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

To solve discrete-continuous dynamic choice models, the quantitative literature has long relied on effective permanent structural smoothing to handle non-differentiable kinks. While traditional methods produce robust qualitative insights, their permanent modifications can alter quantitative predictions. To solve the true, unsmoothed economy, we introduce a soft-to-hard (StH) deep reinforcement learning framework. By systematically annealing temporary smoothing to zero, StH enables gradient propagation across decision boundaries before enforcing hard-switch operators. We implement this framework in two complementary ways. The grid-based StH-Q implementation delivers high numerical precision in the low-dimensional benchmark, whereas the continuous-state, continuous-action StH-AC implementation avoids storing a full state–action table and provides a scalable route to higher-dimensional applications.

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

*This research was supported by NSF grant SES-1949430, National Natural Science Foundation of China (No. 72450002), and Beijing Philosophy and Social Science Foundation (No. 24DTR024).

Email addresses: `libo.econ@pku.edu.cn` (Bo Li), `maliars@stanford.edu` (Serguei Maliar), `zpyuan@pku.edu.cn` (Peiyuan Zhang)

1. Introduction

Discrete-continuous dynamic choice (DCDC) models—widely used across labor economics, industrial organization, public finance, and macroeconomics—combine endogenous qualitative choices with continuous quantitative optimization. However, solving these models presents a fundamental computational bottleneck: discrete switches yield value functions formed by upper envelopes and introduce non-differentiable kinks that propagate exponentially during backward induction. To achieve numerical tractability, the quantitative literature frequently relies on permanent smoothing mechanisms—such as auxiliary income shocks (Chatterjee and Eyigungor, 2012) or extreme-value taste shocks (Gordon, 2019; Dvorkin et al., 2021; Mihalache, 2024)—to eliminate these non-convexities. These innovations have successfully eliminated non-convexities, stabilized spatial grids, and allowed researchers to derive critical qualitative insights.

Yet, this computational tractability requires an uncomfortable compromise: permanent smoothing structurally alters the target economic environment. To solve the exact, unsmoothed economic environment without suffering from the curse of dimensionality, we introduce a novel soft-to-hard (StH) deep reinforcement learning framework. Rather than permanently altering the structural model, our approach utilizes smoothing strictly as a temporary numerical device during the initial learning phase. By temporarily replacing hard discrete switches with soft choice probabilities, we enable gradients to propagate across non-differentiable decision boundaries, overcoming the vanishing gradient problem that typically prevents deep learning algorithms from evaluating discrete choices. Crucially, this artificial softness is systematically annealed to zero, allowing the policy and pricing networks to refine and evaluate decisions strictly under the exact hard-switch operators. Consequently, the StH solver isolates the true economic mechanics of DCDC models, providing a domain-agnostic algorithm that frees economic analysis from the distortions of problem-specific structural smoothing.

Methodologically, our solver bridges structural microeconometrics and deep reinforcement learning; see Fernández-Villaverde (2026) and Li et al. (2026) for recent reviews of the deep reinforcement literature. The log-sum-exp formulation—traditionally used to model Extreme Value Type-I taste shocks in discrete choice models (McFadden, 1974; Rust, 1987; Berry et al., 1995)—mirrors maximum entropy reinforcement learning in artificial intelligence (e.g., Soft Actor-Critic; see Haarnoja et al., 2018). However, our primary methodological contribution is utilizing this structure strictly as a temporary, vanishing regularization device. Unlike Maliar and Maliar (2022), who construct choice boundaries via cross-entropy classification, our architecture embeds an annealing temperature parameter directly into continuous continuation values. This enables smooth gradient propagation across competing value branches during training, before refining the coupled policy, value, and pricing networks under the exact hard-switch operator.

We evaluate the StH solver on sovereign default models—a canonical and challenging laboratory for DCDC environments. First introduced in short-term debt settings by Eaton and Gersovitz (1981) and Arellano (2008), and advanced to long-term debt frameworks by Hatchondo and Martinez (2009), Sánchez et al. (2014) and Chatterjee and Eyigungor (2012) (hereafter CE2012), these models combine a hard binary choice to repay or default with continuous debt choices and non-linear, recursive bond pricing. To address the computational challenges in these settings, the quantitative macroeconomics literature has developed specialized numerical algorithms and structural modifications. One track leverages geometric filtering to handle non-convexities, 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 tractability by permanently smoothing the environment through auxiliary income shocks (Chatterjee and Eyigungor, 2012) or Extreme Value preference shocks (Gordon, 2019; Dvorkin et al., 2021; Mihalache, 2024). Other approaches bypass these discrete-time frictions altogether using random maturity structures (Arellano and Ramanarayanan, 2012) or continuous-time formulations (Aguiar et al., 2013; Aguiar and Amador, 2020).

Although effective, permanent structural modifications alter the economic environment and can shift counterfactual policy implications. Our objective—evaluating policies under original hard-switch decision rules—aligns with a pioneering work of Mateos-Planas et al. (2026), who derive a customized system of generalized Euler equations. However, their framework requires extensive model-specific derivations and remains bound to discrete grids subject to the curse of dimensionality. By contrast, the StH approach is domain-agnostic and highly scalable: it combines neural network function approximation with continuous-action spaces, evaluating exact hard-switch decision rules without relying on dense grids or problem-specific analytical customizations.

We study two complementary implementations of the StH framework. StH-Q is a grid-based tabular implementation that provides a transparent small-model benchmark, whereas StH-AC uses neural function approximation and continuous actions to preserve the framework's route to larger state spaces. Both anneal temporary smoothing to zero and evaluate the original unsmoothed model under hard decisions.

Applying the StH solver across the multi-period sovereign debt environments, we establish four main results:

1. *Permanent smoothing distorts macroeconomic dynamics.* All four implementations yield qualitatively similar predictions, but permanent smoothing is not innocuous. In the high-volatility common-history comparison, the maximum spread gap is 6.04 percentage points, or 22.58% relative to the lowest contemporaneous spread.

2. *Attempting to sharpen default boundaries in smoothing methods destabilizes fixed-point convergence.* Traditional smoothing methods cannot simply be "tuned" to recover the exact hard solution. When smoothing parameters are shrunk to enforce sharper boundaries, both auxiliary-shock and taste-shock solvers suffer fixed-point convergence failure. This numerical breakdown generates massive structural variance, causing predicted debt-to-output ratios to wildly swing by 15% and 22%, respectively—an unacceptable margin of error driven entirely by arbitrary parameter choices.

3. *StH-Q achieves high numerical precision in small problems.* The tabular StH-Q implementation solves the original unsmoothed problem on the common grid. In the sovereign-default benchmark it provides a transparent same-grid validation, attaining highly accurate values, policies, prices, and default boundaries without permanent smoothing.

4. *StH-AC provides a scalable neural implementation.* StH-AC represents policies and equilibrium objects with neural networks over continuous states and actions, rather than storing the full state–action table or restricting debt choice to a finite action grid during policy optimization. It therefore preserves the StH framework's route to higher-dimensional discrete-continuous choice models. Together, the two implementations separate the roles cleanly: StH-Q establishes high precision in the small benchmark, while StH-AC supplies the representation needed for larger applications such as lumpy investment, portfolio choice with transaction costs, and occupational choice.

