannot be performed. A model-based optimization approach requires less parameter tests than an evolutionary approach; however, they are more computationally expensive. But, since the limiting factor here is parameter tests and not computational time, a model-based optimization would be better suited for DBS programming currently. This thesis will present a model based closed-loop optimization approach for DBS programming and begin exploring its feasibility in a computational model.

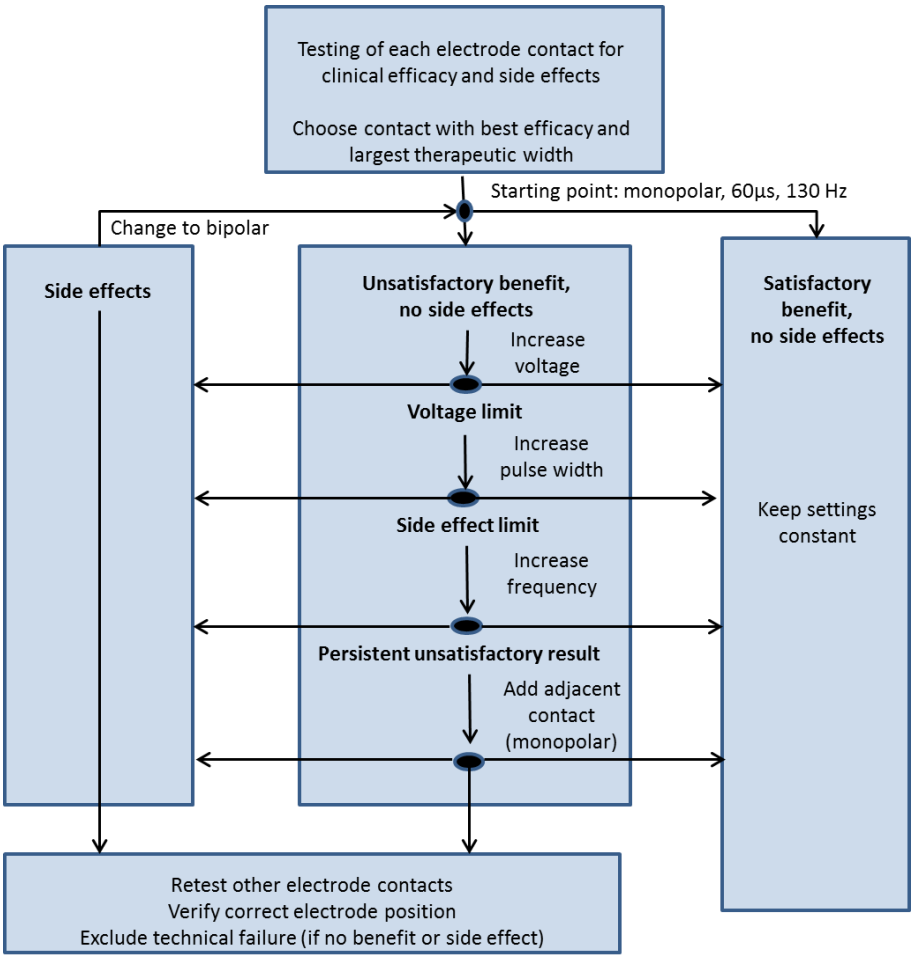

Figure 1: Conventional procedure to program a DBS device. This flowchart demonstrates the typical procedure for programming a DBS device[4]. An initial set of parameters is set, and then one parameter is varied until a side effect limit is reached. Then a different parameter is varied. This process is repeated, and if monopolar stimulation fails, then bipolar stimulation is attempted. This procedure is inefficient and does not attempt to find an optimal set of parameters.

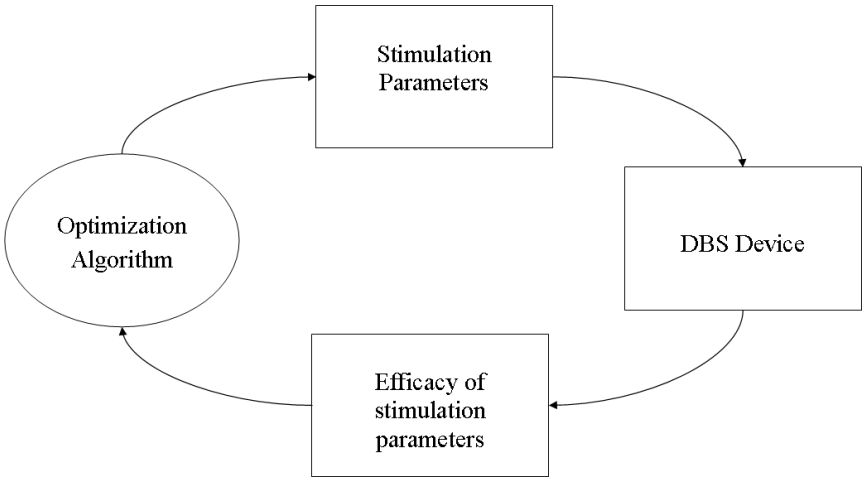

Figure 2: Closed-loop Optimization Scheme for DBS. A general overview of how a closed-loop optimization scheme would be applied to DBS is shown. A set of parameters are chosen by the algorithm which are tested on the patient, and then the efficacy of those parameters is fed back into the optimization algorithm. Then the algorithm can determine the next set of parameters to test to reach the optimum.

