Learning Robotic Manipulation of Natural Materials with Variable Properties for Construction Tasks

Nicolas Kubail Kalousdian[1], Grzegorz Łochnicki,
Valentin N. Hartmann[2], Samuel Leder[3], Ozgur S. Oguz[2], Achim Menges[3], Marc Toussaint[4]

***Abstract*—The introduction of robotics and machine learning to architectural construction is leading to more efficient construction practices. So far, robotic construction has largely been implemented on standardized materials, conducting simple, predictable, and repetitive tasks. We present a novel mobile robotic system and corresponding learning approach that takes a step towards assembly of natural materials with anisotropic mechanical properties for more sustainable architectural construction. Through experiments both in simulation and in the real world, we demonstrate a dynamically adjusted curriculum and randomization approach for the problem of learning manipulation tasks involving materials with biological variability, namely bamboo. Using our approach, robots are able to transport bamboo bundles and reach to goal-positions during the assembly of bamboo structures.**

***Index Terms*—Robotics and Automation in Construction; Hardware-Software Integration in Robotics; AI-Enabled Robotics**

I. Introduction

THERE is an indisputable need for a more sustainable built environment [1]. Enabling robotic construction systems to build with a wide range of materials is a critical step in that direction [2]. Over the past decade, research on robotic construction with standardized materials has led to more efficient construction processes [3]. Natural materials, which are often heterogeneous, deformable, and/or anisotropic, have largely been phased out in favor of highly processed materials. Enabling usage of such non-standard materials increases the range of feasible materials in the construction industry, thus forging new paths towards more diverse, and sustainable processes [4].

Fig. 1: **Left:** Custom mobile robot climbing a bamboo structure, **Right:** Agent learning how to reach the goal (red) by bending bamboo bundles in simulation by swinging.

Natural materials are difficult to physically handle and a challenge to digitally model due to the fact that their mechanical properties and dynamics vary within and across samples. Therefore, robotic construction with such materials requires planning and control strategies that generalize to uncertain properties of a single element and continuously adapt to unpredictable deformations inherent to natural materials in the real world. Research on the topic of robotic construction thus tends to focus on building with idealized and predictable materials [5], are cast as purely geometrical problems [6], or are planned completely in advance leaving no room for adaptation [7]. Such assumptions rarely hold in real-world settings, where varying material properties and environment conditions need to be taken into account. Recent research in machine learning indicates that the challenges of dealing with irregular materials can be tackled with reinforcement learning (RL). RL has been widely used in training control policies for dynamic tasks that can be transferred to real-world robots [8]–[10]. Although robots are increasingly utilized in the architectural construction process [11], and RL has proven robust in training robots for real world applications [12], [13], an approach to training robots to perform construction tasks using materials with biological variability is missing. This work explores the feasibility of combining methods from the fields of machine learning, robotics, and construction from a systems-oriented perspective on the problem of robotically manipulating materials with variable properties in dynamic construction tasks in both simulation and the real world. In doing so, the feasibility of the proposed methods is evaluated, aiming to contribute, as fundamental research, to the eventual transition of research from laboratory settings to real-world applications.

As a case study, we chose the task of assembling bamboo

Manuscript received: September, 16, 2021; Revised December, 17, 2021; Accepted March, 6, 2022. This paper was recommended for publication by Editor Y. Choi upon evaluation of the Associate Editor and Reviewers' comments. This research has been supported by the Cyber Valley Research Fund (CyVy-RF-2021-18) and the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2120/1–390831618 - Cluster of Excellence IntCDC.

[1] N. K. Kalousdian is with the Institute for Computational Design and Construction, University of Stuttgart, Germany, with the Cluster of Excellence IntCDC, and with the Cyber Valley, nicolas.kubail@icd.uni-stuttgart.de

[2] V. N. Hartmann and O. S. Oguz are with the Machine Learning & Robotics Lab, University of Stuttgart, Germany, with the Learning and Intelligent Systems Group, TU Berlin, Germany, and with the Cluster of Excellence IntCDC, valentin.hartmann@ipvs.uni-stuttgart.de

[3] S. Leder and A. Menges are with the Institute for Computational Design and Construction, University of Stuttgart, Germany, and with the Cluster of Excellence IntCDC samuel.leder@icd.uni-stuttgart.de

[4] M. Toussaint is with the Learning and Intelligent Systems Group, TU Berlin, Germany, and with the Cluster of Excellence IntCDC, toussaint@tu-berlin.de

Digital Object Identifier (DOI): see top of this page.

structures with a team of custom mobile robots. Bamboo is an irregular material that is deformable and anisotropic, which adds to the complexity of working both physically and computationally with the material. Bamboo is a rapid-renewable plant with a much shorter crop-cycle than wood making it a particularly important inroad to more sustainable construction [14]. A similar demonstration can be done with other naturally renewable construction materials, such as wood, which exhibits natural variability if it is unprocessed.

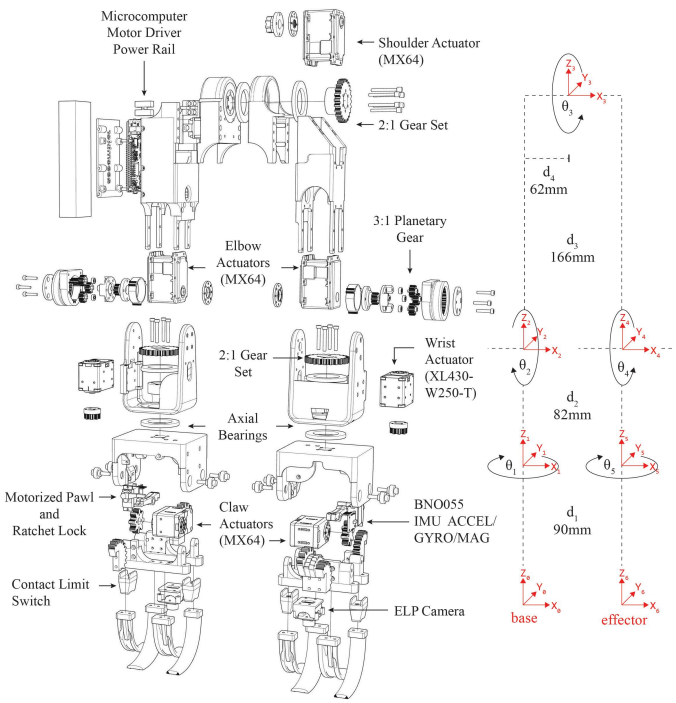


Fig. 2: Kinematic and hardware design of the robot, and gear ratios for each joints. The claws can act both as base link or end effector.

