Solving Discrete-Continuous Dynamic Choice Models: A Soft-to-Hard AI Solver with an Application to Sovereign Default

Bo Li[a], Serguei Maliar[b,c], Peiyuan Zhang[a]

[a] *School of Economics, Peking University, Beijing, China*
[b] *Department of Economics, Santa Clara University, USA*
[c] *Hoover Institution, Stanford University, USA*

Abstract

This paper introduces a soft-to-hard (StH) deep reinforcement learning framework to solve a class of discrete-continuous dynamic choice models. To overcome vanishing gradients caused by non-differentiable kinks, we augment an actor–critic architecture with distinct training and evaluation modes. During training, hard discrete switches are replaced with soft choice probabilities, facilitating gradient penetration into the value and policy networks. This smoothing is strictly temporary; the temperature parameter is systematically annealed to zero across iterations. In evaluation, agents return to the exact hard-switch decision. Using one- and multi-period sovereign default problems, we demonstrate that the StH solver delivers highly accurate, scalable solutions to original, non-smoothed environments. The replication package will be released upon acceptance.

Keywords: Sovereign default, Long-term debt, Reinforcement learning, Actor–critic, Computational economics
JEL classification: C61, C63, C68, F34

1. Introduction

The numerical solution of discrete-continuous dynamic choice (DCDC) models remains a major computational frontier in quantitative economics. Prevalent across labor economics, industrial organization, public finance, and macroeconomics, these frameworks model environments where agents make an endogenous qualitative choice alongside a continuous quantitative optimization. The primary computational roadblock in these settings stems from the severe non-convexity of the value function. When an agent evaluates a discrete switch, the resulting value function

becomes the upper envelope of multiple choice-specific value functions, generating non-differentiable kinks. During backward induction or fixed-point iterations, these kinks propagate and accumulate exponentially, traditionally forcing value function iteration into exhaustive, highly inefficient grid-search operations. Moreover, from a deep learning perspective, these non-differentiable thresholds sever the flow of gradients at the discrete max operator or binary indicator function. If a model with a hard discrete choice is optimized end-to-end, any policy branch that falls below the selection threshold in early iterations receives a zero gradient. This creates a severe vanishing gradient problem: the neural networks lose critical learning signals, stalling convergence and rendering standard gradient-based optimization unviable.

To overcome these numerical hurdles and vanishing gradient problems, we introduce a soft-to-hard (StH) deep reinforcement learning solver explicitly designed for general DCDC models. Building on an actor–critic architecture, we augment the solver with distinct training and evaluation modes to maintain a rigorous separation between the "hard" economic targets of the model and the "soft" differentiable operators required for efficient training. Specifically, we propose a novel gradient penetration mechanism: during training, the non-differentiable hard switches are replaced with a log-sum-exp soft continuation envelope and smooth logistic choice probabilities. This temporary smoothing allows gradients to penetrate the value branches and the policy actor, enabling joint optimization within a fully differentiable structure. As the solution is refined across iterations, the training temperature is systematically annealed to zero. In evaluation mode, the AI agents return to the exact hard-switch choice, ensuring that the reported policies and simulated moments reflect the pure, un-smoothed theoretical economy.

Methodologically, our general solver bridges a broad interdisciplinary literature. The log-sum-exp mathematical structure was historically pioneered in microeconometrics and industrial organization (McFadden, 1974; Rust, 1987; Berry et al., 1995) to model Extreme Value Type-I (Gumbel) taste shocks, where the resulting smooth probabilities reflect genuine variations in agent preferences. Concurrently, this exact mathematical structure was independently derived in the artificial intelligence community under Maximum Entropy Reinforcement Learning (e.g., Soft Actor-Critic; see Haarnoja et al., 2018). In the machine learning domain, the temperature parameter τ serves as the mathematical equivalent to an economic taste shock, utilized to regularize neural networks and encourage policy exploration. Our primary methodological contribution is unifying these concepts: rather than accepting the log-sum-exp operator as a permanent structural modification of the economic environment, our algorithm restricts its use strictly to the neural network training phase. Our analysis aligns with a broader surge of "physics-informed" deep learning methods in economics,

which embed Bellman structures directly into neural architectures to solve complex dynamic models (Maliar et al., 2021; Azinovic et al., 2022; Han et al., 2021); see Fernández-Villaverde (2026) for a recent review. The connection between reinforcement learning and large-scale economic dynamics also has earlier roots: Jirnyi and Lepetyuk (2011) applied reinforcement learning ideas to incomplete-market models with aggregate uncertainty, while Li et al. (2026) develops an informed actor–critic method that embeds known economic structure into actor and critic updates. Within the DCDC literature, the sole precedent is Maliar and Maliar (2022), who introduced a deep learning classification framework for DCDC problems that maps discrete decisions into categorical tasks. Our StH framework introduces a distinct approach with two core advantages. First, by embedding an annealing temperature directly into the continuous continuation values rather than relying on cross-entropy classification, it ensures smooth gradient penetration across value branches. Second, it generalizes to a broader class of DCDC problems where the continuous policies and recursive pricing rules are themselves endogenous, state-dependent outcomes of those sharp boundaries.

We apply this general-purpose StH solver to the sovereign default problem, a quintessential and highly demanding laboratory for DCDC frictions. Pioneered with the one-period model by Eaton and Gersovitz (1981) and Arellano (2008), and advanced to long-term debt by Hatchondo and Martinez (2009), Chatterjee and Eyigungor (2012) (hereafter CE2012) and Sánchez et al. (2014), this environment features a hard binary choice to default or repay coupled with continuous debt issuance and recursive long-term pricing. The resulting methodological approaches for these models have been extended well beyond sovereign debt, enabling the efficient modeling of consumer installment loans (Chatterjee and Mitra, 2024) and facilitating the integration of default frameworks into broader macroeconomic models to assess financial fragility (Gennaioli et al., 2014; van der Kwaak and van Wijnbergen, 2014).

To deal with numerical challenges of sovereign default models, the macroeconomics literature has developed advanced grid algorithms and clever structural modifications. One prominent track leverages geometric filtering to isolate choices, such as the Discrete-Continuous Endogenous Grid Method (DCEGM; Iskhakov et al., 2017; Druedahl, 2021), and Mathematical Programming with Equilibrium Constraints (MPEC) frameworks (Su and Judd, 2012). A second track achieves computational tractability by smoothly approximating the environment, embedding features like auxiliary income shocks (Chatterjee and Eyigungor, 2012) or Extreme Value preference shocks (Gordon, 2019; Dvorkin et al., 2021; Mihalache, 2024). Others creatively bypass these discrete-time frictions entirely through random maturity structures (Arellano and Ramanarayanan, 2012) or continuous-time frameworks (Aguiar

et al., 2013; Aguiar and Amador, 2020).

While these modeling choices are highly effective, they alter the underlying structural environment, which can shift counterfactual policy implications. In seeking to solve the original, un-smoothed problem directly, our objective closely aligns with the pioneering work of Mateos-Planas et al. (2025), who develop a customized system of generalized Euler equations for this exact purpose. However, their analytical approach requires extensive model-specific derivations and remains bound to discrete state-space grids that scale poorly to higher dimensions. An advantage of our StH approach is that it leverages neural network function approximation to bypass grid dependencies entirely, providing a highly scalable, continuous-action architecture that solves the original hard-switch economy without requiring dense, problem-specific analytical customizations.

We demonstrate the efficacy of our StH solver through two sequential applications within this sovereign default domain. First, we evaluate the solver on a one-period sovereign debt benchmark to isolate the discrete default choice from the recursive feedback loops of long-term pricing. We show that the gradient penetration mechanism successfully recovers the true default boundary and asset policies, even when initialized under highly distorted configurations. Second, we analyze the full multi-period CE2012 model to demonstrate how the additional non-convexities of debt dilution and recursive pricing can be resolved. We first verify that neural networks can represent the key CE2012 objects with high accuracy, and then apply the final continuous-action StH actor–critic solver to the hard no-shock economy. The intermediate bridge diagnostics are reported in the appendix. The comparison with the CE2012 auxiliary-shock benchmark and a taste-shock/logit benchmark quantifies how smoothed solution objects differ from the hard StH target, rather than treating those smoothed economies as the target itself. The complete Python implementation of the StH solver and the replication package for the sovereign default application will be released upon acceptance.

The remainder of the paper is organized as follows. Section 2 formulates the general discrete-continuous dynamic choice framework and establishes the theoretical mechanics of the Soft-to-Hard regularizer. Section 3 details the canonical long-term sovereign default model. Section 4 introduces the continuous-action AI solver and gradient penetration losses. Sections 5 and 6 illustrate the application of the StH method to the one-period and multi-period debt models, respectively. Section 7 concludes.

2. A soft-to-hard (StH) solver for DCDC environments

To establish a foundation for our computational method, we first formally define a generic discrete-continuous dynamic choice (DCDC) environment. This abstracts away from the specific mechanics of sovereign default to highlight the universal computational bottleneck inherent in these models, before introducing our Soft-to-Hard (StH) deep reinforcement learning framework to solve it.

2.1. A general discrete-continuous dynamic choice framework

We consider an infinite-horizon, discrete-time Markov decision process where an economic agent makes both qualitative (discrete) and quantitative (continuous) decisions at each period. Let time be indexed by $t = 0, 1, 2, \ldots, \infty$.

State and action spaces.. At the beginning of each period, the agent observes a state vector $s \in \mathcal{S}$, which encapsulates all payoff-relevant endogenous variables (e.g., accumulated assets, current debt) and exogenous stochastic processes (e.g., income shocks, productivity). Given the state s, the agent chooses a combined action pair (d, x):
$\star$ Discrete action (d): A mutually exclusive qualitative choice drawn from a finite choice set $\mathcal{D}$. To align with standard default/repay or entry/exit models, we focus on the binary case where $d \in \{0, 1\}$.
$\star$ Continuous action (x): A quantitative choice drawn from a compact, continuous feasible set $\mathcal{X}(s, d) \subseteq \mathbb{R}^n$, which may depend on both the current state and the chosen discrete action.

Preferences and transitions.. The agent receives an immediate payoff $U(s, d, x)$ and discounts future utility by a factor $\beta \in (0, 1)$. The state evolves according to a Markov transition probability distribution $P(s'|s, d, x)$, which governs the transition to next period's state s' conditional on the current state and the full action profile.

The Bellman equation and the non-convexity problem.. The agent's objective is to maximize the expected discounted sum of future utility. The recursive formulation of this problem is characterized by the overall value function $V(s)$, which is the upper envelope of the choice-specific value functions. The overall value function is defined as:

$$V(s) = \max_{d \in \{0,1\}} \left\{ V^d(s) \right\} \tag{1}$$

where the choice-specific value function (or "branch") for a given discrete choice d is the solution to the continuous optimization problem:

$$V^d(s) = \max_{x \in \mathcal{X}(s,d)} \left\{ U(s,d,x) + \beta \mathbb{E}_{s'|s,d,x} \left[V(s') \right] \right\} \tag{2}$$

Equations (1) and (2) clearly isolate the fundamental computational bottleneck in DCDC models. While the choice-specific value functions $V^0(s)$ and $V^1(s)$ may individually be well-behaved and concave, the overall value function $V(s)$ introduces a hard max operator. This operator creates non-differentiable kinks at the points in the state space where the agent is indifferent between the discrete choices (i.e., where $V^1(s) = V^0(s)$). When iterating backward or optimizing via gradient descent, this non-differentiable upper envelope halts gradient propagation. If one branch currently dominates the other in training, the sub-optimal branch receives a gradient of zero—a phenomenon that prevents neural networks from updating the losing branch and exploring the full optimal policy. The soft-to-hard (StH) solver introduced below is specifically engineered to resolve this exact friction by temporarily smoothing the max operator in $V(s)$ during training, while preserving the exact structural conditions of $V^d(s)$.

2.2. Structure of the StH solver

Our solver utilizes four neural networks:

$$x_\omega(s), \quad V^1_\theta(s), \quad V^0_\psi(s), \quad \Gamma_\phi(s,x) \tag{3}$$

In Equation (3), x_ω is the continuous actor conditional on selecting discrete choice $d = 1$, V^1_θ is the critic for the $d = 1$ branch, V^0_ψ is the critic for the $d = 0$ branch, and Γ_ϕ is an equilibrium expectation network (e.g., a pricing schedule, transition rule, or wage function required by the specific economic environment). We separate these four objects because the continuous policy, the discrete value branches, and the equilibrium conditions are distinct but mutually interacting economic objects that should be represented independently during training and re-coupled through the Bellman equations.

To ensure the continuous actor always satisfies basic domain bounds, it is parameterized via a bounded activation function (such as a sigmoid σ):

$$x_\omega(s) = X_{max} \sigma(f_\omega(s)) \tag{4}$$

Equation (4) guarantees $x \in [0, X_{max}]$. A similar bounding parameterization is applied to the equilibrium network Γ_ϕ based on the theoretical bounds of the specific model.

The methodological distinction of our StH algorithm lies in the location of the smoothing and the representation of the actions. While standard DCDC methods often introduce permanent auxiliary shocks or taste shocks to smooth the policy choice into a probability distribution over a discretized grid, our final AI solver outputs a deterministic continuous action directly via x_ω. Smooth choice probabilities are introduced exclusively into the continuation and expectation targets during training.

