Preference-Based Learning for Exoskeleton Gait Optimization

Maegan Tucker*[1], Ellen Novoseller*[2], Claudia Kann[1],
Yanan Sui[3], Yisong Yue[2], Joel W. Burdick[1,2], and Aaron D. Ames[1,2]

***Abstract*—This paper presents a personalized gait optimization framework for lower-body exoskeletons. Rather than optimizing numerical objectives such as the mechanical cost of transport, our approach directly learns from user preferences, e.g., for comfort. Building upon work in preference-based interactive learning, we present the CoSpar algorithm. CoSpar prompts the user to give pairwise preferences between trials and suggest improvements; as exoskeleton walking is a non-intuitive behavior, users can provide preferences more easily and reliably than numerical feedback. We show that CoSpar performs competitively in simulation and demonstrate a prototype implementation of CoSpar on a lower-body exoskeleton to optimize human walking trajectory features. In the experiments, CoSpar consistently found user-preferred parameters of the exoskeleton's walking gait, which suggests that it is a promising starting point for adapting and personalizing exoskeletons (or other assistive devices) to individual users.**

Fig. 1. Atalante Exoskeleton with and without the user. The user is wearing a mask to measure metabolic expenditure.

I. INTRODUCTION

The field of human-robot interaction is receiving increasing attention in many application domains, from mobility assistance to autonomous driving, and from education to dialog systems. In many such domains, for a robotic system to interact optimally with a human user, it must adapt to user feedback. In particular, learning from user feedback could help to improve robotic assistive devices.

This work focuses on optimizing walking gaits for a lower-body exoskeleton, Atalante, to maximize user comfort. Atalante, developed by Wandercraft [1], uses 12 actuated joints to restore mobility to individuals with lower-limb mobility impairments, which could potentially benefit approximately 6.4 million people in the United States alone [2]. Existing work with Atalante has demonstrated dynamically-stable walking using the method of partial hybrid zero dynamics (PHZD), originally designed for bipedal robots [3]–[5]. While this method generates stable bipedal locomotion, there is no current framework to optimize for comfort; yet, user comfort should be a critical objective of gait optimization for exoskeleton walking. While existing methods [6] can generate human-like walking gaits for bipedal robots, it is unlikely that these methods fulfill the preferences of individuals using robotic assistance.

Existing human-in-the-loop algorithms optimize quantitative metrics such as metabolic expenditure [7]; however, since the goal of this work is to optimize for user comfort, the presented learning approach uses user preferences obtained from sequential gait trials. By directly incorporating personalized feedback, we avoid making overly-strong assumptions about gait preference, or optimizing for a numerical quantity not aligned to personalized comfort.

For exoskeleton gait generation, as in many real-world settings involving people [8]–[10], it is challenging for people to reliably specify numerical scores or provide demonstrations. In such cases, the users' *relative preferences* measure their comfort more reliably. Previous studies have found preferences to be more reliable than numerical scores in a range of domains, including information retrieval [11] and autonomous driving [10].

Building upon techniques from dueling bandits [12]–[14] and coactive learning [15], [16], we propose the CoSpar algorithm to learn user-preferred exoskeleton gaits. CoSpar is a mixed-initiative approach, which both queries the user for preferences and allows the user to suggest improvements. We also validate CoSpar in simulation and human experiments, in which CoSpar finds user-preferred gaits within a gait library. This procedure not only identifies users' preferred walking trajectories, but also provides insights into the users' preferences for certain gaits.

II. GAIT GENERATION FOR BIPEDAL ROBOTS

Many existing lower-body exoskeletons either require the use of arm-crutches [17]–[19] or use slow static gaits with speeds around 0.05 m/s [20]. Using the PHZD method, dynamic crutchless exoskeleton walking has been demonstrated

*These two authors contributed equally to this work.

This research was supported by NIH grant EB007615, NSF NRI award 1724464, NSF Graduate Research Fellowship No. DGE1745301, and the Caltech Big Ideas and ZEITLIN Funds.

This work was conducted under IRB No. 16-0693.

[1]Authors are with the Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125.

[2]Authors are with the Department of Computing and Mathematical Sciences, California Institute of Technology, Pasadena, CA 91125.

[3]Author is with the School of Aerospace Engineering, Tsinghua University, Beijing, China 100084.

to generate dynamically-stable gaits. We briefly explain this method to illustrate how it can be adapted based on user preferences; for more details, refer to [3]–[5].

Partial Hybrid Zero Dynamics Method. Systems with impulse effects, such as ground impacts, can be represented as *hybrid control systems* [21]–[23]. Summarizing from [4], the natural system dynamics can then be represented on an invariant reduced-dimensional surface, termed the *zero dynamics* surface [24], by appropriately defining the *virtual constraints* and using a feedback-linearizing controller to drive them to zero. Since the exoskeleton's desired forward hip velocity is constant and its actual velocity experiences a jump at impact, the *partial zero dynamics* surface is considered. The *virtual constraints* are defined as the difference between the actual and desired outputs:

$$y_1(q, \dot{q}) = y_1^a(q, \dot{q}) - v_d \tag{1}$$

$$y_2(q, \alpha) = y_2^a(q) - y_2^d(\tau(q), \alpha), \tag{2}$$

where the actual outputs y_1^a and y_2^a are velocity-regulating and position-modulating terms, respectively. The output y_1^a is driven to a constant desired velocity v_d, while y_2^a is driven to a vector of desired trajectories, y_2^d. The trajectories y_2^d are represented using a Bézier polynomial with coefficients α and state-based timing variable $\tau(q)$.

