Causal Newton Optimization: Online Calibration with Iterative Local Linear Modeling and Newton Updates *

Daigo Fujiwara[1,3] , **Tomonori Izumitani**[1] and **Shohei Shimizu**[2,3]
[1]NTT DOCOMO BUSINESS, Inc.
[2]The University of Osaka
[3]Shiga University
{d.fujiwara, tomonori.izumitani}@ntt.com, shimizu.shohei@gmail.com

Abstract

Optimization in industrial systems often involves calibrating from a semi-optimized state, where global exploration methods like Reinforcement Learning (RL) or Bayesian Optimization (BO) are inefficient or unsafe. We propose Causal Newton Optimization (CNO), an online algorithm that iteratively calibrates inputs under a known causal graph but unknown structural equations. CNO estimates local linear causal effects via additive interventions and employs a log-linear variance regression to robustly guide Newton-based updates. Evaluations on synthetic systems and a chemical plant simulator demonstrate that CNO achieves the best balance between objective improvement and robustness. While traditional PID control suits standard dynamical systems, CNO significantly outperforms RL and BO in complex structural causal models, providing the robust stability vital for safety-critical real-world applications.

1 Introduction

Optimization under Real-World Setting. In many industrial and economic systems, optimization operations are rarely started from scratch. Instead, they typically operate in a *semi-optimized state*, governed by existing mechanisms such as PID controllers and professional manual control in chemical plants, or established policy guidelines in economics. However, due to complex nonlinear dynamics and unmodeled exogenous disturbances, these systems often suffer from persistent deviations (e.g., steady-state errors or oscillations) from the ideal target [Åström and Hägglund, 1995]. Operators attempt to mitigate these errors through manual tuning, typically analyzing operation logs accumulated over a fixed period and adjusting parameters at regular intervals (e.g., monthly maintenance). This scenario, where calibration is performed after collecting a certain amount of data at a fixed frequency, is standard and natural in process industries such as chemical plants. Relying on trial-and-error in this periodic process is labor-intensive and inconsistent.

To automate this optimization, methods such as Reinforcement Learning (RL) and Bayesian Optimization (BO) are potential candidates. However, these global optimization approaches generally face the *exploration-exploitation trade-off*, requiring extensive exploration to learn the global landscape. In safety-critical systems like chemical plants or healthcare, such exploratory behaviors, especially *hard interventions* that fix inputs and break existing semi-optimization, are often unacceptable due to safety risks and potential economic losses. Furthermore, real-world systems are often non-stationary; dynamics drift due to equipment aging (e.g., catalyst degradation) or seasonal shifts. Global models (e.g., Gaussian Processes in BO) often struggle to adapt quickly to these shifts and suffer from scalability issues regarding historical data storage.

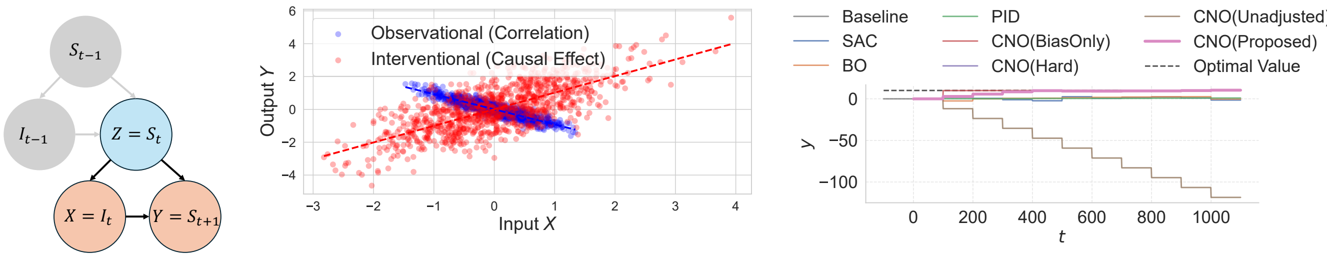
Confounding Bias in Feedback Control. A critical challenge in optimizing such semi-optimized systems is the presence of confounders—variables that influence both the manipulated input and the target output. For instance, in a typical feedback control system, a past system state or an external disturbance affects both the current control input (via existing feedback mechanisms) and the future state of the system (Figure 1a). Ignoring this underlying causal structure leads to biased estimates of the input's true effect on the output (Figure 1b). If an optimizer relies merely on observational correlations rather than causal effects, at worst this confounding bias can flip the apparent direction of the effect, driving a naive optimization algorithm in the completely wrong direction (Figure 1c). We will formally illustrate this phenomenon using a concrete mathematical example in Section 4.

1.1 Proposed Approach

To address these challenges, we propose *Causal Newton Optimization (CNO)*. Unlike global explorers, CNO frames the problem as the *online calibration* of a continuous input variable to fine-tune an existing system. Our approach rests on three key pillars justified by the calibration setting:

Additive Intervention for Safety. Instead of replacing the existing control logic (hard intervention), we employ *additive interventions* that superimpose a correction term onto the natural input value. This preserves the system's inherent response mechanisms (e.g., safety feedback loops) and leverages natural fluctuations for estimation, significantly enhancing safety compared to exploration without prior knowledge.

*The supplementary appendices and full proofs are available at: https://github.com/nttcom/CNO-Supplementary.



(a) Causal graph (b) Confounding bias (c) Optimization results

Figure 1: Example of confounding bias in a feedback control system. (a) The state S_t is a confounder of the next input I_t and the next state S_{t+1}. (b) Observational data from the original Linear SCM and interventional data under overridden random inputs. Dashed lines indicate coefficients from the Ordinary Least Squares. (c) CNO(Unadjusted) diverges due to confounding bias, while CNO(Proposed) converges.

Moreover, CNO's local linear modeling provides high interpretability, allowing operators to understand and trust the optimization process.

Newton's Method for Fast Convergence. We posit that in the neighborhood of a semi-optimal state, system dynamics can be well-approximated by local linear models. By leveraging this inductive bias, CNO estimates local gradients and utilizes Newton's method. This enables CNO to achieve fast convergence based on Newton's method's quadratic characteristics, minimizing the duration of suboptimal operation and adapting rapidly to system drifts without the heavy computational burden of maintaining global models.