2.3. Training targets: soft continuation and Bellman regression

If the original hard-max operator is trained end-to-end directly, gradients are cut off: the discrete branch that loses early in training may become permanently inactive, and the continuous actor may exploit local errors in the critics by proposing pathological actions. To address this, we replace the hard continuation value during training by a log-sum-exp soft continuation:

$$\mathcal{C}_\tau(s) = \tau \log \left[\exp\left(\frac{V^1(s)}{\tau}\right) + \exp\left(\frac{V^0(s)}{\tau}\right) \right] \tag{5}$$

where $\tau > 0$ is the temperature parameter controlling the sharpness of the discrete switch. Equation (5) induces the corresponding soft choice probability for branch $d = 1$:

$$p^1_\tau(s) = \frac{1}{1 + \exp\left(-\frac{V^1(s) - V^0(s)}{\tau}\right)} \tag{6}$$

In mathematical form, Equations (5) and (6) are analogous to the log-sum-exp envelope used in taste-shock methods, but here they are utilized strictly as differentiable operators inside the training targets. Based on this, the Bellman regression targets for the critics are formulated as:

$$V^1(s) = U^1(s, x_\omega(s)) + \beta \mathbb{E}_{s'|s,d=1} \left[\mathcal{C}_\tau(s') \right], \tag{7}$$

$$V^0(s) = U^0(s) + \beta \mathbb{E}_{s'|s,d=0} \left[\mathcal{C}_\tau(s') \right]. \tag{8}$$

with corresponding mean-squared-error losses:

$$L_{V^1} = \mathbb{E}\left[(V^1_\theta(s) - V^1(s))^2 \right], \quad L_{V^0} = \mathbb{E}\left[(V^0_\psi(s) - V^0(s))^2 \right] \tag{9}$$

In Equations (7)–(9), the future continuation terms are provided by target networks to stabilize the interaction among the actor, the critics, and the equilibrium network. The target for the equilibrium network Γ_ϕ similarly replaces hard indicator functions with the differentiable soft choice probability $p^1_\tau(s')$, ensuring it remains fully differentiable with respect to future continuous actions.

2.4. Why gradient penetration can train hard discrete choices

This mechanism must be distinguished from simply "passing gradients through." Define the objective induced by the actor at state s as:

$$V_\omega^1(s) = U^1(s, x_\omega(s)) + \beta \mathbb{E}_{s'} \left[\mathcal{C}_\tau(s') \right] \tag{10}$$

If we train the current decision using the original hard choice objective,

$$J_{hard}(s) = \max\{V_\omega^1(s), V_\psi^0(s)\} \tag{11}$$

then the actor branch value in Equation (10) affects training only when branch 1 is selected. Whenever branch $d = 0$ strictly dominates at the current state (i.e., $V_\omega^1(s) < V_\psi^0(s)$), Equation (11) implies:

$$\nabla_\omega J_{hard}(s) = 0 \tag{12}$$

Equation (12) is the primary "hard-max problem" in discrete-continuous choices: even if branch 1 is only slightly sub-optimal, the continuous actor x_ω receives zero gradient signal on how to improve its policy to make branch 1 competitive.

Gradient penetration replaces $J_{hard}(s)$ with the soft choice objective:

$$J_\tau(s) = \tau \log \left[\exp\left(\frac{V_\omega^1(s)}{\tau} \right) + \exp\left(\frac{V_\psi^0(s)}{\tau} \right) \right] \tag{13}$$

The actor gradient under Equation (13) then becomes:

$$\nabla_\omega J_\tau(s) = p_\tau^1(s) \nabla_\omega V_\omega^1(s) \tag{14}$$

while the competing branch receives updates proportional to:

$$\frac{\partial J_\tau(s)}{\partial V_\psi^0(s)} = 1 - p_\tau^1(s) \tag{15}$$

Equations (14) and (15) show that, as long as $\tau > 0$, both the currently dominant branch and the currently weaker branch maintain positive learning weights. The continuous policy can therefore be optimized gradually near the discrete decision boundary until the hard decision naturally flips.

2.5. Actor objective and feasibility control

In this framework, the actor optimizes the structural economic objective directly. To prevent the actor from exploring economically infeasible regions (e.g., violating budget constraints) during early training phases, we map the proposed action $x_\omega(s)$ into a constraint function $g(s, x_\omega) \leq 0$. We then define a feasibility penalty:

$$pen_\omega(s) = \max\{g(s, x_\omega(s)), 0\}^2 \tag{16}$$

The combined actor loss is defined as:

$$L_{actor} = -\mathbb{E}\left[U^1(s, x_\omega^{safe}(s)) + \beta \mathbb{E}_{s'}\left[\mathcal{C}_\tau(s')\right]\right] + \lambda_{feas}\mathbb{E}[pen_\omega(s)] \tag{17}$$

In Equation (17), x^{safe} ensures the utility function does not crash on invalid inputs, while Equation (16) disciplines economically infeasible actions. The actor is therefore a continuous solver, but it is strictly disciplined by economic feasibility constraints and the equilibrium network.

2.6. Annealing, hard evaluation, and economic interpretation

During training, we utilize a temperature annealing schedule:

$$\tau(t) = \tau_{init}\left(\frac{\tau_{final}}{\tau_{init}}\right)^{\min\{1, t/T_{ann}\}} \tag{18}$$

Equation (18) makes the soft operator gradually approach the true hard comparison. During evaluation, we strictly return to the original hard continuation and hard discrete rule:

$$p^1_{hard}(s) = \mathbf{1}\{V^1(s) \geq V^0(s)\}, \quad \mathcal{C}_{hard}(s) = \max\{V^1(s), V^0(s)\} \tag{19}$$

Equation (19) ensures that training-time softness serves purely as a numerical regularization device, while hard evaluation preserves the exact structural interpretation of the economic model. This design relies on two critical mathematical properties outlined below.

2.7. Mathematical properties and proofs

To ensure that training-time softness serves merely as a numerical regularization device without permanently altering the structural economic model, our framework relies on two critical mathematical properties. The formal proofs are provided in Appendix A.

Proposition 1 (Gradient penetration). Given the soft continuation $\mathcal{C}_\tau(s)$, the partial derivatives with respect to the two value branches are:

$$\frac{\partial \mathcal{C}_\tau(s)}{\partial V^1(s)} = p_\tau^1(s), \quad \frac{\partial \mathcal{C}_\tau(s)}{\partial V^0(s)} = 1 - p_\tau^1(s) \tag{20}$$

Equation (20) implies that, as long as $\tau > 0$, both branches receive strictly positive gradient weights, ensuring that gradients penetrate the default boundary.

Proof sketch. The result follows directly from applying the chain rule to the log-sum-exp function. The derivative of the outer logarithm introduces the sum of the exponentials in the denominator, while the inner derivative of the specific branch provides the numerator. Dividing both by the dominant exponential yields the standard logistic sigmoid probabilities. Because the exponential function is strictly positive, the gradients never exactly reach 0 or 1 for any finite value.

Implication. Unlike the hard max operator—which yields a gradient of 1 for the dominant branch and 0 for the losing branch (causing the "dead neuron" problem)—this soft operator guarantees that gradients dynamically penetrate the default boundary, allowing the neural network to continuously update both choice-specific value functions everywhere in the state space.

Proposition 2 (Recovery of the hard model). (Recovery of the Hard Model). For any state s, the soft continuation satisfies the strict bounds:

$$\max\{V^1(s), V^0(s)\} \leq \mathcal{C}_\tau(s) \leq \max\{V^1(s), V^0(s)\} + \tau \log 2 \tag{21}$$

The approximation bound in Equation (21) vanishes as $\tau \downarrow 0$. Furthermore, at any state s where $V^1(s) \neq V^0(s)$, the soft probability $p_\tau^1(s)$ converges exactly to the hard indicator $1\{V^1(s) \geq V^0(s)\}$ as $\tau \downarrow 0$.

Proof sketch. To bound the approximation error, we factor out the maximum of the two branches, $M \equiv \max\{V^1(s), V^0(s)\}$, from the log-sum-exp argument. This isolates the difference between the smaller branch and the maximum, strictly bounding the residual term within the interval $[0, \tau \log 2]$. By the Squeeze Theorem, as the temperature $\tau \downarrow 0$, the error term vanishes, forcing the soft continuation to converge uniformly to the true hard maximum. Furthermore, evaluating the limit of $p_\tau^1(s)$ as $\tau \downarrow 0$ confirms that the probability collapses to exactly 1 or 0, except at the exact indifference boundary.

Implication. Under standard economic regularity conditions, the indifference boundary constitutes a set of measure zero. Therefore, as the temperature anneals to zero, our soft operators converge to the exact structural hard model almost ev-

erywhere, allowing us to evaluate the final policy without any residual smoothing artifacts.

In sum, the first property means that soft continuation assigns nonzero gradients to both branches, and the second property means that as the temperature approaches zero, the soft operator recovers the original hard comparison. This is exactly what we mean by *gradient penetration*: the soft operator lets training pass through the hard default boundary, while evaluation returns to the original economic objects.

Our framework aligns with the expanding literature on deep learning with structural economic restrictions. This physics-informed paradigm rejects black-box statistical estimation in favor of embedding restrictions such as Bellman operators, Euler equations, and market-clearing conditions directly into the objective functions of deep neural architectures (Maliar et al., 2021; Maliar and Maliar, 2022; Azinovic et al., 2022; Han et al., 2021); see Fernández-Villaverde (2026) for a comprehensive review of this paradigm shift. Deep learning has also successfully scaled continuous-time frameworks (Duarte, 2018; Fernández-Villaverde et al., 2020; Gopalakrishna, 2021; Sauzet, 2021; Achdou et al., 2022; Han et al., 2018; Huang, 2022; Payne et al., 2025a; Gopalakrishna et al., 2024) and search-and-matching models (Payne et al., 2025b). Reinforcement learning approaches have also been widely adopted across economics (Watkins and Dayan, 1992; Sutton and Barto, 2018; Williams, 1992; Konda and Tsitsiklis, 2000; Lillicrap et al., 2015; Feng et al., 2024); see Charpentier et al. (2023); Chen et al. (2021), and Mosavi et al. (2020) for comprehensive surveys. In particular, there is a highly innovative yet underappreciated (and unpublished) working paper by Jirnyi and Lepetyuk (2011) that remarkably preceded the current machine learning boom; it introduced a prescient framework to solve the canonical Krusell–Smith economy by synthesizing reinforcement learning, nonparametric statistics, and nearest-neighbor clustering. Recent work by Li et al. (2026) emphasizes that the success of RL methods in economics hinges on directly embedding structural constraints into the training process. Our StH architecture strictly adheres to this principle by restricting smoothing exclusively to the training operator.

Finally, there is also a recent deep learning framework by Maliar and Maliar (2022) that tackles DCDC problems by treating discrete economic decisions as categorical tasks optimized end-to-end via deep neural networks. The soft-to-hard (StH) framework represents a fundamentally different approach that offers two important advantages. First, while the pure classification framework of Maliar and Maliar (2022) focuses on isolating choice boundaries via cross-entropy operators, our architecture embeds an annealing temperature parameter directly into the underlying continuous continuation values, enabling smooth gradient penetration across competing value branches. Second, the StH approach generalizes to a broader class of

DCDC problems where the continuous policy functions and recursive asset-pricing rules are themselves endogenous, state-dependent functions of the non-differentiable discrete choice boundaries.

3. Application to Sovereign Default: Chatterjee and Eyigungor's (2012) framework

With the general discrete-continuous dynamic choice (DCDC) framework established, we now apply our methodology to the sovereign default model of Chatterjee and Eyigungor (2012)—hereafter CE2012. This framework is central to our analysis because it represents the canonical long-term debt model in the literature. To highlight the exact computational friction our algorithm resolves, we present two distinct versions of this environment: the pure "hard choice" structural model, and the traditional "smoothed" versions (utilizing auxiliary or taste shocks) that the literature has historically relied upon for computational tractability.

3.1. Mapping the core environment

Regardless of the smoothing assumptions, the baseline state and action spaces map to our DCDC framework as follows:

State Space (s):. The sovereign enters each period with $s = (z, B)$, where z is an exogenous endowment realization and $B \geq 0$ is the inherited stock of long-term debt.

Discrete Action (d):. The sovereign chooses $d \in \{0, 1\}$, where $d = 1$ indicates repayment and $d = 0$ indicates default.

Continuous Action (x):. Conditional on repayment ($d = 1$), the sovereign chooses next-period debt $x = B'$ from a bounded interval $[0, B_{max}]$.

Long-term debt matures probabilistically. Each unit of debt matures next period with probability $\lambda \in (0, 1)$, while the non-maturing fraction $1 - \lambda$ pays a coupon $\text{coup} > 0$. We define the per-unit servicing cost as:

$$\kappa \equiv \lambda + (1 - \lambda)\text{coup} \tag{22}$$

If the sovereign repays, the period budget constraint is:

$$c^R(z, B, B') = z - \kappa B + q(z, B')[B' - (1 - \lambda)B] \tag{23}$$

In Equation (23), $q(z, B')$ is the bond price schedule (representing the equilibrium expectation network Γ from our general framework). The term $-\kappa B$ uses the service cost in Equation (22), and $q(z, B')[B' - (1-\lambda)B]$ is the revenue from net new issuance.