According to Theorem 2 in [6], if there exist virtual constraints that yield an impact-invariant periodic orbit on the *partial zero dynamics* surface, then these outputs, when properly controlled on the exoskeleton, yield stable periodic walking. The orbit is *impact-invariant* if it returns to the partial zero dynamics surface $\mathcal{PZ}_\alpha$ after an impact event. To find the polynomials α that yield an impact-invariant periodic orbit on the reduced-order manifold, we formulate an optimization problem of the form:

$$\alpha^* = \underset{\alpha}{\operatorname{argmin}} \quad \mathcal{J}(\alpha), \tag{3}$$

$$\text{s.t.} \quad \Delta(\mathcal{S} \cap \mathcal{PZ}_\alpha) \subset \mathcal{PZ}_\alpha, \tag{4}$$

$$\mathcal{W}_i x \leq b_i, \tag{5}$$

where $\mathcal{J}(\alpha)$ is a user-determined cost, (4) is the impact invariance condition, (5) are other physical constraints, $\mathcal{S}$ is the *guard* defining the conditions under which impulsive behavior occurs, and Δ is the *reset map* governing the system's dynamical response to hitting the guard.

The optimization in (3)-(5) produces a gait that can be altered by varying the cost function $\mathcal{J}(\alpha)$ and/or adding physical constraints. In bipedal walking, this cost is frequently the mechanical cost of transport (COT) defined by Eqs. (17)-(18) in [25]. To create the desired motion, one must add physical constraints such as step length and foot height.

Gait Generation Applied to Lower-Body Exoskeletons. To translate gait generation to lower-body exoskeletons, one must choose the optimization cost function and physical constraints to obtain user-preferred gaits. While it is possible to optimize generated gaits for mechanical properties such as COT, there is currently no well-understood relationship between the parameters of the optimization problem and user preferences. Additionally, due to the time-consuming nature of gait generation—both the time required to tune the optimization problem's constraints and the time required to run the program—the issue of generating human-preferred dynamically-stable walking gaits remains largely unexplored.

Gait Library. It has become increasingly common to pre-compute a set of nominal walking gaits over a grid of various parameters [26]. These pre-computed gaits are combined to form a "gait library," through which gaits can be selected and executed immediately. For the purpose of exoskeleton walking, a gait library allows the operator to select a gait that is comfortable for the patient; however, it is not yet clear how to select an appropriate walking gait to optimize user comfort and preference. Thus, we consider learning from the user's preferences, as discussed below.

III. Preference-based Learning Algorithm

We leverage *preference-based learning* (e.g., does the user prefer gait A over gait B?) to determine the gait parameters most preferred by the user [13], [14], [16], [27]–[29], since preference feedback has been shown to be much more reliable than absolute feedback when learning from subjective human responses [13], [30]. Thus, our goal to personalize the exoskeleton's gait can be framed as *dueling bandit* [13], [14] and *coactive learning* [15], [16] problems.

Our work builds upon the Self-Sparring algorithm, a Bayesian dueling bandits approach that enjoys both competitive theoretical convergence guarantees and empirical performance [12]. Self-Sparring learns a Bayesian posterior over each action's utility to the user and draws multiple samples from the model's posterior to "duel" or "spar" via preference elicitation. The Self-Sparring algorithm iteratively: a) draws multiple samples from the posterior model of the actions' utilities; b) for each sampled model, executes the action with the highest sampled utility; c) queries for preference feedback between the executed actions; and d) updates the posterior according to the acquired preference data.

To collect more feedback beyond just one bit per preference, we also allow the user to suggest improvements during their trials. This approach resembles the *coactive learning* framework [15], [16], in which the user identifies an improved action as feedback to each presented action. Coactive learning has been applied to robot trajectory planning [31], [32], but has not, to our knowledge, yet been applied to robotic gait generation or in concert with preference learning.

The CoSpar Algorithm. To optimize an exoskeleton's gait within the gait library (Section II), we propose the CoSpar algorithm, a *mixed-initiative* learning approach [33], [34] which extends the Self-Sparring algorithm to incorporate coactive feedback. Similarly to Self-Sparring, CoSpar maintains a Bayesian *preference relation function* over the possible actions, which is fitted to observed preference feedback. CoSpar updates this model with user feedback and uses it to select actions for new trials and to elicit feedback. We first define the Bayesian preference model, and then detail the steps of Algorithm 1.

Modeling Utilities from Preference Data. We adopt the preference-based Gaussian process model of [35]. Let $\mathcal{A} \subset \mathbb{R}^d$ be the finite set of available actions with cardinality $A = |\mathcal{A}|$. At any point in time, CoSPAR has collected a preference feedback dataset $D = \{\boldsymbol{x}_{k1} \succ \boldsymbol{x}_{k2} \,|\, k = 1, \ldots, N\}$ consisting of N preferences, where $\boldsymbol{x}_{k1} \succ \boldsymbol{x}_{k2}$ indicates that the user prefers action $\boldsymbol{x}_{k1} \in \mathcal{A}$ to action $\boldsymbol{x}_{k2} \in \mathcal{A}$ in preference k. Furthermore, we assume each action $\boldsymbol{x}_i$ has a latent, underlying utility to the user, $f(\boldsymbol{x}_i)$. For finite action spaces, the utilities can be written in vector form: $\boldsymbol{f} := [f(\boldsymbol{x}_1), f(\boldsymbol{x}_2), \ldots, f(\boldsymbol{x}_A)]^T$. Given preference data D, we are interested in the posterior probability of $\boldsymbol{f}$:

$$P(\boldsymbol{f}|D) \propto P(D|\boldsymbol{f})P(\boldsymbol{f}). \tag{6}$$

We define a Gaussian prior over $\boldsymbol{f}$:

$$P(\boldsymbol{f}) = \frac{1}{(2\pi)^{A/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}\boldsymbol{f}^T\Sigma^{-1}\boldsymbol{f}\right),$$

where $\Sigma \in \mathbb{R}^{A\times A}$, $[\Sigma]_{ij} = \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)$, and $\mathcal{K}$ is a kernel, for instance the squared exponential kernel.