CHAPTER 2

METHODS

This computational study will begin exploring the feasibility of a model based closed-loop optimization approach for DBS programming. This section will present the model of Parkinson's that was used and the closed-loop optimization approach for DBS programming.

Model

A Hodgkin Huxley type model of the basal ganglia was used to model Parkinson's [10]. This model is a further modified version of the original Rubin and Terman model [11, 12]. The nuclei modeled are the subthalamic nucleus (STN), globus pallidus internus (GPi), globus pallidus externus (GPe), and the thalamus (Th). The sensory motor cortex (SMC) is not explicitly modeled. The SMC fires at a certain frequency with noise so that it is not exactly periodic. Below are the equations for the neurons in this model.

$$C_m V'_{Th} = -(I_L + I_{Na} + I_K + I_T + I_{GPi \rightarrow Th}) + I_{SMC}$$

$$C_m V'_{STN} = -(I_L + I_{Na} + I_K + I_T + I_{Ca} + I_{ahp} + I_{GPe \rightarrow STN}) + I_{app} + I_{DBS}$$

$$C_m V'_{GP} = -(I_L + I_{Na} + I_K + I_T + I_{Ca} + I_{ahp} + I_{STN \rightarrow GP} + I_{GPe \rightarrow GPe/GPi}) + I_{app}$$

The thalamus should relay the signals of the SMC in this model, but in the Parkinsonian condition the thalamus is unable to correctly transmit the signals from the SMC. Figure 3 shows the connectivity of the model and figure 4 shows the model's response to different conditions in the model. To switch from the healthy to the

Parkinson's condition, the applied current (I_{app}) which is the net current input from other sources is lowered. In the parkinsonian condition, the GPi cells fire in a more burst-like nature which prevents the thalamus from successfully relaying signals from the SMC (indicated by asterisks in figure 4). However, DBS (figure 3) applied to the STN is able to successfully restore the thalamus's ability to relay signals from the SMC.

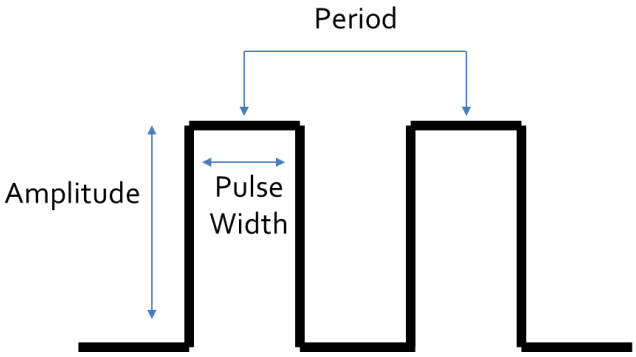

Figure 3: Stimulus parameters for DBS current. The DBS current waveform is a square waveform with an adjustable amplitude, pulse width, and period.

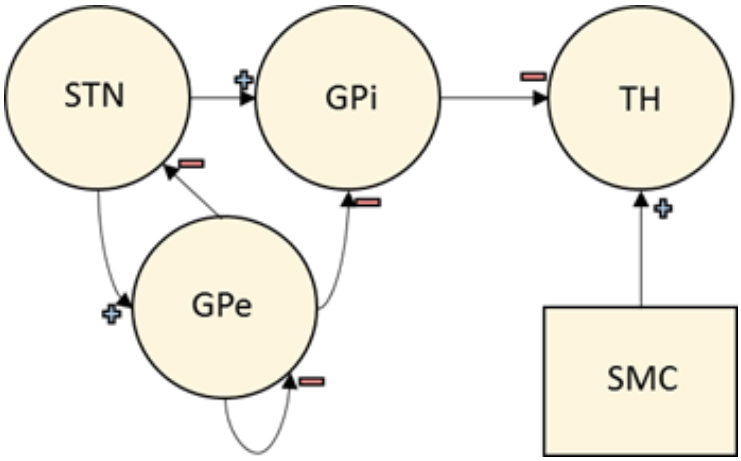

Figure 4: Connectivity of Model. The plus signs indicate excitatory connections and negative signs indicate inhibitory connections.