Causal Inference for Confounding Adjustment. A critical challenge in this setting is *confounding*. When using additive interventions, the input remains correlated with upstream variables (confounders). For instance, external temperature might affect both the control input (via existing feedback) and the target outcome. Ignoring this causal structure leads to biased gradient estimates potentially driving the optimization in the wrong direction (as demonstrated in our motivating example). CNO explicitly addresses this by using a known causal graph to perform back-door adjustment, ensuring valid gradient estimation even under confounding.

In this paper, we propose CNO, an algorithm that iteratively estimates local linear causal effects on expectation and variance from online data to update interventions. We introduce a log-linear variance regression to robustly handle heteroscedasticity. Experiments on synthetic nonlinear systems and a continuous stirred-tank reactor (CSTR) simulator demonstrate that CNO achieves faster convergence and higher robustness compared to RL, BO and PID (a standard feedback control method).

2 Related Work

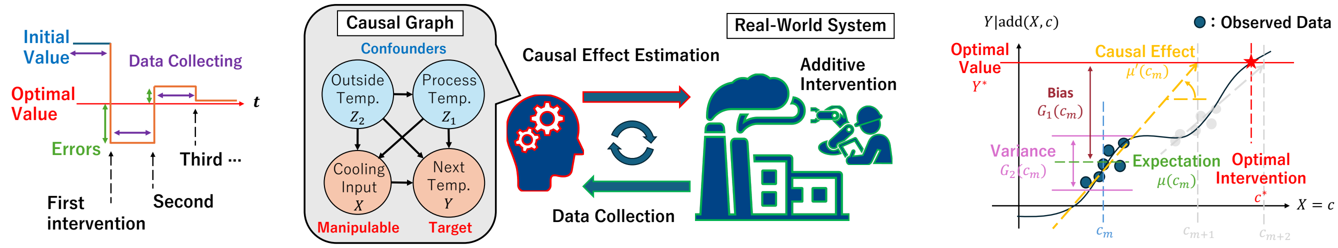
2.1 Online Optimization & Control

Reinforcement Learning (RL), such as Soft Actor-Critic (SAC) [Haarnoja *et al.*, 2018], learns optimal policies for dynamical environments through interaction without explicit system models. While flexible as black-box solvers, RL algorithms often suffer from sample inefficiency, requiring sensitive tuning of numerous hyperparameters and manual reward function design. They typically need many trial-and-error episodes, often requiring pre-training in simulators to achieve acceptable performance in real-world systems. Bayesian Optimization (BO) [Frazier, 2018] optimizes black-box objective functions globally by building probabilistic surrogate models, typically Gaussian Processes (GPs). BO focuses on balancing the exploration-exploitation trade-off and can be sample-efficient by explicitly modeling the objective function. However, BO typically scales poorly with high-dimensional parameter spaces and increasing historical data, and it requires careful kernel selection for accurate modeling. Bandit algorithms [Lattimore and Szepesvári, 2020] also address the exploration-exploitation trade-off but often assume discrete action spaces, making them less effective for continuous parameter tuning. Furthermore, since RL, BO, and Bandits fundamentally require exploratory behavior, safety guarantees are often challenging in real-world applications such as plant engineering. Classical control theories, such as PID control [Åström and Hägglund, 1995], typically assume a known dynamical system structure or rely on feedback error correction without learning from online data distributions. Their primary aim is to maintain system stability and track setpoints (exploitation). While Adaptive Control [Astrom and Wittenmark, 1994] can learn data online, and Model Predictive Control (MPC) [Camacho *et al.*, 2007] explicitly can use pre-trained models of dynamics, they generally require accurate system models and may struggle with unmodeled dynamics or complex disturbances.

CNO bridges the gap between these fields and causal inference. Unlike RL, BO, and Bandit, CNO leverages the *known causal graph* and the inductive bias of *local linear approximations* around a semi-optimal state, resulting in high sample efficiency without extensive exploration phases. Unlike PID, CNO explicitly handles confounding bias through back-door adjustment, proactively learning from online data to correct steady-state error that feedback control alone cannot resolve. Moreover, CNO's local linear modeling allows high interpretability and low computational overhead, adapting to trends and distribution shifts.

2.2 Causal Inspired Optimization

Recently, methods integrating causal inference with online optimizations are emerging. Surveys are available in [Zeng



(a) Repeating interventions (b) Problem setting overview (c) Local estimation and Newton's method

Figure 2: Overview of the proposed method. (a) Optimization reducing errors by repeated interventions. (b) Problem settings of optimization. Collected data by intervention is available online. (c) Combination with local linear estimation of causal effects and Newton's method.

et al., 2024; Subramanian and Ravindran, 2025].

Causal Optimization for Efficient Intervention Exploration. Lattimore *et al.* [2016] extended the multi-armed bandit problem by considering causal relationships between arms to identify optimal interventions. Similarly, Causal Bayesian Optimization (CBO) [Aglietti *et al.*, 2020; Gultchin *et al.*, 2023] propose causal-aware Gaussian Processes model to simultaneously identify the optimal intervention set and values. These methods primarily focus on the *exploration-exploitation trade-off* and address the combinatorial explosion of intervention variables utilizing known causal relationships. In contrast, CNO targets the *calibration* of a specific continuous variable in a semi-optimized system, focusing on fast convergence rather than global exploration.

Causal Optimization for Adjustment of Confounding. Bareinboim *et al.* [2015] focused on adjusting confounding bias in multi-armed bandit problems. They demonstrated that conventional bandit algorithms perform sub-optimally when unobserved confounders influence both the intervention (arm selection) and the outcome. Crucially, they utilized not only interventional data but also **natural observational data** to extract information about unobserved confounders, improving algorithm efficiency. CNO shares the aim of adjusting confounding bias and utilizing natural observational data (via additive interventions). However, while Bareinboim *et al.* address the discrete arm selection problem with exploration-exploitation, CNO focuses on continuous variables tuning with *fast convergence*, and uses inductive bias from local linear approximations around a semi-optimal state.

