High-dimensional Optimistic Safe Optimization with Projection to Distance-preserving, Quasi-physical Spaces

Zeji Yi, Yunyue Wei, Hongda Li, Yanan Sui
National Engineering Research Center of Neuromodulation
School of Aerospace Engineering, Tsinghua University, China
{yizj20,weiyy20,lhd21}@mails.tsinghua.edu.cn, ysui@tsinghua.edu.cn

Abstract

Many real-life sampling problems are high-dimensional and require safety guarantee during optimization. Current safe exploration algorithms ensure safety by conservatively expanding the safe region, leading to inefficiency in large-scale input settings. In this paper, we propose a practical method, which utilizes auto-encoder to link physical prior of certain problems with index-based input space and also projects the original input space into a low-dimensional subspace. The low-dimensional space can be viewed as a quasi-conformal transformation of space with explicit physical meaning. An optimistic safe strategy to efficiently optimize the utility function is carried in the low-dimensional space then. We show in simulation that our method outperforms representative safe exploration algorithms while sacrificing little safety. Clinically, our proposed method also achieved better or competitive performance on two high-dimensional neural stimulation optimization tasks comparing to human experts.

1. Introduction

High-dimensional safety-critical systems widely exist in real-world applications. Many of them correspond to the problem of *safe exploration*, where we need to sequentially optimize an unknown utility function while satisfying some unknown safety constraints. For example, in clinical treatment, physicians need to choose among different therapies while avoiding those that would hurt patients. Existing safe exploration methods discriminate safe regions with estimated function lower confidence bound, ensuring safety with high probability. Such a pessimistic strategy might be inefficient in high-dimensional and large-scale input settings, which are common in real-life scenarios.

In this paper, we proposed a practical method which optimizes the objective function using an *optimistic safe* strategy, discriminating safe regions with estimated function upper confidence bound. To mitigate the high-dimensional issue, we trained an auto-encoder with physics intuition as output and Gaussian process regularization terms to map the original input space into a low-dimensional one. Tolerating a little number of unsafe sampling, our proposed method outperforms current safe optimization methods on synthetic functions, achieving comparable performance against representative unconstrained Bayesian optimization methods.

We also evaluated our method in optimizing spinal neuromodulation therapy for a paraplegic patient. A high-density electrode array was implanted in the dorsal epidural space to apply spinal stimulation. Optimized stimulation configurations can help the patient to stand and walk voluntarily. Excluding the amplitude and frequency, only for the electrodes configuration, there exists over 10^{15} possible selections, which are extremely difficult for human experts to optimize. Rather than selecting from the vast input space, human experts are capable of recognizing unsafe configurations, and denying them before applying stimulation. Our proposed method achieved competitive performance against human experts over all missions in the clinical experiment and even better than human in single muscle activation optimization.

2. Related Work

Bayesian optimization is often used for black-box optimization when the evaluations are expensive. The sequential decision-making problem with safety constraints has been extensively studied. To achieve full safety during exploratory sampling, several algorithms (Sui et al. (2015, 2018); Turchetta et al. (2019)) have been proposed with theoretical guarantee in near-optimality and safety with high probability. Baumann et al. (2021) and Sukhija et al. (2022) achieved global safe exploration by introducing backup policies. Compared to zero-toleration of unsafe actions, Marco et al. (2020) proposed a solution where a pre-defined budget of failures are allowed. Using Lipschitz continuity of functions and confidence interval estimated by Gaussian process (GP, Rasmussen (2003)), these algorithms discriminate safe region of the input space, conservatively exploring functions. Concretely, actions are labeled as safe when their estimated lower confidence bound is above the safety threshold. Such pessimistic safe strategy might be inefficient when exploring large scale input space.