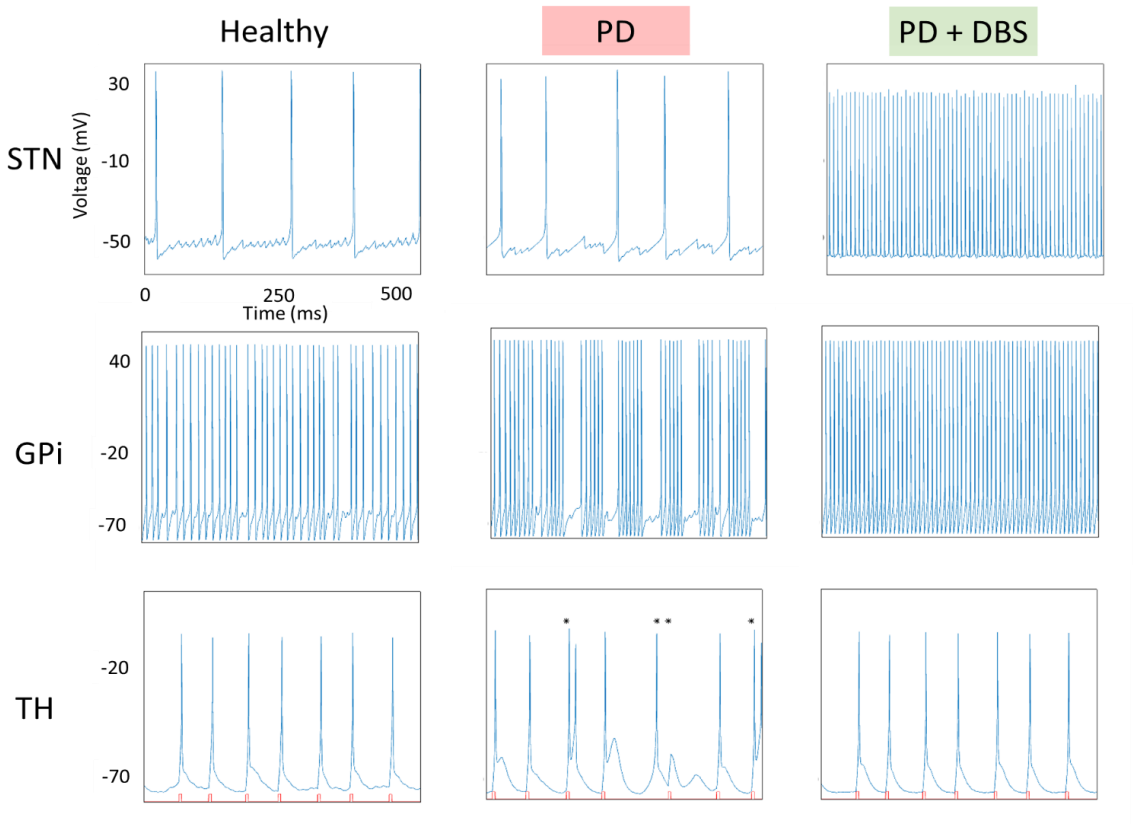

Figure 5. The model's response in different conditions. In the healthy condition, the GPi fires irregularly and the Th successfully relays all signals from the SMC. In the Parkinson's disease (PD) condition, the GPi will become more burst-like and the Th is unable to relay all signals from the SMC. Asterisks indicate a failure in thalamic transmission which is a burst (firing twice for one input), firing without an input, and not firing when an input is given. DBS entrains the STN and GPI to fire at the stimulation frequency which restores thalamic transmission.

Closed-Loop Optimization

The closed-loop optimization procedure utilized here is based off of the Efficient Global Optimization (EGO) algorithm [13]. A detailed overview of EGO is given in Appendix A. In this closed-loop scheme, first several open loop responses are recorded using a method such as a Latin hypercube. Then those open loop responses are used to build a model or response surface of the system using the DACE (Design and Analysis of Computer Experiments) model [14]. After constructing the DACE model, a criterion

known as the maximum expected improvement (max EI) is computed. The max EI is an estimate of what location in the parameter space will lead to the greatest improvement upon the current minimum. After determining where the EI is maximized, those parameters are tested on the system. Then a feedback term (or the cost function) is calculated and used to update the model. Two feedback terms are independently explored here. One feedback term used here is thalamic fidelity which is the number of incorrectly relayed signals over total signals to be transmitted, figure 6. The second feedback (GPiPow) term is two times the power in the GPi cells in the 1-20 Hz frequency range, figure 7. Another term that can be incorporated into the feedback is power consumption. The term used for battery consumption here is called BattC and is the integral of the current over one period which is equal to the pulse width * amplitude * frequency multiplied by a weighting term α. The point of the weighting term is balance how much to emphasize saving battery. If too much emphasis is put on minimizing battery consumption, then the patient may not see enough benefit; therefore, α must be chosen to promote solutions that save battery and do not reduce efficacy. In this study, α is set to 0.001. The Max EI also is useful as a stopping criterion. Whenever the max EI is less than a certain threshold, the best set of parameters evaluated to that point are returned and the loop is terminated. One parameter test is setting a set of DBS parameters and evaluating how well those parameters worked, so going through one complete loop is one parameter test.