3 Proposed Method

3.1 Problem Setting

We assume that the data consist of continuous variables and follow a structural causal model (SCM). In an SCM, the causal generative process of the data is described by structural equations, and the causal relationships are represented by a directed graph [Pearl, 2009; Pearl *et al.*, 2016]. We assume a known, acyclic causal graph $\mathcal{G}$ with unknown functional forms of the structural equations. For $P \in \mathbb{N}$ variables, the structural equations are:

$$V_i = f_i(\mathrm{pa}_i, U_i) \quad (i = 1, 2, \dots, P), \tag{1}$$

where pa_i is the set of direct causes (i.e. parents) of V_i given from the causal graph $\mathcal{G}$, and f_i is an unknown deterministic function. The exogenous variable U_i is a random variable determined independently of the structural equations and representing unobserved disturbances. The endogenous variable V_i is a random variable generated from structural equations. Let $\boldsymbol{V} = \{V_i \mid i = 1, 2, \dots, P\}$. We define $Y\ (\in \boldsymbol{V})$ as a target variable for optimization and $X\ (\in \boldsymbol{V})$ as a variable manipulable through interventions. We evaluate Y against a constant reference value (setpoint) $Y^* \in \mathbb{R}$ using the standard squared error, $L(Y) = (Y - Y^*)^2$.

Calibration Setting via Additive Intervention. We assume the system is currently operating in a *semi-optimized state* governed by existing mechanisms (e.g., manual control or feedback loops)[Åström and Hägglund, 1995; Marlin and Marlin, 1995; Sinha and Mishra, 2018].

Assumption 1 (Calibration Setting for Semi-Optimized)**.** *Let X_{nat} be the natural value of X determined by the structural equation $X_{nat} = f_X(pa_X, U_X)$. We assume X_{nat} is semi-optimized for $L(Y)$, e.g. minimization of its expectation, $\min_{f_X}(\mathbb{E}[L(Y)|\,\mathrm{do}(X = X_{nat})])$. Then the outcome under the natural state, $Y|do(X = X_{nat})$, lies within a neighborhood of Y^* where $G(c)$ is locally convex and the Gauss-Newton approximation holds.*

Rather than the standard hard interventions, we calibrate the existing X_{nat} using an **additive intervention** [Pearl *et al.*, 2016; Sani *et al.*, 2020] with a correction term $c \in \mathbb{R}$:

$$\mathrm{add}(X, c) := \mathrm{do}(X = X_{\mathrm{nat}} + c). \tag{2}$$

This intervention in (2) shifts the mean by c, preserving the dependence of X on exogenous noises U_X and parents pa_X, while standard hard interventions fix X to a constant a and remove the causal links from parents to X. This setting ensures safety by maintaining the system's reaction to disturbances and allows us to leverage the preserved natural variance of X_{nat} for estimating local causal effects. Furthermore, since the system is in a neighborhood of an optimal state, the nonlinearity of the system is often relatively relieved. This property justifies the use of local linear approximations for online estimation. The optimization of the correction term c is formulated as:

$$c^* = \operatorname*{argmin}_{c \in \mathbb{R}}\{G(c)\},\ G(c) = \mathbb{E}\left[L\left(Y\right) \mid \mathrm{add}\left(X, c\right)\right] \tag{3}$$

Since f_i and U_i are unknown, the expectation in (3) cannot be computed analytically. We consider an online setting where system observations are obtained at time steps $t \in \mathbb{N}$:

$$\boldsymbol{v}^{(t)} = \{v_1^{(t)}, v_2^{(t)}, \ldots, v_P^{(t)}\}.$$

However, the correction term c is not updated at every step; instead, c is held constant over a fixed interval $T \in \mathbb{N}$, to collect multiple data points for estimation. We assume that the data points $\boldsymbol{v}^{(t)}$ are i.i.d. samples generated from a time-invariant SCM under the additive intervention. Thus, the distributions of variables depend on time t only through the updated correction term c. Let $m = 0, 1, \ldots$ be the iteration index. The data batch D_m collected under the m-th correction term c_m is defined as:

$$D_m = \{\boldsymbol{v}^{(t)} \mid t = mT+1, \ldots, (m+1)T \; ; \; \mathrm{add}(X^{(t)}, c_m)\}.$$

After obtaining D_m, we estimate local statistics to calculate the next correction term c_{m+1}. c_{m+1} will be applied to the system to collect the subsequent batch D_{m+1}.

3.2 Derivation of Optimization Algorithm

Newton's method. We employ a Newton-based algorithm to solve the optimization problem in (3). Defining the derivative of the objective function as $g(c) := G'(c)$, Newton's method seeks the root $g(c) = 0$ via the following update rule:

$$c_{m+1} = c_m - \frac{g(c_m)}{g'(c_m)}. \tag{4}$$

A Newton-based approach provides clear interpretability of the algorithm's behavior, theoretical basis of fast convergence (approximately quadratic). Moreover it aligns with the linear causal modeling ((9), etc.). See Supplementary Material Appendix A, and [Süli and Mayers, 2003] for detailed assumptions and the statement.

Local Convexity. Although global convexity of $G(c)$ is needed in Newton's method optimization, it is not guaranteed for general nonlinear functions. However, our problem setting calibration of an existing optimization (Assumption 1) justifies the assumption of local convexity.

Lemma 1 (Local Convexity under Calibration Setting)**.** *Under the assumption of semi-optimized setting (Assumption 1), the initial correction term $c_0 = 0$ is within a sufficiently small neighborhood of strict local optimum of c^*. In this region, the objective function $G(c)$ is assumed to be convex and twice differentiable. Under this assumption, the standard convergence properties of Newton's method apply (see Supplementary Material Appendix B for details).*

Decomposition of Objective Function. The expectation in (3) can be transformed by an approach analogous to the standard bias-variance decomposition:

$$\begin{aligned} G(c) &:= \mathbb{E}\left[(Y - Y^*)^2 \mid \mathrm{add}(X, c)\right] \\ &= \underbrace{(Y^* - \mu(c))^2}_{G_1(c)\text{: Bias }^2} + \underbrace{\mathbb{V}[Y \mid \mathrm{add}(X, c)]}_{G_2(c)\text{: Variance}}, \end{aligned} \tag{5}$$

where $\mu(c) = \mathbb{E}\left[Y \mid \mathrm{add}\left(X, c\right)\right]$. Note that if all structural equations f_i are linear, the variance $G_2(c)$ remains constant regardless of c, while general nonlinear structural equations lead to $G_2(c)$ varying with c.