3.2. Version 1: The hard choice baseline (no shocks)

This is the strict economic target our StH solver evaluates against. In this pure structural version, there are no unobserved shocks smoothing the agent's decision. If the sovereign repays, it remains in credit markets and solves:

$$V^R(z,B) = \max_{B' \in [0, B_{max}]} \left\{ u(c^R(z,B,B')) + \beta \sum_{z'} P_z(z,z') V(z',B') \right\} \tag{24}$$

If the sovereign defaults, it is excluded from credit markets and suffers an output loss. Let $h(z) \equiv z - \phi(z)$ denote output under default, where $\phi(z) > 0$ is the penalty. The sovereign re-enters credit markets next period with exogenous probability $\chi \in (0,1)$, starting with zero debt. The default value is:

$$V^D(z) = u(h(z)) + \beta \sum_{z'} P_z(z,z') \left[\chi V(z',0) + (1-\chi) V^D(z') \right]. \tag{25}$$

The overall value function is the non-differentiable upper envelope:

$$V(z,B) = \max\{V^R(z,B), V^D(z)\}. \tag{26}$$

The optimal default decision is a strict, hard indicator:

$$d(z,B) = \mathbf{1}\{V^D(z) > V^R(z,B)\} \tag{27}$$

Because international lenders price debt based on the exact probability of default, the recursive pricing equation incorporates the hard jump in Equation (27):

$$q(z,B') = \frac{1}{1+r_f} \sum_{z'} P_z(z,z') \big(1 - d(z',B')\big) \left[\lambda + (1-\lambda) \left(\text{coup} + q(z', A(z',B')) \right) \right] \tag{28}$$

where $A(z',B')$ denotes the optimal continuous debt policy conditional on repayment. Equations (24) –(28) define the hard-choice CE2012 target that our StH solver evaluates: repayment in Equation (24), default in Equation (25), the upper envelope in Equation (26), and pricing in Equation (28). Solving this exact framework is notoriously difficult because the hard indicator $d(z,B)$ halts gradient propagation and shatters the continuity of the bond price schedule.

3.3. Version 2: Smoothed frameworks

To handle the severe non-convexities generated by the hard choice baseline, the macroeconomic literature has historically embedded permanent smoothing devices directly into the economic model. This fundamentally alters the agent's environment to guarantee computational convergence.

3.3.1. 2a. Auxiliary income shocks (the original CE2012 approach)

To stabilize the fixed-point iteration for the bond price without taste shocks, CE2012 introduces a small, continuous, and i.i.d. transitory income shock m. The state expands to (z, B, m). While the sovereign's default decision at any exact state (z, B, m) remains a hard boundary, integrating over the unobserved shock m generates a smooth expected default probability:

$$\mathbb{E}_m[d(z, B, m)] = \Pr(V^D(z) > V^R(z, B) + m) \tag{29}$$

Equation (29) mechanically transforms the discrete jump in policy into a continuous transition region.

3.3.2. 2b. Taste-shock smoothing (extreme value frameworks)

An alternative, increasingly common route in the literature is to add i.i.d. Extreme Value Type-I "taste shocks" (ϵ^R, ϵ^D) directly to the discrete alternatives, see Gordon (2019), Dvorkin et al. (2021), and Mihalache (2024). The sovereign's true choice-specific values become $V^R(z, B) + \sigma\epsilon^R$ and $V^D(z) + \sigma\epsilon^D$, where $\sigma > 0$ is a scale parameter. Under this assumption, the overall value function evaluates to a closed-form log-sum-exp envelope (the "Emax" operator):

$$V_{taste}(z, B) = \sigma \log \left[\exp\left(\frac{V^R(z, B)}{\sigma}\right) + \exp\left(\frac{V^D(z)}{\sigma}\right) \right] \tag{30}$$

Consequently, the strict default indicator is replaced by a smooth logistic probability everywhere in the model:

$$P(Default|z, B) = \frac{\exp(V^D(z)/\sigma)}{\exp(V^R(z, B)/\sigma) + \exp(V^D(z)/\sigma)} \tag{31}$$

The probability in Equation (31), which is induced by the value in Equation (30), is then directly inserted into the lenders' pricing equation.

3.4. The methodological distinction between smoothing and StH solver

Both auxiliary shocks and taste shocks resolve the hard-max problem by permanently modifying the structural environment. The sovereign is no longer solving the Version 1 baseline; they are solving an alternative Version 2 economy where unobserved forces constantly blur their incentives. By contrast, our StH algorithm isolates the log-sum-exp smoothing strictly within the neural network training operators (as gradient penetration) to stabilize learning. Once training is complete, the final evaluations, moments, and price schedules are generated using the exact hard, no-shock rules of Version 1.

4. Implementation of the StH Solver for the CE2012 Model

Having established the general StH framework and its mathematical properties, we now detail its specific implementation for the CE2012 sovereign default environment. The solver maps the general DCDC operators to the specific economic objects of the sovereign debt model, using four distinct neural networks.

4.1. Network architecture and bounding

All four networks use feedforward multilayer perceptrons with two hidden layers of width 256 and ReLU activations. To ensure the solver respects the economic bounds of the CE2012 model from the first iteration, we apply terminal sigmoid activations $\sigma(\cdot)$ scaled by strict economic limits:

- Continuous Actor (B'_ω): Maps state $(z, B) \mapsto B' \in [0, B_{max}]$. Parameterized as $B'_\omega(z, B) = B_{max}\sigma(f_\omega(z, B))$.
- Repayment Critic (V^R_θ): Maps state $(z, B) \mapsto V^R \in \mathbb{R}$. Unbounded.
- Default Critic (V^D_ψ): Maps state $z \mapsto V^D \in \mathbb{R}$. Unbounded.
- Price Network (q_ϕ): Maps future state (z, B') to the admissible price interval $[0, q^{rf}]$. Parameterized as $q_\phi(z, B') = q^{rf}\sigma(f_\phi(z, B'))$.

The risk-neutral upper bound on the bond price is strictly enforced using $q^{rf} = \frac{\lambda+(1-\lambda)\text{coup}}{r_f+\lambda}$.

4.2. Training targets and feasibility control

Following the StH methodology, we replace the hard continuation value and hard default indicator with the soft operators $\mathcal{C}_{\tau_D}(z, B)$ and $p^R_{\tau_D}(z, B)$, controlled by temperature τ_D. The Bellman regression targets for the critics are adapted to the CE2012 transitions:

$$V^R(z, B) = u(c^{safe}_\omega(z, B)) + \beta \sum_{z'} P_z(z, z')\mathcal{C}_{\tau_D}(z', B'_\omega(z, B)), \tag{32}$$

$$V^D(z) = u(h(z)) + \beta \sum_{z'} P_z(z, z') \left[\chi\mathcal{C}_{\tau_D}(z', 0) + (1 - \chi)V^D_\psi(z')\right]. \tag{33}$$

Similarly, the pricing equation target utilizes the soft repayment probability to ensure the price network remains differentiable with respect to the actor's choices:

$$q(z, B') = \frac{1}{1 + r_f} \sum_{z'} P_z(z, z') p^R_{\tau_D}(z', B') \left[\lambda + (1 - \lambda) \left(\text{coup} + q_\phi(z', B''_\omega(z', B'))\right)\right] \tag{34}$$

where $B''_\omega(z', B')$ denotes the borrowing choice of the target actor in the future state. Equations (32), (33), and (34) are the CE2012 counterparts of the generic soft Bellman targets in Equations (7) and (8). The critics and the price network are trained using standard mean-squared-error losses against these targets.

4.3. Actor loss and feasibility

The actor optimizes the structural economic objective directly. Given the actor's current proposal $B'_\omega(z, B)$, repayment consumption is:

$$c_\omega(z, B) = z - \kappa B + q_\phi(z, B'_\omega(z, B)) \left[B'_\omega(z, B) - (1 - \lambda)B\right] \tag{35}$$

To prevent the actor from exploring economically invalid regions that violate the budget constraint (yielding negative consumption), we define a safe lower bound $c^{safe}_\omega(z, B) = \max\{c_\omega(z, B), c\}$ and a feasibility penalty $pen_\omega(z, B) = \max\{c - c_\omega(z, B), 0\}^2$. The unified actor loss is therefore:

$$L_{actor} = -\mathbb{E}\left[u(c^{safe}_\omega(z, B)) + \beta \mathbb{E}_{z'|z} \mathcal{C}_{\tau_D}(z', B'_\omega(z, B))\right] + \lambda_{feas} \mathbb{E}[pen_\omega(z, B)] \tag{36}$$

Equation (36) applies the generic actor objective in Equation (17) to the CE2012 budget constraint in Equation (35).

4.4. Algorithm

Algorithm 1 summarizes the complete solution procedure, which integrates the deep reinforcement learning updates with the StH temperature annealing.

4.5. Hyperparameters and state sampling

Here, we describe the implementation details.

Training dynamics.. We use the Adam optimizer with a batch size of 256. The learning rates are 10^{-3} for the critics and price network, and 10^{-4} for the actor. Gradients are clipped to a maximum norm of 10.0. Target networks are updated via Polyak averaging with $\tau = 0.001$. The feasibility penalty coefficient is set to $\lambda_{feas} = 10^4$ with a consumption floor of $c = 10^{-6}$.

Algorithm 1 Soft-to-hard actor–critic solver for sovereign default

Initialize B'_ω, V^R_θ, V^D_ψ, q_ϕ, and their Polyak-averaged target networks.
for $t = 1, \ldots, T$ **do**
    Update temperature $\tau_D(t)$ according to the annealing schedule.
    Sample (z, B') points and update q_ϕ using the soft price target.
    Sample (z, B) points and update V^R_θ using the soft Bellman target.
    Sample z points and update V^D_ψ using the soft Bellman target.
    Sample (z, B) points and update B'_ω using L_{actor}.
    Apply gradient clipping.
    Update target networks using Polyak averaging.
end for
Evaluate the final model in exact hard mode using $p^R_{hard}(z, B)$ and $\mathcal{C}_{hard}(z, B)$.

Temperature Annealing.. The temperature $\tau_D(t)$ decays exponentially over $T_{ann} = 400,000$ iterations according to Equation (37):

$$\tau_D(t) = \tau_{D,init} \left(\frac{\tau_{D,final}}{\tau_{D,init}} \right)^{\min\{1, t/T_{ann}\}} \tag{37}$$

with $\tau_{D,init} = 0.05$ and $\tau_{D,final} = 0.002$. The full training run spans $T = 800,000$ iterations, followed by an optional evaluation block of 100,000 iterations locked in strict hard mode.

State Space and Expectations.. The exogenous endowment z follows an AR(1) process in logs: $\log z_t = (1 - \rho_z)\mu_z + \rho_z \log z_{t-1} + \varepsilon_t$, with $\varepsilon_t \sim \mathcal{N}(0, \sigma_z^2)$. The parameters are $\rho_z = 0.9485$, $\sigma_z = 0.02709$, and $\mu_z = 0$. The process is discretized into $N_z = 200$ points spanning ± 3 standard deviations. All expectations over future z' are computed exactly via matrix-vector multiplication with the transition matrix P_z. Debt levels B are sampled uniformly from $[0, B_{max}]$ with $B_{max} = 1.05$ during training, and evaluated on a finer grid of 350 points during final hard evaluation.

5. Solving the one-period sovereign debt model

As a first step, we test our StH solver on the one-period sovereign debt model. This model features the hard default boundary but does not have recursive long-term pricing and debt dilution that are present in multi-period debt model. This allows us to separate the question "Can the discrete default choice itself be trained?" from the question "Can the solver deal with long-term pricing feedback?" We will show

that the StH solver can systematically recover a default boundary, price function, and asset policy.

5.1. The one-period model

In our notation, the one-period model can be understood as the special case $\lambda = 1$. In this case all debt matures next period, and both the coupon and the resale term disappear. The budget constraint becomes

$$c^{R,1}(z, B, B') = z - B + q^1(z, B')B'. \tag{38}$$

Let $V^{R,1}(z, B)$ denote the repayment value in the one-period model, $V^{D,1}(z)$ the default value, and $V^1(z, B)$ the total value. The recursive problem corresponding to Equation (38) is then

$$V^{R,1}(z, B) = \max_{B' \in [0, B_{\max}]} \left\{ u\big(c^{R,1}(z, B, B')\big) + \beta \mathbb{E}_{z'|z} V^1(z', B') \right\}, \tag{39}$$

$$V^{D,1}(z) = u\big(h(z)\big) + \beta \left[\chi \mathbb{E}_{z'|z} V^1(z', 0) + (1 - \chi) \mathbb{E}_{z'|z} V^{D,1}(z') \right], \tag{40}$$

$$V^1(z, B) = \max\{V^{R,1}(z, B), V^{D,1}(z)\}. \tag{41}$$

Because debt is fully short term, the bond price reduces to a one-period pricing relation based on the next-period repayment probability,

$$q^1(z, B') = \frac{1}{1 + r_f} \sum_{z'} P_z(z, z') p^{R,1}(z', B'). \tag{42}$$

where $p^{R,1}(z', B')$ is the repayment event in next period's state (z', B'). These equations show that the one-period model fully preserves the hard repay/default comparison: Equation (39) defines repayment, Equation (40) defines default, Equation (41) defines the upper envelope, and Equation (42) defines pricing.

5.2. Main one-period results

Our analysis of the one-period environment proved that the gradient penetration mechanism can successfully approximate the hard discrete default boundary. Figure 1 shows that the StH solver delivers meaningful policies for consumption and assets, total value, price schedule, and debt-revenue objects.[1]

[1] Labels in the figures such as "idx0" and "idx2" simply refer to selected indices on the discretized income shock grid used for plotting; they correspond to different shock magnitudes and are not additional state variables.

