Towards Safe and Adaptive Human Limb Manipulation

Eric Hu*, Rajat Kumar Jenamani*, Tom Silver,
Skyler Valdez, Carly Jiang, Nikki Hart, Tapomayukh Bhattacharjee
Cornell University

1 Introduction

The World Health Organization reports nearly 190 million individuals have motor impairments [1], necessitating assistance with activities of daily living (ADLs) such as dressing, bathing, and transferring. This constant need for help severely impacts their mental health, intensifying feelings of dependence and shame [2, 3]. Moreover, the acute shortage of caregivers places a heavy demand on those available [4, 5] and pushes care recipients toward institutionalization, limiting personal freedom and increasing isolation. As society ages globally, these challenges will only escalate.

Robotic caregivers have the potential to physically assist with ADLs [6, 7] and enable people with mobility limitations to enhance their independence [8, 9], while also reducing the physical workload on caregivers. A critical aspect of many daily activities involves the safe manipulation of human limbs; for instance, dressing requires lifting a limb to slide a sleeve on, bathing involves raising a limb to wash underneath, and after moving from a bed to a wheelchair, a caregiver must position the recipient's limb on the wheelchair's armrest. However, almost all prior works in these domains assume that humans without active range of motion are static and inarticulate [10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20].

Limb manipulation for care recipients is extremely challenging due to the diverse mobility limitations they may experience. Some individuals may have restricted passive range of motion in specific limb areas, commonly seen in post-stroke recovery or joint arthrofibrosis. Human joints also feature interdependent joint limits, complicating modeling efforts. Conditions like spasticity from cerebral palsy or stroke can cause variable resistance during movement, while others, such as those with muscular dystrophy, might experience hypertonia, leading to counteracting joint forces. Although prior knowledge of muscle tone is possible, it varies significantly even for the same individual due to mental state and environmental factors. Therefore, a limb manipulation system needs to be personalized to each user's functionality and designed to be closed-loop, allowing it to estimate changes in muscle tone in real time and adapt to them.

In this paper, we present a closed-loop approach to limb manipulation. We first introduce a mathematical formulation of the limb manipulation setting, where a robot rigidly grasps a human limb, and derive an analytical dynamics model for this interaction. We then instantiate **LimbManipulationBench**, a simulation benchmark that models manipulation of all four human limbs across two scenarios (wheelchair and hospital bed), with 100 distinct start–goal configuration pairs per setting. We validate the correctness of our analytical model by instantiating the same setup in PyBullet and comparing predicted trajectories against physics-based simulation rollouts, observing exact agreement. Using this analytical model, we show that a naïve sampling-based model predictive control (MPC) approach is prohibitively slow for real-time deployment. To address this limitation, we propose **WOVEN-LM**: $\underline{\text{W}}$hite-b$\underline{\text{o}}$x $\underline{\text{V}}$alidation Int$\underline{\text{e}}$rleaved with Fast $\underline{\text{N}}$eural Predictions for $\underline{\text{L}}$imb $\underline{\text{M}}$anipulation. WOVEN-LM leverages a fast black-box neural dynamics model to efficiently rank sampled trajectories using batched rollouts, while lazily validating candidate best trajectories against the white-box analytical model before execution. This design enables real-time performance while maintaining verifiable safety. Finally, we conduct experiments in one of the LimbManipulationBench environments to demonstrate the feasibility and effectiveness of WOVEN-LM.

Our main contributions are:

- A mathematical formulation for safe and adaptive robotic limb repositioning for individuals with mobility limitations, incorporating human joint dynamics and muscle tone, and informed by stakeholder input.
- **LimbManipulationBench**: A simulation benchmark for human limb manipulation covering all four limbs across two scenarios (wheelchair and hospital bed), with 100 distinct start–goal configuration pairs per setting.
- **WOVEN-LM**: A real-time, closed-loop limb manipulation approach that interleaves fast neural black-box dynamics models with slower white-box analytical models within a model predictive control framework.
- Experiments in LimbManipulationBench demonstrating the feasibility and effectiveness of the proposed MPC framework.

2 Related Work

Manipulation of Human Limbs. Using robotic platforms for human-body manipulation presents significant challenges, primarily due to the critical need for safety. Most prior works assume some active range of motion in humans and focus on developing exoskeletons [21, 22, 23, 24, 25, 26] to assist human motion, with a primary goal of minimizing the 'metabolic cost' of movement to ensure energy efficiency. In contrast, our work aims to enable a general-purpose robotic manipulator to reposition human limbs. Within this domain, various works explore gripper design for securely grasping human forearms [27, 28], which complements our focus on ensuring safe control once a stable grasp is achieved. Wang et al. [29] explore limb repositioning in a user-supervised setting using an interface that allows for selecting desired limb movements [30]. However, their approach relies on predefined movements, whereas our goal is to generalize to various start and goal positions.