Sections 2–4 develop the soft-to-hard framework, specify the sovereign default model, and describe its numerical implementations. Section 5 retains the original numerical discussion and places the updated four-method comparison alongside it. Section 6 concludes. The appendices contain proofs, validation experiments, implementation details, and the supplementary numerical comparisons.

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

We first formally define a generic discrete-continuous dynamic choice (DCDC) environment, formulate the soft-to-hard (StH) computational method, and establish its theoretical properties. Our presentation abstracts from specific application details to highlight the universal computational bottleneck inherent in DCDC models. In the following section, we demonstrate how to apply the StH method to the specific problem of sovereign default.

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):

⋆ 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 $d \in \{0, 1\}$.

⋆ 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 during training, the suboptimal 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 friction by temporarily smoothing the max operator in $V(s)$ during training, while retaining the branch-specific structural conditions of $V^d(s)$.

2.2. Structure of the StH solver

Our solver utilizes four neural networks:

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

where 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):

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

which 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.

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 $p^1_\tau(s')$. These Bellman and equilibrium-network targets are detached regression targets: the soft probability makes the target vary smoothly with future states and actions, but gradients from the regression losses are not propagated through the target networks or target actor.

2.4. Why gradient penetration can train hard discrete choices

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

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

The implemented actor update optimizes this branch value directly; it does not place a current-state log-sum-exp envelope around $V^1_\omega(s)$ and $V^0_\psi(s)$. Gradient penetration instead operates through the next-period continuation. Let s'_ω denote the next state induced by $x_\omega(s)$. Away from ties, the hard continuation $\mathcal{C}_H(s'_\omega) = \max\{V^1(s'_\omega), V^0(s'_\omega)\}$ has the derivative

$$\nabla_\omega \mathcal{C}_H(s'_\omega) = \begin{cases} \nabla_\omega V^1(s'_\omega), & V^1(s'_\omega) > V^0(s'_\omega), \\ \nabla_\omega V^0(s'_\omega), & V^0(s'_\omega) > V^1(s'_\omega). \end{cases} \tag{11}$$

Thus a hard switch can abruptly turn an action-dependent continuation gradient on or off. During soft training, the corresponding derivative is

$$\nabla_\omega \mathcal{C}_\tau(s'_\omega) = p^1_\tau(s'_\omega) \nabla_\omega V^1(s'_\omega) + \left[1 - p^1_\tau(s'_\omega)\right] \nabla_\omega V^0(s'_\omega). \tag{12}$$

Consequently, differentiation of Equation (10) through the online critics gives

$$\nabla_\omega V^1_\omega(s) = \nabla_\omega U^1(s, x_\omega(s)) + \beta \mathbb{E}_{s'}[\nabla_\omega \mathcal{C}_\tau(s'_\omega)]\,. \tag{13}$$

For every $\tau > 0$, the soft weights in Equation (12) change continuously near the future discrete-choice boundary. The actor can therefore improve its repayment-contingent policy without the discontinuous branch selection in Equation (11).

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 $g(s, x_\omega) \le 0$. We then define a feasibility penalty:

$$pen_\omega(s) = \max\{g(s, x_\omega(s)), 0\}^2 \tag{14}$$

The combined actor loss is defined as:

$$L_{actor} = -\mathbb{E}\left[U^1(s, x^{safe}_\omega(s)) + \beta \mathbb{E}_{s'}\left[\mathcal{C}_\tau(s')\right]\right] + \lambda_{feas} \mathbb{E}[pen_\omega(s)] \tag{15}$$

In Equation (15), x^{safe} ensures the utility function does not crash on invalid inputs, while Equation (14) 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 refinement, and evaluation

The first training stage uses the temperature annealing schedule

$$\tau(t) = \tau_{init} \left(\frac{\tau_{final}}{\tau_{init}} \right)^{\min\{1, t/T_{ann}\}} \tag{16}$$

Equation (16) makes the soft operator gradually approach the true hard comparison while retaining a positive soft-stage terminal temperature. In the second training stage, the soft operators are removed and the continuation, choice, pricing, and actor updates take their exact hard forms, corresponding operationally to the zero-temperature limit. After these two training stages, evaluation uses the same hard continuation and discrete rule:

$$p^1_{hard}(s) = \mathbf{1}\{V^1(s) \geq V^0(s)\}, \quad \mathcal{C}_{hard}(s) = \max\{V^1(s), V^0(s)\} \tag{17}$$

Equation (17) ensures that first-stage softness serves purely as a numerical regularization device, while second-stage hard refinement and subsequent hard evaluation preserve the structural interpretation of the economic model. The hard-refinement block reconciles the actor, critics, and equilibrium network with this operator before evaluation and reporting. The properties below characterize the local training mechanism and the operator limit, not convergence of the coupled neural solver.

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: gradient penetration and the hard-operator limit. These properties are outlined below, with formal proofs 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:

$$\frac{\partial \mathcal{C}_\tau(s)}{\partial V^1(s)} = p^1_\tau(s), \quad \frac{\partial \mathcal{C}_\tau(s)}{\partial V^0(s)} = 1 - p^1_\tau(s) \tag{18}$$

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—the soft operator assigns the losing branch a positive analytical weight, as long as $\tau > 0$. This permits local updates near indifference states.

Proposition 2 (Hard-operator limit). For any state s, the soft continuation satisfies the strict bounds:

$$\max\{V^1(s), V^0(s)\} \le \mathcal{C}_\tau(s) \le \max\{V^1(s), V^0(s)\} + \tau \log 2 \tag{19}$$

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 hard maximum. Furthermore, evaluating the limit of $p^1_\tau(s)$ as $\tau \downarrow 0$ confirms that the probability collapses to exactly 1 or 0, except at the exact indifference boundary.

Implication. The approximation bound in Equation (19) vanishes as $\tau \downarrow 0$. Furthermore, at any state s where $V^1(s) \ne V^0(s)$, the soft probability $p^1_\tau(s)$ converges pointwise to the hard indicator $1\{V^1(s) \ge V^0(s)\}$ as $\tau \downarrow 0$.

In sum, gradient penetration assigns positive analytical weights to both branches during training, while the hard-operator limit ensures that as the temperature approaches zero, the soft operator recovers the original hard-switch economic objects.

3. Sovereign default model by Chatterjee and Eyigungor (2012)

We now describe our primary DCDC application: the canonical long-term debt model of Chatterjee and Eyigungor (2012) (hereafter CE2012). To evaluate the StH solution to the original, unsmoothed problem against established benchmarks, we also present two traditional smoothing paradigms—auxiliary income shocks and 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 generic symbol $d \in \{0, 1\}$ remains a branch label in the DCDC framework. In the CE2012 application we use R and D for the two branches and reserve $\delta(z, B) = 1$ for default. This convention avoids reversing the indicator in the lenders' pricing equation.