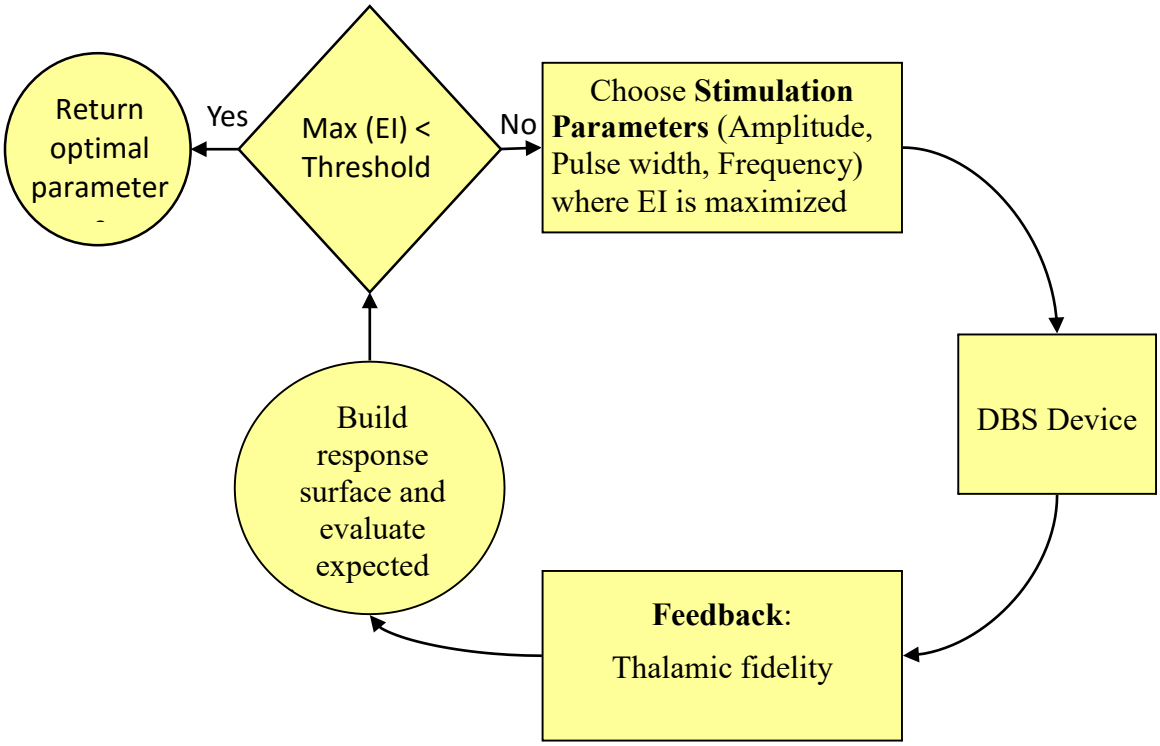

Figure 6. The Closed-Loop Optimization Procedure. A depiction of the closed-loop optimization scheme. The expected improvement (EI) is a metric that helps determine where to sample next by estimating where the most improvement can be found upon the current optimum.

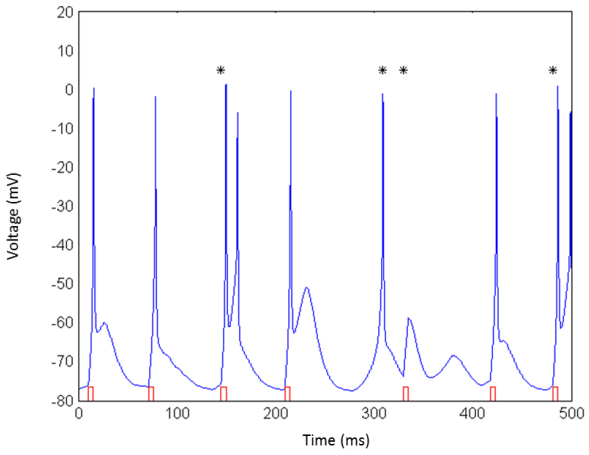

Figure 7. Feedback Parameter Thalamic Fidelity. Above is a display of the voltage trace for the thalamus during the Parkinson's condition. The asterisk denote incorrect relays. The feedback parameter thalamic fidelity is the number of errors (asterisk) divided by total inputs from the SMC.

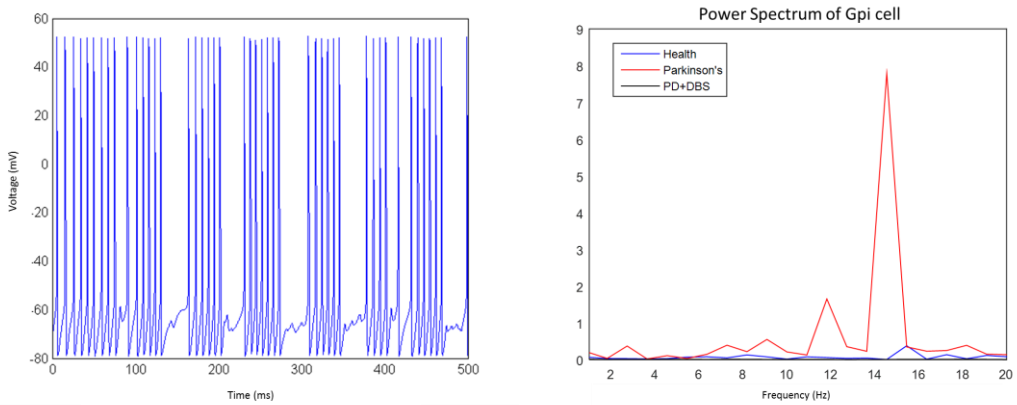

Figure 8. Feedback Parameter GPi Power. Above is a display of the voltage trace for the GPi during the Parkinson's condition. As can be seen in the power spectrum, the GPi fires much more synchronously in the Parkinson's condition making the power in the 1-20 Hz band a useful feedback parameter.

CHAPTER 3

RESULTS

The ability of the DACE model to estimate the true response for both feedback terms is depicted in figure 8. It can qualitatively be seen that the DACE estimate can reasonably predict the true response with a limited number of samples. The estimate will typically improve as more parameter tests are done. For all simulations, there were 33 open loop runs executed using a Latin hypercube. The stopping criterion was the Max EI falling less than 5%. An example optimization is shown in figure 9 where the Max EI is plotted over 60 closed loops. Results from 25 simulations for each feedback term are tabulated in Table 1. The power saved is calculated from the percentage of power saved from the worst case (max amplitude, pulse width, and frequency).