Since the distribution $p(Y \mid \mathrm{add}(X, c))$ is unknown, we cannot analytically compute $G(c)$ or its derivatives. In our proposed method, we approximate and estimate the terms $g(c_m) = g_1(c_m) + g_2(c_m)$ and $g'(c_m) = g'_1(c_m) + g'_2(c_m)$ to compute the update in (4).

Derivatives of Squared Bias. We first consider the squared bias term $G_1(c)$ and its derivatives.

Lemma 2 (Approximation of the Derivative $g'_1(c)$)**.** *Under the calibration setting (Assumption 1), $\mu(c)$ is expected to be sufficiently close to Y^* (i.e., the bias is small) such that $(\mu(c) - Y^*)\mu''(c)$ is negligible compared to $\mu'(c)^2$. Therefore, the following equation and approximation hold:*

$$G'_1(c) = g_1(c) = 2(\mu(c) - Y^*)\mu'(c), \tag{6}$$

$$g'_1(c) = 2\left(\mu'(c)^2 + (\mu(c) - Y^*)\,\mu''(c)\right) \simeq 2\mu'(c)^2. \tag{7}$$

The approximation (7) *ignores second-order derivative, so aligns with the linear causal modeling* ((9)*, etc.).*

Remark. The approximation in (7) is equivalent to the strategy employed in the Gauss-Newton method (see [Wright *et al.*, 1999] for details), which is widely established for solving nonlinear least squares problems. While the standard Gauss-Newton method is typically designed for static optimization with known parametric functions, our approach extends this framework to an online stochastic optimization setting where functions are unknown and must be estimated via causal inference.

Local Causal Effect Estimation. To calculate (6) and (7), we need to estimate the derivative $\mu'(c) = \frac{\partial}{\partial c}\mathbb{E}[Y \mid \mathrm{add}(X, c)]$, which represents the local causal effect of X on Y under the additive intervention. Standard causal inference typically assumes hard interventions $\mathrm{do}(X = a)$, which fix X to a specific value. While each derivative estimated under additive and hard interventions differs in general nonlinear cases, it becomes equivalent in linear models.

Lemma 3 (Equivalence of Linear Hard and Additive Causal Effects)**.** *Under the setting of a constant linear causal effect $w_{YX} = \frac{\partial}{\partial a}\mathbb{E}[Y \mid \mathrm{do}(X = a)]$, the local causal effect of X on Y under hard interventions is equal to that under additive interventions:*

$$\begin{aligned} \frac{\partial}{\partial c}\mathbb{E}[Y \mid \mathrm{add}(X, c)] &= \frac{\partial}{\partial c}\mathbb{E}[Y \mid \mathrm{do}(X = X_{nat} + c)] \\ &= \frac{\partial}{\partial a}\;\mathbb{E}[Y \mid \mathrm{do}(X = a)]|_{a = X_{nat} + c} = w_{YX}. \end{aligned} \tag{8}$$

Thus, under the assumption of *local linearity* (discussed in Lemma 4), we can approximate the causal effect under additive interventions using the standard back-door adjustment (discussed in Proposition 1) under hard interventions.

In linear SCMs, the causal effect w_{YX} is constant and can be estimated via the back-door adjustment [Pearl, 2009; Pearl *et al.*, 2016]. In the back-door adjustment, using a set of covariates $\boldsymbol{Z}$ that satisfies the back-door criterion as explanatory variables in addition to X, the Ordinary Least Squares (OLS) can consistently estimate the causal effect. Specifying the back-door criterion needs the known causal graph $\mathcal{G}$. Refer to the detailed definition in Supplementary Material Appendix C.

Proposition 1 (Back-door Adjustment in Linear Case)**.** *Assume a linear SCM with known causal graph where variables $\boldsymbol{Z} \subset \boldsymbol{V} \setminus \{X, Y\}$ satisfy the back-door criterion relative to (X, Y). Consider the linear regression of Y on X and $\boldsymbol{Z}$:*

$$Y = w_0 + w_{YX} X + \boldsymbol{w}_{Y\boldsymbol{Z}}^\top \boldsymbol{Z} + \epsilon, \tag{9}$$

where we assume $\mathbb{E}[\epsilon] = 0$ (absorbed into w_0). Then, the regression coefficient w_{YX} identifies the constant causal derivative $w_{YX} = \frac{\partial}{\partial a}\mathbb{E}\left[Y \mid \mathrm{do}\left(X = a\right)\right]$. The OLS estimator $\hat{w}_{YX}$ is consistent for w_{YX} as the sample size increases [Stock and Watson, 2015].

Local Linearity of Structural Equations. Since the structural equations f_i in (1) are generally nonlinear, we employ a *local linear approximation* around the current intervention value c_m.

Lemma 4 (Linearization over Structural Equations)**.** *Assuming f_i are twice differentiable, the deviation of $\boldsymbol{\xi} = (pa_i, U_i)^T$ and Hessian $\nabla^2 f_i$ is sufficiently small, the entire SCM can be approximated by a linear model locally. (see Supplementary Material Appendix D for formal conditions and proofs)*

Considering all the above discussion, the approximated estimators for $\mu'(c)$ are given as $\hat{\mu}'_m := \hat{\mu}'(c_m) = \hat{w}_{YX}$. The expectation $\mu(c_m)$ is estimated by the sample mean of observed Y values under additive interventions $\mathrm{add}(X, c_m)$, $\hat{\mu}_m := \frac{1}{T}\sum_{t=mT+1}^{(m+1)T} y^{(t)}$. Considering (6) and (7), we get following estimators:

$$\hat{g}_{1,m} = 2(\hat{\mu}_m - Y^*)\hat{\mu}'_m, \quad \hat{g}'_{1,m} = 2\hat{\mu}'^2_m. \tag{10}$$

Approximated Derivatives of Variance

Estimating the derivatives of the variance term, $g_2(c)$ and $g'_2(c)$, is challenging. The naive finite differences often leads to numerical instability, particularly when the intervention step size $|c_m - c_{m-1}|$ is small in the convergence stages of the algorithm. To overcome this, we propose a robust estimation method based on a local log-linear approximation.