Continuous Action (x):. Conditional on repayment (branch R), 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 coup > 0. We define the per-unit servicing cost as:

$$\kappa \equiv \lambda + (1 - \lambda)\text{coup}. \tag{20}$$

If the sovereign repays, the period budget constraint is:

$$c^R(z, B, B') = z - \kappa B + q(z, B')[B' - (1 - \lambda)B] \tag{21}$$

In Equation (21), $q(z, B')$ 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 (20), and $q(z, B')[B' - (1 - \lambda)B]$ is the revenue from net new issuance.

3.2. The modified economies: artificial smoothing for computational tractability

We first describe two standard approaches in the DCDC literature that permanently modify the economic environment through artificial smoothing: auxiliary income shocks (Chatterjee and Eyigungor, 2012) and extreme-value taste shocks (Gordon, 2019; Dvorkin et al., 2021; Mihalache, 2024).

3.2.1. Auxiliary income shocks

CE2012 introduces a small i.i.d. transitory income shock m. The shock enters repayment consumption, but the original CE2012 code resets the transitory component to its lower support point, $-\bar{m}$, when default occurs:

$$c^R_m(z, B, B') = z + m - \kappa B + q(z, B')[B' - (1 - \lambda)B], \tag{22}$$

$$c^D_{-\bar{m}}(z) = h(z) - \bar{m}. \tag{23}$$

Let $\bar{V}(z,B) = \int V(z,B,m)\,dF_m(m)$ and $\bar{X}(z) = \int X(z,m)\,dF_m(m)$ denote next-period values before the future transitory shock is realized. The current choice-specific objects are

$$V^R(z,B,m) = \max_{B' \in [0,B_{max}]} \left\{ u(c_m^R(z,B,B')) + \beta E_{z'|z} \bar{V}(z',B') \right\}, \tag{24}$$

$$X(z,m) = u(h(z)+m) + \beta E_{z'|z} \left[\chi \bar{V}(z',0) + (1-\chi)\bar{X}(z') \right]. \tag{25}$$

Upon entry into default, the code evaluates the second equation at $m = -\bar{m}$. The realized decision therefore remains hard, but the default branch in that comparison is independent of the currently realized m:

$$\delta(z,B,m) = \mathbf{1}\{X(z,-\bar{m}) > V^R(z,B,m)\}. \tag{26}$$

Lenders integrate this decision over the auxiliary shock,

$$\bar{\delta}(z,B) = \int \delta(z,B,m)\,dF_m(m), \tag{27}$$

and use $1-\bar{\delta}$ in the repayment-weighted price update. The integration smooths expected repayment. The shock m enters income and does not enter additively in the value function. This timing and reset convention matches the original CE2012 implementation used for the auxiliary results below.

3.2.2. Taste-shock smoothing

The taste-shock method used below has two smoothing scales. The borrowing scale ρ_B smooths the debt choice on a finite grid $\mathcal{G}_B$. For the repayment return $Q^R(z,B,B')$, define

$$V^R_{\rho_B}(z,B) = \rho_B \log \sum_{B' \in \mathcal{G}_B} \exp\left(\frac{Q^R(z,B,B')}{\rho_B} \right), \tag{28}$$

$$\pi_{\rho_B}(B' \mid z,B) = \frac{\exp(Q^R(z,B,B')/\rho_B)}{\sum_{\widetilde{B} \in \mathcal{G}_B} \exp(Q^R(z,B,\widetilde{B})/\rho_B)}. \tag{29}$$

The default scale ρ_D smooths the repay/default choice:

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

$$P_{\rho_D}(\text{default} \mid z,B) = \frac{\exp(V^D(z)/\rho_D)}{\exp(V^R_{\rho_B}(z,B)/\rho_D) + \exp(V^D(z)/\rho_D)}. \tag{31}$$

We normalize the common additive extreme-value constant to zero. The limits $\rho_B \downarrow 0$ and $\rho_D \downarrow 0$ recover the debt-grid maximum and hard default choice when the relevant maximizers are unique. Following Mihalache (2024, Table 1), the main economic comparison uses $(\rho_D, \rho_B) = (5 \times 10^{-4}, 10^{-5})$. The sensitivity experiment scales both temperatures by the same multiplier, preserving $\rho_D/\rho_B = 50$.

3.3. The original hard-switch unsmoothed economy

We now describe the original, unsmoothed version of the economy, which we solve using the StH method. In this structural version, there are no unobserved shocks smoothing the agent's decision. If the sovereign repays, it remains in credit markets and solves:

$$V^R(z,B) = \max_{B' \in [0, B_{max}]} \left\{ u(c^R(z,B,B')) + \beta \sum_{z'} P_z(z,z') V(z',B') \right\} \tag{32}$$

If the sovereign defaults, it is excluded from credit markets and loses output. The CE2012 default-output function used in the replication is

$$\widetilde{h}(z) = z - \max\{0, d_0 z + d_1 z^2\}, \tag{33}$$

$$z^* \in \arg\max_{z \in \mathcal{G}_z} \widetilde{h}(z), \tag{34}$$

$$h(z) = \begin{cases} \widetilde{h}(z), & z < z^*, \\ \widetilde{h}(z^*), & z \geq z^*. \end{cases} \tag{35}$$

Here $\mathcal{G}_z$ is the endowment grid. The final line reproduces the upper-output plateau in the CE2012 code and is inactive when the maximum is at the top grid point. The sovereign re-enters credit markets next period with exogenous probability $\chi \in (0,1)$ and zero debt. Its default value is:

$$V^D(z) = u(h(z)) + \beta \sum_{z'} P_z(z,z') \left[\chi V(z',0) + (1-\chi) V^D(z') \right]. \tag{36}$$

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

$$V(z,B) = \max\{V^R(z,B), V^D(z)\}. \tag{37}$$

The optimal default decision is a strict, hard indicator:

$$\delta(z,B) = \mathbf{1}\{V^D(z) > V^R(z,B)\} \tag{38}$$

Because international lenders price debt based on the hard next-period default event, the recursive pricing equation incorporates the hard jump in Equation (38):

$$q(z, B') = \frac{1}{1 + r_f} \sum_{z'} P_z(z, z')\big(1 - \delta(z', B')\big) \left[\lambda + (1 - \lambda) \left(\text{coup} + q(z', A(z', B'))\right)\right] \tag{39}$$

where $A(z', B')$ denotes the optimal continuous debt policy conditional on repayment. Equations (32) –(39) define the hard-switch CE2012 baseline used for the StH solver's hard evaluation: repayment in Equation (32), default in Equation (36), the upper envelope in Equation (37), and pricing in Equation (39). Solving this hard-switch framework is difficult because the hard indicator $\delta(z, B)$ halts gradient propagation and shatters the continuity of the bond price schedule.

4. Implementation of the StH solver for the sovereign default model

We now describe an implementation of the StH solver for the CE2012 sovereign default environment. The reported implementation has two training stages followed by separate evaluation and reporting. The first stage performs 1,000,000 soft-training updates while annealing the smoothing operator to a low positive terminal temperature. The second stage performs 100,000 hard-refinement updates of the actor, repayment and default values, and bond-price network using the exact hard operators, corresponding operationally to zero temperature. After training, policies, prices, residuals, and simulations are evaluated and reported under the hard repay/default rule.