For computing the likelihood $P(D|\boldsymbol{f})$, we assume feedback may be corrupted by i.i.d. Gaussian noise: when presented with action $\boldsymbol{x}_i$, the user determines her internal valuation $y(\boldsymbol{x}_i) = f(\boldsymbol{x}_i) + \varepsilon_i$, where $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$. Then,

$$\begin{aligned} P(\boldsymbol{x}_{k1} \succ \boldsymbol{x}_{k2}|\boldsymbol{f}) &= P(y(\boldsymbol{x}_{k1}) > y(\boldsymbol{x}_{k2})|f(\boldsymbol{x}_{k1}), f(\boldsymbol{x}_{k2})) \\ &= \Phi\left[\frac{f(\boldsymbol{x}_{k1}) - f(\boldsymbol{x}_{k2})}{\sqrt{2}\sigma}\right], \end{aligned}$$

where Φ is the standard normal cumulative distribution function, and $y(\boldsymbol{x}_{kj}) = f(\boldsymbol{x}_{kj}) + \varepsilon_{kj}$, $j \in \{1, 2\}$. Thus, the full expression for the likelihood is:

$$P(D|\boldsymbol{f}) = \prod_{k=1}^{N} \Phi\left[\frac{f(\boldsymbol{x}_{k1}) - f(\boldsymbol{x}_{k2})}{\sqrt{2}\sigma}\right]. \tag{7}$$

The posterior $P(\boldsymbol{f}|D)$ can be estimated via the Laplace approximation as a multivariate Gaussian distribution; see [35] for details. Finally, in formulating the posterior, preferences can be weighted relatively to one another if some are thought to be noisier than others. This is accomplished by changing σ to σ_k in (7) to model differing values of the preference noise parameter among the data points, and is analogous to weighted Gaussian process regression [36].

The Learning Algorithm. Let (Σ, σ) represent the prior parameters of the Bayesian preference model, as outlined above. From these parameters, one obtains the prior mean and covariance, $(\boldsymbol{\mu}_0, \Sigma_0)$ (Line 3 in Alg. 1). In each iteration, CoSPAR updates the utility model (Line 21) via the Laplace approximation to the posterior in (6) to obtain $\mathcal{N}(\boldsymbol{\mu}_t, \Sigma_t)$.

To select actions in the t^{th} iteration (Lines 5-8), the algorithm first draws n samples from the posterior, $\mathcal{N}(\boldsymbol{\mu}_{t-1}, \Sigma_{t-1})$. Each of these is a utility function f_j, giving a utility value for each action in $\mathcal{A}$. The corresponding selected action is simply the one maximizing f_j (Line 7).

The n actions are executed (Line 9), and the user provides pairwise preference feedback between pairs of actions (the user can always state "no preference"). We extend Self-Sparring [12] to extract more preference comparisons from the available trials by assuming that the user can remember the b actions *preceding* the current n actions. The user thus provides preferences between any combination of the current n actions and the previous b actions. For instance, for $n = 1$, $b > 0$, one can interpret b as a buffer of previous trials that the user remembers. For $n = b = 1$, the user can report preferences between any pair of two consecutive trials, i.e., the user is asked, "Did you like this trial more or less than the last trial?" We expect that setting $n = 1$ while increasing b to as many trials as the user can accurately remember would minimize the trials required to reach a preferred gait. In Line 9, the pairwise preferences from iteration t form a matrix $R \in \mathbb{R}^{n\times(n+b)}$, where $r_{jk} \in \{0, 1, \emptyset\}$; the values 0 and 1 express preference information, while $\emptyset$ denotes the lack of a preference between the actions concerned.

Finally, the user can suggest improvements in the form of coactive feedback (Line 16). For example, the user could request a longer or shorter step length. In Line 16, $\emptyset$ indicates that no coactive feedback was provided. Otherwise, the user's suggestion is appended to the data D as preferred to the previously-tested action. In learning the model posterior, one can assign the coactive preferences a smaller weight relative to pairwise preferences via the input parameter $\beta > 0$.