The mobile robot was custom-designed to deal with the varying properties of bamboo and to perform two tasks involved in assembling bamboo elements: namely *transporting* and *reaching*. Our aim is to show that the robots can learn to use their weight, movement, and momentum to swing and bend bamboo elements into designed configurations while hanging on to unstable and partially observable structures (Fig. 1). In pursuing this goal, our contributions are:

- We co-design both hardware, and learning and control software for a novel mobile robotic system that can be employed in bamboo construction.
- We present a dynamically adjusted curriculum approach for the problem of learning manipulation of natural materials with variable properties for building construction.
- We extend the application of automatic domain randomization (ADR) to model and compensate for uncertainty in material parameters during domain transfer.
- We demonstrate that ADR can serve as an exploratory method for identifying the feasible parameter space of the robot-policy pairing.

II. Related Work

There is a growing interest in robotics in construction, mainly due to the possibilities of increasing efficiency and sustainability, and decreasing cost. We discuss relevant work in construction robotics, particularly distributed robotic systems, and learning for material manipulation.

a) Robots in Construction: While brick-laying robots [15], robots to extrude concrete to build houses [16], or robotic excavators and bulldozers [17] were deployed on real construction sites, and some even integrated in standardized processes such as BIM [18], [19], most ongoing research is acting in laboratory-environments, such as drones used for assembly of a foam tower [20] or assembly of a timber structure [21].

Additionally, most work is focused on highly regular building materials, such as bricks, or standardized wooden struts [22]. Conversely, the work presented in [23] uses stones to build a tower, however, only demonstrates the concept on a simple stacking task. The potential of digital building tools with respect to highly amorphous materials is investigated in [24], and [25] gives a more exhaustive review on the possibilities of collective robotic construction.

Other work in building construction and digital fabrication focuses on the robotic planning aspect for co-design, by attempting to evaluate feasibility of a robotic building process [7]. The project presented in [26] specifically designs timber structures to enable a team of distributed robots to efficiently fabricate them. The review in [27] notes that while various use-cases for prefabrication (such as [28], [29]) and mobile robotics in construction exist, few of them are economically viable due to their required supervision, and high specialization. Instead of fitting the material to our expectations, we enable the usage of irregular building materials by explicitly accounting for uncertainty in material parameters, and by doing so enable a less wasteful construction process.

b) Learning for Material Manipulation: In most manipulation research (such as [7], [30]), the assumption is made that the robots deal with rigid bodies, and do not modify the structure and shape of the objects itself. It is then possible to leverage trajectory optimization algorithms to synthesize a motion for the manipulation tasks. The work in [31] deals with learning motion primitives to e.g. pour fluids reliably, and to enable common task and motion planning approaches to use these primitives. This work does however not explicitly represent the fluid, and rather hides the actual material manipulation in a black-box primitive. If manipulation of the material into a certain shape is the goal, this approach does not work.

More recent work used deep reinforcement learning to manipulate non-rigid bodies [32], [33], which requires efficient simulation tools, and usually intricate prior knowledge of the models, and involved actions. Previous work on learning material manipulation was applied on amorphous materials in a kitchen scenario [34], with the purpose of character-animation (e.g. to spread rice on a cooking surface). As such, it is possible to separate the consideration of the body from the motion of the tool. Another approach to increase efficiency of the learning process is the use of differentiable (learned)

models [35] or to introduce physics into the model [36]. In the construction setting, a similar approach was followed in [37] to learn how to deform sheet metal. In digital fabrication, a common approach to deal with irregular materials is to extensively scan the available natural, irregular materials, and use traditional construction processes based on the generated model of the material [4]. This approach is time-consuming and not scalable.

Contrary to the previously mentioned work, we propose jointly learning the behavior of the material and the motion of the agent to influence the behavior of the material in a single policy. While there are similarities to [32], [33], our work deals with deformable, not amorphous materials, and is transferred to real hardware.

III. Problem Description

Robotic construction of bamboo structures requires many tasks, and thus various skills to be planned for. We tackle two of these tasks: *reaching* towards, and *transportation* of bamboo bundles. Both tasks can be described as stochastic, partially observable Markov Decision Processes (POMDPs).

- *Reaching*: the robot has to reach a goal with its end-effector (Fig. 1, right).
- *Transportation*: the robot holds a bamboo element in its free end-effector, and has to reach a goal position with the tip of the bamboo element (Fig. 3).

We describe the motion each task deals with: Reaching involves bending of the bamboo bundle that the robot is holding on to, i.e. the goal positions might be out of reach if the bamboo-bundle which the robot holds on would be static. Transportation involves reacting to the deformations and spring-back of the root bundle as the held bamboo element is manipulated in addition to the bending behaviour described above. In both cases, the goal is stationary, and is sampled depending on the allowable bending radius of the bamboo, and the robot morphology.