Safe exploration algorithms, as well as other Bayesian Optimization (BO) methods, are restricted to low-dimensional problems, often below 20. The candidate selections grow exponentially with the dimension, and is impossible to enumerate when maximizing the acquisition function. To tackle the curse of dimensionality, many methods map the original input space to low-dimensional subspace (Garnett et al. (2013); Wang et al. (2016); Nayebi et al. (2019); Moriconi et al. (2020a,b)). They apply BO in feature space with feasible dimension, and evaluate new candidates after reconstruction. Functions are assumed to have intrinsic low-dimensional structure when applying these methods.

To the best of our knowledge, there does not exist any work about dimension reduction in safe optimization problems, especially in high-dimensional space, as safety constraints limit the efficiency of optimization in high-dimensional space significantly. Besides, the considerably amount of data demand for auto-encoder is contradict with the original intention of Bayesian Optimization.

3. Problem Formulation

Theoretically and practically, the elevation of dimension would lead current safety-constrained algorithms to be more inefficient. On one hand, high-dimensional Bayesian optimization is still an open problem. On the other hand, more samples would be wasted on the exploration

of the safe region. Here, we propose an optimistic safe optimization algorithm for complex hybrid high-dimensional systems, under a certain degree of safety restraint.

We aim to optimize an unknown reward function $f : \mathcal{X} \to \mathbb{R}$ with decision $\boldsymbol{x}_t \in \mathcal{X}$ where t represents the t_{th} round. Concretely, we pick a new decision point given current decision's feedback $y_t = f(\boldsymbol{x}_t) + n_t$ and all the history. Besides, we can also get a noise-perturbed observation of safety value from another unknown function $g(\boldsymbol{x}_t)+n_t$. We call the decisions that satisfy $g(\boldsymbol{x}_t) \geq h$ safe. In this paper, we wish to improve the sample efficiency by slightly loosing the safety constraint that allow a small number of unsafe decision. In this sense, we give a probabilistic version of safety constraint that,

$$Pr(g(\boldsymbol{x}_t) \geq h) \geq \alpha \text{ for } i = 1, \ldots, n \tag{1}$$

To be more specific, the input is formulated as hybrid variable $\boldsymbol{x}_t = \begin{bmatrix} {}^d\boldsymbol{x}_t \\ {}^c\boldsymbol{x}_t \end{bmatrix} \in \mathbb{R}^{k+l}$, Where ${}^d\boldsymbol{x}_t \in \mathbb{R}^k$ is the discrete variable and ${}^c\boldsymbol{x}_t \in \mathbb{R}^l$ is the continuous variable. We use an encoder $\mathcal{E} : \boldsymbol{x}_t \in \mathcal{X} \to \boldsymbol{s}_t \in \mathcal{S}$ to reduce the dimension of input by nonlinear mapping and then carry out optimistic safe optimization in space $\mathcal{S}$.

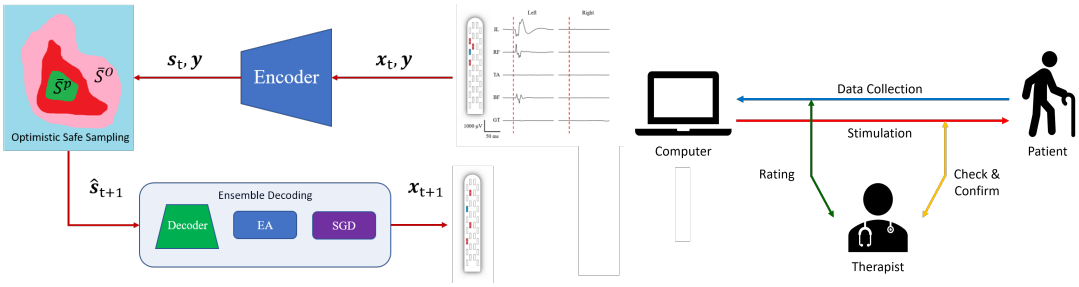
More specific explanation of neural stimulation case is attached in Appendix.A.