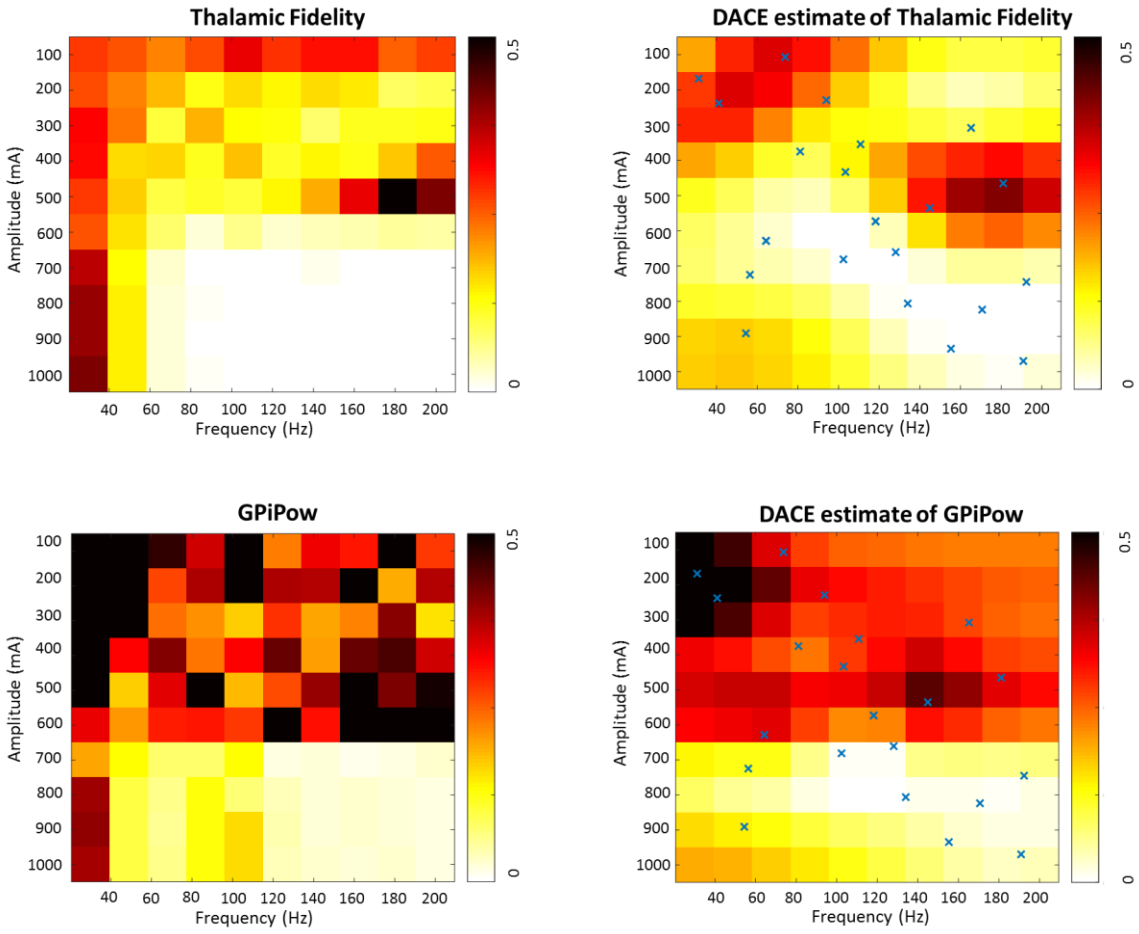

Figure 9. **Response surface estimate of parameter space.** On the left is a plot of the true parameter space for each respective objective function (feedback parameter). On the right are the response surfaces built with 20 parameter tests depicted by blue Xs using the DACE model. The DACE model serves as the response surface that estimates the DBS space in the closed-loop optimization procedure presented here.

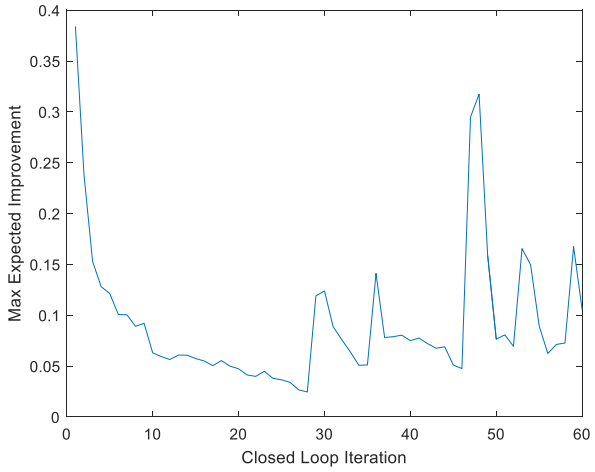

Figure 10. Example Optimization. The maximum expected improvement across 60 closed loop iterations is show for one optimization simulation where the feedback term was GPpow + BattC.

Feedback	Average Power Saved	Closed-Loop iterations	Thalamic Errors
Thalamic Fidelity	63.5 ± 18.5 %	62.8 ± 16.9	0 ± 0
Thalamic Fidelity + BattC	85.6 ± 2.1 %	63.8 ± 16.1	0 ± 0
GpiPow	70.5 ± 16.9 %	63.3 ± 12.8	0.3 ± 0.3 %
GpiPow + BattC	81.1 ± 5.6 %	67.4 ± 15.8	0.3 ± 0.3 %

Table 1. Optimization Results. Results are shown as mean ± standard deviation.

CHAPTER 4

DISCUSSION

The current approach for DBS programming can be time consuming and suboptimal. This work puts forward a model based closed-loop optimization approach as an alternative way to program DBS devices for Parkinson's disease. A model based approach is better suited than alternative approaches like evolutionary algorithms that have been proposed previously for DBS Programming. The primary advantage of the model based approach is that it will require less parameter tests. An evolutionary algorithm will typically beat a model based optimization approach in computation time; however, a model based approach will typically need less samples or parameter tests to find a reasonable optimum. For DBS programming, the primary limiting factor is how many parameters you can test. Testing a parameter is time consuming and fatigues the patient; therefore any programming approach for DBS needs to require as few parameter test's as possible. The constraint on parameter tests makes a model based approach better suited for DBS programming.