For real world tasks, the robot does not have access to the full state information, but just relies on its observations. We can then formally define a POMDP as a six-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \Omega, \mathcal{O})$, representing states, actions, transitions, rewards (as in an MDP), and finite set of observations, and observation dynamics, respectively. Given such a POMDP, we employ RL methods, in particular domain randomization and curriculum learning (Section VI), to find a policy π that fulfills the task. The observed features o_t are taken from internal and external sensor readings, and describe the environment's state s_t at time t (see Eq. (1)). The feature vector o_t is then used to build the observation history $o_{1:t} = \{o_1, ..., o_t\}$ from the start time, to the current time t. The policy's outputs a_t are joint position targets. Rewards are calculated cumulatively in each training episode as described in Eq. (2). To simulate the transition function $p^*(s_{t+1}|s_t, a_t)$ of the real environment, we describe the bamboo material system as a discrete chain of rigid-bodies connected by damped harmonic oscillators. Together with a Featherstone solver to simulate the robotic hardware, this describes the transition function $p'(s_{t+1}|s_t, a_t)$ of the training environment. Thus, the goal is to train the policies in p' such that they can be transferred to p^*.

TABLE I: Mechanical properties (M) of Arundinaria Amabilis samples.

Property	Mean	Std. Deviation
Length (l)	1.8 m	0.1 m
Cross Section (c)	1.4 cm	0.4 cm
Wall Thickness (w)	2 mm	1 mm
Elastic Modulus (E)	17.6 GPa	1.4 GPa
Shear Modulus (Γ)	6.62 GPa	0.53 GPa
Bend Radius (κ)	9.7 mm	3 mm
Poisson's Ratio (υ)	0.3	NA

IV. Material Description

Bamboo rods are the base element of the construction system. The physical setup of the bamboo structure consists of the bamboo *bundles*, connection mechanisms, and the anchoring system. The bamboo rods are joined together with metal zip-ties into short bundle assembly groups of 1.8 meters length. In order to achieve longer lengths we shift bundle assembly groups half of their length apart and join them further with metal zip ties (Fig. 4). The difference in rod count of each bundle assembly group is described by the bundle density discretization (a full bundle may be made up of 3 assembly groups, containing 8, 5, and 3 rods, respectively, and is thus tapering towards its tip). Bundles are attached to the ground with steel anchors and clamps.

a) Mechanical Parameters: We describe the space of mechanical parameters $\mathcal{M}$ of *Arundinaria Amabilis* bamboo rods, the species we use in this work, in Table I.

b) Geometric Parameters: The space of geometric parameters $\mathcal{B}$ is a set of parameter-ranges that describe possible material configurations. This space is defined by the inclination or pitch of the bundle, the rotation around the world z vector (yaw), the total bundle length (as a multiple of single rod lengths l), and the bundle density discretization (Fig. 4). Thus, the set of possible configurations is $\mathcal{C} = \mathcal{M} \times \mathcal{B}$.

V. Hardware System Design

For this work we developed a mobile robot for brachiation on the bamboo, which was co-designed relative to the material system described in Section IV.

a) Hardware: Figure 2 illustrates the mechatronic design of the robot, including its actuators, controllers, and sensors. The robot's state is estimated with a combination of internal and external sensors: Internally, the robot keeps track of its orientation as a quaternion q, using an accelerometer, a gyroscope, and a magnetometer embedded in each end-effector. The joints' positions θ_i are monitored through the servos' built-in encoders. Externally, a multi-camera tracking system localizes the robot root link position p_r, in the construction site. The end-effector claws have interlocking fingers and are controlled with a force-feedback strategy. This helps the robot to maintain a tight grip on the bamboo bundles while allowing it to adapt its grip to various bundle cross-section geometries that result from the unpredictable arrangement of rods within a bundle, and the varying cross-section of each rod. The inside of each claw is equipped with a limit switch to detect

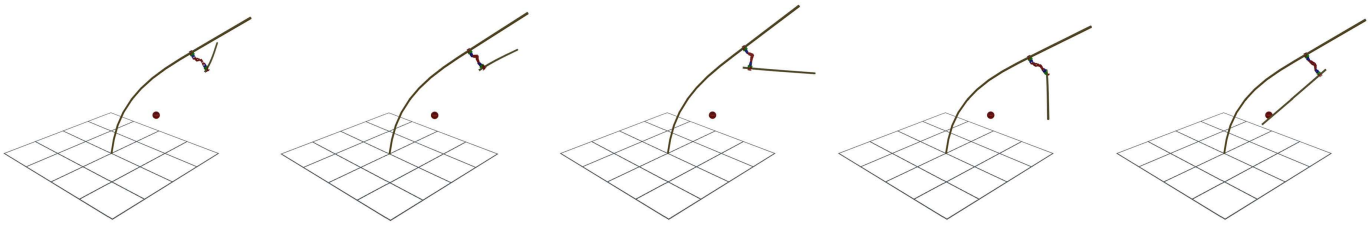


Fig. 3: Transportation task: the robot begins the episode holding a rod at a random position, and has to reach the sphere with the tip of the rod, while accounting for the deformation in the root bundle caused by the shifting center of gravity.

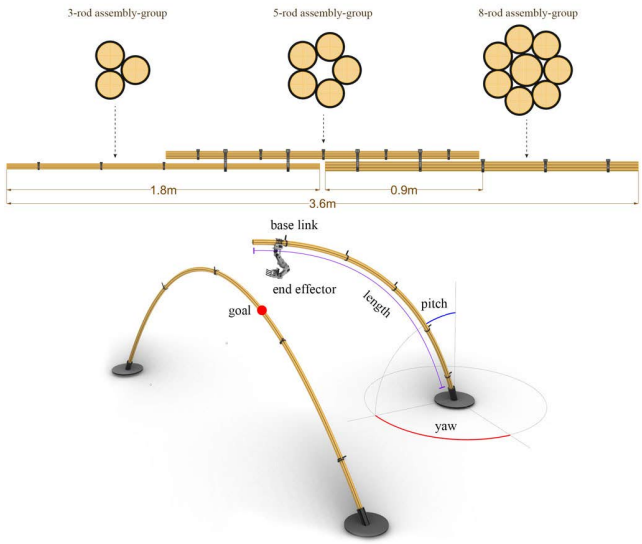


Fig. 4: Example of bamboo bundle assembly, and geometric parameters visualization for reaching task.

contact. Thus, the robot hardware state $\mathcal{O}_t^{\text{hw}}$ at time t is the concatenation of all sensor-readings from above.