Algorithm 1 CoSPAR

1: **procedure** CoSPAR($\mathcal{A}$ = action set, n = number of actions to select at each iteration, b = buffer size, (Σ, σ) = utility prior parameters, β = coactive feedback weight)
2: $D = \emptyset$ ▷ Initialize preference dataset
3: Obtain prior $(\boldsymbol{\mu}_0, \Sigma_0)$ over $\mathcal{A}$ from (Σ, σ)
4: **for all** $t = 1, 2, \ldots$ **do**
5: **for all** $j = 1, \ldots, n$ **do**
6: Sample utility function f_j from $(\boldsymbol{\mu}_{t-1}, \Sigma_{t-1})$
7: Select action $a_j(t) = \text{argmax}_{x\in\mathcal{A}} f_j(x)$
8: **end for**
9: Execute n actions; observe pairwise feedback matrix $R = \{r_{jk} \in \{0, 1, \emptyset\}\}_{n\times(n+b)}$
10: **for all** $j = 1, \ldots, n; k = 1, \ldots, n + b$ **do**
11: **if** $r_{jk} \neq \emptyset$ **then**
12: Append preference to dataset D
13: **end if**
14: **end for**
15: **for all** $j = 1, \ldots, n$ **do**
16: Obtain coactive feedback $\tilde{a}_j(t) \in \mathcal{A} \cup \emptyset$
17: **if** $\tilde{a}_j(t) \neq \emptyset$ **then**
18: Add to D: $\tilde{a}_j(t)$ preferred to $a_j(t)$, weight β
19: **end if**
20: **end for**
21: Update Bayesian posterior over D to obtain $(\boldsymbol{\mu}_t, \Sigma_t)$
22: **end for**
23: **end procedure**

IV. Simulation Results

The performance of CoSPAR is evaluated in two sets of simulations: (1) the compass-gait (CG) biped's COT,[1] and (2)

[1]Bayesian model's kernel: squared exponential with lengthscale = 0.025, signal variance = 0.0001, noise variance = 1e-8; preference noise (σ) = 0.01

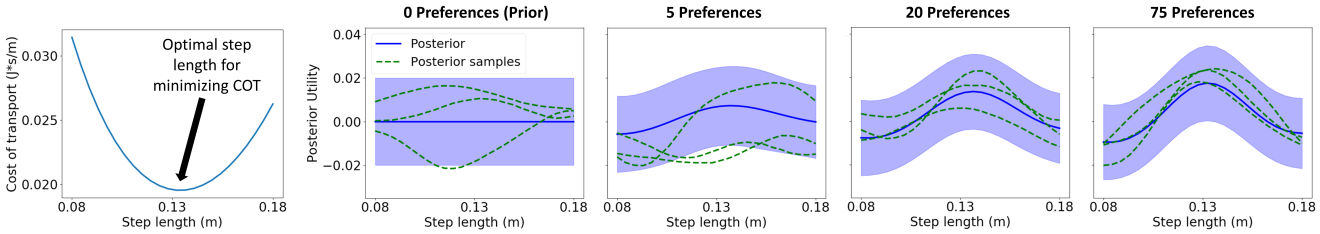
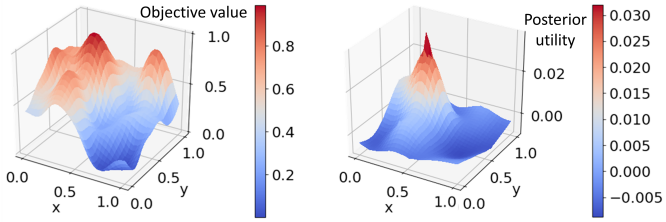


Fig. 2. Leftmost: COT for the CG biped at different step lengths and a fixed 0.2 m/s velocity. Remaining plots: posterior utility estimates of CoSpar ($n = 2$, $b = 0$; without coactive feedback) after varying iterations of learning (posterior mean +/- 2 standard deviations). The plots each show 3 posterior samples, which lie in the high-confidence region (mean +/- 2 stds) with high probability. The posterior utility estimate quickly converges to identifying the optimal action.



(a) Objective function (b) Model posterior

Fig. 3. a) Example synthetic 2D objective function. b) Utility model posterior learned after 150 iterations of CoSpar in simulation ($n = 1$; $b = 1$; coactive feedback). CoSpar prioritizes identifying and exploring the optimal region, rather than learning a globally-accurate utility landscape.

a set of synthetic optimization objective functions.[2] In both cases, CoSpar efficiently converges to the optimum.

Optimizing the Compass-Gait Biped's Cost-of-Transport. We first evaluate our approach with a simulated CG biped, optimizing its COT over the step length via preference feedback (Fig. 2). Preferences are determined by comparing COT values, calculated by simulating gaits for multiple step lengths, each at a fixed forward hip velocity of 0.2 m/s. These simulated gaits were synthesized via a single-point shooting partial hybrid zero dynamics method [24].

We use CoSpar to minimize the one-dimensional objective function in Fig. 2, using pairwise preferences obtained by comparing COT values for the different step lengths. Here, we use CoSpar with $n = 2$, $b = 0$, and without coactive feedback. Note that without a buffer or coactive feedback, CoSpar reduces to Self-Sparring [12]. At each iteration, two new samples are drawn from the Bayesian posterior, and the resultant two step lengths are compared to elicit a preference. Using the new preferences, CoSpar updates its posterior over the utility of each step length.

Fig. 2 depicts the evolution of the posterior preference model, where each iteration corresponds to a preference between two new trials. With more preference data, the posterior utility increasingly peaks at the point of lowest COT. These results suggest that CoSpar can efficiently identify high-utility actions from preference feedback alone.