4. Algorithm

The optimistic safe optimization algorithm intended to optimize the utility function under probabilistic safety constraints. Unlike most safe optimization problems, we allow certain extent of unsafe evaluations. The optimistic opinion towards safety allows more bold and efficient samples of potential high value points. Obviously, the optimum gets better with more arms in the pool as alternatives. Meanwhile, the optimistic safe method focuses on the problem where the overall environment is safer, i.e., the safety threshold is lower than the mean value of safety function g. Because the optimistic method guarantees the lower bound of safe probability for each candidate point, the safety probability of the selected point is always higher than the threshold α. Therefore, the overall occurrence frequency of violation samples is much smaller than α. Compared to the original safe optimization method, optimistic safe method allows exploration outside the pessimistic safe region which also overcomes the pain point that the selection is always stuck in the isolated safe island of the original safe seed and the disconnected safe regions can never be explored. In a word, the optimistic safe algorithm solves the following problem

$$\max_{\boldsymbol{x}_t \in \mathcal{X}} f(\boldsymbol{s}_t) \quad \text{subject to } Pr(g(\boldsymbol{x}_t) \geq h) \geq \alpha \text{ for } t = 1, \ldots, n \tag{2}$$

Compared to the original GP-UCB algorithm (Srinivas et al. (2010)), we define a safe-set whose safety probability is above α, where α is usually smaller than 0.5. By choosing β^O which makes that $\Psi(\beta^O) = 1 - \alpha$. Then the candidate arms are the ones that satisfy $\mu_{t-1}(\boldsymbol{s}) + \beta^O \sigma_{t-1}(x) \geq h$. An overview of optimistic safe exploration is illustrated in Algorithm 1,Appendix.A. The experimental safety violation frequency depends on both optimism β^O and the safety threshold h. As for the regularized safety function, lower h implies a smaller unsafe region.



(a) Algorithm Pipeline (b) Clinical Pipeline

Figure 1: As shown in (a), at each round, the algorithm takes the feedback of the new evaluation which could be processed value of data recorded or rating of performance from therapist as shown in (b). Then the encoder takes the high-dimensional hybrid input and generates low-dimensional representation. In the low-dimensional space we do optimistic safe optimization and pass a recommended representation to ensemble decoding to get a recommended output.

Applying the optimistic safe algorithm to the specific problem directly as in section 3 is infeasible for the following reasons. Firstly, it is easy to notice that the underlying physical meaning is not clear for the discrete value. Especially, their value may just stand for index, which makes it difficult to select an appropriate kernel and construct the Gaussian process. Secondly, the enormous scale of candidate set is also difficult to deal with. Both problems remind us to learn a more appropriate representation for the high-dimensional hybrid space $\mathcal{X}$.

Here, we leverage an auto-encoder to carry out nonlinear mapping of the original space and build the representation. In the encoder part, hybrid high-dimensional variable $\boldsymbol{x}_t$ is the input and the code is the low-dimensional representation $\boldsymbol{s}_t$. As for the decoder part, the target output is some prior knowledge of the domain. For the neural stimulation case, a good target can be finite element method results or synthetic approximations for electric field. Unlike the inequality of continuous value which implies explicit physical meaning, the discrete variables are often index terms instead of value. After going through the encoder, the outputted prior knowledge from the decoder endows the representation code with reasonable physical meaning. We use simple Resnet as the network for the encoder and decoder. After the encoder-decoder model being established, optimistic safe optimization can be carried out in the low-dimensional representation space. The stimulation pattern space writes as $\mathcal{X}$, the low-d representation writes as $\mathcal{S}$, and the Encoder writes as $\mathcal{E} : x \in \mathcal{X} \to s \in \mathcal{S}$.

The reconstruction loss $Loss_{rec}$ is defined as the MSE loss between the original variable and the reconstructed one. The domain-knowledge integration loss $Loss_{KI}$ is defined as the MSE loss between the target prior knowledge and the one generated by the decoder. With these two terms, the auto-encoder is forced to learn a representation with physical meaning.