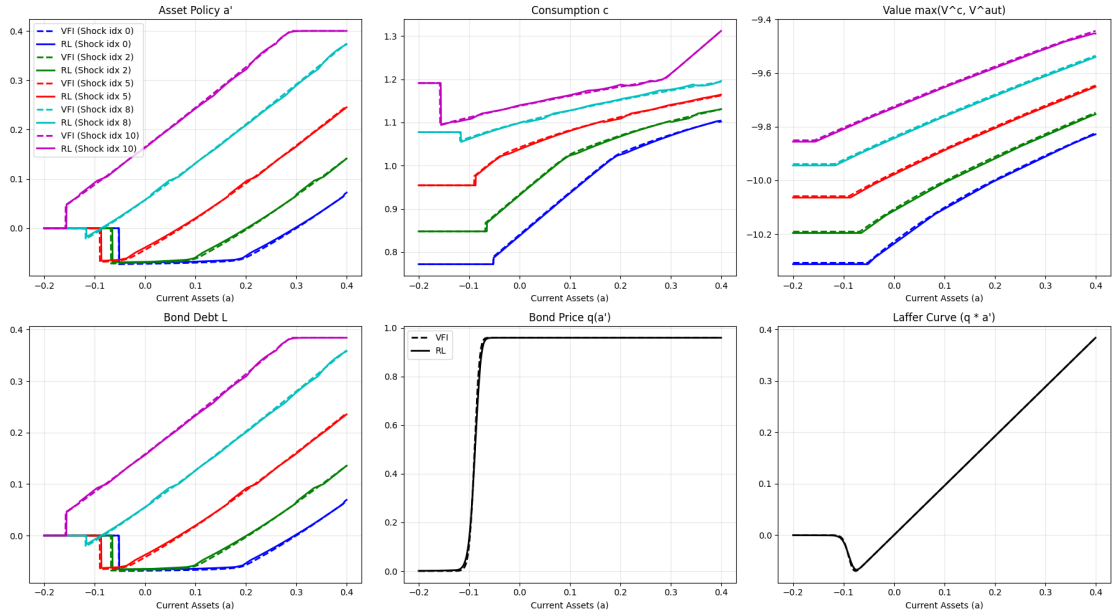

Fig. 1: The StH solver successfully recovers the default boundary, price schedule, and policy functions in the one-period sovereign debt model.

The figure is the main-text evidence for the mechanism: the soft training operator keeps gradients alive on both sides of the default boundary, while the reported solution is evaluated under the original hard comparison. The more demanding poor-initialization exercise is reported in Appendix B. There, the solver first stagnates far from the benchmark but then escapes because the soft continuation and repayment probabilities continue to assign nonzero gradients to both value branches.

6. Solving the long-term default model

The previous section established that our Soft-to-Hard (StH) solver can successfully recover the exact default boundary in a one-period debt model. In this section, we demonstrate that the StH solver can also handle the severe non-convexities introduced by debt dilution and recursive pricing—two fundamental challenges intrinsic to the long-term debt environment.

6.1. From neural representation to the final StH solver

To keep the main text focused on the central economic and computational objects, we compress the long-term application into two main steps. First, we verify that neural networks can accurately represent the core CE2012 solution objects when the

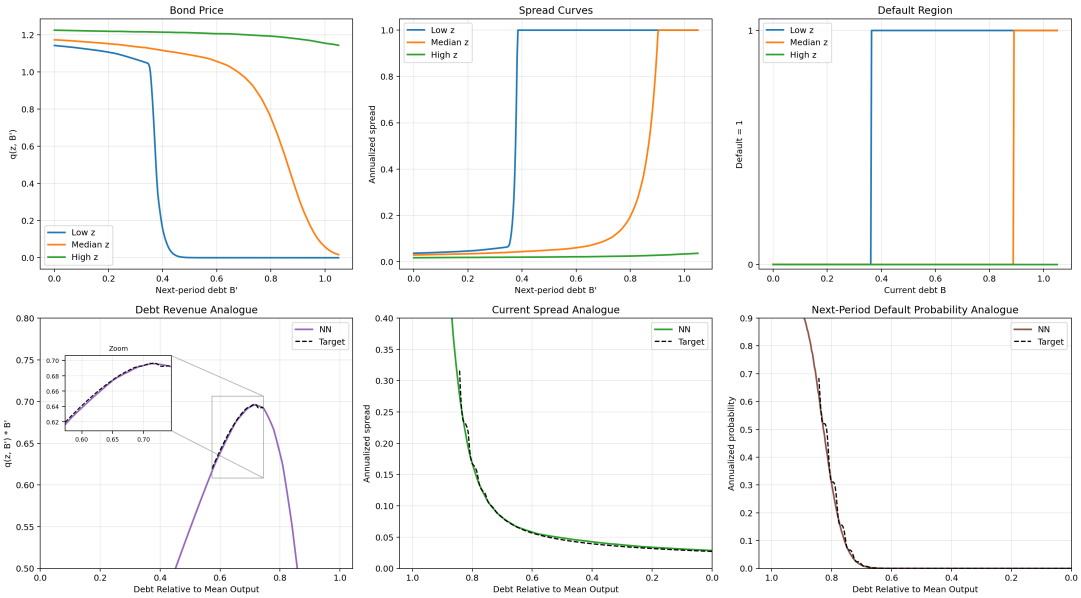

Fig. 2: Neural approximation of the CE2012 auxiliary-shock operator. The panels compare the neural-network analogue with the benchmark objects exported or reconstructed from the original CE2012 MATLAB replication scripts.

original auxiliary-shock environment is preserved. Second, we present the finalized continuous-action StH solver evaluated under the hard no-shock default rule. The intermediate bridge diagnostics that move the solver from grid-based soft Bellman training to actor-based control are reported in Appendix C.

6.1.1. Neuralizing the original CE2012 operator

This representation check preserves the original CE2012 structural environment: the debt action is chosen on a discrete grid, the original default operator is left intact, and the auxiliary income-shock structure is used to guarantee convergence. Our only change to the CE2012 methodology is a replacement of grid values with deep neural network approximation. In Figure 2, we show the most important objects: the price schedule, the spread curve, the default region, and the revenue/spread/default diagnostics corresponding to the benchmark.

In the upper panels, the labels low, median, and high z denote the first, middle, and last grid points of the discretized persistent endowment process. In the lower diagnostic panels, the "target" curves are reconstructed from the original CE2012 data using their exact replication formulas: debt revenue is $q(z, B')\, B$, the value panel varies the candidate debt choice B' at a fixed current debt position, the spread panel

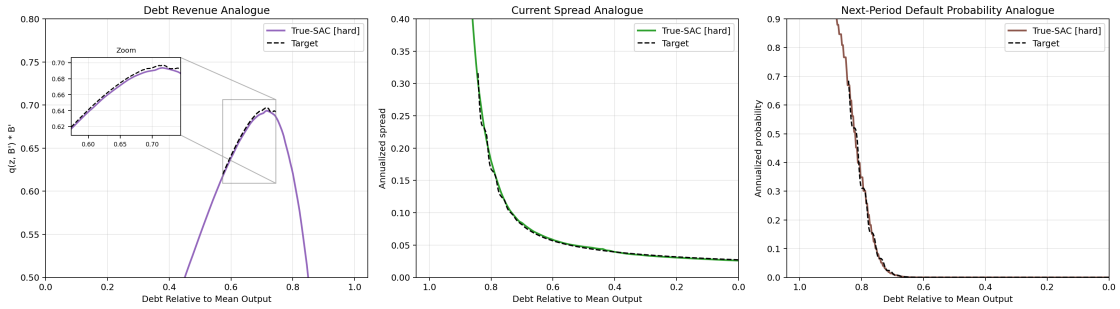

Fig. 3: Final continuous-action StH actor–critic solution under hard evaluation. The panels report the primary economic objects after all permanent smoothing devices are removed from the evaluated economy.

maps the long-term bond pricing formula, and the default panel reports the annualized next-period default probability. The tight alignment between the target curves and our neural network "analogues" confirms that deep neural networks possess the representational capacity required to approximate the complex, non-linear objects of the baseline model. However, despite this representational success, the fundamental limitation of this representation exercise remains: the solver still depends entirely on a computationally expensive global discrete action search.

6.1.2. Final continuous actor–critic StH solver

The final specification implements the methodology of the paper. We combine continuous actions, the baseline long-term economic environment, soft training operators, and strict hard evaluation into a unified Soft-to-Hard (StH) continuous actor–critic solver. Global action-grid searches are entirely eliminated. The auxiliary income shocks and structural smoothing parameters inherited from the baseline literature are discarded. In their place, our soft repayment probabilities penetrate the sharp default boundary during training while enforcing the strict hard-switch decision during evaluation. Figure 3 collects the core equilibrium results from this final continuous actor–critic model.

Our numerical experiments indicate that the StH solver is stable and generates accurate solutions for the long-term hard-switch default model. Because the soft-training outcomes and hard-evaluation outputs are virtually indistinguishable visually, we report the rigorous hard-evaluation results as our primary evidence.

6.2. Comparison of the StH target and smoothed benchmarks

This section evaluates the finalized StH continuous solver as the target solution of the hard, no-shock economy. The CE2012 solution is an essential benchmark

because it is the canonical long-term default computation, but it is not our target object: it uses auxiliary income shocks to smooth the price/default mapping. To quantify how different smoothing devices move the solution away from the hard target, we compare the final StH solution both with the original CE2012 auxiliary-shock benchmark and with a standard taste-shock/logit benchmark computed under the same CE2012 calibration.

6.2.1. Direct comparison and the limits of taste shocks

Figure 4 provides a three-way comparison between the original CE2012 auxiliary-shock benchmark, a taste-shock/logit benchmark under the same CE2012 calibration, and our final StH solution evaluated under strict hard-switch rules. All economic primitives are held fixed at the CE2012 values; the taste-shock comparison only adds the logit smoothing scale to the repayment/default alternatives, as in logit-based default models (Gordon, 2019; Dvorkin et al., 2021; Mihalache, 2024). The distinction is therefore about where smoothing enters the computation: CE2012 and taste shocks solve smoothed benchmark economies, while StH uses smoothing only during training and reports the hard-evaluation solution. The figure shows that the taste-shock/logit benchmark is close to the other objects in median and high endowment states, but shifts the low-endowment default region, where default incentives are most sensitive. The detailed temperature diagnostics in Appendix D show that this low-state displacement is not merely a matter of selecting a different smoothing scale. For the main text, the key point is simple: taste shocks regularize the model by changing the solved economic environment, whereas StH regularizes only the training problem and then evaluates the no-shock hard model.

6.2.2. Dynamic performance and moments

To verify the framework's long-run dynamic accuracy, Figure 5 plots the simulated time series of the sovereign spread.

The StH hard-evaluation path closely mirrors the CE2012 benchmark series over the full simulated horizon. Table 1 presents the corresponding ergodic moments where final AC denotes the final actor–critic StH solution evaluated in hard mode.

Consistent with the graphical evidence, the continuous StH solver generates canonical moments—including the spread, debt ratio, and default frequencies—with high fidelity. The marginal differences that remain between the hard StH target and the CE2012 benchmark are theoretically informative. Because the StH framework constructs the solution to the pure, unsmoothed model, this comparison allows us to isolate and measure how much the solution object moves when the computation relies on permanent structural smoothing.

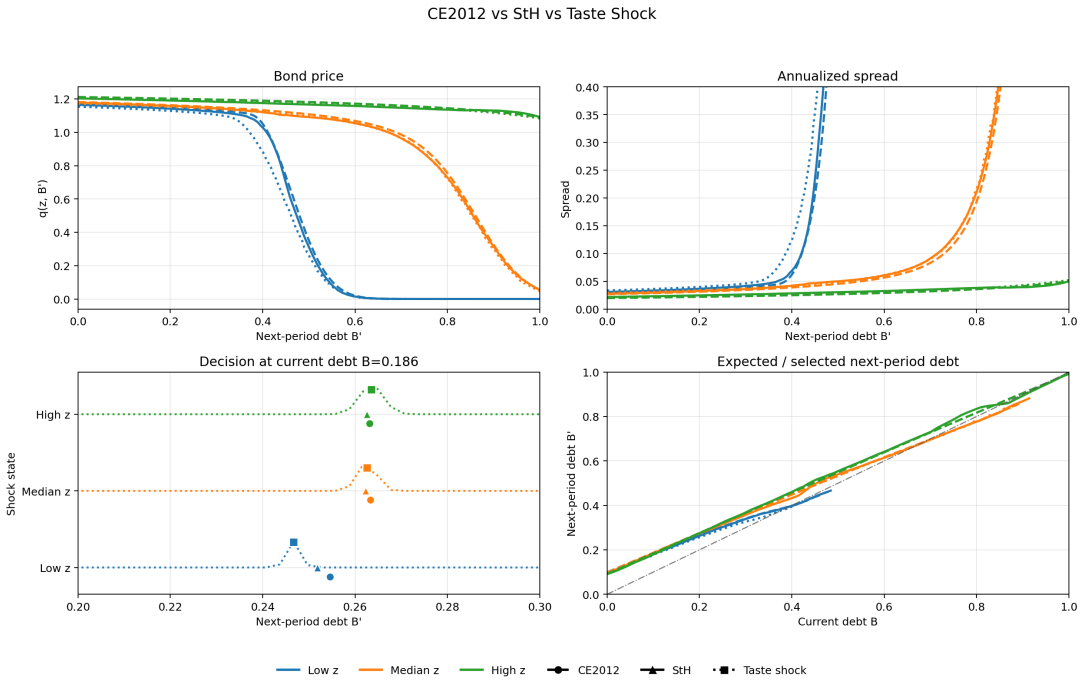

Fig. 4: CE2012 auxiliary-shock benchmark, final StH hard-model solution, and taste-shock/logit benchmark under the same CE2012 calibration. The upper-left panel reports bond prices $q(z, B')$ and the upper-right panel reports annualized spreads. The lower-left panel compares the selected next-period debt at a fixed current debt level, while the lower-right panel reports expected or selected next-period debt over the current-debt grid.

Table 1: Targeted and untargeted economic moments generated by the final actor–critic StH solver.

Core moments			**Non-core moments**		
Indicator	Target	Final AC	Indicator	Target	Final AC
Mean spread	0.0815	0.0875	Average debt service	0.0550	0.0548
Std. spread	0.0443	0.0486	Std. C/Std. Y	1.11	1.1066
Mean debt / output	0.7000	0.6997	Std. NX/Y/Std. Y	0.20	0.2087
Default frequency	0.0680	0.0745	Corr(C, Y)	0.99	0.9847
			Corr(Spread, Y)	-0.65	-0.6611
			Corr$(NX/Y, Y)$	-0.44	-0.4204

Note: Target reports the target moments in the original CE2012 paper. Final AC reports the moments generated by the final actor–critic StH solution evaluated in hard mode.