The joints are actuated by a series of chained Dynamixel motors. Motor torque requirements were derived from a Newton-Euler formulation in worst case loading scenarios. The shoulder and elbow joints experience the most torque requirements, and use the MX64 model with a 2 to 1 and 3 to 1 gear ratio respectively. The wrist joints and claws require less force, and are actuated by XL430-W250-T motors with a 2 to 1 gear ratio. Furthermore the claws contain a pawl and ratchet safety mechanism to mechanically lock the grip around the bamboo in case of motor failure.

b) Kinematics Model: The robot has five degrees of freedom (see Fig. 2). Figure 5b lists the kinematic constraints for each joint, with axis vectors described in relation to the parent joint. The kinematic chain, $\mathcal{K} \in \mathbb{R}^5$, was designed to allow for a motion space that matches the complexity induced by the geometric and mechanical parameters of the material specifications (Section IV). We aimed to minimize the motor count while still keeping the DoFs required by the assembly tasks. The morphology is symmetrical: two axial joints on the bases (wrists θ_1 and θ_5), connect to two revolute joints, (elbows θ_2 and θ_4), that meet at a central revolute joint (shoulder, θ_3).

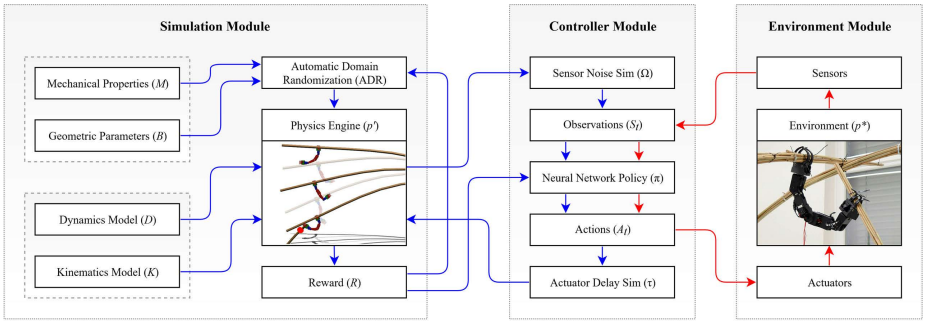
Algorithm 1 ADR for Variable Materials

Require: $\mathcal{M}, \mathcal{B}$ ▷ Mechanical and geometric param.
Require: D, K ▷ Robotic dynamics and kinematics
Require: ϕ_M, ϕ_B, d ▷ Initial param. values, and count
Require: m, n ▷ Buffer size, sample boundary (%)
Require: Δ, ϱ ▷ Expansion rate, expansion start
Require: σ^L, σ^H ▷ Expansion thresholds
Require: $\{r_i^L, r_i^H\}_{i=1}^d$ ▷ High and Low Reward Buffers
Require: π ▷ Policy in training
Require: E, i ▷ Episodes per epoch, current episode
$\mu \leftarrow \mathcal{M}(\phi_M), \mathcal{B}(\phi_B)$
for $i = 1, .., E$ **do**
  $c_i \sim \mathcal{U}(\mu^H, \mu^L)$
  $R_i \leftarrow \texttt{EvaluateModel}(c_i, D, K, \pi)$
  if $\max(R_i, R_{i-1} ... R_0) \geq \varrho$ **then**
    for $j = 1, .., |c_i|$ **do**
      if $c_i^j >= n\mu_j^H$ **then**
        append R_i to r_j^H
      if $c_i^j <= (1-n)\mu_j^L$ **then**
        append R_i to r_j^L
      if $\texttt{length}(r_j^L) \geq m$ **then**
        $r_{\text{avg}} = \frac{1}{m}\sum_{t=0}^{m} r_j^L[t]$
        clear buffer r_j^L
        if $r_{\text{avg}} \geq \sigma^H$ **then**
          $\mu_j^L \leftarrow \mu_j^L - \Delta$
        else if $r_{\text{avg}} \leq \sigma^L$ **then**
          $\mu_j^L \leftarrow \mu_j^L + \Delta$
      if $\texttt{length}(r_j^H) \geq m$ **then**
        $r_{\text{avg}} = \frac{1}{m}\sum_{t=0}^{m} r_j^H[t]$
        clear buffer r_j^H
        if $r_{\text{avg}} \geq \sigma^H$ **then**
          $\mu_j^H \leftarrow \mu_j^H + \Delta$
        else if $r_{\text{avg}} \leq \sigma^L$ **then**
          $\mu_j^H \leftarrow \mu_j^H - \Delta$

VI. Learning System Architecture

The proposed system architecture consists of three interconnected parts: simulation, controller, and environment. They interact within the *learning*- and the *execution*-pipeline (Fig. 5a). For training we opt to implement and use Proximal Policy Optimization as in [38], since it's a robust and efficient model-free RL algorithm.

a) Domain Randomization (DR): RL algorithms are often sample inefficient and the costs of training policies on real robots too expensive and time consuming. Therefore training is usually conducted in simulation. Furthermore, transferring a policy from simulation to a real-world robot, known as sim-to-real, often fails due to inconsistencies between the models and calibration used to simulate a training environment and



(a)

Joint	Axis	Range
θ_1 Wrist	Z	$[-170°, 170°]$
θ_2 Elbow	X	$[-90°, 90°]$
θ_3 Shoulder	X	$[-135°, 135°]$
θ_4 Elbow	X	$[-90°, 90°]$
θ_5 Wrist	Z	$[-170°, 170°]$

(b)

Fig. 5: **(a)** System architecture for problems of learning manipulation of materials with variable properties in construction. Learning in simulation pipeline is shown in blue, and the real world execution pipeline is shown in red. **(b)** Kinematic description of the robot and constraints for each joint.

the physical world [10].