In addition, we would like the representation space to be a Reproducing Kernel Hilbert Space(RKHS) that we can easily apply Gaussian process regression. Therefore, the choice of the kernel is fetal. Without any further constraint, the underlying Gaussian process on the

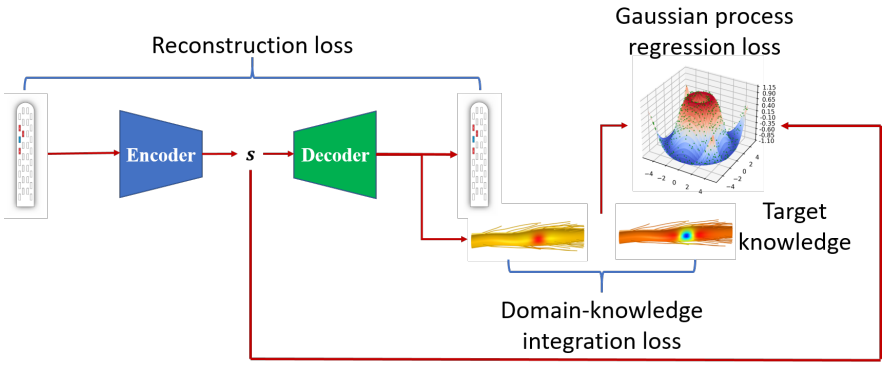


Figure 2: Trainging procedure: After the input hybrid variable being transformed into code, one decoder manages to reconstruct the original hybrid variable. The other managed to approach the prior knowledge.

representation space is arbitrary and unpredictable, which is not limited to common RBF, Matérn, e.t.c., Kernels. To shape and regular the space, we optimize the log-likelihood of the Gaussian process with the processed target knowledge as y and the code as s. Here we fix the kernel parameter and shaping the space to optimize the likelihood. Specifically, $Loss_{GP} = -log(P\left(y \mid X, \beta, \theta, \sigma^2\right))$. The overall loss is the weighted sum of the three

$$Loss = Loss_{rec} + 0.5 * Loss_{KI} + 0.1 * Loss_{GP} \tag{3}$$

We call the above auto-encoder Gaussian Process Amendment Auto-encoder. It codes high-dimensional hybrid parameters with prior domain knowledge and force the discrete variables with similar physical meaning to be closer in the code space.

With the established autoencoder, we can complete the high-dimensional hybrid optimistic safe optimization algorithm. Different from the vanilla Bayesian optimization setting that generates the candidate set from SOBOL or just equidistant sampling, our candidate set comes from both SOBOL sequence of the representation space and the coded local variation of current data. Then the autoencoder decodes the recommended s_t of each round. For the ones come directly from encoded x, it is straightforward. While for an arbitrary point in the representation space, we assemble two methods to get the nearest point for it. One is the reconstruction decoder from the auto-encoder. Another method is quasi-gradient by optimizing the discrete and continuous value of $\mathcal{X}$ with evolutionary algorithm and gradient respectively in rotation. For each candidate s_t, assemble the two methods and choose the better result. Eventually, we get a feasible test point in the original space.

5. Experiment

5.1 Synthetic Data

We compare our proposed method against two popular safe exploration algorithms: SafeOpt and GoOSE. We also add GP-UCB into baselines, which neglects safety constraints. We sampled 100 random functions from zero-mean GP with radial basis function kernel over input space $[-5, 5]^4$, setting safety threshold to -0.5. Fig 3 shows that our proposed methods

outperforms SafeOpt and GoOSE with a large margin. The algorithm achieves comparable performance against GP-UCB, with much fewer unsafe selections. Our method achieved efficient optimization while sacrificing little safety.

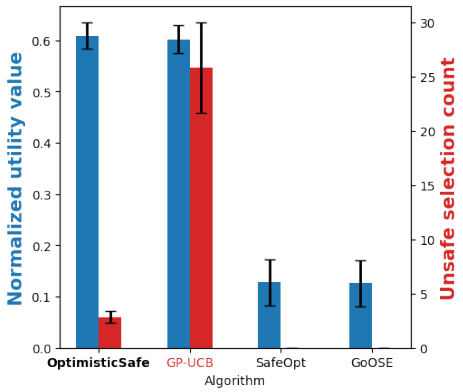


Figure 3: Algorithm performance on synthetic data.