More recently, some completely autonomous methods have also been proposed [31, 32]. Chow et al. [31] propose a model-predictive approach to limb manipulation in simulation, but they assume the robot embodiment to be two end effectors (modeled as flat plates) that have unrestricted movement in space, and thus do not reason about the mechanics of the robot's movements. More recently, Peiros et al. [32] model human manipulation as a constrained motion planning problem, and demonstrate execution feasibility on a real human. However, none of these works consider severe mobility limitations such as muscle spasms, muscle tone and limited passive range of motion which demand closed-loop control with adaptive cost functions.

Articulated Object Manipulation. Some research in articulated object manipulation focuses on directly learning policies that deduce action sequences from raw sensory input [33, 34]. Different from these black-box approaches, others propose perception methods for constructing an articulation model, such as joint parameter prediction [35] or dynamic property estimation [36, 37, 38]. With an established model of an object, recent approaches have utilized sampling-based motion planners [39] or model-predictive control [40] for action generation. However, these techniques generally assume uniform properties for the articulated object, a presumption that does not hold in cases like human limb dynamics where variability in muscle tone or spasms might occur. In our work, we formulate limb repositioning as a model-based control problem to ensure safety. We assume certain components of the object model, such as the joint-link configuration, are available apriori. Meanwhile, dynamic aspects like muscle tone require online adaptation to accurately reflect real-time conditions.

3 Problem Formulation

In this work, we focus on the task of limb manipulation of human limbs. In this section, we first model the world of our limb manipulation experiments, and then we describe the limb manipulation task and the characteristics of a robot trajectory we would deem as successfully completing the task.

3.1 World Modeling

We model the human arm H as a serial chain of three rigid bodies: the upper arm, the forearm, and the wrist. The upper arm is attached to a fixed torso through a 3-Degree-of-Freedom (3-DoF) spherical shoulder joint. The upper arm and forearm are connected by a 1-DoF revolute elbow joint, and the forearm is connected to the wrist through a 2-DoF revolute wrist joint, resulting in a total of $j_H = 6$ joints and $l_H = 3$ links. The human is seated in a wheelchair W next to a 7-DoF Franka Emika Panda robot arm R, which has $j_R = 6$ joints and $l_R = 6$ links (we fix one of the joints of the panda to remove kinematic redundancy).

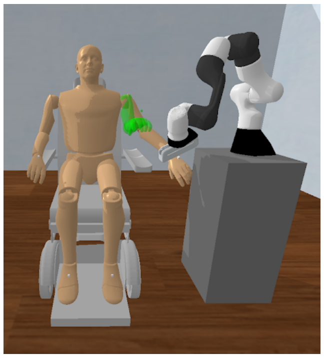

Figure 1: We model limb manipulation as a robot rigidly grasping a human limb at the wrist or ankle. The target limb configuration is shown in green.

We account for muscle tone in our model of the human arm. Muscle tone is the passive resistance to movement in a joint and is dependent on joint position and velocity. Accordingly, we model it as torques on each joint, where the torque at joint j is represented as a spring-damper system:

$$\tau_j(\theta, \dot{\theta}) = \max(0, -(b_j - k_j|\theta - m_j|) - c_j\dot{\theta})$$

where k_j is the spring constant, m_j is the spring center, b_j is the maximum torque of the muscle tone being modeled, and c_j is the damping coefficient, all of which are specified for each joint j. Muscle tone may vary over long periods of time, so we start with an initial estimation and converge on a more accurate model online. Lastly, we assume that the user's muscle tone does not change during the short duration of limb manipulation.

3.2 Limb Manipulation Task

We define the limb manipulation task as finding a *feasible* trajectory $\zeta^{0\to T}$ that moves a human limb from an initial configuration $q_H^{(0)}$ to a target configuration $q_H^{(T)}$ at time T. At any time $t \in [0, T]$, the trajectory state $\zeta^{(t)} = (R^{(t)}, H^{(t)})$ consists of the robot state $R^{(t)}$ and the human state $H^{(t)}$. Specifically, $R^{(t)}$ includes joint torques $\tau_R^{(t)}$, joint configuration $q_R^{(t)}$, joint velocity $\dot{q}_R^{(t)}$, and joint acceleration $\ddot{q}_R^{(t)}$. $H^{(t)}$ consists of an analogous tuple $(\tau_H^{(t)}, q_H^{(t)}, \dot{q}_H^{(t)}, \ddot{q}_H^{(t)})$.

For a trajectory to be *feasible*, $\zeta^{(t)}$ must satisfy the following kinematic and dynamic constraints:

Kinematic Constraints: We require that a trajectory ζ describes no collisions, does not violate joint limits for either the human arm or robot.

Collisions: The robot and human should not collide with each other, themselves (no self-collisions), or objects in the environment.

Joint Limits: At any time $t \in [0, T]$, we enforce human joint limits such that $q_H^{(t)} \in Q_H$, where $Q_H \subseteq \mathbb{R}^{j_H}$ denotes all valid, or physically attainable, human limb configurations.