As one of our aims is to show the proposed system physically, we employ domain randomization in order to overcome the sim-to-real gap. Domain randomization is a technique used in RL problems where a model or policy trained in a source domain, the dynamics denoted by the environment transition function p', is made transferable to a target domain, the real world denoted by p^*, by randomizing parameters of the source domain during training [39].

To do this, we generate instances $c_i \in \mathcal{C}$ by sampling from a uniform distribution $c_i \sim \mathcal{U}(\mu^L, \mu^H)$ defined by high threshold μ^H and low threshold μ^L, with $\mu^H, \mu^L \in \mathbb{R}^{|\mathcal{C}|}$ (line 3). This causes the policy to adapt to variability in the the training environment, and changes the learning goal to maximize the expected return across a distribution of environments C [10].

b) Curriculum Learning: When applying DR to the problem of building using materials with variable properties we quickly face the challenge of manually tuning randomization ranges that meaningfully capture the space of environments $\mathcal{C}$. Furthermore, we must ensure that the agent is still able to learn a successful policy in a reasonable amount of time without getting stuck in local minima within such a large variation of training environments.

In order to overcome this challenge we drive domain randomization with a curriculum. Curriculum Learning is a strategy for policies to progressively learn from simple tasks to more difficult problems. This is done by gradually increasing the difficulty of training samples in relation to agent performance. In our case the difficulty is defined as the size of the configuration space that we are sampling from. First, we initialize each randomized parameter with a corresponding low variance μ (line 1), then the agent performance is evaluated in line 4 based on a training environment defined by the sampled parameters in line 3. Once the reward reaches a minimum threshold of ϱ (line 5) the expansion algorithm kicks in. Line 6 iterates through each parameter in $\mathcal{M}, \mathcal{B}$, and checks whether it was boundary sampled (lines 7 and 9), if so then it adds the reward of the current episode to that parameter's buffer r_j^H and r_j^L. Lines 11 and 18 check if the buffers are full, and if so we take the average reward and extend or contract the variance of the parameter's distribution in relation to the policy's performance that was accumulated during the learning process (lines 12-17, and 19-24). With each increase in variance the task gets harder, because the policy needs to generalize to a larger distribution of randomized environments. The goal is to have a policy which can incrementally generalize across a large distribution of parameter configurations.

VII. Experiments and Results

We conduct a series of experiments and ablation studies to evaluate the learned reaching and transportation policies. These experiments aim to (*i*) evaluate the ability of the proposed system to learn two different tasks with bamboo, (*ii*) test the ability of the learned policies to generalize to unseen material and geometric configurations, (*iii*) compare these policies' generalization ability to a baseline DR method, (*iv*) evaluate the performance of the hardware, and learning and control software design of our novel mobile robotic system.

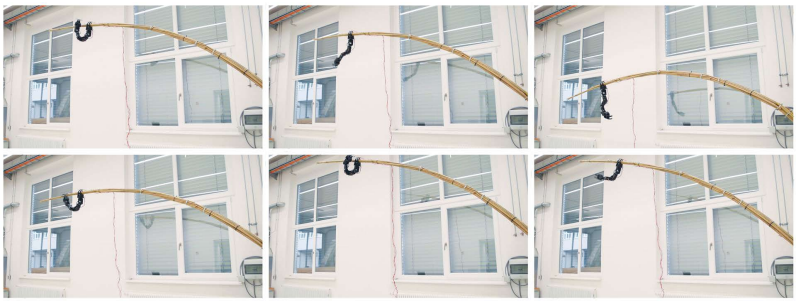
a) Training Details: In order to achieve the high level of interactivity and stability required for reinforcement learning, simulations of material-robot interactions are conducted within the PhysX 4 engine. The physical setup follows the description in Section IV. All policies are feed-forward networks with three layers of 128 units each, and they are trained using asynchronous gradient descent [40] to keep training stable.

b) Input: The policy's input at timestep t consists of the history $o_{1:t}$ of the observation vectors which are computed from $\mathcal{O}_t^{\text{hw}}$:

$$\mathcal{O}_t = (n \in \mathbb{R}^3, q \in SO(3), \theta \in \mathbb{R}^5, \omega \in \mathbb{R}^3, v \in \mathbb{R}^3), \quad (1)$$

where n is a vector from the free end-effector (respectively from the tip of the bundle) to the goal, q is the orientation of the robot's base link as a quaternion, θ are the joint angles, ω is the angular velocity of the base link, and v is the linear velocity of the base link[1]. In order to better represent motion information to the policy, the past l observations (i.e. the truncated history $o_{t-l+1:t}$) at each time step are fed to the policy. To reduce the sim-to-real gap, Gaussian noise is applied to each feature in order to model uncertainty in the hardware sensors, as shown in [10].

[1]Note that we do not need any explicit estimation of the state of the bamboo bundle in our formulation.