4.1. Network architecture and bounding

All four networks $B'_\omega(z, B)$, $V^R_\theta(z, B)$, $V^D_\psi(z)$, and $q_\phi(z, B')$, corresponding to the generic objects in Equation (3), 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 actor's current proposal implies repayment consumption

$$c_\omega(z, B) = z - \kappa B + q_\phi(z, B'_\omega(z, B))\left[B'_\omega(z, B) - (1-\lambda)B\right]. \tag{40}$$

For all training targets, we use the positive consumption floor $\underline{c} > 0$ and define $c^{safe}_\omega(z, B) = \max\{c_\omega(z, B), \underline{c}\}$ and $pen_\omega(z, B) = \max\{\underline{c} - c_\omega(z, B), 0\}^2$. The Bellman regression targets for the critics are adapted to the CE2012 transitions:

$$V^R(z, B) = u(c^{safe}_\omega(z, B)) + \beta \sum_{z'} P_z(z, z')\mathcal{C}_{\tau_D}(z', B'_\omega(z, B)), \tag{41}$$

$$V^D(z) = u(h(z)) + \beta \sum_{z'} P_z(z, z')\left[\chi \mathcal{C}_{\tau_D}(z', 0) + (1-\chi)V^D_\psi(z')\right]. \tag{42}$$

Similarly, the pricing equation target utilizes the soft repayment probability to make the detached lender-pricing target vary smoothly with the actor's future choices:

$$q(z, B') = \frac{1}{1+r_f} \sum_{z'} P_z(z, z')p^R_{\tau_D}(z', B')\left[\lambda + (1-\lambda)\left(\text{coup} + q_\phi(z', B''_\omega(z', B'))\right)\right] \tag{43}$$

where $B''_\omega(z', B')$ denotes the borrowing choice of the target actor in the future state. Equations (41), (42), and (43) 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 using the repayment consumption and feasibility definitions above. The unified actor loss is therefore:

$$L_{actor} = -\mathbb{E}\left[u(c_\omega^{safe}(z,B)) + \beta \mathbb{E}_{z'|z} \mathcal{C}_{\tau_D}(z', B'_\omega(z,B))\right] + \lambda_{feas}\mathbb{E}[pen_\omega(z,B)] \tag{44}$$

Equation (44) applies the generic actor objective in Equation (15) to the CE2012 budget constraint in Equation (40). During an actor update, the price and value functions in this objective are the online networks. Since $V^D(z')$ does not depend on the debt choice B', gradient penetration has the model-specific form

$$\frac{\partial \mathcal{C}_{\tau_D}(z', B')}{\partial B'} = p^R_{\tau_D}(z', B') \frac{\partial V^R_\theta(z', B')}{\partial B'}, \tag{45}$$

$$\frac{\partial \mathcal{C}_H(z', B')}{\partial B'} = \mathbf{1}\{V^R_\theta(z', B') > V^D_\psi(z')\} \frac{\partial V^R_\theta(z', B')}{\partial B'} \tag{46}$$

away from ties. These expressions concern the next-period continuation gradient; the actor loss is not multiplied by a current-period repayment probability. During hard refinement, the actor remains free and the soft continuation in Equation (44) is replaced by the hard upper envelope:

$$\begin{aligned} L^H_{actor} = & -\mathbb{E}\big[u(c_\omega^{safe}(z,B)) + \beta E_{z'|z} \max\big\{V^R_\theta(z', B'_\omega(z,B)), V^D_\psi(z')\big\}\big] \\ & + \lambda_{feas}\mathbb{E}[pen_\omega(z,B)]. \end{aligned} \tag{47}$$

In Equation (47), the actor differentiates through the online price and value networks, following the selected next-period continuation branch of the hard maximum except at ties. The actor is trained as a repayment-contingent action at every sampled state; there is no current-period hard default indicator in its loss. By contrast, the hard critic and price-network regression targets are detached and use target-network copies (denoted by a superscript minus); the hard price target uses the next-period indicator $\mathbf{1}\{V^{R,-} \geq V^{D,-}\}$. Thus, the hard block continues to update the actor, critics, price network, and target networks; it is not merely an evaluation pass.

4.4. Algorithm

Algorithm 1 summarizes our solution procedure. The pseudocode distinguishes the two training stages—soft initial training and hard refinement—from the separate final hard evaluation.

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

Require: Model primitives, networks $\{B'_\omega, V^R_\theta, V^D_\psi, q_\phi\}$, soft budget $T_S = 10^6$, hard budget $T_H = 10^5$

Ensure: Hard-evaluation policies, prices, residuals, and simulated moments

Initialize online networks and target copies.
Fit the actor to $B' = (1-\lambda)B$ for 10,000 initialization updates at $\tau_D = 0.05$.
Hold the actor fixed and pretrain V^R, V^D, and q for 40,000 further updates.
Synchronize target networks, clear Adam states, and reset the main training key to seed 88.
for $t = 1, \ldots, T_S$ **do**
    Draw a state minibatch with boundary-focused observations
    Set $\tau_D \leftarrow \tau_D(t)$, annealed over the first 3×10^5 updates
    if $(t-1) \bmod 5000 < 1000$ **then**
        Update B'_ω through online q_ϕ and the online soft next-period continuation, with the feasibility penalty
    else
        Form detached target-network soft Bellman and lender-pricing targets
        Update V^R_θ, V^D_ψ, and q_ϕ
    end if
    Clip active gradients and Polyak-update target networks
end for
Replace the soft operators used in the actor and regression targets by their exact hard counterparts
Multiply all learning rates by 0.02
for $k = 1, \ldots, T_H$ **do**
    Draw a state minibatch with boundary-focused observations
    if $(k-1) \bmod 5000 < 1000$ **then**
        Update the free actor through online q_ϕ and the online hard next-period upper envelope
    else
        Form detached target-network hard Bellman and indicator-based pricing targets
        Update V^R_θ, V^D_ψ, and q_ϕ
    end if
    Clip active gradients and Polyak-update target networks
end for
Evaluate once under the exact repay/default rule at step $T_S + T_H$

4.5. StH-Q: tabular learning from simulated experience

StH-Q replaces the neural value functions and actor with a table of repayment action values, $Q^R(z, B, B')$, and a greedy debt choice. It also maintains the default value $V^D(z)$ and lender price $q(z, B')$. The upper-case Q^R is an action value; the lower-case q is a bond price. Simulated repayment transitions supply experience for updating the table. Randomly drawn states and a small number of model-based updates supplement this experience, so the implementation is a model-assisted, or Dyna-style, tabular method.