The goal of this study was to show through computational simulations the potential of a model based closed loop optimization algorithm to improve the DBS programming procedure to motivate more computational and clinical studies. As can be seen from figure 9, the Max EI can be quite volatile. This volatility is due to the model finding a new region of interest after sampling at a particular point. For the stopping criterion used in this paper of .05, the optimization would have stopped around iteration 20. But, perhaps, the new area of the model sampled at around iteration 47 found an area of interest with a better optimum. Therefore, it would be useful to explore different

stopping criteria to balance finding the best point possible in as few parameter tests as possible. The results from Table 1 show that the feedback terms that accounted for power consumption did save more power than the feedback terms that did not account for power. Accounting for battery consumption did take more closed-loop iterations, but not many more and the benefit of using less battery may be worth requiring a few extra parameter tests. Also when accounting for battery consumption, it has to be carefully considered how much to emphasize battery consumption. If battery consumption is emphasized too strongly then the efficacy of the treatment may suffer (figure 11). Thalamic errors were not completely removed for some simulations where GPiPow was in the feedback. This is likely due to GPiPow being a more indirect measurement of disease state. One drawback of these results is that they are only from 25 simulations in each feedback term. For a better indication of the efficacy of this approach, a larger number of simulations should be done.

The primary barrier to implement the proposed closed-loop optimization within the clinic is an automated way to measure some type of feedback. The feedback parameters used in this work are not currently measurable in the clinic. Therefore, identifying an objective function that is quantitative and easy to measure is necessary to implement this procedure. Recent clinical work has utilized an automated quantitative method to assess Parkinson's symptoms in an open loop fashion [7]. It would be possible to implement the closed-loop optimization proposed here within that setup; however, there is still much work to be done in identifying objective functions to optimize against in Parkinson's and other neurodegenerative diseases.

Another interesting way to implement this methodology in the clinic may be by utilizing the biophysical model used here in conjunction with the black box model. Physiological measurements made from an individual could be used to adapt the biophysical model to better fit that individual. Then this optimization approach could be used to build an estimate of the parameter space in the fitted model. Then the optimum of the estimated parameter space can be used as a possible set of parameters to test.

This computational study focused on the monopolar case for DBS. Accounting for stimulation with more poles would be useful for DBS since multipolar stimulation can better localize the stimulating current which can help limit side effects. To account for the multipolar case in simulation, a finite element model would be required to better model the effects of multiple poles. Also, certain aspects of the proposed optimization scheme can be further optimized such as the number of open loops that are initially run and the best threshold for maximum expected improvement to stop the optimization.

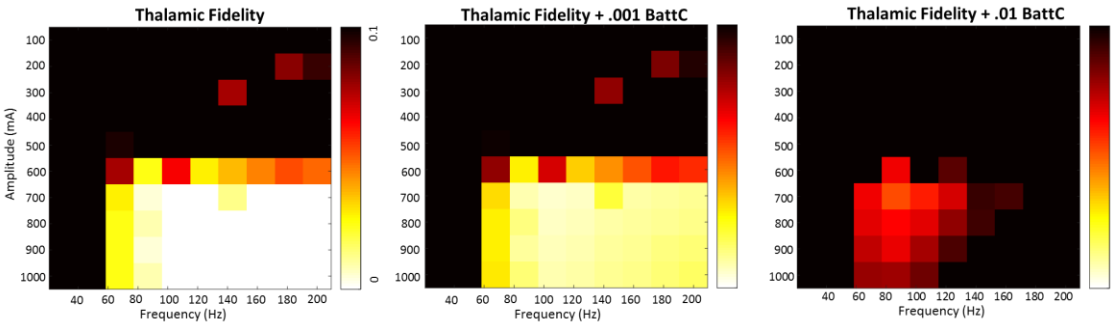

Figure 11. Effect of incorporating battery consumption into the feedback. The parameter space for thalamic fidelity without any power incorporated into it can be seen on the left. The middle figure shows the effect of weighing the battery consumption term by .001. This weighting helps discriminate between equally effective parameters by penalizing parameters that consume more battery. On the far right is weighing the battery consumption term by .01. This weighting now allows parameters at a frequency of 60 Hz which are clearly not as effective at reducing symptoms as higher frequencies appear to be the better choice because they consume less battery. The weighting of the battery consumption term must be carefully chosen so to find parameters that provide the most benefit and consume the least amount of power.

CHAPTER 5

CONCLUSION

The current procedure for DBS programming is time consuming and sub optimal. To improve the current procedure, a model based closed-loop global optimization approach was presented. Model based approaches can typically require much less parameter tests to find an optimum than evolutionary approaches, the primary approach investigated for DBS previously. For an optimization to be used in the clinic currently, it must use as few parameter tests as possible because it is time consuming and fatiguing to the patient to test a parameter set. Therefore, model based approaches currently are more advantageous than evolutionary approaches to do a closed-loop optimization of DBS programming. This work shows that the model based approach presented here can be effective for DBS programming and should be investigated further through computation and clinical studies.