Dynamic Constraints: To ensure safety and comfort, we impose torque constraints on the human limb. We limit the torque applied by the robot such that $\tau_r < \tau_{r_{max}}$.

We consider a vector τ_h of torques on each human joint, and decompose it into torques caused by muscle tone, gravity, and the robot:

$$\tau_h = \tau_t + \tau_g + \tau_r.$$

A trajectory is considered *feasible* if the cumulative torque magnitude on the human limb remains below a safety threshold $\tau_{h_{safe}}$ at all times:

$$|\tau_h| < \tau_{h_{safe}}, \quad \forall t \in [0, T].$$

Among all feasible trajectories, we define the *optimal* trajectory as the one that minimizes the time to reach the goal configuration:

$$\min_{\zeta^{0\to T}} ; T.$$

4 Method

We propose **WOVEN-LM**: **W**hite-b**o**x **V**alidation Int**e**rleaved with Fast **N**eural Predictions for **L**imb **M**anipulation. Our method consists of four core components:
(1) deriving an analytical dynamics model for limb manipulation,
(2) learning a neural dynamics model of coupled human–robot motion,
(3) interleaving fast neural predictions with slow white-box verification for trajectory planning, and
(4) (future work) actively refining a user-specific muscle-tone model through safe exploration.

4.1 Analytical Dynamics Model for Limb Manipulation

Given the robot joint positions and velocities $(q_r, \dot{q}_r)$, the human joint positions and velocities $(q_h, \dot{q}_h)$, and the applied robot joint torques $\boldsymbol{\tau}_r$, we model the limb manipulation setup as a coupled human-robot system under a rigid grasp.

We begin with the Euler-Lagrange dynamics of the robot arm, which relate the robot joint accelerations $\ddot{q}_r$ to the applied torques and interaction forces:

$$\mathbf{M}_r(q_r)\ddot{q}_r + \mathbf{C}_r(q_r, \dot{q}_r)\dot{q}_r + \mathbf{g}_r(q_r) + \boldsymbol{\tau}_{r,\text{ext}} = \boldsymbol{\tau}_r \tag{1}$$

Here, $\mathbf{M}_r(\cdot)$ denotes the robot mass (inertia) matrix, $\mathbf{C}_r(\cdot, \cdot)$ captures Coriolis effects, $\mathbf{g}_r(\cdot)$ represents gravitational torques, and $\boldsymbol{\tau}_{r,\text{ext}}$ denotes external joint torques arising from contact.

Next, we model the human arm dynamics under the assumption that the human arm is not actively actuated (no muscle tone) and that all joint torques arise from interaction forces at the grasp:

$$\mathbf{M}_h(q_h)\ddot{q}_h + \mathbf{C}_h(q_h, \dot{q}_h)\dot{q}_h + \mathbf{g}_h(q_h) = \boldsymbol{\tau}_{h,\text{ext}} \tag{2}$$

Here, $\mathbf{M}_h(\cdot)$, $\mathbf{C}_h(\cdot, \cdot)$, and $\mathbf{g}_h(\cdot)$ denote the corresponding mass, Coriolis, and gravity terms for the human arm, and $\boldsymbol{\tau}_{h,\text{ext}}$ represents external torques induced by contact.

The coupling between the human and robot arms is expressed through the grasp constraint, which maps the contact wrench at the grasp point to joint-space torques on each arm:

$$\boldsymbol{\tau}_{h,\text{ext}} = \mathbf{J}_h(q_h)^\top \mathbf{F}_c, \qquad \boldsymbol{\tau}_{r,\text{ext}} = -\mathbf{J}_r(q_r)^\top \mathbf{F}_c \tag{3}$$

Here, $\mathbf{J}_h(\cdot)$ and $\mathbf{J}_r(\cdot)$ are the human and robot end-effector Jacobians, respectively, and $\mathbf{F}_c$ denotes the contact wrench.

Finally, we impose a rigid grasp assumption, which enforces equality of the human and robot end-effector velocities at the contact point:

$$\mathbf{J}_h(q_h)\,\dot{q}_h = \mathbf{J}_r(q_r)\,\dot{q}_r \tag{4}$$

Together, equations (1)–(4) define a coupled dynamical system that can be solved to recover $(\ddot{q}_h,\ \ddot{q}_r,\ \mathbf{F}_c)$ given $(q_h,\ \dot{q}_h,\ q_r,\ \dot{q}_r,\ \boldsymbol{\tau}_r)$; for examples we can solve:

$$\begin{aligned}
\ddot{q}_{r(t)} = {} & \Big(\mathbf{J}_{r(t)}^\top \big(\mathbf{J}_{h(t)}^\top\big)^{\#}\big(\mathbf{M}_{h(t)} + \mathbf{C}_{h(t)}\,\Delta t\big)\mathbf{J}_{h(t)}^{\#}\mathbf{J}_{r(t)} + \mathbf{M}_{r(t)} + \mathbf{C}_{r(t)}\,\Delta t\Big)^{\#} \\
& \times \Big(\boldsymbol{\tau}_{r(t)} - \mathbf{J}_{r(t)}^\top \big(\mathbf{J}_{h(t)}^\top\big)^{\#}\Big(\big(\mathbf{M}_{h(t)}\,\Delta t^{-1} + \mathbf{C}_{h(t)}\big)\mathbf{J}_{h(t)}^{\#}\mathbf{J}_{r(t)}\,\dot{q}_{r(t-1)} \\
& \qquad - \mathbf{M}_{h(t)}\,\Delta t^{-1}\dot{q}_{h(t-1)} + \mathbf{g}_{h(t)}\Big) - \mathbf{C}_{r(t)}\,\dot{q}_{r(t-1)} - \mathbf{g}_{r(t)}\Big)
\end{aligned} \tag{5}$$

However, computing this solution is too slow for use as a dynamics model within a model-predictive control framework, motivating the use of a learned, faster neural dynamics model.

4.2 Learning a Neural Dynamics Model

We train a neural regressor f_θ to predict coupled human–robot joint accelerations,

$$(\ddot{q}_h,\ \ddot{q}_r) = f_\theta(q_h, \dot{q}_h, q_r, \dot{q}_r, \boldsymbol{\tau}_r, \phi),$$

where ϕ optionally encodes limb-specific properties such as link lengths, masses, and inertias. The model is trained entirely on analytically generated data that captures the coupled human–robot dynamics under rigid grasp constraints.

To generate training data, we synthesize state–torque–acceleration tuples using the analytical dynamics model. We first sample human joint positions and velocities, then solve for compatible robot joint configurations under the grasp constraint. Specifically, we compute robot joint positions via inverse kinematics and robot joint velocities using the grasp-induced velocity relationship. Samples that violate robot joint limits are discarded and resampled. We then sample a random robot joint torque and compute the corresponding joint accelerations using the analytical forward dynamics model. The data generation procedure is summarized below:

```
$(q_h, \dot{q}_h) \leftarrow$ sample human joint positions and velocities
$q_r \leftarrow \mathrm{IK}(q_h)$
$\dot{q}_r \leftarrow \mathbf{J}_r^{\#}(\mathbf{J}_h \dot{q}_h)$
if $q_r$ violates joint limits: resample
$\boldsymbol{\tau}_r \leftarrow$ sample random robot torque
$(\ddot{q}_r, \ddot{q}_h) \leftarrow \mathrm{AnalyticalDynamicsModel}(q_r, \dot{q}_r, q_h, \dot{q}_h, \boldsymbol{\tau}_r)$
dataset $\leftarrow \big([\boldsymbol{\tau}_r, q_r, \dot{q}_r, q_h, \dot{q}_h], [\ddot{q}_r, \ddot{q}_h]\big)$
```

We generate a synthetic dataset using the analytical model and train a multi-layer perceptron via supervised regression. Although the learned model is computationally efficient and supports batched inference, it offers no formal guarantees on prediction accuracy, motivating the need for verification.

4.3 Learning–Verification Synergy

At execution time, f_θ is embedded within a sampling-based model-predictive control (MPC) framework. The planner samples candidate torque trajectories, rolls them out using f_θ to obtain predicted future states, and ranks them via a cost function. Our *white-box* analytical dynamics model then acts as a verifier and rejects trajectories that violate analytic safety constraints (joint limits or torque thresholds). If the top candidate is infeasible, the next-best is tested. This interleaved strategy enables fast, data-driven planning while preserving verifiability and safety.

4.4 Future Work: Active Learning for Muscle-Tone Modeling

Muscle tone governs the counter-torques $\tau_{h,\text{counter}}$ exerted by the human limb in response to robot-applied torques. Because these torques vary across individuals and can change over time, we introduce a separate estimator g_ψ that predicts the expected counter-torque given the current limb configuration and motion history. The predicted $\widehat{\tau}_{h,\text{counter}} = g_\psi(q_h, \dot{q}_h, q_r, \dot{q}_r)$ is then provided as an input to the neural dynamics model f_θ during planning.

We frame the online refinement of g_ψ as an *active learning* problem integrated within the MPC loop. In addition to standard task and safety objectives, the MPC cost includes an auxiliary term that encourages trajectories expected to reduce uncertainty in the torque estimation, guiding the robot to collect data that is both informative and safe. After each control cycle, the true counter-torques measured from interaction are used to update g_ψ via a supervised loss. This design enables the robot to continuously personalize its muscle-tone model for safe, adaptive control.

Due to time constraints, we leave this component for future work.

