A. Implementation Details for Experiment Discovery

A.1. Planner

The Experiment Planner consisted of a uniform distribution planner for a horizon of 6 control signals. The planner was trained using the Cross Entropy Method Model Predictive Control (Camacho & Alba, 2013; De Boer et al., 2005) on the true environment. We sampled 30 plans per iteration from the distribution initialized to uniform $\mathcal{U}(controlLow, controlHigh)$. Each of the sampled plans are applied to each of the training environments and the top 10% of the plans are used to update the distribution. The CEM training required 10 iterations.

A.2. Training Environments

The training environments vary in each experiment. In Section 4.3, we utilize 3 setups, `Mass`, `SizeMass` and `ShapeSizeMass`. For `Mass`, we allow the agent to access 5 environments with masses varying from 0.1 kg to 0.5 kg. In `SizeMass`, the agent has access to 30 environments with masses varying uniformly from 0.1 to 0.5 kg and sizes from 0.05 to 0.1 meters. Finally, in `ShapeSizeMass`, the agent has access to 60 environments, with masses varying uniformly from 0.1 to 0.5 kg, sizes from 0.05 to 0.1 meters and shapes either being cubes or spheres. During experiment discovery, in each environment, the agent has access to the position of the block in the environment along with its quaternion orientation.

The total number of causal causal factors of each environment are rather large in number due to the fact that the simulator is a complex realistic physics engine. Examples of the causal factors in the environment include gravity, friction coefficients between all on interacting surfaces, shapes, sizes and masses of blocks, control signal frequencies of the environment. However, we only vary 1 during `Mass`, 2 during `SizeMass` and 3 during `ShapeSizeMass`.

A.3. Curiosity Reward Calculation

We predetermine the minimum description length of the clustering model $L(\mathbf{M})$ by assuming that the observations $\mathbf{o}_{0:T}$, obtained by applying experimental behavior $\mathbf{a}_{0:T}$ are produced by a bi-modal generator distribution, where each mode corresponds to either a low or high (quantized) value of a causal factor. This also ensures that $L(\mathbf{M})$ is as small as possible. The planner, eq. (7) solves the following optimization problem:

$$\begin{aligned}\arg\max_{\mathbf{a}_{0:T}\in\mathcal{A}^T}[&\min\{d(\mathbf{o}_{0:T},\mathbf{o'}_{0:T}):\mathbf{o}_{0:T}\in C_1,\mathbf{o'}_{0:T}\in C_2\}-\\&\max\{d(\mathbf{o}_{0:T},\mathbf{o''}_{0:T}):\mathbf{o''}_{0:T},\mathbf{o}_{0:T}\in C_1\}-\\&\max\{d(\mathbf{o'}_{0:T},\mathbf{o'''}_{0:T}):\mathbf{o'}_{0:T},\mathbf{o'''}_{0:T}\in C_2\}]\end{aligned} \tag{10}$$

the distance function $d(\cdot,\cdot)$ in the space of trajectories is set to be Soft Dynamic Time Warping (Cuturi & Blondel, 2017). The trajectory length T is 6 control steps long. The objective is a modified version of the Silhouette Score (Rousseeuw, 1987).

Intuitively, Objective (10) expresses the ability of a low complexity model, assumed to be bi-modal, to encode the state $\mathbf{O}=\mathbf{o}_{0:T}$. If multiple causal factors control $\mathbf{O}$, then the Minimum Description Length of $L(\mathbf{O})$ will be high. Subsequently, since $\mathbf{M}$ is a simple model, the deviation of $\mathbf{O}$ from $\mathbf{M}$ will be high i.e. $L(\mathbf{O}|\mathbf{M})$ will be high resulting in a low value of the optimization objective. C_1 and C_2 correspond to clusters of outcomes which quantize the values of a causal factor isolated by $\mathbf{a}_{0:T}$. $\mathbf{o}_{0:T},\mathbf{o''}_{0:T}\in C_1$ correspond to trajectories of states i.e. observations obtained by applying $\mathbf{a}_{0:T}$ to environments with say, low values of a causal factor while $\mathbf{o'}_{0:T},\mathbf{o'''}_{0:T}\in C_2$ correspond to trajectories of observations i.e. state obtained by applying $\mathbf{a}_{0:T}$ to environments with say, high values of the same causal factor. Objective (10) attempts to ensure that these clusters are far apart from each other and are tight i.e. a simple model $\mathbf{M}$ encodes $\mathbf{O}$ well.

B. Implementation Details for Transfer

In Section 4.4, we show the utility of learning causal representations in 2 separate experimental setups. During `TransferMass`, the agent has access to 10 environments during training, with masses ranging from 0.1 to 0.5 kg. At test time, the agent is required to perform the place-and-orient task masses 2 masses - 0.7 kg and 0.75 kg. During `TransferSizeMass`, the agent has access to 10 environments during training, with sizes from either 0.01 or 0.05 m and masses ranging from 0.1 to 0.5 kg. At test time the agent is asked to perform the task on 2 environments with masses 0.7 kg and 0.75 kg with sizes = 0.05 m.