(a)

Task	Seen	Unseen	Diff.
Reaching (ADR)	**91.0%**	**83.5%**	**7.5%**
Reaching (DR)	83.4%	66.7%	16.7%
Transp. (ADR)	92.9%	**88.9%**	**4.0%**
Transp. (DR)	**93.4%**	83.8%	9.6%

(b)

Fig. 6: (a) Snapshots of the robot bending the bamboo bundle in the *reaching* task. (b) Performance of the ADR-policies compared against the DR-policies.

c) Actions: The policy outputs position targets described in degrees as offsets from the current joint rotations. This results in a 5D action space. Valid actions are constrained to be within the ranges specified in the corresponding joints in Fig. 5b. Furthermore, we apply a phase shift τ to the control signal coming from the policy during training. This serves to further reduce the sim-to-real gap by modelling the latency present in the physical controller [10].

d) Reward: The reward is a function of the distance between the end-effector position p_t (respectively bundle tip for the transp. task) at time t, and the goal position g (Eq. (2)). We use an intermittent reward strategy, accumulating the reward at each timestep t until we reach an end condition of $f(t) = 0$ or the maximum amount of steps per episode T in contrast to only rewarding the agent when the goal is reached to compensate for learning difficulties in problems with sparse-reward environments. Thus, the total reward is $R = R_i + R_g - R_p$, where $R_g = 1$ is added once the goal was reached[2], and R_p is a penalty that is added at each step to encourage shorter solutions.

$$R_i = \sum_{t=0}^{T} \max\left(0, \min\left(f(t_0), \ldots, f(t-1)\right) - f(t)\right), \quad \text{with} \quad f(t) = \frac{||g - p_t||_2}{||g - p_0||_2}. \tag{2}$$

e) Transfer to Unseen Configurations: In the first ablation study we compare the performance of policies trained with ADR against those trained with DR. The unseen configurations we test this on are parameter sets that were not encountered before (i.e. during training). Both the low reward threshold σ^L and expansion start ϱ are set to 0.9, in order to keep the policy achieving a minimum of 90% accuracy during ADR expansion. Policies trained with ADR outperformed those trained with DR when transferring to unseen configurations. The ADR policies achieved a success rate above 80% on both seen and unseen configurations. In the reaching task the ADR policy had a 7.5% drop in performance, whereas the DR policy had a 16.7% drop. In the transportation task the performance drops were 4.0% and 9.6% respectively (Fig. 6b). This confirms our contribution of extending the application of automatic domain randomization to compensate for uncertainty in material parameters.

[2]Shaping the reward function like this is necessary due to the intermittent reward: If $f(t)$ is used as reward directly, the agent can exploit this by 'not quite' reaching the goal, thus accumulating reward until the end.

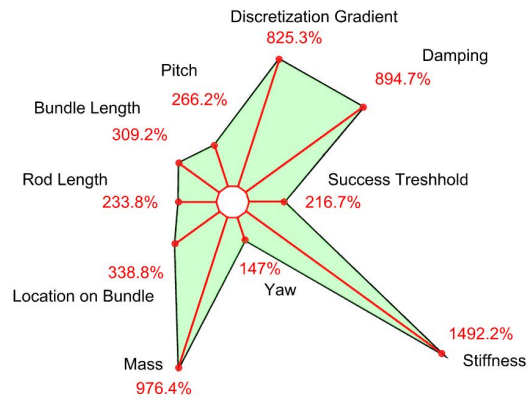


Fig. 7: Feasible parameter space of the learned transportation policy. Numbers are the percentage of expansion in relation to the starting range.

f) Sample Efficiency and Motion: In the second ablation study we compare two network architectures for the ADR policy, one that observes temporally stacked frames, and one that only observes single instances in time. When comparing sample efficiency of the two network architectures, we observe that the temporally stacked frames approach has the least variance across the two tasks (Fig. 9). In the reaching task, temporal stacks train almost twice as fast as the single frame variant. In transportation, temporal stacks are also more efficient. Therefore we can conclude that the more a task is time dependent and impacted by nonlinear motion, the more useful stacked frames becomes. This suggests that motion history representation is a key feature in the observation space of deformable material manipulation problems.

g) ADR Expansion Rate and Learning Decay: The learning rate is kept constant during training to prevent the policy from overfitting to the initial un-expanded local minimum. As the expansion curve stabilizes (i.e., reward increase is marginal), we also decay the learning rate to let the policy converge to a stable state (Fig. 9a). We observed a sharp drop in performance when the expansion rate was set above 2% (see Fig. 8).

h) Feasible parameter space: The resulting parameter space C_{feas} defined by the upper and lower thresholds from

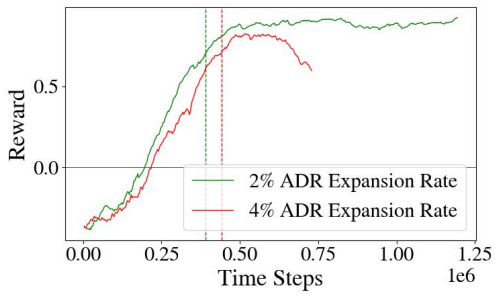


Fig. 8: Learning curves with different ADR expansion rates.

each learning session represents the feasible permutations of mechanical and geometric parameters that the resulting robot-policy pairing can solve at the 90% success rate (Fig. 7). This space can be used as a feasibility check when designing structures to be built by this system, thus serving as a valuable tool in co-design pipelines.

i) Proof of Concept Hardware Evaluation: Experiments were conducted to determine the performance of the hardware using policies from various stages of development and demonstrate a proof of concept of the system.

The robot was able to transport bamboo pieces successfully while hanging on to various parts of the demonstrator structure (see supplementary video). The cantilever formed by denser bundle assembly-groups was bigger than the simulated counterpart, and led to some discrepancies between simulation and real tests. Reaching was also possible, and the robot reacted quickly to material deformations. However the amplitude of the deformations caused by the robot's swinging was smaller than those seen in simulation Fig. 6a. The sizing of the bamboo rods acquired caused excessive damping when bundled, and the imperfections in the printed body of the robot caused greater friction in the joints than anticipated. Subsequent prototypes will be made of aluminum parts and tests will be conducted on rods with various diameters to determine appropriate robot-material hardware pairings.