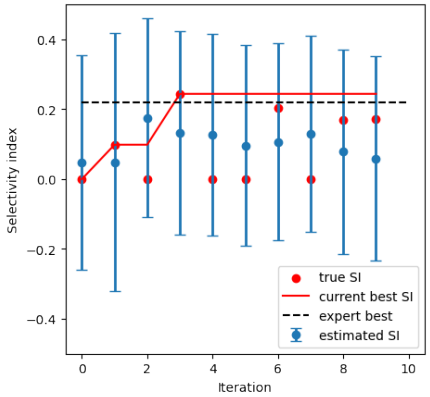


Figure 4: Optimization of Iliopsoas Muscle Selectivity

5.2 Clinical Experiment

We applied our algorithm to neural stimulation therapy optimization. Our first patient has motor complete spinal cord injury, no lower limb voluntary movement can be achieved without stimulation. The patient achieved full-weight walking with a walking frame after configuration optimization and training for a few months. The treatment/optimization procedure is illustrated in Figure 1 (b). Pain feedback scored by the patient was taken as safety function value. Clinicians can also deny recommendations they consider unsafe. Algorithm performance was evaluated in two tasks: single muscle activation optimization and standing/walking optimization. The results of the latter one are left to Appendix B.2.2 due to page limitation.

5.2.1 Muscle Selectivity Optimization

Swing is an important phase in the walking loop. To achieve such joint movement, the iliopsoas muscle should be activated while antagonistic muscles (e.g. rectus femoris) need to be suppressed. Selectivity Index (SI) is a measurement of single muscle activation calculated from electromyographic (EMG) data (Eq 4). Larger SI indicates better muscle selectivity. In this experiment, the patient was seated in a wheelchair and EMG data of 12 lower limb muscles was recorded. Before formal optimization, we tried all single-cathode configurations as the initial dataset and reference for restricting input space. Optimization performance between the algorithm and human expert was compared when optimizing iliopsoas SI, shown in Figure 4. Our algorithm outperformed human expert by 10.47% in SI optimization. Most of the true values lie in the confidence region of GP prediction, indicating the correctness of model estimation.

6. Conclusion

We proposed a practical method to tackle high-dimensional safe exploration problems. We used an autoencoder to learn a mapping between original input space and low-dimensional subspace with physical intuition and Gaussian process regularization. Our optimistic safe exploration strategy efficiently optimizes the objective function while sacrificing little safety. On synthetic data, we demonstrated that our proposed method outperformed representative safe exploration methods, achieving similar efficiency and fewer unsafe selections against unconstrained algorithms. On clinical evaluations of neural stimulation optimization, our methods achieved competitive performance compared to human experts, showing better exploration behavior. The algorithm has the potential to be employed in many real-life applications with high-dimensional settings and safety restrictions.

Acknowledgments

Zeji Yi and Yunyue Wei contributed equally to this work.

References

Dominik Baumann, Alonso Marco, Matteo Turchetta, and Sebastian Trimpe. Gosafe: Globally optimal safe robot learning. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4452–4458. IEEE, 2021.

Roman Garnett, Michael A Osborne, and Philipp Hennig. Active learning of linear embeddings for gaussian processes. *arXiv preprint arXiv:1310.6740*, 2013.

Alonso Marco, Alexander von Rohr, Dominik Baumann, José Miguel Hernández-Lobato, and Sebastian Trimpe. Excursion search for constrained bayesian optimization under a limited budget of failures. *arXiv preprint arXiv:2005.07443*, 2020.

Riccardo Moriconi, Marc Peter Deisenroth, and KS Sesh Kumar. High-dimensional bayesian optimization using low-dimensional feature spaces. *Machine Learning*, 109(9):1925–1943, 2020a.