We find that testing with large and light blocks increase the chances of accidental goal completions. Thus, during test-time, we use environments with high masses for out-of-distribution testing. The causal representation is concatenated to the state of the environment as a contextual input and supplied to a PPO-Optimized Actor-Critic Policy i.e., it receives 57 dimensional input for `TransferMass`, and a 58 dimensional for `TransferSizeMass`). The policy network consists of 2 hidden layers with 256 and 128 units respectively. The experiments are parallelized

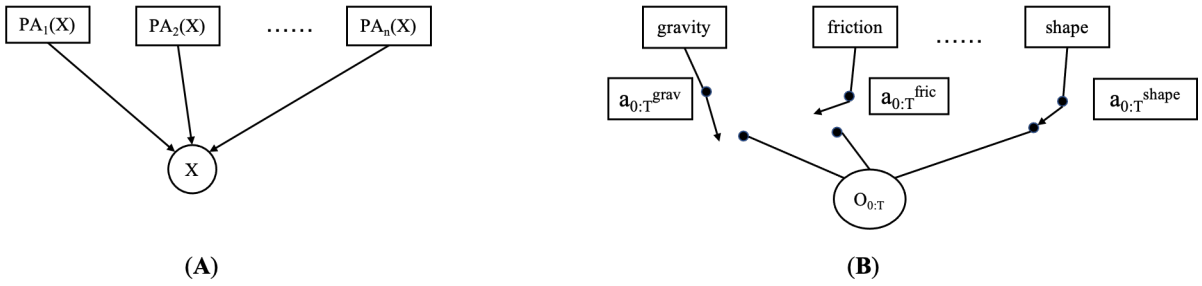

Figure 8. Directed Graphical Model. A Directed Acyclic Graph (DAG) visually represents the causal dependencies of observed and unobserved variables. In **(A)**, an observed variable $\mathbf{X}$ is caused by unobserved causal variables, $PA_i(\mathbf{X})$. In **(B)**, describes the scenario modeled in the paper, where a subset of the unobserved parent causal variables influence the observed variable $\mathbf{O}$. The action sequence $\mathbf{a}_{0:T}$ serves a gating mechanism, allowing or blocking particular edges of the causal graph.

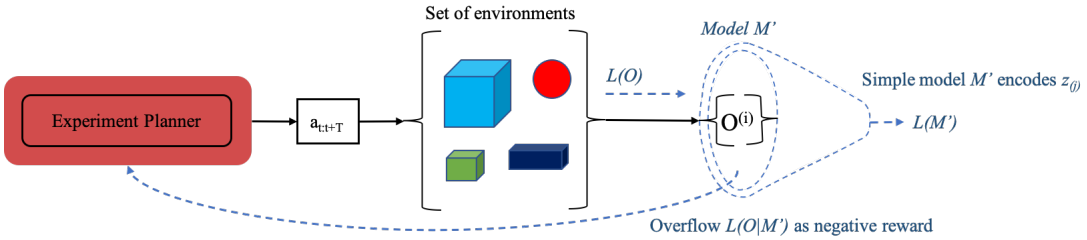

Figure 9. Overview of training. The experiment planner generates a trajectory of actions which is applied to each of the environments with varying causal factors namely mass, shape and size of blocks. For each environment, an observation trajectory or state $\mathbf{O}^i \in O$ is obtained. A simple model with fixed low expressive power is used to approximate the generative model for $\mathbf{O}$. The "information overflow" $L(\mathbf{O}|\mathbf{M})$ is returned as negative reward forcing O to be caused by few causal factors.

on 10 CPUs and implemented using stable baselines (Hill et al., 2018). The PPO configuration was {"gamma":0.9995, "n_steps": 5000, "ent_coef": 0, "learning_rate": 0.00025, "vf_coef": 0.5, "max_grad_norm": 10, "nminibatches": 1000, "noptepochs": 4}

The agent receives a dense reward at each time step during the maximizing external reward phase (Figure 1), the negative of the distance of the block from the goal position scaled by factor of 1000. The control signal was repeated 10 times to the actuators of the motors on each finger.

C. Implementation Details for Pre-trained Behaviors

In section 4.2, we study how the acquired experimental behaviors obtained through Causal Curiosity can be used as pre-training for a variety of downstream tasks. The Vanilla CEM depicts the cost of training an experiment planner from scratch to maximize an external dense reward where the agent minimizes the distance between the position of a block in an environment from the goal in the Lifting setup and imparts a velocity to the block along a particular direction in the Travel setup.

$$R(\mathbf{a}_{0:T}) = -\sum_t dist(\mathbf{goal}_t - \mathbf{block}_t) \tag{11}$$

The second baseline (Additive Reward) studies the setup when the agent receives both the curiosity signal and the external reward and attempts to maximize both. The agent receives access all the training environments with varying causal factors and must simultaneously maximize both curiosity and the task reward. The equation below shows the reward maximized for the `Lifting` task.