Instead of estimating variance and derivatives directly, we assume the conditional variance locally follows a log-linear relationship with the intervention c, i.e., $\ln G_2(c) \approx \omega_0 + \omega c$. Under this assumption, the derivatives satisfy $g_2(c) = \omega G_2(c)$ and $g'_2(c) = \omega^2 G_2(c)$ (see Supplementary Material Appendix E for derivation). This formulation ensures consistency with the positivity of variance. To estimate the coefficient ω, we employ heteroscedastic regression using the residuals $r^{(t)} = y^{(t)} - \hat{w}_0 - \hat{w}_{YX} x^{(t)} - \hat{\boldsymbol{w}}_{Y\boldsymbol{Z}}^\top \boldsymbol{z}^{(t)}$. Note that, although the sample mean $\hat{\mu}_m$ is sufficient for an unbiased estimation of $g_1(c)$ and biased but stable estimation of $g'_1(c)$, using it for residual estimation introduces crucial bias due to the local mean shift. Therefore, we use regression-based residuals (see Supplementary Material Appendix F for details). Thus, we solve the following regression problem:

$$\ln((r^{(t)})^2) = \omega_0 + \omega_{YX} x^{(t)} + \boldsymbol{\omega}_{Y\boldsymbol{Z}}^\top \boldsymbol{z}^{(t)} + \varepsilon^{(t)}. \tag{11}$$

Using the same adjustment set Z as in the regression, confounding biases are theoretically adjusted for the variance as well [Pearl, 2009; Pearl *et al.*, 2016] (see Supplementary Material Appendix G). Moreover, the OLS (11) provides a consistent and unbiased estimator of the local variance coefficient ω_{YX}, although biased estimator for the intercept ω_0 and the variance $G_2(c)$ (see Supplementary Material Appendix H). Finally, combining $\hat{\omega}_{YX}$ with the current variance estimate $\hat{G}_2(c_m) := \frac{1}{T}\sum_{t=mT+1}^{(m+1)T}(r^{(t)})^2$, we compute the derivatives for the Newton update as:

$$\hat{g}_{2,m} = \hat{\omega}_{YX}\hat{G}_2(c_m), \quad \hat{g}'_{2,m} = \hat{\omega}^2_{YX}\hat{G}_2(c_m). \tag{12}$$

We provide a detailed discussion about other variance estimators in Supplementary Material Appendix J.

Resulting Update Equation

Considering (4) $\sim$ (12), the algorithm we developed for the proposed method is given as the following update equation:

$$c_{m+1} = c_m - \frac{\hat{g}_{1,m} + \hat{g}_{2,m}}{\hat{g}'_{1,m} + \hat{g}'_{2,m}}. \tag{13}$$

Note that initial correction term is set as $c_0 = 0$. If the structural equations are completely linear or ignoring $G_2(c)$, where $\hat{g}_{2,m} = \hat{g}'_{2,m} = 0$, we can obtain the following more interpretable form: $c_{m+1} - c_m = \frac{Y^* - \hat{\mu}_m}{\hat{\mu}'_m}$. This form indicates the division of the offset $Y^* - \hat{\mu}_m$ by the local causal effect $\hat{\mu}'_m$, which is illustrated in the first concept of Figure 2c. The pseudo-code of the entire algorithm is summarized in Supplementary Material Appendix I.

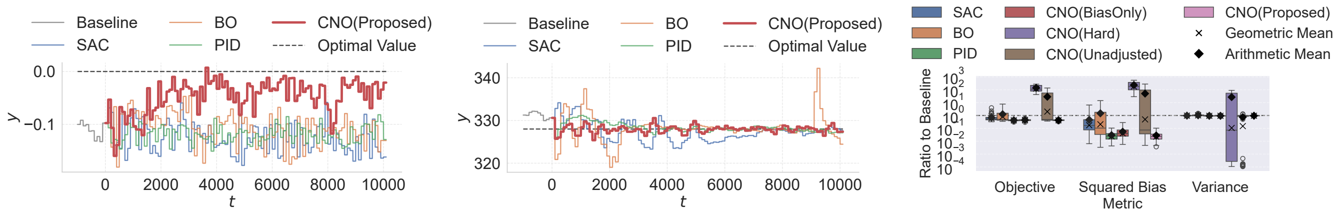
4 Numerical Experiments

To evaluate the effectiveness of our proposed method, Causal Newton Optimization (CNO), we conducted a series of experiments across three distinct environments with varying complexities. We compare CNO against representative algorithms from system control and online optimization. In every experiment, we set the time interval for each intervention as $T = 100$.

Linear SCM: Confounding Bias in Feedback Control. To formally illustrate the confounding bias discussed in Section 1, we evaluate a simple linear SCM representing the feedback control system depicted in Figure 1a. Consider a state variable to be controlled $S^{(t)}$. In a typical semi-optimal steady-state, $S^{(t)}$ fluctuates around a constant $\tilde{r}$ (= a reference value r + steady-state error). The system dynamics, including the feedback control input $I^{(t)}$, can be modelled as:

$$\begin{cases} S^{(t)} \simeq \tilde{r} + U_S^{(t)} & \text{(Steady-state)} \\ I^{(t)} = -K_p(S^{(t)} - r) + U_I^{(t)} & \text{(Feedback)} \\ S^{(t+1)} = \alpha I^{(t)} + \beta S^{(t)} + U_S^{(t+1)} & \text{(Dynamics)} \end{cases}$$
$$\simeq \begin{cases} Z = U_Z & (U_Z \sim \mathcal{N}(0, \sigma^2_{U_Z})) \\ X = -K_p Z + U_X & (U_X \sim \mathcal{N}(0, \sigma^2_{U_X})) \\ Y = \alpha X + \beta Z + U_Y & (U_Y \sim \mathcal{N}(0, \sigma^2_{U_Y})) \end{cases},$$

where $K_p = 0.5$, $\alpha = 1.0$, $\beta = 1.0$, $\sigma_{U_Z} = 1.0$, $\sigma_{U_X} = 0.1$, and $\sigma_{U_Y} = 0.1$, $r = Y^* = 10.0$. For simplicity, variables are assumed centered. We aim to optimize the output $Y = S^{(t+1)}$ by manipulating the input $X = I^{(t)}$, adjusting for $Z = S^{(t)}$.