5 Experiments

5.1 Experimental Setup: LimbManipulationBench

To evaluate limb manipulation across diverse scenarios, we propose **LimbManipulationBench**, a PyBullet-based simulation benchmark consisting of eight environments. Each environment features

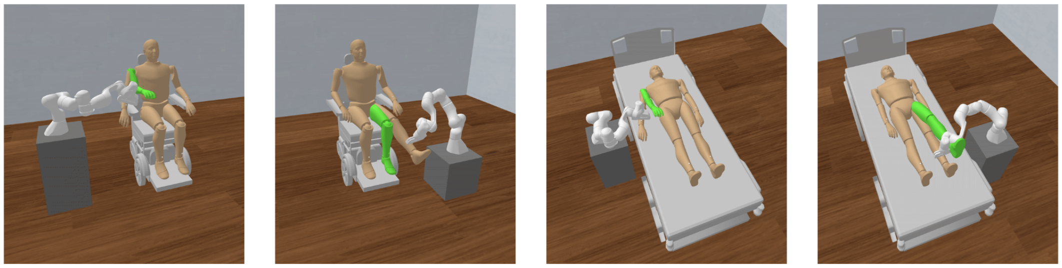

Figure 2: **LimbManipulationBench.** We propose a PyBullet-based benchmark with eight environments where a Franka Emika Panda grasps a human avatar's limb across four limbs and two settings: wheelchair and hospital bed. We instantiate 100 start-goal pairs in each of the eight environments.

a Franka Emika Panda robot grasping a human avatar's limb in one of two settings: a wheelchair or a hospital bed, and across all four human limbs. Figure 2 illustrates representative environments from the benchmark. For each environment, we evaluate 100 distinct start–goal configurations of the human arm. Unless otherwise stated, the remainder of this section focuses on the wheelchair-left-arm env, in which the human is seated in a wheelchair and the robot grasps the left arm (Figure 1).

5.2 Validating our Analytical Dynamics Model

To validate the correctness of our analytical dynamics model, we apply random joint torques to the robot arm for 2,000 timesteps in the wheelchair–left-arm environment and compare the resulting robot and human joint positions and velocities against those obtained from PyBullet. Torques are sampled uniformly within the robot's actuation limits. The analytical model is evaluated in an open-loop manner: it is initialized with the PyBullet state at timestep zero and subsequently rolled out without feedback, allowing errors to accumulate over time. As shown in Figure 3, the predicted robot and human joint positions and velocities from the analytical model match the PyBullet simulation exactly over all 2,000 timesteps, validating the correctness of our derivation.

5.3 Training a Neural Dynamics Model

We collect a dataset of 20 million datapoints, with candidate human joint configurations sampled within ±30° for each joint angle of the nominal arm pose shown in Figure 1. Data generation required approximately 10 hours using 64 parallel worker threads on a 64-core CPU cluster node.

We train a 4-layer multilayer perceptron with approximately 8.4 million parameters, using ReLU activations and linear input normalization. The network takes 30-dimensional inputs (robot joint positions, velocities, torques, and human joint positions and velocities) and outputs 12-dimensional

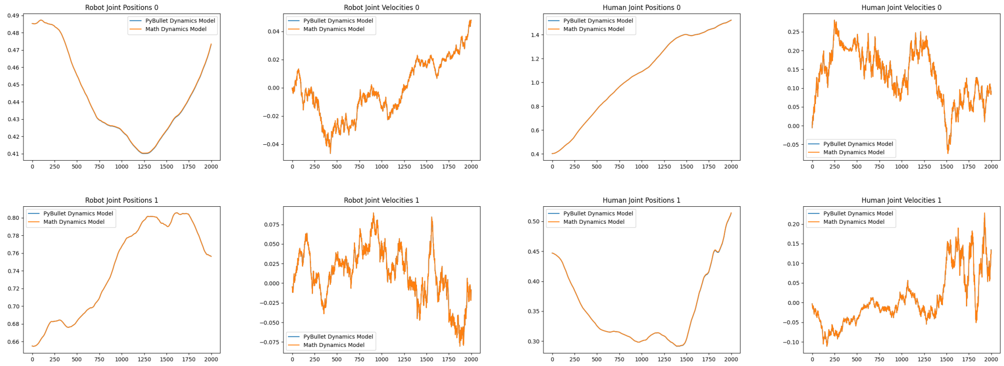

Figure 3: We validate our analytical dynamics model (orange) by comparing it to PyBullet rollouts (blue) from the same initial state over 2000 timesteps. Joint positions and velocities for both the robot and human match exactly across all joints (first two joints starting from base shown).