[2]Kernel: squared exponential with lengthscale = $[0.15, 0.15]$, signal variance = 0.0001, noise variance = 1e-5; preference noise (σ) = 0.01

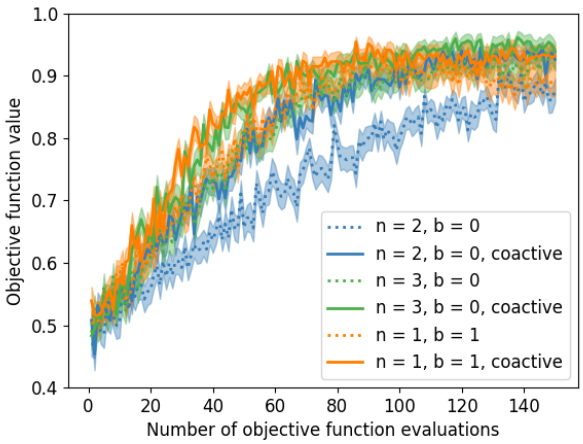


Fig. 4. CoSpar simulation results on 2D synthetic objective functions, comparing CoSpar with and without coactive feedback for three parameter settings n and b (see Algorithm 1). Mean +/- standard error of the objective values achieved over 100 repetitions. The maximal and minimal objective function values are normalized to 0 and 1. We see that coactive feedback always helps, and that $n = 2$, $b = 0$—which receives the fewest preferences—performs worst.

Optimizing Synthetic Two-Dimensional Functions. We next test CoSpar on synthetic 2D utility functions, such as the one shown in Fig. 3a. Each utility function was generated from a Gaussian process prior on a 30-by-30 grid. These experiments evaluate the potential to scale CoSpar to higher dimensions and the advantages of coactive feedback.

We compare three settings for CoSpar's (n, b) parameters: $(2, 0)$, $(3, 0)$, $(1, 1)$ as explained in Sec. III. For each setting—as well as with and without coactive feedback—we simulate CoSpar on each of the 100 random objective functions. In each case, the number of objective function evaluations, or experimental trials, was held constant at 150.

Coactive feedback is simulated using a 2nd-order differencing approximation of the objective function's gradient. If CoSpar selects a point at which both gradient components have magnitudes below their respective 50$^{\text{th}}$ percentile thresholds, then no coactive feedback is given. Otherwise, we consider the higher-magnitude gradient component, and depending on the highest threshold that it exceeds (50$^{\text{th}}$ or 75$^{\text{th}}$), simulate coactive feedback as either a 5% or 10% increase in the appropriate direction and dimension.

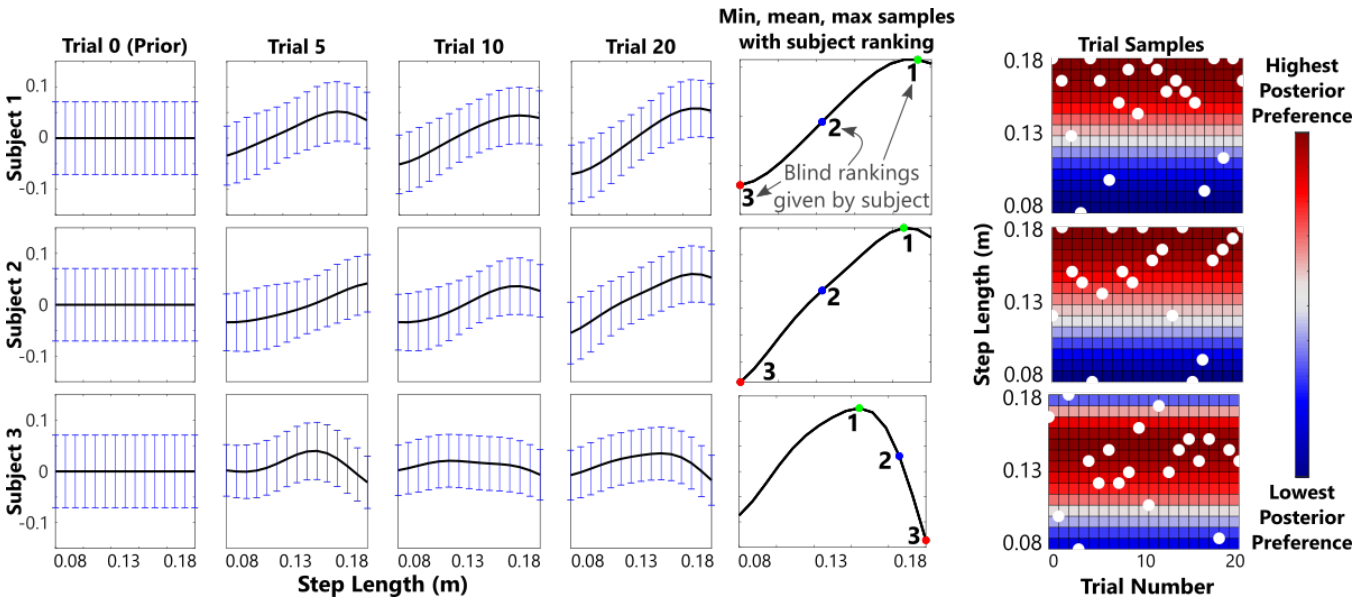


Fig. 5. Experimental results for optimizing step length with three subjects (one row per subject). Columns 1-4 illustrate the evolution of the preference model posterior (mean +/- standard deviation), shown at various trials. CoSpar converges to similar but distinct optimal gaits for different subjects. Column 5 depicts the subjects' blind ranking of the 3 gaits sampled after 20 trials. The rightmost column displays the experimental trials in chronological order, with the background depicting the posterior preference mean at each step length. CoSpar draws more samples in the region of higher posterior preference.

Fig. 4 shows the simulation results. In each case, the mixed-initiative simulations involving coactive feedback improve upon those with only preferences. Learning is slowest for $n = 2, b = 0$ (Fig. 4), since that case elicits the fewest preferences. Fig. 3b depicts the utility model's posterior mean for the objective function in Fig. 3a, learned in the simulation with $n = 1$, $b = 1$, and mixed-initiative feedback. In comparing Fig. 3b to Fig. 3a, we see that CoSpar learns a sharp peak around the optimum, as it is designed to converge to sampling preferred regions, rather than giving the user undesirable options by exploring elsewhere.