APPENDIX A

THE EFFICIENT GLOBAL OPTIMIZATION ALGORITHM (EGO)

The algorithm that will form the foundation of the closed-loop optimization done in this study is the efficient global optimization algorithm (EGO) (Fig. 12) [13]. An overview of EGO will be given here. For a more detailed description of EGO, readers are suggested to [13]. In EGO, a model is built that treats each point as the result of a stochastic process known as the DACE (Design and Analysis of Computer Experiments) model [14]. By treating points this way, it provides tools for quantifying the amount of uncertainty at points that have not been evaluated which can help in determining where in the design space to search next. Also in this model, it is assumed that points that are spatially closer to each other have a higher correlation in their errors than points that are farther away from each other, which is intuitive. The following formulas are used to calculate a distance, which is not the usual Euclidean distance, between two points, and this distance is used to calculate the correlation of the error between the two points.

$$d(\boldsymbol{x}_1, \boldsymbol{x}_2) = \sum_{h=1}^{k} \theta \left| x_{1(h)} - x_{2(h)} \right|^p \qquad \theta \geq 0 \;\; p \in [1,2] \qquad \text{Eq.1}$$

$$Corr(x_1, x_2) = \; e^{-d(x_1, x_2)} \qquad \text{Eq.2}$$

K represents the number of variables, or dimensions, h. In my case, k is 4 (amplitude, frequency, pulse width, and electrode). Depending on how the space is parameterized, k could be greater than 4. θ is a measure of the 'sensitivity' of a variable. So if a variable, like amplitude, has a large sensitivity, then a small change in amplitude would lead to a large change in the value of the objective function, and if amplitude has a small sensitivity, then a large change in amplitude would lead to a small change in the value of

the objective function. The parameter p is related to the smoothness of the function. The correlation matrix, $\boldsymbol{R}$, is a n by n matrix which at row i and column j has the value $Corr(x_i, x_j)$, and n is the number of observed points. Then values for μ and σ^2 can be estimated by maximizing the likelihood function (Eq.3).

$$\frac{1}{(2\pi)^{n/2}(\sigma^2)^{n/2}|\boldsymbol{R}|^{1/2}} \exp\left[-\frac{(\boldsymbol{y}-\boldsymbol{1}\mu)'\boldsymbol{R}^{-1}(\boldsymbol{y}-\boldsymbol{1}\mu)}{2\sigma^2}\right] \quad \text{Eq.3}$$

$$\hat{\mu} = \frac{\boldsymbol{1}'\boldsymbol{R}^{-1}\boldsymbol{y}}{\boldsymbol{1}'\boldsymbol{R}^{-1}\boldsymbol{1}} \quad \text{Eq.4}$$

$$\hat{\sigma}^2 = \frac{(\boldsymbol{y}-\boldsymbol{1}\hat{\mu})'\boldsymbol{R}^{-1}(\boldsymbol{y}-\boldsymbol{1}\hat{\mu})}{n} \quad \text{Eq.5}$$

The vector $\boldsymbol{y}$ is the n observations of the objective function values. The vector $\boldsymbol{1}$ is a n long vector of ones. Then the values of $\hat{\mu}$ and $\hat{\sigma}^2$ are plugged back into the likelihood function to find estimates of θ and p. Then the following equation can be used as a predictor of the objective function. Eq.6

$$\hat{y}(x^*) = \hat{\mu} + \boldsymbol{r}\boldsymbol{R}^{-1}(\boldsymbol{y}-\boldsymbol{1}\hat{\mu})$$

The parameter x^* is the point at which the prediction is being done. The vector $\boldsymbol{r}$ is a n length vector of correlation of errors between the point of prediction, x^*, and the already sampled points, $\boldsymbol{x}$, so element i in $\boldsymbol{r}$ can be calculated as $\boldsymbol{r}_i(x^*) = Corr(x^*, x_i)$. The mean squared error for this predictor, $s^2(x^*)$ can be determined as shown below.

$$s^2(x^*) = \hat{\sigma}^2\left[1 - \boldsymbol{r}'\boldsymbol{R}^{-1}\boldsymbol{r} + \frac{(1-\boldsymbol{1}'\boldsymbol{R}^{-1}\boldsymbol{r})^2}{\boldsymbol{1}'\boldsymbol{R}^{-1}\boldsymbol{1}}\right] \quad \text{Eq.7}$$

The root mean squared error (RMSE) is designated as follows $s(x) = \sqrt{s^2(x^*)}$. This completes the DACE model. Next, a criterion known as expected improvement is calculated to determine where to sample for the next point. Expected improvement

combines the uncertainty of not knowing what the objective function is at a certain point along with the probability that a particular point could improve upon the current best point for the minimum, f_{min}. The expected improvement is calculated as shown below where PDF is the probability density function and the CDF is the cumulative distribution function.

$$EI(\boldsymbol{x}) = \left(f_{min} - \hat{y}(\boldsymbol{x})\right) PDF\left(\frac{f_{min} - \hat{y}(\boldsymbol{x})}{s(\boldsymbol{x})}\right) + s(\boldsymbol{x}) CDF\left(\frac{f_{min} - \hat{y}(\boldsymbol{x})}{s(\boldsymbol{x})}\right) \qquad \text{Eq.8}$$

The expected improvement also provides a good stopping rule. For example, if the expected improvement is less than 0.1%, then return the current optimum. This is just an example of a stopping rule and not necessarily the stopping rule that will be used in this study. An overview of the EGO algorithm has been given. For a more detailed overview of EGO, readers are recommended to the following paper [13]. The EGO algorithm has been further enhanced to account for stochastic systems[15]. A stochastic system here is a system that gives different results each time an experiment is repeated. This study will look at employing another modified version of EGO that can account for stochastic systems and an objective function that changes with time.