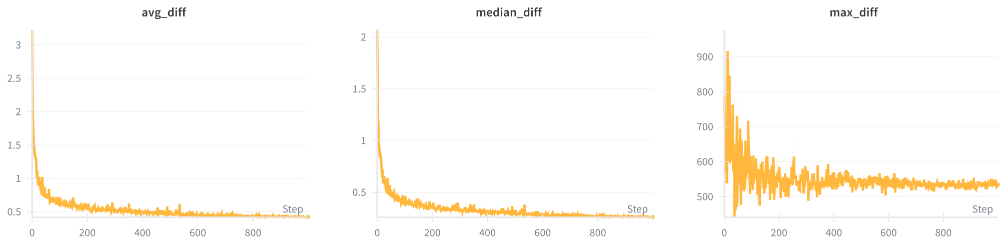

Figure 4: We train a four-layer MLP neural dynamics model on a synthetic dataset of 20M samples. Although the mean (left) and median (center) errors between predicted and ground-truth human/robot accelerations remain high, they are dominated by hard-to-learn regions near kinematic singularities, as reflected by the non-decreasing maximum error (right). In practice, we show that the model is sufficient across diverse start-goal configurations. Link to WandB Run.

joint accelerations (6 human and 6 robot). Training is performed using supervised regression with mean-squared error loss. Because both inputs and outputs are low dimensional, the full dataset is stored on the GPU, enabling large batch sizes. After hyperparameter sweeps, we select a learning rate of 3×10^{-4}, batch size 16,384, and 1,000 training epochs, with hidden layer sizes of 1024–2048–2048–1024. Training takes 8.6 hours on an NVIDIA RTX 6000 Ada Generation GPU.

Figure 4 shows the training results from our best Weights & Biases run. The final mean prediction error (on a held-out set) is approximately 0.4 radians (24.1 degrees) / s^2, with a median error of 0.26 radians (14.9 degrees) / s^2. While these errors appear large, we observe that the maximum error is dominated by rare but extreme outliers (up to 534 radians / s^2), which the network fails to approximate accurately. We hypothesize that these outliers arise near kinematic singularities. In such regions, small changes in joint configuration can induce large changes in the quantities used to compute accelerations, due to the ill-conditioning of the robot Jacobian as its smallest singular value approaches zero. As a result, the target accelerations exhibit extremely large magnitudes and high sensitivity in a small subset of the dataset, disproportionately inflating the maximum error and, to a lesser extent, the mean error. To validate this hypothesis, we train a separate network to predict the 6×6 robot Jacobian given robot joint positions alone. As shown in this separate Weights & Biases run, this model exhibits similar behavior: while the mean and median errors decrease substantially during training, the maximum error remains large due to near-singular configurations. These results suggest that the reported mean and median errors are inflated by near-singularity samples, which make even a simple robot jacobian very hard to learn.

In practice, the model performs well away from singularities, as validated in later experiments and illustrated by this open-loop rollout of the trained neural dynamics over 500 timesteps. The human and robot end effectors initially move together before gradually diverging as errors accumulate.

5.4 WOVEN-LM vs MPC with Analytical Dynamics Model

We compare WOVEN-LM against a baseline MPC approach that uses the analytical dynamics model, in order to evaluate real-time feasibility. Both methods are evaluated on the task shown in Figure 1 and use the same MPC formulation, with cost terms for goal reaching and human joint limit violations. The planning horizon is 25 timesteps, with 50 trajectory samples per planning step.

Analytical dynamics MPC. Using the analytical dynamics model within MPC results in a total of 658 execution timesteps, with an average planning time of 1.97 seconds per timestep and a total execution time of 21.6 minutes. Due to this computational cost, the analytical dynamics MPC cannot be deployed on a real robot.

WOVEN-LM. Replacing the analytical dynamics model with WOVEN-LM yields 656 execution timesteps under the same MPC configuration. The average planning time per timestep is reduced to 0.103 seconds, enabling real-time performance. This corresponds to a $19\times$ speedup over the analytical dynamics baseline. For verification, we use a tolerance of 0.1 radians ($\approx 6^\circ$), which is substantially smaller than the median and mean prediction errors observed during training. This reduction in planning time enables real-time execution, signaling the practicality of WOVEN-LM for deployment on physical robots.

5.5 WOVEN-LM on LimbManipulationBench

We evaluate WOVEN-LM on the 100 wheelchair–left-arm tasks in LimbManipulationBench. For 7 tasks, inverse kinematics for the initial human arm configuration is infeasible, leaving 93 valid problems. Among these, WOVEN-LM successfully solves 71 tasks, corresponding to a success rate of 76.3%. Videos of all 71 successful trials are available here.

The remaining 22 failures occur when the robot–human system enters states in which all trajectories proposed by the neural dynamics model fail the verification step, resulting in no feasible trajectory. In future work, such cases could be addressed by falling back to the analytical dynamics model to escape these locally challenging regions where the learned model performs poorly.