$$\begin{aligned} R(\mathbf{a}_{0:T}) = \sum_{envs}\sum_{t}^{T} -dist(\mathbf{goal}_t - \mathbf{block}_t) + \\ [\min\{d(\mathbf{o}_{0:T}, \mathbf{o'}_{0:T}) : \mathbf{o}_{0:T} \in C_1, \mathbf{o'}_{0:T} \in C_2\} - \\ \max\{d(\mathbf{o}_{0:T}, \mathbf{o''}_{0:T}) : \mathbf{o''}_{0:T}, \mathbf{o}_{0:T} \in C_1\} - \\ \max\{d(\mathbf{o'}_{0:T}, \mathbf{o'''}_{0:T}) : \mathbf{o'}_{0:T}, \mathbf{o'''}_{0:T} \in C_2\}] \end{aligned} \tag{12}$$

The curious agent first acquired the experimental behavior by interacting with multiple environments with varying causal factors. The lifting skill was obtained during `Mass`, when the agent attempted to differentiate between multiple blocks of varying mass. The curious agent trained for $600,000$ time steps on the curiosity reward. The acquired

behavior was then applied to the downstream lifting task and fine tuned to external rewards. The Vanilla CEM baseline had an identical structure to that of the Curious agent, and received only external reward as in Equation (11). The additive agent simultaneously optimized both external reward and the curiosity reward as in Equation (12).

We find that maximizing the curiosity reward in addition to simultaneously maximizing external rewards results in sub-optimal performance due to our formulation of the curiosity reward. To maximize curiosity, the agent must discover behaviors that divide environments into 2 clusters. Thus in the context of the experimental setups, this corresponds to acquiring a lifting/pushing behavior that allows the agent to lift/impart horizontal velocity to blocks in half of the environments, while not being able to do so in the remaining environments. However, the explicit external reward incentivizes the agent to lift/impart horizontal velocity blocks in all environments. Thus these competing objectives result in sub-par performance.

D. Intuition for Definition of Causal Factors

We begin with a simple example of a person walking on earth. This person experiences various physical processes while interacting in their world, for example gravity, friction, wind etc. These physical processes affect the outcome of interactions of the person with their environment. For example, while jumping on earth, the human experiences gravity which affects the outcome of their jump, the fact that they falls back to the ground. Additionally, these physical processes (or causal mechanisms) are parameterized by causal factors, for example, acceleration constant due to gravity $g = 9.8m/s^2$ on earth, or coefficients of friction between their feet and the ground which assume particular numerical values.

These causal factors may vary across multiple environments. For example, the person may walk on sand or on ice, surfaces with varying frictional values. Thus the outcome of running on such surfaces will vary, running on sand will require significant effort, while running on ice may result in the person slipping. Thus the coefficient of friction between the person's feet and the surface they walk on affects the outcome of a particular behavior in said environment. In our definition, $\mathbf{h}_j$ are causal factors such as friction or gravity etc. H is the global set containing all such causal factors.

Now we ask the question (which we subsequently answer), given multiple environments, how would a human characterize each of them depending on the value of a causal factor? Through experimental behaviors. The human in the above example would attempt to run in each of the environments she encountered, be it on sand, on ice, in mud etc. If they slipped in an environment, she would characterize it as slippery. If they didn't, they would characterize it as non-slippery. We attempt to equip our agent with similar logic. The "sequence of actions" ($\mathbf{a}_{0:T}$) described in our paper corresponds to the human running. The sequence of observations ($\mathbf{o}_{0:T}$) corresponds to the outcome of running "experiment". $\mathbf{o}_{0:T}$ might belong to either of the clusters of outcomes C_a or C_b corresponding to slipping or not slipping.

Algorithm 2 Inference Loop

Input: Unseen Test Environment env, trained Planner and Causal Inference Module
Initialize $\mathbf{causalRep} = [\,]$
Initialize training environment set $Envs$
for k in range(K) **do**
  Reset env
  Sample experimental behavior $\mathbf{a}_{0:T} \sim$ CEM($\cdot|\,\mathbf{causalRep}$)
  Apply $\mathbf{a}_{0:T}$ to env
  Collect $\mathbf{O} = \mathbf{o}_{0:T}$
  Use learnt $q_M(\mathbf{z}|\mathbf{O}, \mathbf{causalRep})$ for cluster assignment i.e. $\mathbf{z}_k = q_M(\mathbf{z}|\mathbf{O}, causalRep)$
  Append z_k to $causalRep$
end for
Learn a policy conditioned on causal factors $\mathbf{a}_t \sim \pi(\cdot|\mathbf{o}_t, \mathbf{z}_{0:K})$ to maximize external reward.

E. Scalability Limitation

We utilize the extremely popular One-Factor-at-a-time (OFAT) general paradigm of scientific investigation, as an inspiration for our method. In the case of many hundreds of causal factors, the complexity of this method will scale exponentially. However, we believe that this would indeed be the case given a human experimenter attempting to discover the causation in any system she is studying. Learning about causation is a computationally expensive affair. We point the reader towards a wealth of material on the design of scientific experiments and more specifically the lack of scalability of OFAT (Fisher, 1936; Hicks, 1964; Czitrom, 1999). Nevertheless, OFAT remains the de facto standard for scientific investigation.