We also show locomotion across a 9 meter long bundle. In this case, a learned policy was not needed, as an inverse kinematics algorithm was able to accomplish the task (see supplementary video), since the goal for the next step can always be described relative to the base link of the robot, and thus the dynamics of the bamboo do not matter as much.

VIII. Discussion and Conclusion

We have presented a novel mobile robotic system that contributes to transition experimentation towards assembly of natural materials with variable mechanical properties for architectural construction. We conducted experiments both in simulation and in the real world to demonstrate the feasibility of our system architecture in learning manipulation tasks involving bamboo, a material with biological variability. Using our approach, robots are able to reach towards and transport bamboo bundles for the assembly of bamboo structures. We also show successful initial experiments for more complex behaviours involving two collaborating robots (see supplementary video). Once a larger range of behaviors is developed, we can start evaluating more global metrics such as effective time for constructing a structure.

Although our work is specific to architectural construction with the aim of enabling the use of natural building materials for construction, we hope that our system serves as a guide for successful robotic manipulation of irregular and variable materials in other domain-specific use cases. We believe that construction site related challenges, such as weather related external disturbances, could also be accounted for using similar methods in future work. Additionally, we want to study multi-robot collaboration with naturally variable materials, and analyze the complex relationships between multiple robots and materials. Initial results on such complex, collaborative tasks can be seen in the supplementary video.

Acknowledgment

The authors would like to thank Ozgur Oguz, Achim Menges, and Marc Toussaint for their valuable supervision.

References

[1] McKinsey & Company, "The next normal in construction: How disruption is reshaping the world's largest ecosystem," 2020.

[2] J. Knippers, C. Kropp, A. Menges, O. Sawodny, and D. Weiskopf, "Integratives computerbasiertes planen und bauen: Architektur digital neu denken," *Bautechnik*, vol. 98, no. 3, pp. 194–207, 2021.

[3] A. Menges, "Integrative design computation: Integrating material behaviour and robotic manufacturing processes in computational design for performative wood constructions," in *In Proc. of the Conf. of the Association For Computer Aided Design In Architecture*, 2011, pp. 78–81.

[4] P. Devadass, F. Dailami, Z. Mollica, and M. Self, "Robotic fabrication of non-standard material," in *Proc. of the Conf. of the Association for Computer Aided Design in Architecture (ACADIA)*, 2016.

[5] K. H. Petersen, R. Nagpal, and J. Werfel, "Termes: An autonomous robotic system for three-dimensional collective construction," in *Proc. of Robotics: Science and Systems (R:SS)*, 2011.

[6] A. Gandia, S. Parascho, R. Rust, G. Casas, F. Gramazio, and M. Kohler, "Towards automatic path planning for robotically assembled spatial structures," in *Robotic Fabrication in Architecture, Art and Design*, J. Willmann, P. Block, M. Hutter, K. Byrne, and T. Schork, Eds., Cham: Springer International Publishing, 2019, pp. 59–73.

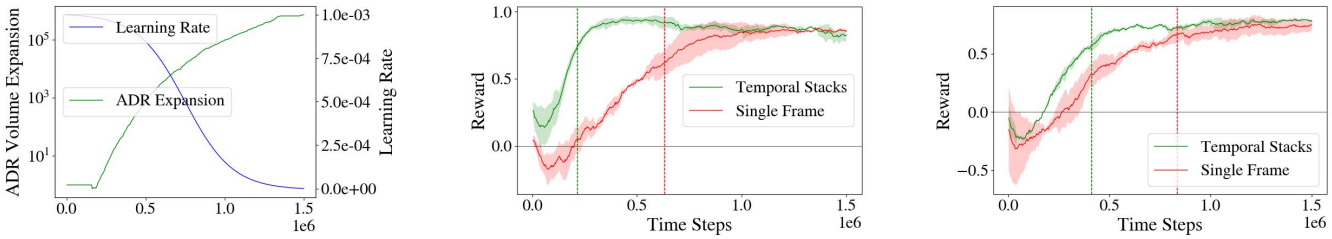
[7] V. N. Hartmann, O. S. Oguz, D. Driess, M. Toussaint, and A. Menges, "Robust task and motion planning for long-horizon architectural construction planning," in *Proc. of the IEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[8] J. Siekmann, S. Valluri, J. Dao, L. Bermillo, H. Duan, A. Fern, and J. Hurst, *Learning memory-based control for human-scale bipedal locomotion*, 2020. arXiv: 2006.02402 [cs.RO].

[9] A. Nagabandi, I. Clavera, S. Liu, R. S. Fearing, P. Abbeel, S. Levine, and C. Finn, *Learning to adapt in dynamic, real-world environments through meta-reinforcement learning*, 2019. arXiv: 1803.11347 [cs.LG].

[10] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, May 2018.

[11] J. M. Davila Delgado, L. Oyedele, A. Ajayi, L. Akanbi, O. Akinade, M. Bilal, and H. Owolabi, "Robotics and automated systems in construction: Understanding industry-specific challenges for adoption," *Journal of Building Engineering*, vol. 26, p. 100 868, 2019, ISSN: 2352-7102.



(a) (b) (c)

Fig. 9: (a) Decay of the learning rate as the expansion curve starts to stabilize (reaching task). Learning curves for (b) reaching and (c) transportation tasks using ADR policies. Vertical lines denote the point which training crosses the initial ADR threshold ϱ and the expansion begins.

[12] *Covariant — Solutions*, https://covariant.ai/solutions, Accessed: 2021-03-17, 2021.

[13] A. Zeng, S. Song, K.-T. Yu, E. Donlon, F. R. Hogan, M. Bauza, D. Ma, O. Taylor, M. Liu, E. Romo, *et al.*, "Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2018, pp. 3750–3757.