(a) Complex Synthetic Environment (1 trial) (b) CSTR simulator (1 trial) (c) CSTR steady-state results (100 trials)

Figure 3: Numerical experiment results. (a)(b) Target variable Y under different algorithms in a trial. Random seeds are fixed across algorithms. (c) Box-and-whisker plot showing the Ratio of Objective Functions for each algorithm in steady-state.

While the true interventional coefficient from $X \to Y$ is α, the unadjusted estimation yields the observational coefficient:

$$\hat{\alpha} = \frac{\mathrm{Cov}(X,Y)}{\mathbb{V}(X)} = \frac{\alpha\sigma_{U_X}^2 - K_p\left(\beta - \alpha K_p\right)\sigma_{U_Z}^2}{\sigma_{U_X}^2 + K_p^2\sigma_{U_Z}^2} \simeq \alpha - \frac{\beta}{K_p}$$

This approximation holds when $\sigma_{U_Z} \gg \sigma_{U_X}$. Crucially, when $\beta/K_p > \alpha$, which holds for the adopted parameters, this apparent coefficient becomes negative ($\hat{\alpha} < 0$), mathematically demonstrating the confounding bias (Figure 1b). The total number of interventions is $M = 10$.

Complex Synthetic Environment. We randomly generated 100 different SCMs with $P = 10$ variables and random Directed Acyclic Graphs (DAG), where each edge is created with a probability of $p = 0.3$ after arranging a random causal order to prevent cycles. If there is no back-door set under a generated graph, the graph is rejected and resampled. The functional forms in the structural equations consisted of linear combinations of parental variables and their polynomials, scaled to be order 1, as follows:

$$\begin{aligned}
&V_i = f_i(pa_i) + U_i \quad (i = 1, 2\ldots, P)\\
&f_i(pa_i) = \begin{cases} f_{\mathrm{pol}}\left(\frac{1}{|pa_i|}\sum_{V_j\in \mathrm{pa}_i} b_{ij}V_j\right) & (\text{if } |pa_i| > 0)\\ \frac{a_{i0}}{K+1} & (\text{if } |pa_i| = 0)\end{cases}\\
&f_{\mathrm{pol}}(v) = \frac{1}{K+1}\sum_{k=0}^{K}\frac{a_{ik}}{k!}v^k, \quad U_i \sim \mathcal{N}(0, \sigma_i^2), \qquad (14)
\end{aligned}$$

where a_{ik} and b_{ij} are constant values sampled from the distribution: $\mathrm{Uniform}([-1, -0.5] \cup [0.5, 1])$, and $K = 3$ as the range of Y is $[-\infty, \infty](\ni Y^*)$. σ_i is randomly sampled from $\mathrm{Uniform}([0.1, 0.3])$ so that $3\sigma_i \leq 1$, guaranteeing $P(|U_i| \leq 1) > 0.997$. The total number of interventions is set as $M = 100$, the optimal value is set as $Y^* = 0$ for scale-free evaluation. Note that in this environment, X_{nat} is not in a semi-optimized state under a existing mechanism, but randomly sampled. This setup tests the algorithms' robustness to varying and complex nonlinear relationships and violations of Assumption 1.

CSTR Simulator. To assess practical applicability in real-world scenarios, we used a Continuous Stirred-Tank Reactor (CSTR) simulator, a well-known benchmark in process control and machine learning tasks with strong nonlinear dynamics [Uppal *et al.*, 1974; Marlin and Marlin, 1995; Sinha and Mishra, 2018]. The implementation used python control library [Fuller *et al.*, 2021]. In this environment, the goal is to regulate the reactor temperature by manipulating the cooling jacket temperature. We set the situation that PID control is already applied but the accuracy is not sufficient. PID control is a method typically suitable for linear systems, not for nonlinear systems such as CSTR, thus this setting is reasonable. Algorithms can start from steady-state by using a stabilization period without interventions. Note that this setting is consistent with the assumption of semi-optimized state (Assumption 1). See Supplementary Material Appendix N for details. The total number of interventions is set as $M = 100$, the optimal value is set as $Y^* = 328.0$[K] under physical limitations.

Compared Methods. Some representative methods from classical system control and modern online optimization were selected to compare with our proposed method, CNO. Compared methods are applied to under the same conditions as CNO, such as the same T, M, and additive intervention settings. **(Outer Loop) PID Controller:** As a representative of classical system control, we implemented a velocity-form PID controller. PID control gain parameter was well hand-tuned. Further details are provided in Appendix M. Note that in CSTR Simulator, another PID control with different parameters and a different intervention interval is already applied to the plant. **Soft Actor-Critic (SAC):** As a state-of-the-art model-free reinforcement learning method, we used SAC. The specific architecture and reward function design are described in Appendix L. **Bayesian Optimization (BO):** We also implemented a BO method using Gaussian Process (GP) regression as a surrogate model and Lower Confidence Bound (LCB) as an acquisition function. Note that SAC and BO have sensitive intervention range parameters. We use $\pm 5\sigma$ (σ is standard deviation) of initial state X_{nat} as the intervention range. **Proposed Methods (CNO Variants):** To analyze the contribution of each component of our proposed method, we evaluate several CNO variants with different variance estimators and intervention types (hard or additive). Note that, hard interventions require excitation noise to identify the causal effect, while additive does not. And there are variants that do not perform back-door adjustment (Unadjusted) and that only adjust squared bias $G_1(c)$ (BiasOnly).