V. Human Subject Experiments

After its validation in simulation, CoSpar was deployed on a lower-body exoskeleton, Atalante, in two personalized gait optimization experiments with human subjects (video: [37]). Both experiments aimed to determine gait parameter values that maximize user comfort, as captured by preference and coactive feedback. The first experiment,[3] repeated for three able-bodied subjects, used CoSpar to determine the user's preferred step length, i.e., optimizing over a one-dimensional feature space. The second experiment[4] demonstrates CoSpar's effectiveness in two-dimensional feature spaces, and optimizes simultaneously over two different gait feature pairs. Importantly, CoSpar operates independently of the choice of gait features. The subjects' metabolic expenditure was also recorded via direct calorimetry as shown in Fig. 1, but this data was uninformative of user preferences, as users are not required to expend effort toward walking.

Learning Preferences between Step Lengths. In the first experiment, all three subjects walked inside the Atalante exoskeleton, with CoSpar selecting the gaits. We considered 15 equally-spaced step lengths between 0.08 and 0.18 meters, each with a precomputed gait from the gait library. Feature discretization was based on users' ability to distinguish nearby values. The users decided when to end each trial, so as to be comfortable providing feedback. Since users have difficulty remembering more than two trials at once, we used CoSpar with $n = 1$ and $b = 1$, which corresponds to asking the user to compare each current trial with the preceding one. Additionally, we query the user for coactive feedback: after each trial, the user can suggest a longer or shorter step length ($\pm$20% of the range), a *slightly* longer or shorter step length ($\pm$10%), or no feedback.

Each participant completed 20 gait trials, providing preference and coactive feedback after each trial. Fig. 5 illustrates the posterior's evolution over the experiment. After only five exoskeleton trials, CoSpar was already able to identify a relatively-compact preferred step length subregion. After the 20 trials, three points along the utility model's posterior mean were selected: the maximum, mean, and minimum. The user walked in the exoskeleton with each of these step lengths in a randomized ordering, and gave a blind ranking of the three, as shown in Fig. 5. For each subject, the blind rankings match the preference posterior obtained by CoSpar, indicating effective learning of individual user preferences.

Learning Preferences over Multiple Features. We further demonstrate CoSpar's practicality to personalize over mul-

[3]Kernel: squared exponential with lengthscale = 0.03, signal variance = 0.005, noise variance = 1e-7; preference noise (σ) = 0.02

[4]Same parameters as in [3] except for step duration lengthscale = 0.08 and step width lengthscale = 0.03

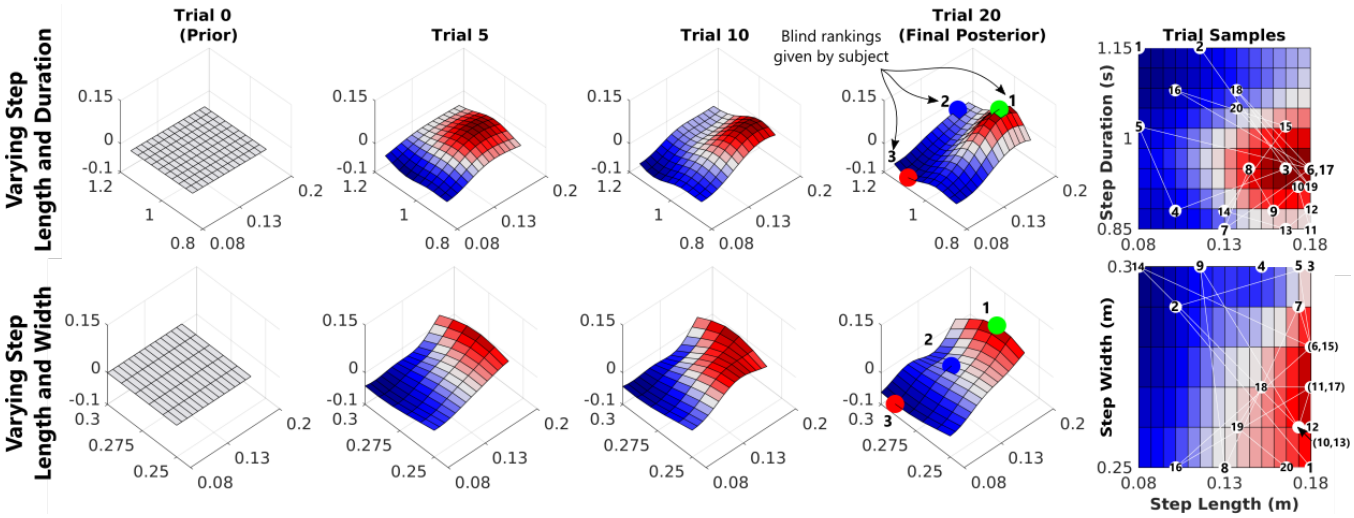


Fig. 6. Experimental results from two-dimensional feature spaces (top row: step length and duration; bottom row: step length and width). Columns 1-4 illustrate the evolution of the preference model's posterior mean. Column 4 also shows the subject's blind ranking of the 3 gaits sampled after 20 trials. Column 5 depicts the experimental trials in chronological order, with the background as in Fig. 5. CoSpar draws more samples in the region of higher posterior preference.