[14] P. van der Lugt, *Booming Bamboo: The (re)discovery of a sustainable material with endless possibilities*. Materia, 2017.

[15] *Hadrian x — outdoor construction & bricklaying robot from fbr*, https://www.fbr.com.au/view/hadrian-x, Accessed: 2021-03-17, 2021.

[16] *Technology — icon*, https://www.iconbuild.com/technology, Accessed: 2021-03-17, 2021.

[17] *Built robotics*, https://www.builtrobotics.com/, Accessed: 2021-03-17, 2021.

[18] L. Ding, W. Jiang, Y. Zhou, C. Zhou, and S. Liu, "Bim-based task-level planning for robotic brick assembly through image-based 3d modeling," *Advanced Engineering Informatics*, vol. 43, p. 100 993, 2020, ISSN: 1474-0346.

[19] S. Kim, M. Peavy, P.-C. Huang, and K. Kim, "Development of bim-integrated construction robot task planning and simulation system," *Automation in Construction*, vol. 127, p. 103 720, 2021, ISSN: 0926-5805.

[20] F. Augugliaro, S. Lupashin, M. Hamer, C. Male, M. Hehn, M. W. Mueller, J. S. Willmann, F. Gramazio, M. Kohler, and R. D'Andrea, "The flight assembled architecture installation: Cooperative construction with flying machines," *IEEE Control Systems Magazine*, vol. 34, no. 4, pp. 46–64, 2014.

[21] A. Thoma, D. Jenny, M. Helmreich, A. Gandia, F. Gramazio, and M. Kohler, "Cooperative robotic fabrication of timber dowel assemblies," in *Research Culture in Architecture: Cross-Disciplinary Collaboration*. 2020, pp. 77–88.

[22] S. Leder, H. Kim, O. S. Oguz, V. N. Hartmann, M. Toussaint, A. Menges, and M. Sitti, *Co-design in architecture: A modular material-robot kinematic construction system*, IROS: Workshop on Construction and Architecture Robotics, 2020.

[23] F. Furrer, M. Wermelinger, H. Yoshida, F. Gramazio, M. Kohler, R. Siegwart, and M. Hutter, "Autonomous robotic stone stacking with online next best object target pose planning," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2017, pp. 2350–2356.

[24] R. L. Johns, "Augmented materiality: Modelling with material indeterminacy," *Fabricate. Negotiating Design and Making*, pp. 216–223, Jan. 2014.

[25] K. H. Petersen, N. Napp, R. Stuart-Smith, D. Rus, and M. Kovac, "A review of collective robotic construction," *Science Robotics*, vol. 4, no. 28, 2019.

[26] A. Adel, A. Thoma, M. Helmreich, F. Gramazio, and M. Kohler, "Design of robotically fabricated timber frame structures," in *Recalibration: On Imprecision and Infidelity. Proceedings of the Conf. of the Association for Computer Aided Design in Architecture*, 2018, pp. 394–403.

[27] T. Bock, "Construction robotics," *Autonomous Robots*, vol. 22, no. 3, pp. 201–209, 2007.

[28] H. J. Wagner, M. Alvarez, O. Kyjanek, Z. Bhiri, M. Buck, and A. Menges, "Flexible and transportable robotic timber construction platform – TIM," *Automation in Construction*, vol. 120, p. 103 400, 2020.

[29] P. Eversmann, F. Gramazio, and M. Kohler, "Robotic prefabrication of timber structures: Towards automated large-scale spatial assembly," *Construction Robotics*, vol. 1, no. 1, pp. 49–60, 2017.

[30] M. Toussaint, J.-S. Ha, and D. Driess, "Describing physics for physical reasoning: Force-based sequential manipulation planning," *IEEE Robotics and Automation Letters*, 2020.

[31] Z. Wang, C. R. Garrett, L. P. Kaelbling, and T. Lozano-Pérez, "Active model learning and diverse action sampling for task and motion planning," in *Proc. of the IEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2018, pp. 4107–4114.

[32] A. Clegg, W. Yu, J. Tan, C. K. Liu, and G. Turk, "Learning to dress: Synthesizing human dressing motion via deep reinforcement learning," *ACM Trans. Graph.*, vol. 37, no. 6, Dec. 2018, ISSN: 0730-0301.

[33] S. Elliott and M. Cakmak, "Robotic cleaning through dirt rearrangement planning with learned transition models," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2018, pp. 1623–1630.

[34] Y. Zhang, W. Yu, C. K. Liu, C. Kemp, and G. Turk, "Learning to manipulate amorphous materials," *ACM Trans. Graph.*, vol. 39, no. 6, Nov. 2020, ISSN: 0730-0301.

[35] C. Schenck, J. Tompson, S. Levine, and D. Fox, "Learning robotic manipulation of granular media," in *Conference on Robot Learning*, PMLR, 2017, pp. 239–248.

[36] Y. Hu, J. Liu, A. Spielberg, J. B. Tenenbaum, W. T. Freeman, J. Wu, D. Rus, and W. Matusik, "ChainQueen: A Real-Time Differentiable Physical Simulator for Soft Robotics," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2019, pp. 6265–6271.

[37] M. Zwierzycki, P. Nicholas, and M. Ramsgaard Thomsen, "Localised and learnt applications of machine learning for robotic incremental sheet forming," in *Humanizing Digital Reality: Design Modelling Symposium*. Singapore: Springer Singapore, 2018, pp. 373–382.

[38] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, *Proximal policy optimization algorithms*, 2017. arXiv: 1707.06347 [cs.LG].

[39] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, Second. The MIT Press, 2018.

[40] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, *Asynchronous methods for deep reinforcement learning*, 2016. arXiv: 1602.01783 [cs.LG].