Evaluation Metrics. We assess the performance of all algorithms using the following metrics. **Objective Function, Squared Bias, Variance**: We evaluate the time-averaged values of the objective function $G(c)$, which is equivalent to the

	Complex Env.					CSTR Env.	
	Obj. Ratio (SS) ↓	Bias^2 Ratio (SS) ↓	Var. Ratio (SS) ↓	Imp. ↑	Imp. (SS) ↑	Obj. Ratio (SS) ↓	Imp. (SS) ↑
SAC	1.02 ± 0.137	1.09 ± 0.762	1.01 ± 0.0625	45/100	53/100	0.682 ± 0.464	88/100
BO	1.01 ± 0.0783	1.01 ± 0.167	1.00 ± 0.0631	51/100	47/100	1.26 ± 1.44	69/100
PID	0.998 ± 0.0728	1.01 ± 0.221	**1.00 ± 0.0619**	49/100	52/100	**0.427 ± 0.0453***	**100/100**
CNO (BiasOnly)	$> 10^{20}$	$> 10^{20}$	$> 10^{20}$	61/99	72/99	0.443 ± 0.0546	**100/100**
CNO (Hard)	0.943 ± 0.26	0.925 ± 0.524	1.00 ± 0.0696	60/100	66/100	141 ± 72.8	0/100
CNO (Unadjusted)	837 ± 8320	57.9 ± 570	1240 ± 12300	76/100	77/100	28.7 ± 48.6	72/100
CNO (Proposed)	**0.762 ± 0.225***	**0.516 ± 0.4**	1.06 ± 0.347	**85/100**	**84/100**	0.433 ± 0.0459**	**100/100**

Table 1: Comparison of performance across different environments. Arithmetic mean $\pm$ standard deviation of the Ratio of Objective Functions (after/before), and the Improved Sample Proportion over 100 trials. 'SS' means steady-state metrics. The arrows $\downarrow$ and $\uparrow$ indicate lower and higher is better, respectively. For pairwise statistical tests (SAC, BO, PID vs. CNO (Proposed), Wilcoxon signed-rank test with Holm-Bonferroni correction, $\alpha = 0.05$): * indicates that the method significantly outperforms all other alternatives in that column; ** indicates that CNO significantly outperforms SAC and BO but is statistically outperformed by PID (detailed test statistics are provided in Appendix O).

Mean Squared Error (MSE) from Y^* over the experiment's duration, and the squared bias $G_1(c)$ and variance $G_2(c)$. **Ratio of Objective Functions**: The ratio of each objective function $G(c), G_1(c), G_2(c)$ before and after applying the algorithm (after/before). Thus a value less than 1 indicates improvement. **Improved Sample Proportion**: The proportion of experimental trials in which the ratio of $G(c)$ is less than 1, excluding any trials that exhibited divergence. **Steady-State (SS) Metrics**: The Objective Functions, Ratios, Improved Sample Proportion in the stable state after the transient response has settled. These are computed by taking the last 10% of overall time length ($M \times T$) from the time series of the target variable Y.

Results. Linear SCM: As shown in Figure 1c, only our CNO considering confounding successfully calibrated the target variable Y. CNO(Unadjusted) drove the optimization in the wrong direction due to the confounding effect. **Complex Synthetic Environment:** As Table 1 shows, CNO is superior to other methods in terms of the Improvement Ratio of Objective Function and Squared Bias, as well as the Improved Sample Proportion, while maintaining comparable variance. Figure 3a shows a representative trial where only CNO successfully optimizes the target variable while other methods struggle in complex SCMs. **CSTR Simulator:** Looking at the results in Figures 3b and 3c and Table 1, our CNO demonstrates superior performance to SAC and BO, and performs as well as PID in steady state and Improved Sample Proportion.

Overall, CNO achieves the best balance between objective improvement and robustness. While the CSTR environment favors PID control, CNO excels in complex environments following general SCMs, demonstrating the robust stability vital for safety-critical real-world applications.

5 Discussion and Limitations

Local Optimization. Since CNO employs a Newton-style update, it is inherently a local optimizer designed for the *calibration* of semi-optimized systems (Assumption 1), rather than global exploration. Consequently, convergence to local optima or transient overshoot may occur if the system is initialized far from the optimal basin of attraction.

Dependence on Known Causal Structure. Accurate gradient estimation relies on the correctness of the causal graph $\mathcal{G}$ for confounding adjustment. While CNO is robust to parametric uncertainty, structural misspecification (e.g., unmeasured confounders) can bias estimates. Addressing unknown structures via causal discovery [Spirtes and Glymour, 1991; Shimizu *et al.*, 2006; Shimizu *et al.*, 2011] remains a future challenge. Furthermore, the assumption that all system variables are observed (causal sufficiency) is implicitly adopted, which is a very strong condition. Therefore, robustness against other factors such as unobserved confounders must also be verified.

Univariate Limitation. The current formulation focuses on a single manipulatable variable. Extending CNO to multivariate interventions requires estimating the full Hessian matrix (including off-diagonal interaction terms), which significantly increases sample complexity and computational cost (corresponding to the quasi-Newton methods).

Approximation and Dynamics. The Gauss-Newton approximation omits the second-order term $(\mu(c) - Y^*)\mu''(c)$, which is theoretically valid only when residuals are small. Furthermore, accurate causal effect estimation assumes the system reaches a quasi-steady state within the intervention interval T; rapid non-stationary drifts or insufficient settling times may violate this assumption.

6 Conclusion

In this paper, we proposed Causal Newton Optimization (CNO), an online algorithm designed for the calibration of semi-optimized systems. Unlike global exploration methods, CNO leverages prior knowledge of existing semi-optimization and iterative local linear causal modeling and Newton-based updates to achieve fast convergence while maintaining safety through additive interventions. Crucially, CNO explicitly handles confounding bias and incorporates a log-linear variance regression to minimize target variance. These result in robust performance. Our evaluation demonstrates that CNO achieves the best balance between objective improvement and robustness. While standard dynamical systems favor traditional PID control, CNO significantly outperforms RL and BO methods, and excels in complex environments governed by general SCMs. This highlights CNO's robust stability for unknown systems, which is vital for safety-critical, real-world applications.