At a feasible state–action triple, the expected temporal-difference update is

$$Y^n(z, B, B') = u(z - \kappa B + q^n(z, B')[B' - (1-\lambda)B]) + \beta \sum_{z'} P_z(z, z') V_\tau^n(z', B'), \tag{48}$$

$$Q^{R,n+1}(z, B, B') = Q^{R,n}(z, B, B') + \alpha_Q \big[Y^n(z, B, B') - Q^{R,n}(z, B, B')\big], \tag{49}$$

$$V^{R,n}(z, B) = \max_{B' \in \mathcal{B}(z,B;q^n)} Q^{R,n}(z, B, B'), \tag{50}$$

$$g^n(z, B) = \operatorname*{arg\,max}_{B' \in \mathcal{B}(z,B;q^n)} Q^{R,n}(z, B, B'). \tag{51}$$

Here κ is the debt-service coefficient in Equation (20), and $\mathcal{B}$ excludes choices with nonpositive consumption. The envelope V_τ smooths the repayment/default maximum during initial training and becomes the exact maximum when $\tau = 0$. Default values and lender prices are updated from the corresponding model equations. Using P_z in the TD target integrates out next-period output risk; the state–action observations are still sampled. The method therefore uses the known transition law rather than estimating it from experience alone.

The experiment uses 200 output points and 350 current-debt and debt-action points. After a temporary soft initialization, it runs 26,000 iterations with simulated experience and 32 sampled-state planning updates per iteration. The first 1,000 iterations anneal the repayment/default temperature; the following 25,000 iterations use hard decisions. The 225 snapshots collected every 40 iterations over the final 9,000 main iterations are averaged and initialize 1,000 additional hard TD iterations at fixed price, without planning. The reported policy is extracted from the stored Q table, not reconstructed by a final full-action Bellman solve. Appendix E records the configuration.

Algorithm 2 Tabular StH-Q with simulated experience

```
Initialize Q^R, V^D, q from the model's temporary soft solution; use training seed 88.
for n = 1, ..., 26,000 do
    Simulate repayment, default, exclusion, and reentry with ϵ-greedy debt choices.
    Draw a minibatch combining valid repayment experience and random states.
    Update sampled Q entries using Equation (49); refresh their greedy policies.
    Update default values and lender prices; back up all actions at 32 sampled states.
    Use τ > 0 for the first 1,000 iterations and τ = 0 thereafter.
    Accumulate tables every 40 iterations after iteration 17,000.
end for
Average the accumulated tables and hold the averaged price fixed.
Perform 1,000 hard TD iterations without planning; export the stored-Q greedy policy.
Evaluate borrower and lender equations and simulate economic outcomes separately.
```

4.6. Hyperparameters and state sampling

Here, we describe further implementation details.

Computational environment.. StH-AC is implemented in JAX and the reported training runs use an RTX 5090. StH-Q is implemented in PyTorch and the reported runs use an RTX 3080. The local StH-AC one-step diagnostic uses JAX 0.11.1 on CPU. Table 1 states the hardware and timing scope for all four implementations.

Training dynamics.. We use Adam with batch size 256. Soft-phase learning rates are 3×10^{-4} for both critics and the price network and 3×10^{-5} for the actor. The hard phase multiplies these rates by 0.02. Gradients are clipped to maximum norm 10. Target networks use Polyak coefficients $\alpha = 0.0005$ in the soft phase and $\alpha = 0.001$ in the hard phase, under the convention $\vartheta^- \leftarrow (1-\alpha)\vartheta^- + \alpha\vartheta$. The feasibility penalty is $\lambda_{feas} = 10^4$ with consumption floor $\underline{c} = 10^{-6}$. General state batches allocate 25% of draws to small repay/default value gaps; 20% of price batches target states with next-period default probability between 0.05 and 0.95.

Temperature annealing.. The temperature $\tau_D(t)$ decays exponentially over $T_{ann} = 300,000$ iterations according to Equation (52):

$$\tau_D(t) = \tau_{D,init} \left(\frac{\tau_{D,final}}{\tau_{D,init}} \right)^{\min\{1, t/T_{ann}\}} \tag{52}$$

with $\tau_{D,init} = 0.05$. The reported specification fixes $\tau_{D,final} = 0.002$. The temperature remains at this value for the rest of the one-million-update soft stage. The following 100,000 updates constitute the second training stage and use exact hard continuation, choice, pricing, and actor objectives. The actor remains free throughout this block, with no initial freeze and no anchor penalty. The reported endpoint is step 1,100,000, after which evaluation and reporting are performed without further training.

Calibration, solution domain, and expectations. The exogenous endowment z follows an AR(1) process in logs:

$$\log z_t = (1 - \rho_z)\mu_z + \rho_z \log z_{t-1} + \varepsilon_t, \qquad \text{with } \varepsilon_t \sim \mathcal{N}(0, \sigma_z^2), \tag{53}$$

where the parameters are $\rho_z = 0.9485$, $\sigma_z = 0.027092$, and $\mu_z = 0$. The process is discretized into $N_z = 200$ points spanning ± 3 standard deviations, with levels defined as $z = \exp\{\log z\}$. All expectations over future z' are computed exactly via matrix-vector multiplication with the transition matrix P_z. Debt states B are sampled continuously from $[0, B_{max}]$, with $B_{max} = 1.05$, during both the soft-training and final hard-refinement stages.

4.7. Preliminary validation and staged implementation

Here, we present two exercises that validate the core mechanics of the StH method and demonstrate its progressive implementation.

4.7.1. One-period model validation

Before extending the soft-to-hard (StH) solver to the full infinite-horizon environment, in Appendix B.3 we evaluate its mechanics in a canonical one-period sovereign default model (Arellano, 2008). By removing multi-period pricing feedback, debt dilution, and long-term bond resale dynamics, this baseline setting isolates discrete-continuous boundary optimization from recursive dynamics, allowing for direct comparison against a high-precision grid-based value function iteration (VFI) benchmark. The one-period analysis yields two primary findings:

- *Benchmark Accuracy:* Under exact hard evaluation, the StH solver closely tracks the grid-based VFI reference across decision rules, value functions, and bond-pricing schedules.

- *Robustness via Gradient Penetration:* When the solver is subjected to severe initial parameter distortions—causing policy functions to temporarily degenerate into flat, unresponsive lines—the soft continuation operator maintains a continuous flow of non-zero gradients to both choice branches. This gradient penetration enables the neural networks to escape local stagnation and reliably recover the true hard default boundary.

4.7.2. Four-stage progression to the full solver

Having confirmed that the StH solver accurately resolves non-differentiable boundary optimization in isolation, we transition to the dynamic, multi-period setting. To systematically isolate each computational bottleneck in moving from standard grid-based solutions to the continuous-action StH solver, Appendix C documents a four-stage numerical progression:

- *Stage I (Neural Representation):* Verifies that neural networks have sufficient capacity to approximate standard CE2012 value and price objects while preserving the original discrete grid search and auxiliary shocks.

- *Stage II (Training-Layer Smoothing):* Shifts smoothing out of the model's economic structure and into the training operator via a grid-based soft Bellman update, while maintaining exact hard evaluation.

- *Stage III (Actor-Critic Bridge):* Introduces a continuous policy network (actor) trained against soft Bellman targets over the debt grid, stabilizing learning before removing the grid entirely.

- *Stage IV (Full Continuous StH Solver):* Eliminates the action grid entirely, using soft training operators for gradient penetration, hard refinement for tuning, and exact hard rules for evaluation.