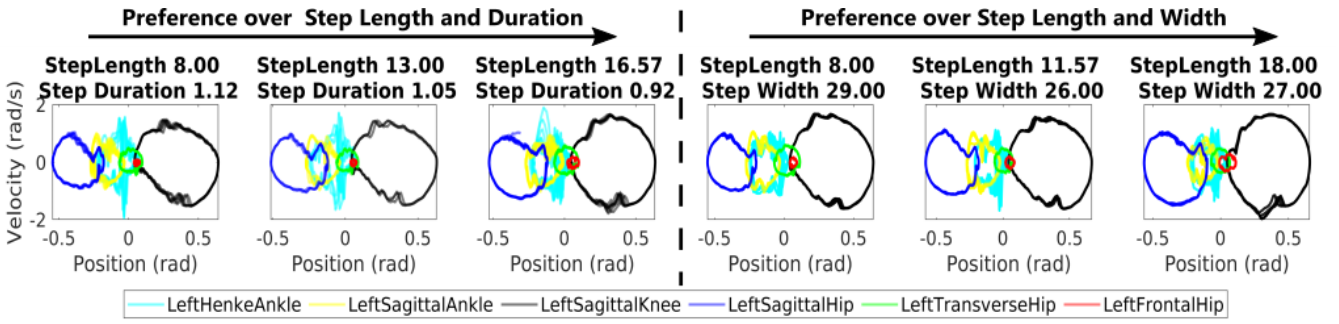


Fig. 7. Experimental phase diagrams of the left leg joints over 10 seconds of walking. The gaits shown correspond to the maximum, mean, and minimum preference posterior values for both of subject 1's 2D experiments. For instance, subject 1 preferred gaits with longer step lengths, as shown by the larger range in sagittal hip angles in the phase diagram.

tiple features, by optimizing over two different feature pairs: 1) step length and step duration and 2) step length and step width. The protocol of the 1D experiment was repeated for subject 1, with step lengths discretized as before, step duration discretized into 10 equally-spaced values between 0.85 and 1.15 seconds (with 10% and 20% modifications under coactive feedback), and step width into 6 values between 0.25 and 0.30 meters (20% and 40%). After each trial, the user was queried for both a pairwise preference and coactive feedback. Fig. 6 shows the results for both feature spaces. The estimated preference values were consistent with a 3-sample blind ranking evaluation, suggesting that CoSpar successfully identified user-preferred parameters. Fig. 7 displays phase diagrams of the gaits with minimum, mean, and maximum posterior utility values to illustrate the difference between preferred and non-preferred gaits.

VI. CONCLUSIONS

This work develops and demonstrates (video: [37]) the CoSpar interactive learning framework for optimizing gaits with respect to user comfort, using human preferences as feedback. We demonstrate the algorithm in simulation, showing that it efficiently learns to select optimal actions. We next apply CoSpar in a user study with the Atalante lower-body exoskeleton, demonstrating the first application of preference-based learning for optimizing dynamic crutchless walking. CoSpar successfully models the users' preferences, identifying compact subregions of preferred gaits.

In the future, we plan to apply CoSpar toward optimizing over larger sets of gait parameters; this will likely require integrating the algorithm with techniques for learning over high-dimensional feature spaces [38]. The method could also be extended beyond working with precomputed gait libraries to generating entirely new gaits or controller designs (e.g., via preference-based reinforcement learning [29], [39]).

ACKNOWLEDGMENTS

The authors would like to thank the volunteers who participated in the experiments, as well as the entire Wandercraft team that designed Atalante and continues to provide technical support for this project.

REFERENCES

[1] Wandercraft, http://www.wandercraft.eu/, Last accessed on 2017-09-15.

[2] A. M. Dollar and H. Herr, "Active orthoses for the lower-limbs: Challenges and state of the art," in *2007 IEEE 10th International Conference on Rehabilitation Robotics*. IEEE, 2007, pp. 968–977.

[3] O. Harib, A. Hereid, A. Agrawal, T. Gurriet, S. Finet, G. Boeris, A. Duburcq, M. E. Mungai, M. Masselin, A. D. Ames *et al.*, "Feedback control of an exoskeleton for paraplegics: Toward robustly stable, hands-free dynamic walking," *IEEE Control Systems Magazine*, vol. 38, no. 6, pp. 61–87, 2018.

[4] T. Gurriet, S. Finet, G. Boeris, A. Duburcq, A. Hereid, O. Harib, M. Masselin, J. Grizzle, and A. D. Ames, "Towards restoring locomotion for paraplegics: Realizing dynamically stable walking on exoskeletons," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2804–2811.

[5] A. Agrawal, O. Harib, A. Hereid, S. Finet, M. Masselin, L. Praly, A. D. Ames, K. Sreenath, and J. W. Grizzle, "First steps towards translating HZD control of bipedal robots to decentralized control of exoskeletons," *IEEE Access*, vol. 5, pp. 9919–9934, 2017.

[6] A. D. Ames, "Human-inspired control of bipedal walking robots," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1115–1130, 2014.

[7] J. Zhang, P. Fiers, K. A. Witte, R. W. Jackson, K. L. Poggensee, C. G. Atkeson, and S. H. Collins, "Human-in-the-loop optimization of exoskeleton assistance during walking," *Science*, vol. 356, no. 6344, pp. 1280–1284, 2017.

[8] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané, "Concrete problems in AI safety," *arXiv preprint arXiv:1606.06565*, 2016.

[9] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 469–483, 2009.