Ethical Statement

The proposed Causal Newton Optimization (CNO) aims to enhance the efficiency and safety of industrial and economic systems by providing a robust online calibration framework. By employing additive interventions, CNO explicitly preserves existing safety mechanisms and feedback loops, significantly reducing the risks associated with raw exploration in safety-critical environments such as chemical plants. Furthermore, the use of local linear causal modeling enhances the interpretability of the optimization process, allowing human operators to monitor and verify algorithmic updates effectively. While the automated nature of CNO could potentially lead to over-reliance on the system, we emphasize that it is designed to complement existing professional control and should be integrated with appropriate monitoring safeguards to mitigate risks from extreme non-stationary drifts.

Acknowledgements

This work was supported by JST CREST Grant Number JPMJCR22D2 and NTT DOCOMO BUSINESS Inc. The authors would like to thank our colleagues at NTT DOCOMO BUSINESS for their insightful discussions and continuous support throughout this project. The authors are also grateful to Shiga University for providing the academic environment and resources that facilitated this research.

References

[Aglietti *et al.*, 2020] Virginia Aglietti, Xiaoyu Lu, Andrei Paleyes, and Javier González. Causal Bayesian Optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 3155–3164. PMLR, 2020.

[Åström and Hägglund, 1995] Karl Johan Åström and Tore Hägglund. PID controllers: theory, design, and tuning. *The international society of measurement and control*, 1995.

[Astrom and Wittenmark, 1994] Karl Johan Astrom and Bjorn Wittenmark. *Adaptive control*. Addison-Wesley Longman Publishing Co., Inc., 1994.

[Bareinboim *et al.*, 2015] Elias Bareinboim, Andrew Forney, and Judea Pearl. Bandits with unobserved confounders: A causal approach. *Advances in Neural Information Processing Systems*, 28, 2015.

[Camacho *et al.*, 2007] Eduardo F Camacho, Carlos Bordons, Eduardo F Camacho, and Carlos Bordons. *Constrained model predictive control*. Springer, 2007.

[Frazier, 2018] Peter I Frazier. A tutorial on Bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

[Fuller *et al.*, 2021] Sawyer Fuller, Ben Greiner, Jason Moore, Richard Murray, René van Paassen, and Rory Yorke. The Python Control Systems Library (python-control). In *60th IEEE Conference on Decision and Control (CDC)*, pages 4875–4881. IEEE, 2021.

[Gultchin *et al.*, 2023] Limor Gultchin, Virginia Aglietti, Alexis Bellot, and Silvia Chiappa. Functional causal Bayesian optimization. In Robin J. Evans and Ilya Shpitser, editors, *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, volume 216 of *Proceedings of Machine Learning Research*, pages 756–765. PMLR, 31 Jul–04 Aug 2023.

[Haarnoja *et al.*, 2018] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. PMLR, 2018.

[Lattimore and Szepesvári, 2020] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[Lattimore *et al.*, 2016] Finnian Lattimore, Tor Lattimore, and Mark D Reid. Causal bandits: Learning good interventions via causal inference. *Advances in neural information processing systems*, 29, 2016.

[Marlin and Marlin, 1995] Thomas E Marlin and TE Marlin. *Process control: designing processes and control systems for dynamic performance*, volume 2. McGraw-Hill New York, 1995.

[Pearl *et al.*, 2016] Judea Pearl, Madelyn Glymour, and Nicholas P Jewell. *Causal inference in statistics: A primer*. John Wiley & Sons, 2016.

[Pearl, 2009] Judea Pearl. *Causality*. Cambridge University Press, 2009.

[Sani *et al.*, 2020] Numair Sani, Jaron Lee, and Ilya Shpitser. Identification and estimation of causal effects defined by shift interventions. In *Conference on Uncertainty in Artificial Intelligence*, pages 949–958. PMLR, 2020.

[Shimizu *et al.*, 2006] Shohei Shimizu, Patrik O Hoyer, Aapo Hyvärinen, Antti Kerminen, and Michael Jordan. A linear non-Gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(10), 2006.

[Shimizu *et al.*, 2011] Shohei Shimizu, Takanori Inazumi, Yasuhiro Sogawa, Aapo Hyvärinen, Yoshinobu Kawahara, Takashi Washio, Patrik O Hoyer, and Kenneth Bollen. DirectLiNGAM: A direct method for learning a linear non-gaussian structural equation model. *The Journal of Machine Learning Research*, 12:1225–1248, 2011.

[Sinha and Mishra, 2018] Abhinav Sinha and Rajiv Kumar Mishra. Control of a nonlinear continuous stirred tank reactor via event triggered sliding modes. *Chemical Engineering Science*, 187:52–59, 2018.

[Spirtes and Glymour, 1991] Peter Spirtes and Clark Glymour. An algorithm for fast recovery of sparse causal graphs. *Social Science Computer Review*, 9(1):62–72, 1991.

[Stock and Watson, 2015] James H Stock and Mark W Watson. Introduction to econometrics (3rd updated edition). *Age (X3)*, 3(0.22), 2015.

[Subramanian and Ravindran, 2025] Chandrasekar Subramanian and Balaraman Ravindran. Causality in bandits: A survey. *ACM Computing Surveys*, 2025.

[Süli and Mayers, 2003] Endre Süli and David F Mayers. *An introduction to numerical analysis*. Cambridge University Press, 2003.

[Uppal *et al.*, 1974] Ashok Uppal, Willis Harmon Ray, and Aubrey B Poore. On the dynamic behavior of continuous stirred tank reactors. *Chemical Engineering Science*, 29(4):967–985, 1974.

[Wright *et al.*, 1999] Stephen Wright, Jorge Nocedal, et al. Numerical optimization. *Springer Science*, 35(67-68):7, 1999.

[Zeng *et al.*, 2024] Yan Zeng, Ruichu Cai, Fuchun Sun, Libo Huang, and Zhifeng Hao. A survey on causal reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2024.