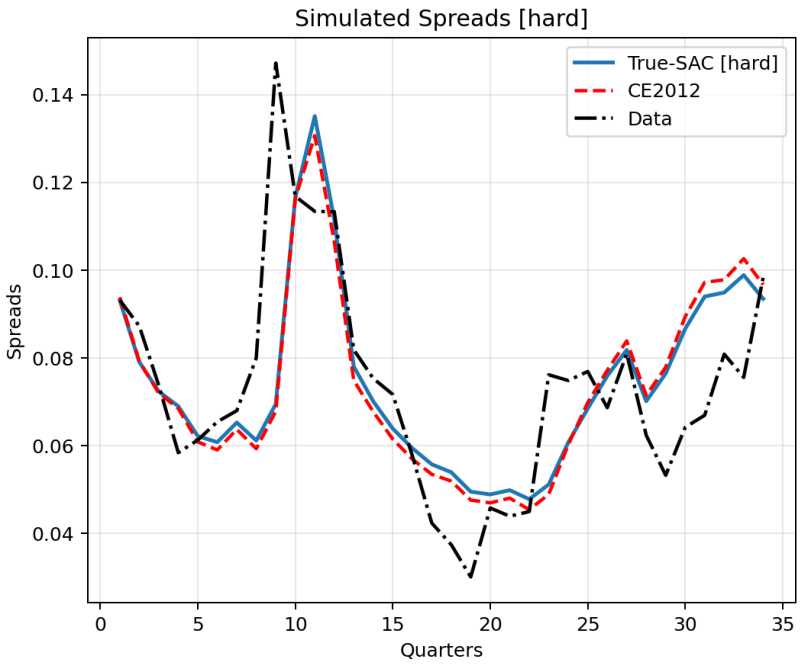

Fig. 5: The StH hard-evaluation path remains close to the CE2012 benchmark over the simulated spread series.

While the historical use of auxiliary threshold shocks in CE2012 provided a remarkably effective practical approximation, the numerical distortions they introduce are not negligible. In applications highly sensitive to exact default timing, spread volatility, or strict welfare calculations, the divergence between structurally smoothed approximations and the true non-smoothed model becomes quantitatively significant. Ultimately, these results underscore the primary value of the StH algorithm: it completely resolves the highly non-linear, long-term sovereign default environment, empowering researchers to evaluate exact theoretical boundaries without settling for structural compromise.

7. Conclusion

This paper has presented an alternative paradigm for solving discrete-continuous dynamic choice models. By decoupling the mathematical smoothing required for numerical optimization from the strict structural boundaries of the target economy, the soft-to-hard (StH) solver demonstrates that deep reinforcement learning can seamlessly navigate sharp, non-differentiable thresholds. The core contribution is not merely that neural networks can approximate traditional choices, but that an AI agent can be trained globally across continuous state and action spaces without losing its learning signal at a discrete boundary. This establishes a mesh-free computational baseline, removing the exponential gridlocks that have constrained the quantitative DCDC analysis.

The quantitative application to sovereign default underscores a broader methodological issue in structural macroeconomics. In discrete-continuous dynamic choice analysis, researchers have long faced an uneasy compromise: either accept the severe dimensionality constraints of a rigid grid, or introduce permanent smoothing devices—such as taste or auxiliary income shocks—to secure numerical tractability. Our comparison with the smoothed benchmark confirms that while these historical workarounds are remarkably effective at approximating aggregate simulated moments, they fundamentally alter the structural primitives of the economic environment. By restricting smooth operators strictly to a temporary training scaffolding, the StH framework solves the hard no-shock economy as the target object and ensures that policy simulations, counterfactual boundaries, and welfare calculations are driven by the economic mechanisms intended by the theorist, rather than by regularizing assumptions required by the computational algorithm.

Looking forward, the scalability of the StH continuous-action architecture opens several distinct computational horizons. First, because the solver completely bypasses grid dependencies, it is uniquely suited for high-dimensional life-cycle settings

and heterogeneous-agent macro models (such as HANK) where lumpy, discrete portfolio adjustments between liquid and illiquid assets generate severe non-convexities. Second, the ability of the gradient penetration mechanism to escape numerical dead zones suggests immense utility in multi-agent strategic environments—such as dynamic games of firm entry and exit or sovereign default with risk-averse lenders—where price schedule discontinuities frequently cause standard fixed-point algorithms to diverge. Ultimately, by shifting the focus from grid management to deep policy training, the StH framework provides a robust, highly adaptable foundation for exploring richer, un-compromised structural environments across the discipline.

Appendix A. Proofs of Propositions 1 and 2

Appendix A.1. Proof of proposition 1 (gradient penetration)

We begin with the definition of the soft continuation value:

$$\mathcal{C}_\tau(s) = \tau \log \left[\exp\left(\frac{V^1(s)}{\tau}\right) + \exp\left(\frac{V^0(s)}{\tau}\right) \right]$$

To find the gradient with respect to $V^1(s)$, we apply the chain rule. First, taking the derivative of the outer logarithmic function $\log(u)$, where $u = \exp(V^1(s)/\tau) + \exp(V^0(s)/\tau)$:

$$\frac{\partial \mathcal{C}_\tau(s)}{\partial V^1(s)} = \tau \cdot \left(\frac{1}{\exp\left(\frac{V^1(s)}{\tau}\right) + \exp\left(\frac{V^0(s)}{\tau}\right)} \right) \cdot \frac{\partial}{\partial V^1(s)} \left[\exp\left(\frac{V^1(s)}{\tau}\right) + \exp\left(\frac{V^0(s)}{\tau}\right) \right]$$

Next, we evaluate the inner derivative. Because $V^0(s)$ is treated as a constant with respect to $V^1(s)$, its derivative is zero:

$$\frac{\partial \mathcal{C}_\tau(s)}{\partial V^1(s)} = \tau \cdot \left(\frac{1}{\exp\left(\frac{V^1(s)}{\tau}\right) + \exp\left(\frac{V^0(s)}{\tau}\right)} \right) \cdot \left(\frac{1}{\tau} \exp\left(\frac{V^1(s)}{\tau}\right) \right)$$

The temperature parameter τ cancels out:

$$\frac{\partial \mathcal{C}_\tau(s)}{\partial V^1(s)} = \frac{\exp\left(\frac{V^1(s)}{\tau}\right)}{\exp\left(\frac{V^1(s)}{\tau}\right) + \exp\left(\frac{V^0(s)}{\tau}\right)}$$

To map this to a standard sigmoid function, we divide the numerator and the denominator by $\exp(V^1(s)/\tau)$:

$$\frac{\partial \mathcal{C}_\tau(s)}{\partial V^1(s)} = \frac{1}{1 + \frac{\exp\left(\frac{V^0(s)}{\tau}\right)}{\exp\left(\frac{V^1(s)}{\tau}\right)}} = \frac{1}{1 + \exp\left(-\frac{V^1(s) - V^0(s)}{\tau}\right)} = p_\tau^1(s)$$

By symmetric application of the chain rule with respect to $V^0(s)$, we obtain:

$$\frac{\partial \mathcal{C}_\tau(s)}{\partial V^0(s)} = \frac{\exp\left(\frac{V^0(s)}{\tau}\right)}{\exp\left(\frac{V^1(s)}{\tau}\right) + \exp\left(\frac{V^0(s)}{\tau}\right)} = 1 - p_\tau^1(s)$$

Implication: Because the exponential function is strictly positive, for any finite values of $V^1(s)$, $V^0(s)$, and strictly positive temperature $\tau > 0$, the probabilities satisfy $0 < p_\tau^1(s) < 1$. Unlike the hard max operator—which yields a gradient of 1 for the dominant branch and 0 for the losing branch—this soft operator guarantees that both value branches receive a non-zero learning signal everywhere in the state space.■

Appendix A.2. Proof of proposition 2 (recovery of the hard model)

First, we establish the bounds on the soft continuation value. Let us define the maximum of the two branches as $M \equiv \max\{V^1(s), V^0(s)\}$. We can rewrite the argument of the log-sum-exp function by factoring out the dominant exponential $\exp(M/\tau)$:

$$\mathcal{C}_\tau(s) = \tau \log\left[\exp\left(\frac{M}{\tau}\right)\left(\exp\left(\frac{V^1(s) - M}{\tau}\right) + \exp\left(\frac{V^0(s) - M}{\tau}\right)\right)\right]$$

Using the property of logarithms $\log(a \cdot b) = \log(a) + \log(b)$:

$$\mathcal{C}_\tau(s) = \tau \log\left(\exp\left(\frac{M}{\tau}\right)\right) + \tau \log\left[\exp\left(\frac{V^1(s) - M}{\tau}\right) + \exp\left(\frac{V^0(s) - M}{\tau}\right)\right]$$

$$\mathcal{C}_\tau(s) = M + \tau \log\left[\exp\left(\frac{V^1(s) - M}{\tau}\right) + \exp\left(\frac{V^0(s) - M}{\tau}\right)\right]$$

By definition, one of the branches equals M, making its corresponding exponent zero (so $\exp(0) = 1$). The other branch must be less than or equal to M. Let $\Delta \leq 0$ denote the difference between the smaller branch and M. The equation simplifies to:

$$\mathcal{C}_\tau(s) = M + \tau \log\left(1 + \exp\left(\frac{\Delta}{\tau}\right)\right)$$

Because $\Delta \leq 0$ and $\tau > 0$, the exponential term satisfies $0 < \exp(\Delta/\tau) \leq 1$. Adding 1 to all sides gives:

$$1 < 1 + \exp\left(\frac{\Delta}{\tau}\right) \leq 2$$

Taking the logarithm (which is strictly increasing) and multiplying by τ preserves the inequalities:

$$0 < \tau \log\left(1 + \exp\left(\frac{\Delta}{\tau}\right)\right) \leq \tau \log 2$$

Substituting this back into our expression for $\mathcal{C}_\tau(s)$ yields the bounded envelope:

$$0 \leq \mathcal{C}_\tau(s) - \max\{V^1(s), V^0(s)\} \leq \tau \log 2$$

By the Squeeze Theorem, as the temperature $\tau \downarrow 0$, the term $\tau \log 2 \to 0$, forcing the soft continuation $\mathcal{C}_\tau(s)$ to converge uniformly to the true hard maximum $\max\{V^1(s), V^0(s)\}$.

Second, we analyze the soft choice probability $p^1_\tau(s)$ as $\tau \downarrow 0$. Let $D \equiv V^1(s) - V^0(s)$ denote the advantage of branch 1. The probability is given by:

$$p^1_\tau(s) = \frac{1}{1 + \exp\left(-\frac{D}{\tau}\right)}$$

We evaluate the limit in three distinct cases:

- Strict Preference for Branch 1 ($D > 0$): As $\tau \downarrow 0$, the exponent $-D/\tau \to -\infty$. Consequently, $\exp(-\infty) \to 0$, and $p^1_\tau(s) \to \frac{1}{1+0} = 1$.
- Strict Preference for Branch 0 ($D < 0$): As $\tau \downarrow 0$, the exponent $-D/\tau \to +\infty$. Consequently, $\exp(+\infty) \to +\infty$, and $p^1_\tau(s) \to \frac{1}{1+\infty} = 0$.
- Indifference ($D = 0$): For any $\tau > 0$, $-D/\tau = 0$, meaning $\exp(0) = 1$. The probability remains exactly $p^1_\tau(s) = \frac{1}{1+1} = 0.5$.

Thus, pointwise convergence to the hard indicator $\mathbf{1}\{V^1(s) \geq V^0(s)\}$ holds exactly at every state except the exact indifference boundary. Under standard economic regularity conditions, this boundary constitutes a set of Lebesgue measure zero. Therefore, the soft probability converges to the hard indicator almost everywhere in the state space. ■

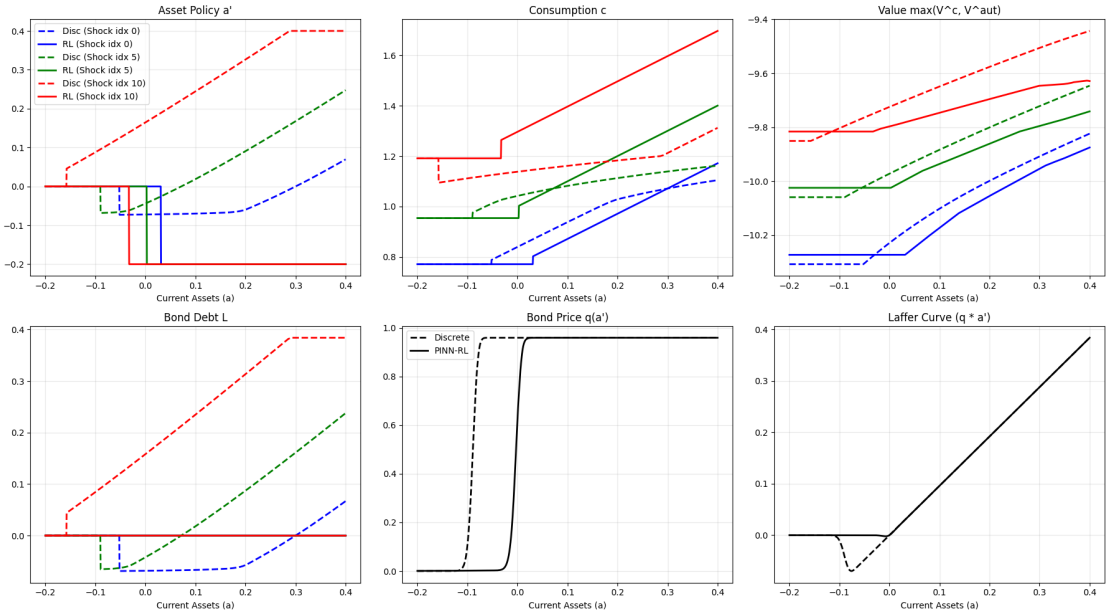

Fig. B.6: Under very poor initialization, the policy function and value objects temporarily fall into a stagnation phase — a region clearly disjoint from the benchmark.