6 Conclusion

In this paper, we presented a closed-loop framework for safe manipulation of human limbs. By deriving an exact analytical dynamics model and combining it with a fast learned neural dynamics predictor within the WOVEN-LM framework, we demonstrated that it is possible to achieve real-time model predictive control while retaining verifiable safety guarantees. Simulation experiments in LimbManipulationBench show that WOVEN-LM enables closed-loop control across diverse limb configurations and significantly outperforms MPC which solely uses an analytical dynamics in terms of real-time feasibility. Looking forward, an important direction is to tackle user-specific muscle tone, which varies across individuals and over time and can substantially affect interaction forces during limb manipulation. Finally, deploying WOVEN-LM on a physical robot interacting with human participants remains a key next step to validate safety, performance, and practicality in real-world caregiving scenarios.

References

[1] W. H. Organization and W. Bank. World report on disability 2011, 2011.

[2] J. Lynch and R. Cahalan. The impact of spinal cord injury on the quality of life of primary family caregivers: a literature review. *Spinal cord*, 55(11):964–978, 2017.

[3] A. Nanavati, P. Alves-Oliveira, T. Schrenk, E. K. Gordon, M. Cakmak, and S. S. Srinivasa. Design principles for robot-assisted feeding in social contexts. In *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, pages 24–33, 2023.

[4] L. E. Dreer, T. R. Elliott, R. Shewchuk, J. W. Berry, and P. Rivera. Family caregivers of persons with spinal cord injury: Predicting caregivers at risk for probable depression. *Rehabilitation Psychology*, 52(3):351, 2007.

[5] A. R. Darragh, C. M. Sommerich, S. A. Lavender, K. J. Tanner, K. Vogel, and M. Campo. Musculoskeletal discomfort, physical demand, and caregiving activities in informal caregivers. *Journal of Applied Gerontology*, 34(6):734–760, 2015.

[6] R. Madan, R. K. Jenamani, V. T. Nguyen, A. Moustafa, X. Hu, K. Dimitropoulou, and T. Bhattacharjee. Sparcs: Structuring physically assistive robotics for caregiving with stakeholders-in-the-loop. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 641–648. IEEE, 2022.

[7] S. W. Brose, D. J. Weber, B. A. Salatin, G. G. Grindle, H. Wang, J. J. Vazquez, and R. A. Cooper. The role of assistive robotics in the lives of persons with disability. *AJPM&R*, 2010.

[8] R. K. Jenamani, D. Stabile, Z. Liu, A. Anwar, K. Dimitropoulou, and T. Bhattacharjee. Feel the bite: Robot-assisted inside-mouth bite transfer using robust mouth perception and physical interaction-aware control. In *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, pages 313–322, 2024.

[9] J. Broekens, M. Heerink, H. Rosendal, et al. Assistive social robots in elderly care: a review. *Gerontechnology*, 2009.

[10] T. Tamei, T. Matsubara, A. Rai, and T. Shibata. Reinforcement learning of clothing assistance with a dual-arm robot. In *2011 11th IEEE-RAS International Conference on Humanoid Robots*, pages 733–738, 2011. doi:10.1109/Humanoids.2011.6100915.

[11] A. Kapusta, W. Yu, T. Bhattacharjee, C. K. Liu, G. Turk, and C. C. Kemp. Data-driven haptic perception for robot-assisted dressing. In *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 451–458, 2016. doi:10.1109/ROMAN.2016.7745158.

[12] Y. Wang, Z. Sun, Z. Erickson, and D. Held. One policy to dress them all: Learning to dress people with diverse poses and garments. *arXiv preprint arXiv:2306.12372*, 2023.

[13] N. Koganti, T. Tamei, T. Matsubara, and T. Shibata. Real-time estimation of human-cloth topological relationship using depth sensor for robotic clothing assistance. In *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, pages 124–129, 2014. doi:10.1109/ROMAN.2014.6926241.

[14] K. Yamazaki, R. Oya, K. Nagahama, K. Okada, and M. Inaba. Bottom dressing by a dual-arm robot using a clothing state estimation based on dynamic shape changes. *International Journal of Advanced Robotic Systems*, 13(1):5, 2016. doi:10.5772/61930.

[15] N. K. R. P. Joshi and T. Shibata. A framework for robotic clothing assistance by imitation learning. *Advanced Robotics*, 33(22):1156–1174, 2019. doi:10.1080/01691864.2019.1636715.

[16] S. Kotsovolis and Y. Demiris. Model predictive control with graph dynamics for garment opening insertion during robot-assisted dressing. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[17] J. Borràs. Effective grasping enables successful robot-assisted dressing. *Science Robotics*, 7 (65):eabo7229, 2022. doi:10.1126/scirobotics.abo7229. URL `https://www.science.org/doi/abs/10.1126/scirobotics.abo7229`.

[18] R. Madan, S. Valdez, D. Kim, S. Fang, L. Zhong, D. T. Virtue, and T. Bhattacharjee. Rabbit: A robot-assisted bed bathing system with multimodal perception and integrated compliance. In *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, pages 472–481, 2024.

[19] C.-H. King, T. L. Chen, A. Jain, and C. C. Kemp. Towards an assistive robot that autonomously performs bed baths for patient hygiene. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 319–324, 2010. doi:10.1109/IROS.2010.5649101.