[10] C. Basu, Q. Yang, D. Hungerman, M. Sinahal, and A. D. Dragan, "Do you want your autonomous car to drive like you?" in *2017 12th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2017, pp. 417–425.

[11] O. Chapelle, T. Joachims, F. Radlinski, and Y. Yue, "Large-scale validation and analysis of interleaved search evaluation," *ACM Transactions on Information Systems (TOIS)*, vol. 30, no. 1, p. 6, 2012.

[12] Y. Sui, V. Zhuang, J. W. Burdick, and Y. Yue, "Multi-dueling bandits with dependent arms," in *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2017.

[13] Y. Sui, M. Zoghi, K. Hofmann, and Y. Yue, "Advancements in dueling bandits," in *IJCAI*, 2018, pp. 5502–5510.

[14] Y. Yue, J. Broder, R. Kleinberg, and T. Joachims, "The k-armed dueling bandits problem," *Journal of Computer and System Sciences*, vol. 78, no. 5, pp. 1538–1556, 2012.

[15] P. Shivaswamy and T. Joachims, "Online structured prediction via coactive learning," in *Proceedings of the 29th International Conference on Machine Learning*. Omnipress, 2012, pp. 59–66.

[16] ——, "Coactive learning," *Journal of Artificial Intelligence Research*, vol. 53, pp. 1–40, 2015.

[17] E. Bionics, https://eksobionics.com/, Last accessed on 2019-09-14.

[18] ReWalk, https://rewalk.com/, Last accessed on 2019-09-14.

[19] Indego, http://www.indego.com/indego/en/home, Last accessed on 2019-09-14.

[20] R. Bionics, https://www.rexbionics.com/, Last accessed on 2019-09-14.

[21] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, "Hybrid zero dynamics of planar biped walkers," *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 42–56, 2003.

[22] D. D. Bainov and P. S. Simeonov, *Systems with impulse effect: Stability, theory and applications*. Wiley, 1989.

[23] H. Ye, A. N. Michel, and L. Hou, "Stability theory for hybrid dynamical systems," *IEEE transactions on Automatic Control*, vol. 43, no. 4, pp. 461–474, 1998.

[24] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback control of dynamic bipedal robot locomotion*. CRC press, 2018.

[25] J. P. Reher, A. Hereid, S. Kolathaya, C. M. Hubicki, and A. D. Ames, "Algorithmic foundations of realizing multi-contact locomotion on the humanoid robot DURUS," in *Twelfth International Workshop on Algorithmic Foundations on Robotics*, 2016.

[26] X. Da, R. Hartley, and J. W. Grizzle, "Supervised learning for stabilizing underactuated bipedal robot locomotion, with outdoor experiments on the wave field," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 3476–3483.

[27] J. Fürnkranz and E. Hüllermeier, *Preference learning*. Springer, 2010.

[28] D. Sadigh, A. D. Dragan, S. Sastry, and S. A. Seshia, "Active preference-based learning of reward functions," in *Robotics: Science and Systems (RSS)*, 2017.

[29] J. Fürnkranz, E. Hüllermeier, W. Cheng, and S.-H. Park, "Preference-based reinforcement learning: A formal framework and a policy iteration algorithm," *Machine Learning*, vol. 89, no. 1-2, pp. 123–156, 2012.

[30] T. Joachims, L. A. Granka, B. Pan, H. Hembrooke, and G. Gay, "Accurately interpreting clickthrough data as implicit feedback," in *SIGIR*, vol. 5, 2005, pp. 154–161.

[31] A. Jain, S. Sharma, T. Joachims, and A. Saxena, "Learning preferences for manipulation tasks from online coactive feedback," *The International Journal of Robotics Research*, vol. 34, no. 10, pp. 1296–1313, 2015.

[32] T. Somers and G. A. Hollinger, "Human–robot planning and learning for marine data collection," *Autonomous Robots*, vol. 40, no. 7, pp. 1123–1137, 2016.

[33] S. A. Wolfman, T. Lau, P. Domingos, P. Domingos, and D. S. Weld, "Mixed initiative interfaces for learning tasks: SMARTedit talks back," in *Proceedings of the 6th International Conference on Intelligent User Interfaces*. ACM, 2001, pp. 167–174.

[34] J. C. Lester, B. A. Stone, and G. D. Stelling, "Lifelike pedagogical agents for mixed-initiative problem solving in constructivist learning environments," *User Modeling and User-Adapted Interaction*, vol. 9, no. 1-2, pp. 1–44, 1999.

[35] W. Chu and Z. Ghahramani, "Preference learning with Gaussian processes," in *Proceedings of the 22nd International Conference on Machine Learning*. ACM, 2005, pp. 137–144.

[36] X. Hong, L. Ren, L. Chen, F. Guo, Y. Ding, and B. Huang, "A weighted Gaussian process regression for multivariate modelling," in *2017 6th International Symposium on Advanced Control of Industrial Processes (AdCONIP)*. IEEE, 2017, pp. 195–200.

[37] "Video of the experimental results." https://youtu.be/-27sHXsvONE.

[38] J. Kirschner, M. Mutny, N. Hiller, R. Ischebeck, and A. Krause, "Adaptive and safe Bayesian optimization in high dimensions via one-dimensional subspaces," in *International Conference on Machine Learning*, 2019, pp. 3429–3438.

[39] E. R. Novoseller, Y. Sui, Y. Yue, and J. W. Burdick, "Dueling posterior sampling for preference-based reinforcement learning," *arXiv preprint arXiv:1908.01289*, 2019.