Across all stages, the solver successfully reconstructs the core equilibrium objects, including price schedules, spread curves, and default regions. This step-by-step validation ensures the final continuous StH solution preserves the underlying economic mechanics while eliminating the distortions inherent in global grid search.

5. The multi-period sovereign default model: numerical solution

We next evaluate the StH framework within the canonical multi-period sovereign debt framework of Chatterjee and Eyigungor (2012), comparing the auxiliary-shock and taste-shock solvers with the grid-based StH-Q and continuous-action StH-AC implementations. All four produce broadly similar macroeconomic moments and qualitative implications, making each suitable for this specific application. However, the traditional smoothing methods introduce persistent numerical distortions that alter the model's precise predictions. While detectable under the baseline calibration ($\sigma_z = 0.027092$), these distortions amplify significantly under high volatility ($\sigma_z = 0.054184$). By pushing default risk deeper into nonlinear regions, this increased volatility acts as a stress test that exposes structural distortions along the decision boundary. StH-Q supplies a fast, high-accuracy benchmark for this low-dimensional grid problem, while StH-AC retains the neural representation needed for larger applications; both evaluate the original unsmoothed economy under hard decision rules. We therefore report high-volatility results in the main text and baseline results in Appendix D. Our comparative analysis yields three main conclusions:

i) Permanent smoothing introduces systematic structural distortions that remain contained under the baseline calibration but compound as economic volatility rises.

ii) Shrinking smoothing parameters in traditional solvers fails to safely approximate the unsmoothed limit, causing numerical instability, fixed-point non-convergence, and kinked equilibrium schedules.

iii) The StH framework achieves competitive computational accuracy relative to grid-based methods while successfully solving the exact, unsmoothed economy.

All four methods use the identical economic simulation seed, 20260830, and common primitive random inputs. StH-Q and StH-AC each use training seed 88; no policy, moment vector, or historical path is formed by averaging across trained solutions. The main Auxiliary and Taste simulations retain their native smoothing shocks. Moments use 256 independent paths of 5,000 quarters, discard 500 quarters as burn-in, and apply the same sample-selection and reentry rules to every method.

Table 1: Recorded computational times

Calibration	Method	Seconds	Device
Baseline	Auxiliary shocks	151.4	CPU, 8 MPI ranks
	Taste shocks	263.3	CPU, 16 threads
	StH-Q	200.0	RTX 3080
	StH-AC	1728.7	RTX 5090
High volatility	Auxiliary shocks	245.8	CPU, 8 MPI ranks
	Taste shocks	765.7	CPU, 16 threads
	StH-Q	198.7	RTX 3080
	StH-AC	1730.3	RTX 5090

Note: StH-Q includes soft initialization, main learning, and the fixed-price TD finish; StH-AC includes pretraining and main training. Large simulation evaluations are separate. Auxiliary- and taste-shock entries are archived native runs. Hardware and timing scopes differ, so the table records practical cost rather than a controlled hardware speedup.

StH-Q solves the two calibrations in approximately 199–200 seconds and remains substantially faster than the neural StH-AC implementation in this benchmark. StH-AC trains in approximately 1,729–1,730 seconds per calibration and preserves the economically relevant solution patterns. The two implementations therefore make the StH advantage explicit: StH-Q supplies exceptional grid-level accuracy and speed for small problems, while StH-AC supplies the neural representation required for scalable applications; neither needs permanent smoothing in the evaluated economy.

5.1. Permanent smoothing generates systematic structural distortions

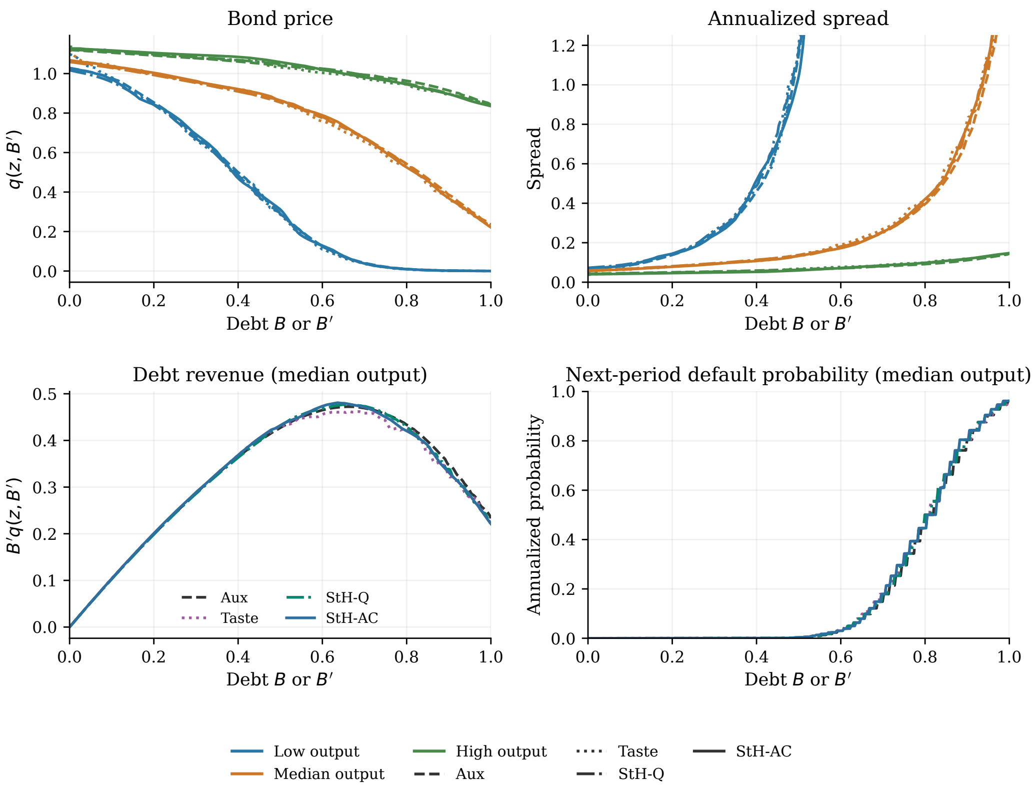
We demonstrate that permanent smoothing alters the target economy, systematically distorting equilibrium functions, business cycle statistics, and aggregate dynamics, although these quantitative discrepancies remain relatively modest under our calibrations.

5.1.1. Equilibrium functions

Figures 1 and 2 present the key equilibrium objects and debt policy functions generated by the four implementations.