Appendix B. Robustness to poor initialization in the one-period model

An important test of any numerical method is its robustness to poor initialization. We therefore investigate whether the gradient penetration mechanism can recover the solution when initialized far from the correct economic region. We observe that under poor initial conditions, the solver first enters a clear stagnation phase. As illustrated in Figure B.6, the asset policy initially degenerates into a flat line unresponsive to the state space, and the value and consumption objects deviate visibly from the benchmark solution.

However, as the soft continuation and soft repayment probabilities continuously allocate non-zero gradients to both branches, the network successfully escapes this stagnation. Figure B.7 demonstrates the recovery stage: continued gradient updates reliably guide the neural networks back to the correct economic region, ultimately converging to the benchmark solution.

We conclude that the StH solver not only successfully recovers a hard default boundary when given a well-designed initial guess, but its gradient penetration mechanics also allow it to systematically recover the structural solution from virtually any initial condition.

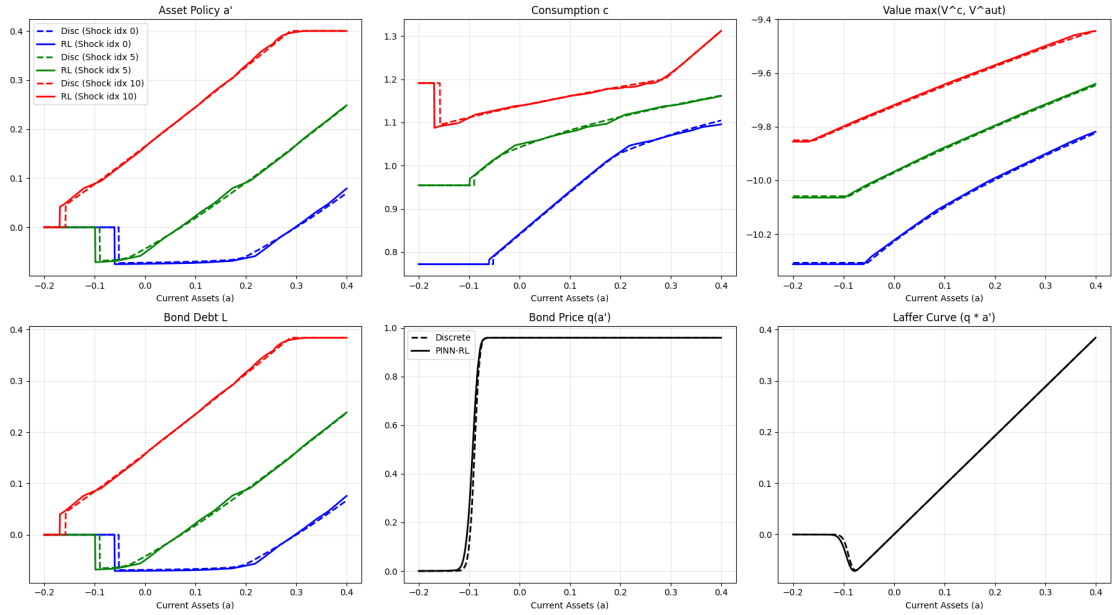

Fig. B.7: Continued gradient penetration brings the training path out of stagnation and converges tightly to the benchmark solution.

Appendix C. Intermediate bridge diagnostics for the long-term solver

This appendix reports the intermediate bridge diagnostics that connect the neural representation exercise in the main text to the final continuous StH actor–critic solver. These experiments are not the main economic results. Their role is to verify that the numerical transition from grid-based soft Bellman training to actor-based control is stable before the final solver removes permanent smoothing devices and evaluates the hard no-shock model.

Appendix C.1. Grid-based soft Bellman training with hard-model evaluation

The first bridge step relocates smoothing from the final economic decision layer to the training layer. Operators are softened during training to maintain gradient flow, but the exact, un-smoothed discrete choice remains the evaluation target. We refer to this intermediate step as Soft Q-Learning (SQL), which here denotes a grid-based soft Bellman update. Specifically, the hard maximization over next-period debt choices is replaced by a log-sum-exp operator over the discrete grid, and the binary repay/default choice is similarly smoothed during backpropagation. This step does not yet introduce a continuous policy actor.

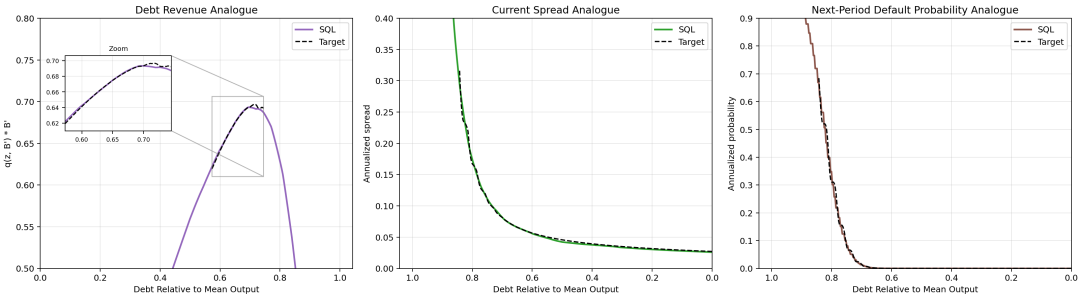

Fig. C.8: Grid-based soft Bellman training with hard-model evaluation. The panels show that relocating smoothing to the training operator preserves the main CE2012 equilibrium objects while maintaining gradient flow.

As demonstrated in Figure C.8, the main equilibrium objects—including the price schedule, spread curve, and default schedules—remain closely aligned with the benchmark. This confirms that the core philosophy of soft training paired with hard evaluation targets is numerically viable. However, because actions remain discrete and the policy still relies entirely on a pre-defined grid, the primary action-space bottleneck has not yet been resolved.

Appendix C.2. Actor-based control with a soft Bellman bridge

Transitioning from a global, value-based grid search to an actor-based policy method requires an intermediate structural reorganization. We introduce a dedicated policy network to handle the action-choice step while retaining the soft Bellman evaluation structure from the grid-based bridge. This diagnostic serves strictly as an actor–critic bridge rather than the final solver. Unlike standard Soft Actor-Critic (SAC) algorithms in the reinforcement learning literature—which rely on stochastic actors, permanent entropy regularization, and unconstrained exploration—this bridge utilizes a deterministic policy network explicitly trained to map states to the optimal choices induced by the underlying soft Bellman grid evaluations.

As illustrated in Figure C.9, the primary solution objects remain tightly aligned with the benchmark. Crucially, our experiments during this bridge reveal that if the soft Bellman training architecture is prematurely or abruptly replaced with a hard discrete choice, the actor's learning dynamics immediately destabilize. This confirms that a differentiable smoothing mechanism is strictly necessary to bridge the policy network across non-differentiable thresholds in the long-term sovereign default environment.

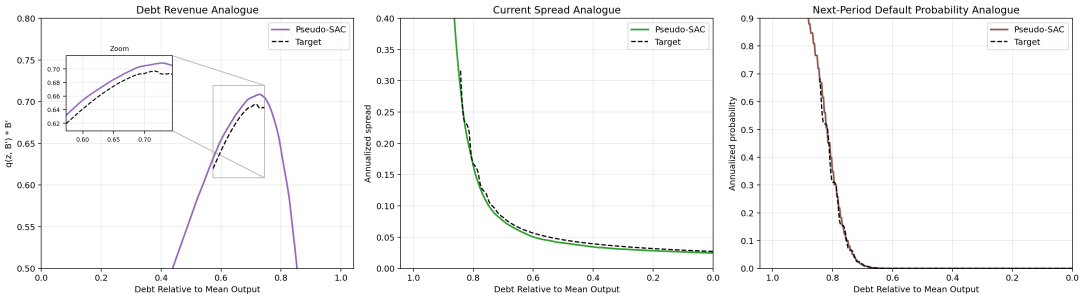

Fig. C.9: Actor-based control with a soft Bellman bridge. The panels confirm that the transition from value-based grid search to a policy network remains stable when the actor is trained through differentiable soft-Bellman targets.

Appendix D. Supplemental diagnostics and temperature sensitivity

This appendix provides extended diagnostic evidence regarding the structural limitations of permanent taste-shock smoothing, alongside intermediate validation checks for the internal mechanics of the StH solver. The first set of figures evaluates whether the taste-shock displacement in the main comparison is robust across endowment states and temperature values. The second set verifies that the intermediate grid-based soft-Bellman bridge can replicate the smoothed taste-shock benchmarks before the final StH solver returns to hard evaluation.

Throughout these diagnostics, economic primitives remain at the CE2012 calibration. The temperature exercises vary only the logit smoothing scales used in the taste-shock/logit benchmark. Thus, the appendix does not introduce a new calibration; it shows how the same economy is represented when the permanent smoothing device is made more or less diffuse.

Appendix D.1. Taste-shock diagnostics and the role of StH

The three-way comparison above also clarifies the role of taste-shock smoothing. Taste shocks are a useful and familiar way to regularize the choice problem, but the resulting equilibrium is a smoothed-choice economy. Reducing the taste-shock temperature sharpens the implied choice distribution, but it does not by itself recover the CE2012 threshold solution in the low-endowment region. An important interpretational detail is that StH does not have a primitive taste-shock choice distribution. The final StH actor is continuous and deterministic. When the figures report a StH soft distribution, it is only a diagnostic softmax over candidate B' values, constructed from the StH soft continuation value. It is therefore not directly comparable to the taste-shock model's primitive `bPol` object as a structural policy probability.

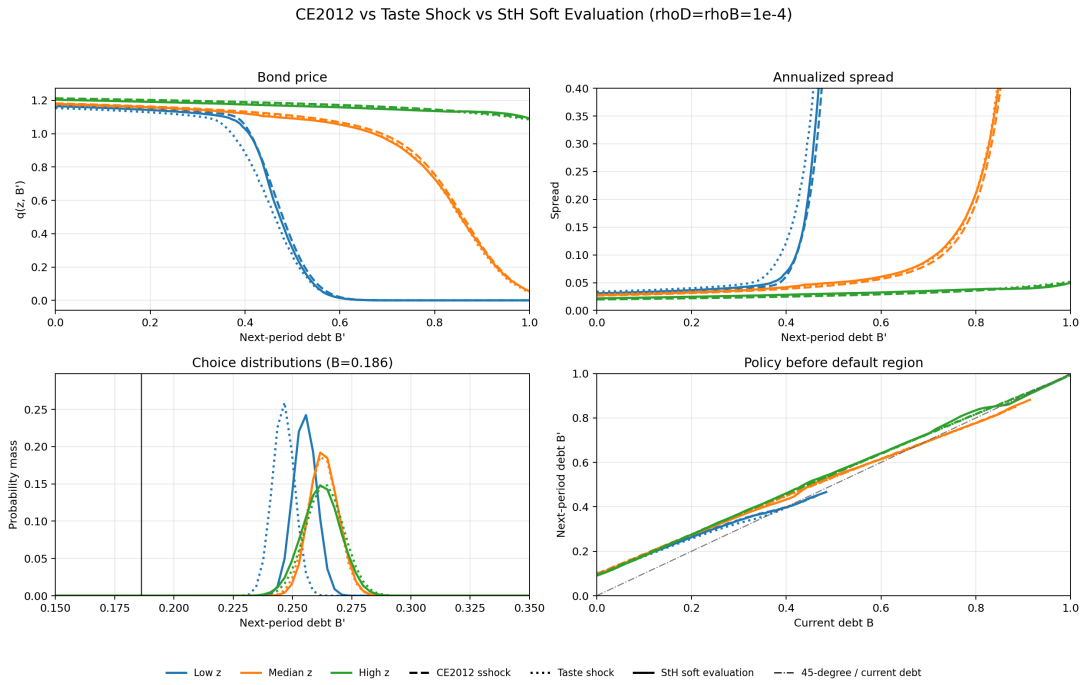

Fig. D.10: CE2012, taste-shock, and StH soft-evaluation diagnostics at $\rho_D = \rho_B = 10^{-4}$. The figure compares three representative endowment states and contains the same core information as the direct taste-shock versus StH comparison, while also displaying the CE2012 reference curves.

In the figures below, ρ_D and ρ_B denote the logit smoothing scales for the default and borrowing blocks. Larger values place more weight on nearby alternatives and spread the default boundary over a wider region of the state space. Smaller values sharpen the choice probabilities and move the object closer to a hard threshold, but they do not remove the fact that the benchmark is still solving a permanently smoothed economy.

Figure D.10 gives the main three-state comparison. It shows CE2012, the taste-shock solution, and the StH soft-evaluation diagnostic at the intermediate temperature $\rho_D = \rho_B = 10^{-4}$. The StH objects remain close to CE2012 in the price and spread panels, and the diagnostic StH soft distribution is close to the taste-shock distribution for the median and high endowment states. The largest remaining displacement occurs in the low endowment state, where default incentives are most sensitive. Figure D.11 shows that this low-to-middle-state displacement is not driven by a single selected state: it persists across a broader set of endowment slices.