Riccardo Moriconi, KS Kumar, and Marc Peter Deisenroth. High-dimensional bayesian optimization with projections using quantile gaussian processes. *Optimization Letters*, 14 (1):51–64, 2020b.

Amin Nayebi, Alexander Munteanu, and Matthias Poloczek. A framework for bayesian optimization in embedded subspaces. In *International Conference on Machine Learning*, pages 4752–4761. PMLR, 2019.

Carl Edward Rasmussen. Gaussian processes in machine learning. In *Summer school on machine learning*, pages 63–71. Springer, 2003.

Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings*

of the 27th International Conference on International Conference on Machine Learning, ICML'10, page 1015–1022, 2010.

Yanan Sui, Alkis Gotovos, Joel Burdick, and Andreas Krause. Safe exploration for optimization with gaussian processes. In *International conference on machine learning*, pages 997–1005. PMLR, 2015.

Yanan Sui, Vincent Zhuang, Joel Burdick, and Yisong Yue. Stagewise safe bayesian optimization with gaussian processes. In *International conference on machine learning*, pages 4781–4789. PMLR, 2018.

Bhavya Sukhija, Matteo Turchetta, David Lindner, Andreas Krause, Sebastian Trimpe, and Dominik Baumann. Scalable safe exploration for global optimization of dynamical systems. *arXiv preprint arXiv:2201.09562*, 2022.

Matteo Turchetta, Felix Berkenkamp, and Andreas Krause. Safe exploration for interactive machine learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando de Feitas. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research*, 55:361–387, 2016.

Appendix A.

In the specific neural stimulation optimization, R_d contains the configuration of anode and cathode, and R_c contains the other information as amplitude. Furthermore, anode, cathode, and open circuit are denoted by 1,-1,0 respectively. Given each X we observe $y_t = f(\boldsymbol{x}_t) + n_t$, where f could be all kinds of missions, selectivity of single muscle, movement of specific joint, performance of gait,etc. Generally speaking, neural stimulation is a safe and effective clinical treatment, but there are a minority group of patterns that could bring short-term side effect as mild labor pains or flatulence. Crucially, we want to avoid this kind of 'unsafe' points. Meanwhile, the evaluation cost is significant, we don't want the safety constraint to lower the efficiency too much. To balance the trade-off, we urge that each selected point is safe with the given extent of confidence. I.e., for all rounds t, it holds that $Pr(f(x_t) \leq h) \geq \alpha$, where$h \in \mathbb{R}$ s a mission-specific safety threshold. For the simplicity of future proof for convergence, the interval Q_t is set to be monotonic contracting for each round.

Algorithm 1 Optimistic Safe Optimization in representation space

Input Sample set $\mathcal{X}$,Encoder $\mathcal{E}$, Decoder $\mathcal{D}$, GP prior (μ, k, σ_0)
seed set $S_0 \in \mathcal{X}$, safety threshold h, safety probability α,safety constant β^O

Calculate β^O with CDF $\Psi(\beta^O) = 1 - \alpha$
$\mathcal{E}(S_0) \to S_0^{\mathcal{S}}$
$C_0(\boldsymbol{s}) \leftarrow [h, \infty)$,for all $\boldsymbol{s} \in S_0^{\mathcal{S}}$
$C_0(\boldsymbol{s}) \leftarrow \mathbb{R}$,for all $\boldsymbol{s} \in \mathcal{S} \backslash S_0^{\mathcal{S}}$
$Q_0(\boldsymbol{s}) \leftarrow \mathbb{R}$, for all $\boldsymbol{s} \in \mathcal{S}$
repeat
  $C_t(\boldsymbol{s}) \leftarrow C_{t-1}(\boldsymbol{s}) \cap Q_{t-1}(\boldsymbol{s})$
  $S_t^{\mathcal{S}} \leftarrow \bigcup_{\boldsymbol{s} \in S_{t-1}^{\mathcal{S}}} \left\{ \boldsymbol{s}' \in \mathcal{S} \mid \mu_{t-1}(\boldsymbol{s}) + \beta^O \sigma_{t-1}(x) \geq h \right\}$
  $\hat{\boldsymbol{s}}_t \leftarrow \operatorname{argmax}_{\boldsymbol{s} \in S_t^{\mathcal{S}}} (u_t(\boldsymbol{s}))$
  $\boldsymbol{x}_t \leftarrow$ Ensemble decoding $\hat{\boldsymbol{s}}_t$
  $y_t \leftarrow f(\boldsymbol{x}_t) + n_t$
  Update the GPR model and Q_t
until EPOCH ENDS