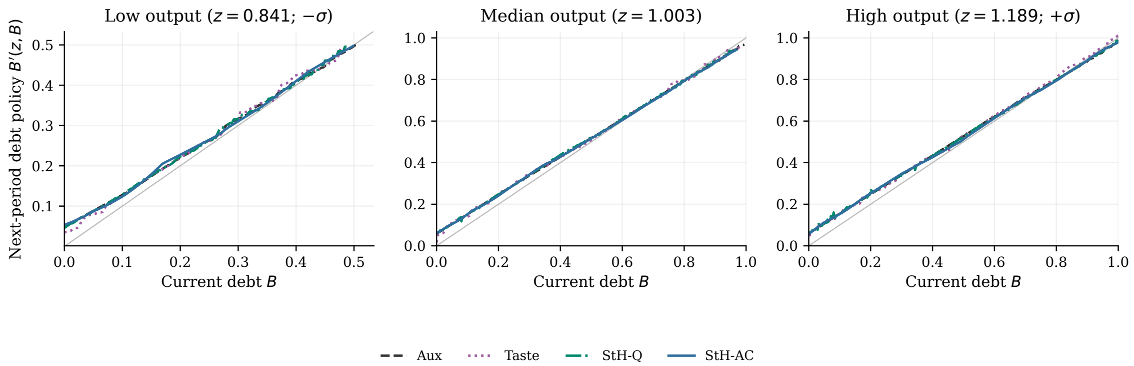
All four implementations reproduce the characteristic geometry of long-term debt models: bond prices fall and spreads rise with debt, and debt revenue follows the standard Laffer curve (Figure 1). The debt policy functions remain continuous within each repayment branch (Figure 2). Differences are concentrated near the price cliff and default boundary, precisely where permanent smoothing changes repayment probabilities. Both StH-Q and StH-AC recover the sharp hard-boundary structure without permanent shocks, showing that the result is a property of the StH construction rather than of one particular numerical representation.

Fig. 1: Equilibrium functions under high-volatility calibration. *Four-method comparison.*



Note: Compares bond-price schedules and annualized spreads at representative low, median, and high output states, together with debt revenue and annualized next-period default probabilities at median output, across the four implementations. In the upper panels, colors distinguish output states and line styles distinguish solution methods; in the lower panels, colors and line styles jointly identify methods. The low and high states are the grid points closest to one unconditional standard deviation of log output below and above the median. The default-probability panel annualizes the quarterly conditional probability p_q as $1 - (1 - p_q)^4$.

Fig. 2: Debt policy functions.



Note: Compares debt policy functions across the four implementations under the high-volatility calibration. Panel titles report the exact output grid point. The low and high panels use the grid points closest to one unconditional standard deviation of log output below $(-\sigma)$ and above $(+\sigma)$ the median, respectively. Each panel is shown over its state-specific repayment region, and the gray line denotes $B' = B$.

5.1.2. Simulated statistics

Business cycle moments are computed across 256 independent simulation paths of 5,000 quarters per method, dropping the initial 500 quarters as a burn-in sample. Table 2 summarizes the resulting business cycle statistics.

All four implementations generate the same qualitative implications, while the high-volatility calibration produces visible quantitative separation in the three core targets. Mean annual spreads range from 16.19% to 18.20%, spread standard deviations from 14.82% to 17.34%, and mean debt/output from 57.76% to 61.87%. StH-AC is lowest on all three, with StH-Q close in the same low-dimensional benchmark. The smoothed solutions retain their native artificial shocks, whereas both StH implementations use hard final decision rules.

5.1.3. Simulated path

Figure 3 traces the four annualized spread paths along the same 33-quarter output history.

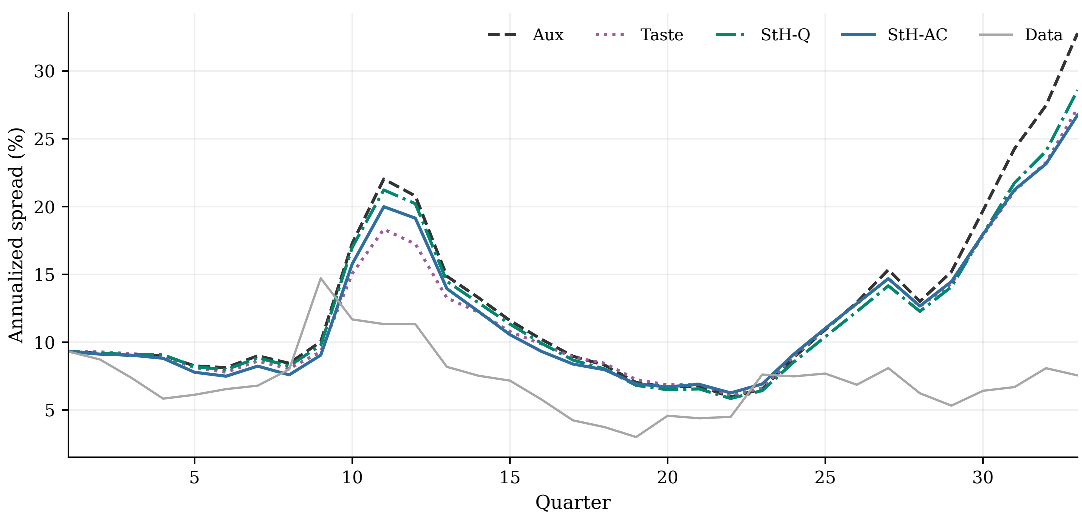
All four paths capture the same cyclical movements and the sharp rise in borrowing costs. The largest separation occurs at quarter 33, where the annualized spreads are 32.79% for auxiliary shocks, 27.26% for taste shocks, 28.59% for StH-Q, and 26.75% for StH-AC. The resulting range is 6.04 percentage points, or 22.58% relative to the lowest contemporaneous spread. The two StH implementations preserve the nonlinear response without artificial shocks in final simulation.

Table 2: Selected business cycle moments under high-volatility calibration.

Indicator	*Auxiliary shocks*	*Taste shocks*	*StH-Q*	*StH-AC*
First moments				
Mean spread	*0.17519*	*0.18198*	*0.16976*	*0.16186*
Std. spread	*0.16475*	*0.17344*	*0.16099*	*0.14815*
Mean debt/output	*0.61866*	*0.61677*	*0.60539*	*0.57763*
Second moments				
Default frequency	*0.12791*	*0.12912*	*0.12572*	*0.11369*
Average debt service	*0.04835*	*0.04803*	*0.04746*	*0.04527*
$\sigma(\log C)/\sigma(\log Y)$	*1.05096*	*1.04935*	*1.04801*	*1.04893*
$\sigma(NX/Y)/\sigma(\log Y)$	*0.10830*	*0.10879*	*0.10302*	*0.10218*
Corr$(\log C, \log Y)$	*0.99544*	*0.99531*	*0.99586*	*0.99599*
Corr$(\text{Spread}, \log Y)$	*-0.73055*	*-0.68833*	*-0.73201*	*-0.74110*
Corr$(NX/Y, \log Y)$	*-0.41765*	*-0.40069*	*-0.41677*	*-0.43027*

Note: Presents simulated macroeconomic business cycle statistics across the four implementations using the common simulation protocol. The first block contains the three calibration targets emphasized by Chatterjee and Eyigungor (2012); default frequency and the remaining statistics are reported in the second block.

Fig. 3: Simulated annualized spread trajectories under high-volatility calibration. *Four-method comparison.*



Note: Evaluates annualized spread dynamics across the four implementations under the high-volatility calibration using a shared exogenous output history. The gray Data series is an empirical visual reference; the largest four-method separation occurs at quarter 33.