We next examine whether the taste-shock displacement is simply a matter of temperature choice. Figures D.12 and D.13 compare the same objects after increasing

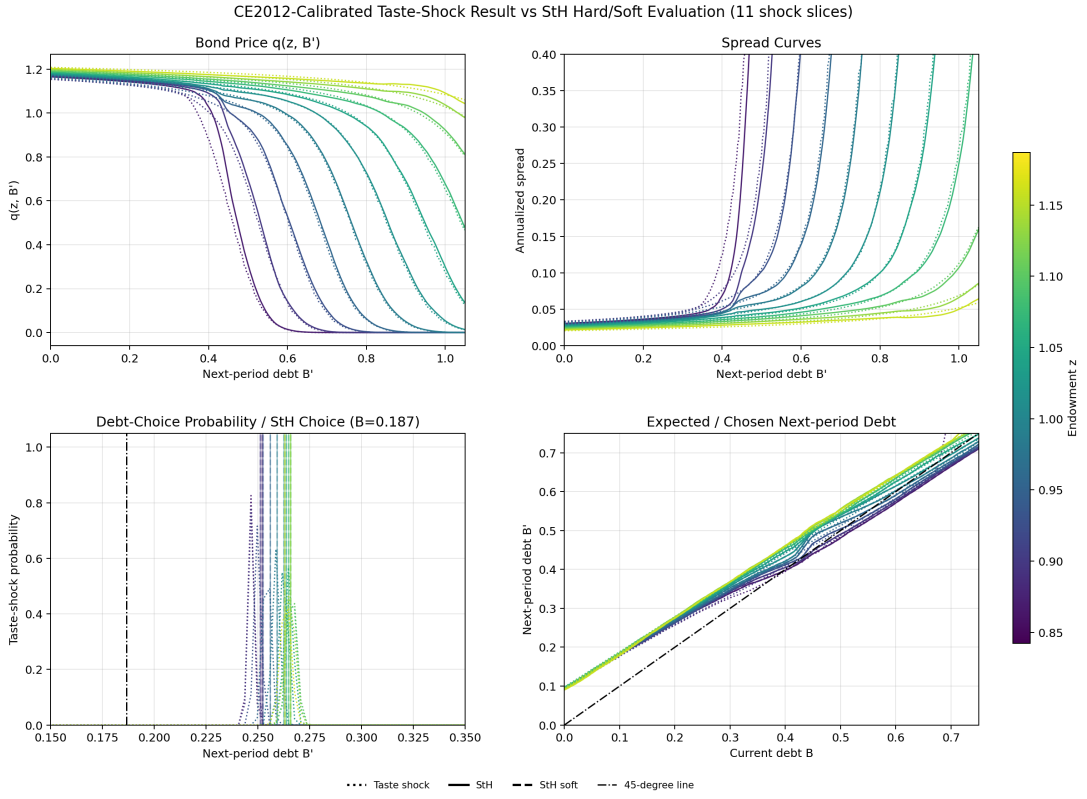

Fig. D.11: Taste-shock solution versus StH hard and soft evaluations for eleven endowment slices. The figure shows that the low-state displacement is not an artifact of selecting only one endowment state.

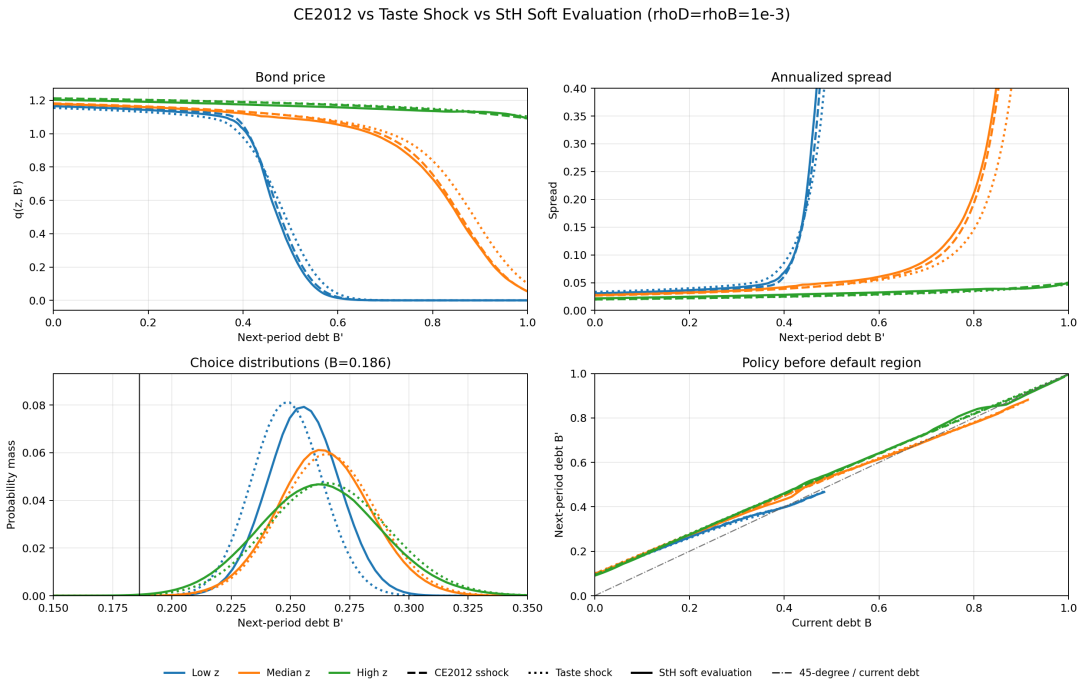

Fig. D.12: CE2012, taste-shock, and StH soft-evaluation diagnostics at $\rho_D = \rho_B = 10^{-3}$. The StH distribution in the lower-left panel is a diagnostic softmax over candidate B' values constructed from the StH soft continuation value, not a primitive actor probability.

and decreasing the smoothing scales around the intermediate case in Figure D.10. Raising the scales to 10^{-3} makes the choice probabilities more diffuse and widens the transition band. Lowering them to 10^{-5} sharpens the implied default and borrowing boundaries. In both cases, the underlying CE2012 calibration is unchanged. Across these temperature checks, the StH diagnostics remain close to CE2012, and the taste-shock displacement in the low endowment region is not mechanically eliminated by changing the smoothing scale.

These diagnostics should not be read as a failure of taste shocks as a numerical device. They show that the taste-shock/logit solution and the StH solution answer different computational questions. The taste-shock approach places smoothing directly into the economic environment. StH uses smooth probabilities only inside the training operator, where they provide gradient penetration, and then removes the smoothing from the final evaluation. This explains why StH can remain close to the CE2012 benchmark without requiring taste shocks in the final solution object. The separation between soft training and hard evaluation is the key point of the method.

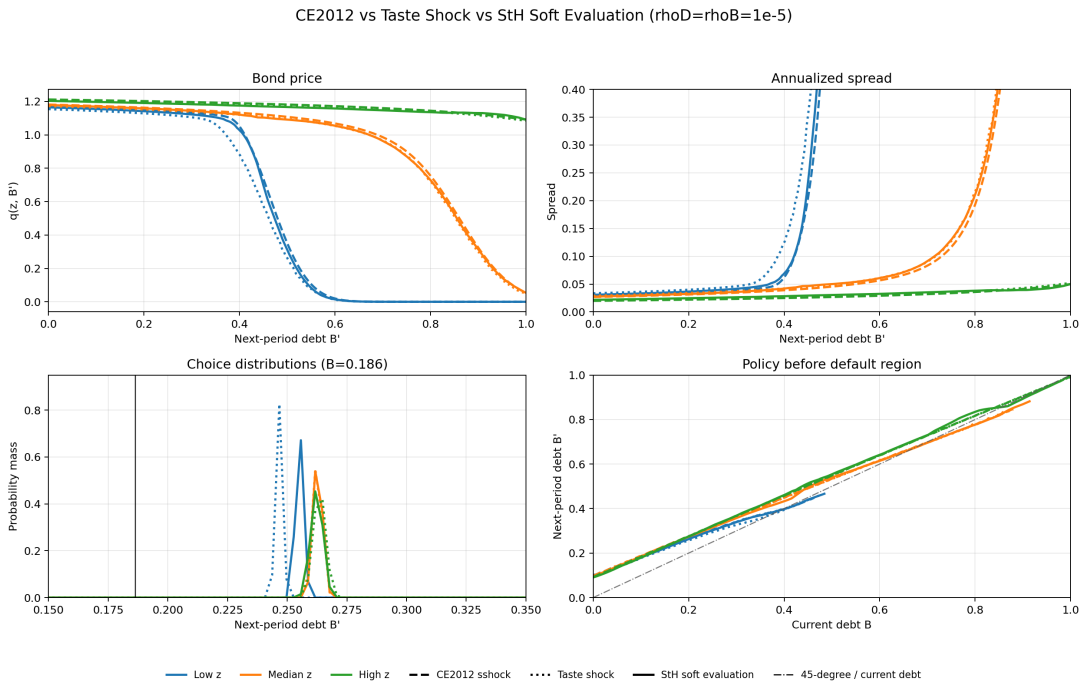

Fig. D.13: CE2012, taste-shock, and StH soft-evaluation diagnostics at $\rho_D = \rho_B = 10^{-5}$. Lowering the taste-shock temperature further sharpens the choice distribution, but the low-state taste-shock boundary does not mechanically converge to the CE2012 threshold solution.

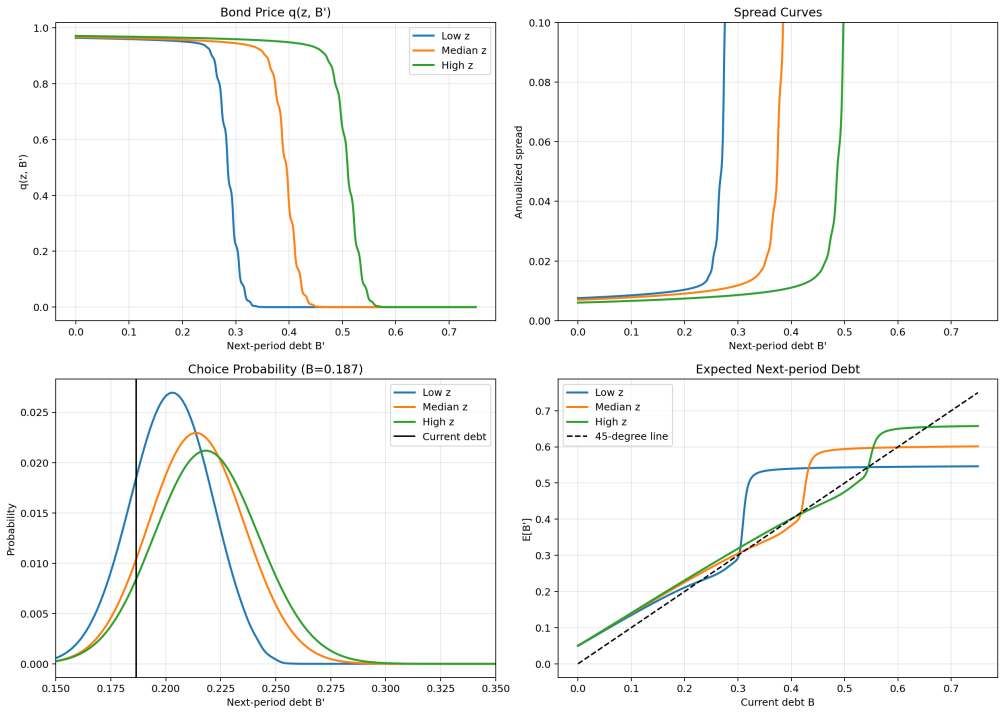

Fig. D.14: Taste-shock benchmark under a high-temperature setting. The panels report the benchmark price schedule, spread curve, choice probability, and expected next-period debt. The higher temperature produces a smoother benchmark geometry and a wider transition region.

Appendix D.2. Numerical illustration of taste-shock smoothing

Figure D.14 and Figure D.15 report the taste-shock benchmark under high-temperature and low-temperature settings. The four reported objects are: the bond-price schedule, spread curves, choice probabilities, and expected next-period debt. Under a high-temperature setting ($\sigma = 10^{-3}$ for both the borrowing and the default choice), the transition region of soft choice becomes wider, the probability mass in the choice probabilities becomes more dispersed, and the price cliff and spread explosion are stretched into a smoother transition band. Under a low-temperature setting ($\sigma = 10^{-5}$ for borrowing, $\sigma = 5 \times 10^{-4}$ for default), the switch becomes steep again, the default threshold and borrowing threshold move closer to a hard boundary, and the corresponding price cliff returns more quickly to the steep shape of the original model. These figures illustrate that the taste-shock benchmark can move between a smoother numerical object and a geometry closer to the hard-boundary model by changing the temperature. However, once smoothing is embedded in the benchmark, the solution object itself is smoothed.