Appendix B.

.1 Synthetic Data

Since our proposed method and compared safe exploration method use different bound to recognize safety, we set different β parameter. For GP-UCB and our method, we set β to 0.5. For SafeOpt and GoOSE, we set β to 2.

Figure 3: Blue bars show the mean utility performance, values normalized by maximum value of each utility function. Red bars show the mean failures. Errorbars indicate one standard error.

.2 Clinical Experiment

.2.1 Muscle Selectivity

Selectivity Index (SI) is a measurement of single muscle activation, computed by the following equation:

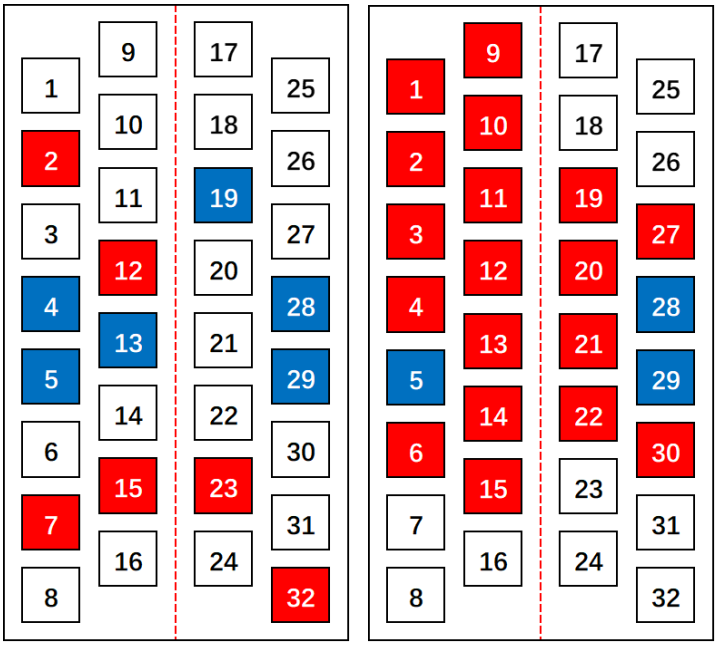
$$\mathrm{SI}_i = \frac{a_i}{1 + \sum_{j \neq i}^{m} a_j} \tag{4}$$

where m is the number of muscles, and a_i is the activation of the i-th muscle, often calculated from EMG. Error bar in plots shows GP estimated $\mu \pm \sigma$. We set β to 1 during optimization. Due to clinical constraints, double-blind experiment are not allowed to test efficiency between human experts and our algorithm.

.2.2 Standing and Walking Optimization

During standing optimization, the patient was initially seated and practiced standing with a standing frame to keep balance. Using muscle SI optimization results from section 5.2.1, we tried several configuration combinations as the initial dataset. Score evaluated by patient and expert was taken as utility function value. Our algorithm found configurations with good scores, which could help the patient to support most of his weight with his feet. As shown in Figure 5, our algorithm is capable to find multiple-cathode configurations and exploring large-scale input space.

During walking optimization, the patients practiced walking with a walking frame under stimulation. We want to optimize two main phases in the patient's walking loop: swing and stance, with two different configurations. Movement quality was scored by human experts, and taken as utility values. We optimized configurations to achieve full-weight walking.



(a) Algorithm (b) Human Expert

Figure 5: Configuration Comparison