[20] Z. Erickson, H. M. Clever, V. Gangaram, G. Turk, C. K. Liu, and C. C. Kemp. Multidimensional capacitive sensing for robot-assisted dressing and bathing. In *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, pages 224–231. IEEE, 2019.

[21] H. Kazerooni, J.-L. Racine, L. Huang, and R. Steger. On the control of the berkeley lower extremity exoskeleton (bleex). In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 4353–4360, 2005. doi:10.1109/ROBOT.2005.1570790.

[22] K. Kong and D. Jeon. Design and control of an exoskeleton for the elderly and patients. *IEEE/ASME Transactions on Mechatronics*, 11(4):428–432, 2006. doi:10.1109/TMECH.2006.878550.

[23] A. M. Dollar and H. Herr. Lower extremity exoskeletons and active orthoses: Challenges and state-of-the-art. *IEEE Transactions on Robotics*, 24(1):144–158, 2008. doi:10.1109/TRO.2008.915453.

[24] L. Chen, C. Chen, Z. Wang, X. Ye, Y. Liu, and X. Wu. A novel lightweight wearable soft exosuit for reducing the metabolic rate and muscle fatigue. *Biosensors*, 11(7):215, 2021.

[25] J. Ortiz, T. Poliero, G. Cairoli, E. Graf, and D. G. Caldwell. Energy efficiency analysis and design optimization of an actuation system in a soft modular lower limb exoskeleton. *IEEE Robotics and Automation Letters*, 3(1):484–491, 2017.

[26] L. Zhou, Y. Li, and S. Bai. A human-centered design optimization approach for robotic exoskeletons through biomechanical simulation. *Robotics and Autonomous Systems*, 91: 337–347, 2017. ISSN 0921-8890. doi:https://doi.org/10.1016/j.robot.2016.12.012. URL `https://www.sciencedirect.com/science/article/pii/S0921889016301877`.

[27] J. M. Gandarias, F. Pastor, A. J. Muñoz-Ramírez, A. J. García-Cerezo, and J. M. Gómez-de Gabriel. Underactuated gripper with forearm roll estimation for human limbs manipulation in rescue robotics. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5937–5942. IEEE, 2019.

[28] Q. Zhao, R. Roy, C. Spurlock, K. Lister, and L. Wang. A high-fidelity simulation framework for grasping stability analysis in human casualty manipulation. *arXiv preprint arXiv:2404.03741*, 2024.

[29] X. Wang, K. Krishnaswamy, M. E. Cabrera, and M. Cakmak. Robotic limb repositioning with supervised autonomy. In *Companion of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, pages 511–513, 2020.

[30] K. Krishnaswamy, R. Kuber, and T. Oates. Developing a limb repositioning robotic interface for persons with severe physical disabilities. *Universal Access in the Information Society*, 15: 609–627, 2016.

[31] K. Chow and C. C. Kemp. Robotic repositioning of human limbs via model predictive control. In *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 473–480. IEEE, 2016.

[32] E. Peiros, Z.-Y. Chiu, Y. Zhi, N. Shinde, and M. C. Yip. Finding biomechanically safe trajectories for robot manipulation of the human body in a search and rescue scenario. *arXiv preprint arXiv:2309.15265*, 2023.

[33] K. Mo, L. J. Guibas, M. Mukadam, A. Gupta, and S. Tulsiani. Where2act: From pixels to actions for articulated 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6813–6823, 2021.

[34] Z. Xu, Z. He, and S. Song. Universal manipulation policy network for articulated objects. *IEEE robotics and automation letters*, 7(2):2447–2454, 2022.

[35] Z. Jiang, C.-C. Hsu, and Y. Zhu. Ditto: Building digital twins of articulated objects from interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5616–5626, 2022.

[36] E. Heiden, Z. Liu, V. Vineet, E. Coumans, and G. S. Sukhatme. Inferring articulated rigid body dynamics from rgbd video. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8383–8390. IEEE, 2022.

[37] K. Hausman, S. Niekum, S. Osentoski, and G. S. Sukhatme. Active articulation model estimation through interactive perception. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3305–3312. IEEE, 2015.

[38] A. Jain, R. Lioutikov, C. Chuck, and S. Niekum. Screwnet: Category-independent articulation model estimation from depth images using screw theory. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13670–13677. IEEE, 2021.

[39] G. Rizzi, J. J. Chung, A. Gawel, L. Ott, M. Tognon, and R. Siegwart. Robust sampling-based control of mobile manipulators for interaction with articulated objects. *IEEE Transactions on Robotics*, 39(3):1929–1946, 2023. doi:10.1109/TRO.2022.3233343.

[40] M. Mittal, D. Hoeller, F. Farshidian, M. Hutter, and A. Garg. Articulated object interaction in unknown scenes with whole-body mobile manipulation. In *2022 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 1647–1654. IEEE, 2022.