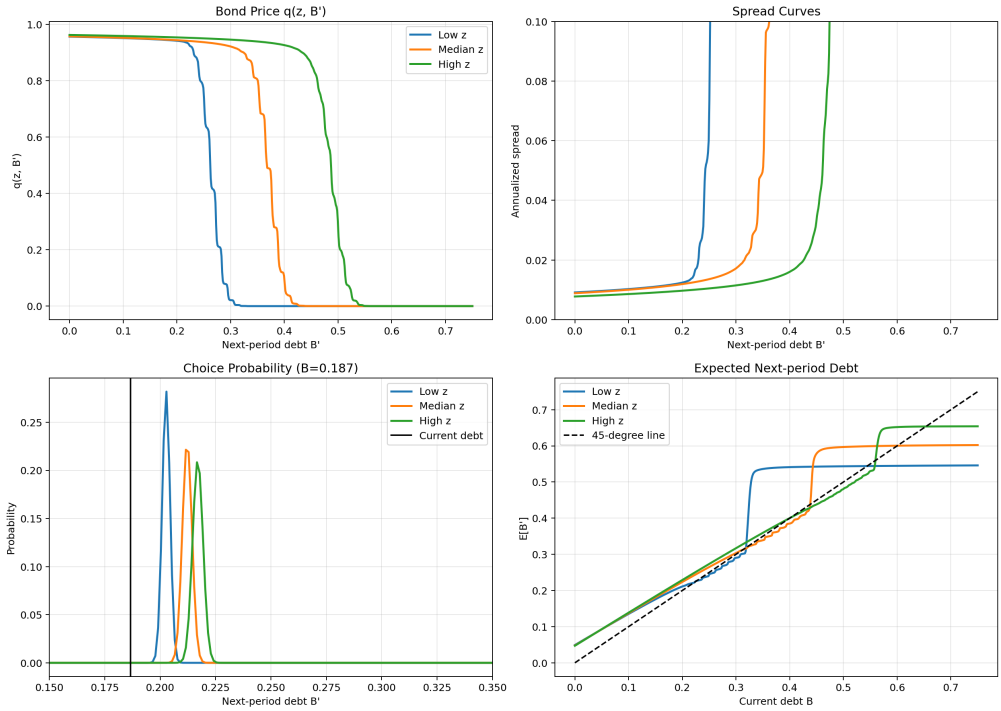

Fig. D.15: Taste-shock benchmark under a low-temperature setting. The same benchmark objects move closer to a hard-boundary geometry, with a steeper switch and a sharper price cliff.

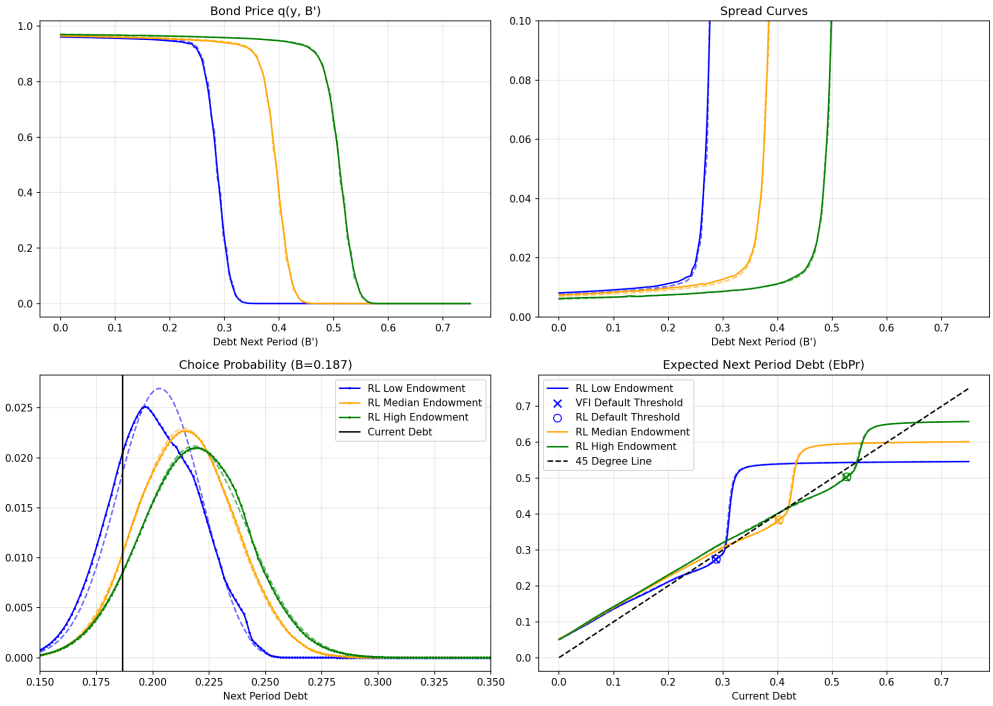

Fig. D.16: Soft-Bellman bridge diagnostic under the high-temperature taste-shock benchmark. The figure compares the grid-based soft-Bellman bridge solution with the smoothed benchmark for prices, spreads, choice probabilities, and expected next-period debt.

We also keep the earlier taste-shock comparison diagnostics in this stage rather than in Section 3.3.2. These figures are bridge diagnostics: they show that the SQL / soft-Bellman training route can match the smoothed taste-shock benchmark before we return to the hard no-shock target. Their role is diagnostic, while the final hard-model evidence is reported in the hard-evaluation figures. In these legacy plots, the label y denotes the endowment state z.

References

Achdou, Y., Han, J., Lasry, J.-M., Lions, P.-L., and Moll, B. (2022). Income and wealth distribution in macroeconomics: A continuous-time approach. *The review of economic studies*, 89(1):45–86.

Aguiar, M. and Amador, M. (2020). Self-fulfilling debt dilution: Maturity and multiplicity in debt models. *American Economic Review*, 110(9):2783–2818.

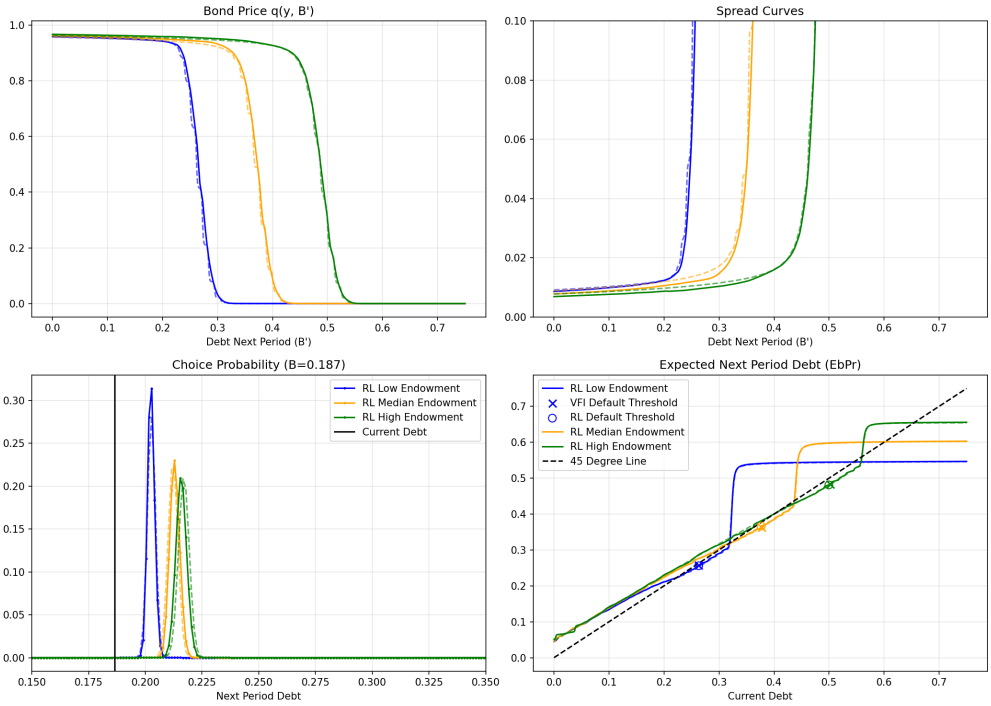

Fig. D.17: Soft-Bellman bridge diagnostic under the low-temperature taste-shock benchmark. The same comparison is reported for the lower-temperature benchmark, where the choice probabilities and price schedules are closer to the hard-boundary geometry.

Aguiar, M., Amador, M., Hopenhayn, H. A., and Werning, I. (2013). Take the short route: How to repay and restructure sovereign debt with multiple maturities. In *Handbook of International Economics*, volume 4, pages 1697–1755. Elsevier.

Arellano, C. (2008). Default risk and income fluctuations in emerging economies. *American Economic Review*, 98(3):690–712.

Arellano, C. and Ramanarayanan, A. (2012). Default and the maturity structure in sovereign bonds. *Journal of Political Economy*, 120(2):187–232.

Azinovic, M., Gaegauf, L., and Scheidegger, S. (2022). Deep equilibrium nets. *International Economic Review*, 63(4):1471–1525.

Berry, S., Levinsohn, J., and Pakes, A. (1995). Automobile prices in market equilibrium. *Econometrica*, 63(4):841–890.

Charpentier, A., Élie, R., and Remlinger, C. (2023). Reinforcement learning in economics and finance. *Computational Economics*, 62(1):425–462.

Chatterjee, S. and Eyigungor, B. (2012). Maturity, indebtedness, and default risk. *American Economic Review*, 102(6):2674–2699.

Chatterjee, S. and Mitra, N. (2024). A quantitative theory of installment loans. Working paper.

Chen, M., Joseph, A., Kumhof, M., Pan, X., and Zhou, X. (2021). Deep reinforcement learning in a monetary model. *arXiv preprint arXiv:2104.09368*.

Druedahl, J. (2021). A guide on solving non-convex consumption-saving models. *Computational Economics*, 58(3):747–775.

Duarte, V. F. (2018). Machine learning for continuous-time economics. Working paper.

Dvorkin, M., Sánchez, J. M., Sapriza, H., and Yurdagul, E. (2021). Sovereign debt restructurings. *American Economic Journal: Macroeconomics*, 13(2):26–77.

Eaton, J. and Gersovitz, M. (1981). Debt with potential repudiation: Theoretical and empirical analysis. *The Review of Economic Studies*, 48(2):289–309.

Feng, Z., Han, J., and Zhu, S. (2024). Optimal taxation with incomplete markets—an exploration via reinforcement learning. Working paper.

Fernández-Villaverde, J. (2026). Deep learning for solving economic models. *Journal of Economic Literature*. Forthcoming.

Fernández-Villaverde, J., Nuno, G., Sorg-Langhans, G., and Vogler, M. (2020). Solving high-dimensional dynamic programming problems using deep learning. *Unpublished working paper*.

Gennaioli, N., Martin, A., and Rossi, S. (2014). Sovereign default, domestic banks, and financial institutions. *The Journal of Finance*, 69(2):819–866.

Gopalakrishna, G. (2021). Aliens and continuous time economies. *Swiss Finance Institute Research Paper*, 21(21-34).

Gopalakrishna, G., Payne, J., and Gu, Z. (2024). Asset pricing, participation constraints, and inequality. *Participation Constraints, and Inequality (November 8, 2024)*.

Gordon, G. (2019). Efficient computation with taste shocks. Working Paper 19-15, Federal Reserve Bank of Richmond.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1861–1870.

Han, J., Jentzen, A., and E, W. (2018). Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*, 115(34):8505–8510.

Han, J., Yang, Y., et al. (2021). Deepham: A global solution method for heterogeneous agent models with aggregate shocks. *arXiv preprint arXiv:2112.14377*.

Hatchondo, J. C. and Martinez, L. (2009). Long-duration bonds and sovereign defaults. *Journal of International Economics*, 79(1):117–125.

Huang, J. (2022). A probabilistic solution to high-dimensional continuous-time macro and finance models. *Available at SSRN 4122454*.

Iskhakov, F., Jørgensen, T. H., Rust, J., and Schjerning, B. (2017). The endogenous grid method for discrete-continuous dynamic choice models with (or without) taste shocks. *Quantitative Economics*, 8(2):317–365.

Jirnyi, A. and Lepetyuk, V. (2011). A reinforcement learning approach to solving incomplete market models with aggregate uncertainty. Working Paper WP-AD 2011-21, Instituto Valenciano de Investigaciones Económicas.

Konda, V. R. and Tsitsiklis, J. N. (2000). Actor-critic algorithms. In *Advances in Neural Information Processing Systems 12*, pages 1008–1014.

Li, B., Maliar, S., and Zhang, P. (2026). An informed actor–critic method for economic dynamics. Working paper.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Maliar, L. and Maliar, S. (2022). Deep learning classification: Modeling discrete labor choice. *Journal of Economic Dynamics and Control*, 135:104295.

Maliar, L., Maliar, S., and Winant, P. (2021). Deep learning for solving dynamic economic models. *Journal of Monetary Economics*, 122:76–101.

Mateos-Planas, X., McCrary, S., Ríos-Rull, J.-V., and Wicht, A. (2025). Commitment in the canonical sovereign default model. *Journal of International Economics*, 157:104120.

McFadden, D. (1974). Conditional logit analysis of qualitative choice behavior. In Zarembka, P., editor, *Frontiers in Econometrics*, pages 105–142. Academic Press, New York.

Mihalache, G. (2024). Solving default models. In preparation for the Oxford Research Encyclopedia of Economics and Finance.

Mosavi, A., Faghan, Y., Ghamisi, P., Duan, P., Ardabili, S. F., Salwana, E., and Band, S. S. (2020). Comprehensive review of deep reinforcement learning methods and applications in economics. *Mathematics*, 8(10):1640.

Payne, J., Rebei, A., and Yang, Y. (2025a). Deep learning for search and matching models. *Available at SSRN 5123878*.

Payne, J., Rebei, A., and Yang, Y. (2025b). Deep learning for search and matching models. Technical Report 25-05, Swiss Finance Institute Research Paper Series. First draft February 2024.

Rust, J. (1987). Optimal replacement of GMC bus engines: An empirical model of harold zurcher. *Econometrica*, 55(5):999–1033.

Sánchez, J. M., Sapriza, H., and Yurdagul, E. (2014). Sovereign default and the choice of maturity. Working Paper 2014-031, Federal Reserve Bank of St. Louis.

Sauzet, M. (2021). Projection methods via neural networks for continuous-time models. *Available at SSRN 3981838*.

Su, C.-L. and Judd, K. L. (2012). Constrained optimization approaches to estimation of structural models. *Econometrica*, 80(5):2213–2230.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 2 edition.

van der Kwaak, C. and van Wijnbergen, S. (2014). Financial fragility, sovereign default risk and the limits to commercial bank bail-outs. *Journal of Economic Dynamics and Control*, 43:218–240.

Watkins, C. J. C. H. and Dayan, P. (1992). Q-learning. *Machine Learning*, 8(3–4):279–292.

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3–4):229–256.