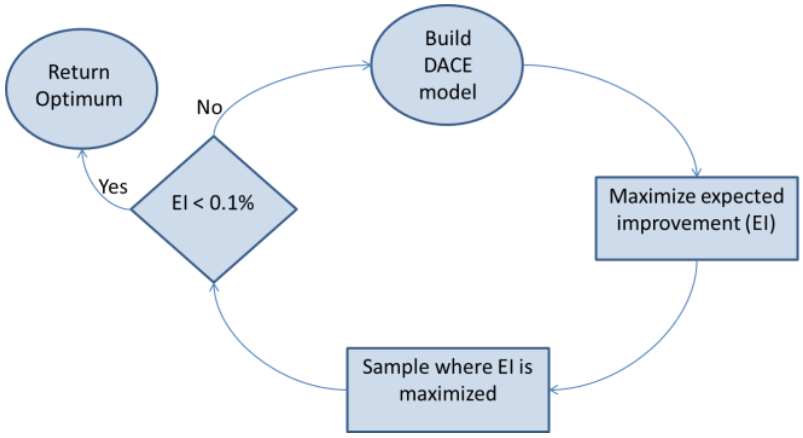

Figure 12: The efficient global optimization (EGO) algorithm. In the EGO algorithm, first a set of parameters for the DACE model are found using maximum likelihood estimation. Then, after the DACE model is built, a criterion known as expected improvement is maximized. Then a sample is taken where the EI is maximum, and the process iterates until a stopping criterion is met. In this figure, the stopping criterion is that the maximum EI is less than 0.1%.

REFERENCES

[1] A. L. Benabid, "Deep brain stimulation for Parkinson's disease," *Current opinion in neurobiology,* vol. 13, pp. 696-706, 2003.

[2] D.-B. S. f. P. s. D. S. Group, "Deep-brain stimulation of the subthalamic nucleus or the pars interna of the globus pallidus in Parkinson's disease," *The New England journal of medicine,* vol. 345, p. 956, 2001.

[3] K. Hunka, O. Suchowersky, S. Wood, L. Derwent, and Z. H. Kiss, "Nursing time to program and assess deep brain stimulators in movement disorder patients," *Journal of Neuroscience Nursing,* vol. 37, pp. 204-210, 2005.

[4] J. Volkmann, J. Herzog, F. Kopper, and G. Deuschl, "Introduction to the programming of deep brain stimulators," *Movement disorders,* vol. 17, pp. S181-S187, 2002.

[5] M. S. Okun, M. Tagliati, M. Pourfar, H. H. Fernandez, R. L. Rodriguez, R. L. Alterman, *et al.*, "Management of referred deep brain stimulation failures: a retrospective analysis from 2 movement disorders centers," *Archives of neurology,* vol. 62, pp. 1250-1255, 2005.

[6] C. L. Pulliam, D. A. Heldman, T. H. Orcutt, T. O. Mera, J. P. Giuffrida, and J. L. Vitek, "Motion sensor strategies for automated optimization of deep brain stimulation in Parkinson's disease," *Parkinsonism & related disorders,* vol. 21, pp. 378-382, 2015.

[7] D. A. Heldman, C. L. Pulliam, E. Urrea Mendoza, M. Gartner, J. P. Giuffrida, E. B. Montgomery, *et al.*, "Computer-Guided Deep Brain Stimulation Programming

for Parkinson's Disease," *Neuromodulation: Technology at the Neural Interface,* 2015.

[8] X. J. Feng, B. Greenwald, H. Rabitz, E. Shea-Brown, and R. Kosut, "Toward closed-loop optimization of deep brain stimulation for Parkinson's disease: concepts and lessons from a computational model," *J Neural Eng,* vol. 4, pp. L14-21, Jun 2007.

[9] X.-J. Feng, E. Shea-Brown, B. Greenwald, R. Kosut, and H. Rabitz, "Optimal deep brain stimulation of the subthalamic nucleus—a computational study," *Journal of computational neuroscience,* vol. 23, pp. 265-282, 2007.

[10] R. Q. So, A. R. Kent, and W. M. Grill, "Relative contributions of local cell and passing fiber activation and silencing to changes in thalamic fidelity during deep brain stimulation and lesioning: a computational modeling study," *J Comput Neurosci,* vol. 32, pp. 499-519, Jun 2012.

[11] D. Terman, J. Rubin, A. Yew, and C. Wilson, "Activity patterns in a model for the subthalamopallidal network of the basal ganglia," *The Journal of neuroscience,* vol. 22, pp. 2963-2976, 2002.

[12] J. E. Rubin and D. Terman, "High frequency stimulation of the subthalamic nucleus eliminates pathological thalamic rhythmicity in a computational model," *Journal of computational neuroscience,* vol. 16, pp. 211-235, 2004.

[13] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global optimization,* vol. 13, pp. 455-492, 1998.

[14] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, "Design and analysis of computer experiments," *Statistical science,* pp. 409-423, 1989.

[15] D. Huang, T. T. Allen, W. I. Notz, and N. Zeng, "Global optimization of stochastic black-box systems via sequential kriging meta-models," *Journal of global optimization,* vol. 34, pp. 441-466, 2006.