5.1.4. Smoothing shocks in simulated moments and histories

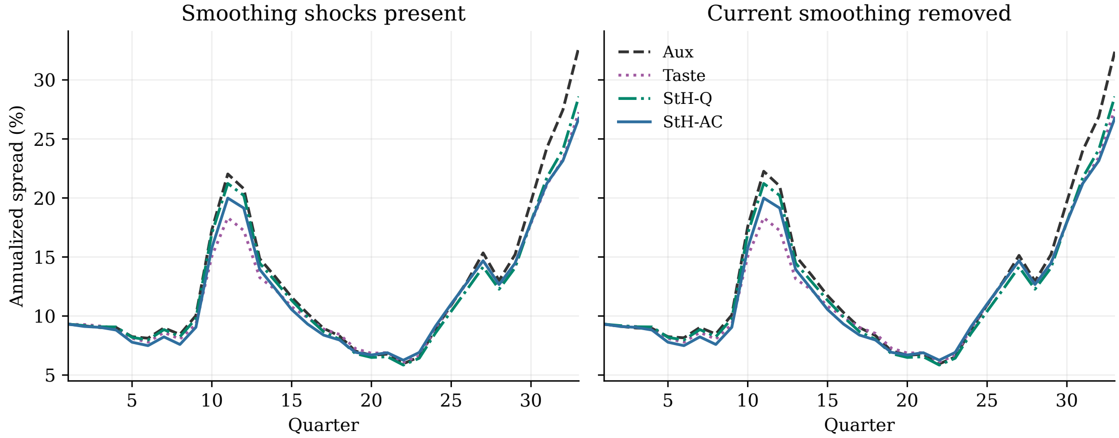
For the auxiliary-shock solution, “zero draw” sets the realized income shock to zero but retains the native default-entry income adjustment, whereas “OFF” removes both current smoothing components. For the taste-shock solution, “OFF” replaces both current taste-choice layers by their hard choices. Output risk, default costs, reentry, prices, and future continuation values remain unchanged. StH-Q and StH-AC contain no smoothing shocks in final evaluation and therefore require no alternative convention.

Table 3: High-volatility moments with smoothing shocks present and omitted

Moment	Aux ON	Aux zero	Aux OFF	Taste ON	Taste OFF	StH-Q	StH-AC
Mean annual spread	0.17519	0.17519	0.16681	0.18198	0.18195	0.16976	0.16186
Std. annual spread	0.16475	0.16433	0.14917	0.17344	0.17408	0.16099	0.14815
Mean debt/output	0.61866	0.61816	0.61790	0.61677	0.61563	0.60539	0.57763
Annual default frequency	0.12791	0.12758	0.13330	0.12912	0.12868	0.12572	0.11369
Mean debt service	0.04835	0.04831	0.04831	0.04803	0.04794	0.04746	0.04527
$\sigma(\log C)/\sigma(\log Y)$	1.05096	1.05034	1.05163	1.04935	1.04887	1.04801	1.04893
$\sigma(NX/Y)/\sigma(\log Y)$	0.10830	0.10732	0.10729	0.10879	0.10779	0.10302	0.10218
$\mathrm{corr}(\log C, \log Y)$	0.99544	0.99551	0.99559	0.99531	0.99540	0.99586	0.99599
$\mathrm{corr}(s, \log Y)$	-0.73055	-0.72973	-0.75299	-0.68833	-0.68676	-0.73201	-0.74110
$\mathrm{corr}(NX/Y, \log Y)$	-0.41765	-0.41674	-0.42967	-0.40069	-0.40078	-0.41677	-0.43027

Note: All columns use the same economic seed, sample-selection rules, and moment definitions as the main comparison. Within each smoothed method, the solved price and continuation functions are held fixed while only the current artificial-shock convention changes.

Fig. 4: High-volatility histories with smoothing shocks present and omitted



Note: Each panel reports one path for each of the four methods under the common output history. OFF applies the current-shock removal convention without rematching initial debt. StH-Q and StH-AC are unchanged because their final decision rules contain no smoothing shocks.

Removing both current auxiliary-shock components lowers the mean annual spread from 17.52% to 16.68% and its standard deviation from 16.48% to 14.92%, while annual default frequency rises from 12.79% to 13.33%. Merely setting the auxiliary draw to zero leaves the mean spread almost unchanged, showing that the default-entry adjustment is economically relevant. Taste ON and OFF differ by only 0.003 percentage points in mean spread. In the common history, ON/OFF spread RMSE is 0.166 percentage points for auxiliary shocks and 0.085 for taste shocks. The experiment shows that simulation treatment itself changes reported predictions; StH avoids this additional convention because its reported moments and histories are generated directly from hard unsmoothed rules.

5.2. Shrinking smoothing parameters fails to reach the unsmoothed limit

Because smoothing parameters are mere computational tools, it is natural to ask whether shrinking them toward zero recovers the exact, zero-shock economy. We find that it does not. Decreasing these parameters fails to provide a stable path to the hard-switch limit; instead, it causes severe numerical deterioration, non-convergence, and kinked equilibrium schedules that ultimately render the solver outputs unreliable.

5.2.1. Numerical deterioration and non-convergence

Smoothing is governed by the auxiliary income shock standard deviation (σ_{aux}) in the auxiliary-shock model, and by default (ρ_D) and borrowing (ρ_B) temperatures

Table 4: Algorithmic convergence diagnostics for smoothing methods under high-volatility calibration.

Method	σ_{aux}	ρ_D	ρ_B	K	terminal err_V	terminal err_q
Auxiliary shocks	0.012000	–	–	702	2.577×10^{-6}	9.974×10^{-6}
Auxiliary shocks	0.007500	–	–	3000	6.201×10^{-6}	1.258×10^{-4}
Auxiliary shocks	0.003000	–	–	3000	4.827×10^{-4}	8.771×10^{-3}
Auxiliary shocks	0.001500	–	–	3000	5.255×10^{-4}	1.163×10^{-2}
Auxiliary shocks	0.000750	–	–	3000	7.187×10^{-4}	1.480×10^{-2}
Taste shocks	–	0.002	4×10^{-5}	1000	2.156×10^{-5}	5.144×10^{-4}
Taste shocks	–	0.00125	2.5×10^{-5}	1000	2.413×10^{-3}	7.284×10^{-2}
Taste shocks	–	5×10^{-4}	10^{-5}	1000	3.720×10^{-3}	7.348×10^{-2}
Taste shocks	–	2.5×10^{-4}	5×10^{-6}	1000	2.926×10^{-3}	9.415×10^{-2}
Taste shocks	–	1.25×10^{-4}	2.5×10^{-6}	1000	1.277×10^{-3}	4.336×10^{-2}

Note: Reports preset iteration caps K and successive-iteration sup-norm movements err_V and err_q for the auxiliary ($K = 3000$) and taste ($K = 1000$) fixed-point solvers. StH-Q and StH-AC are omitted because their fixed-budget stochastic procedures do not generate comparable successive-grid convergence statistics; their hard-equation accuracy is reported in Table 6.

in the taste-shock model; reducing these parameters toward zero sharpens decision boundaries toward the exact hard limit. For these methods, we report iterations to convergence (K)—capped at 3000 and 1000 as is prescribed by their foundational papers, respectively—and the terminal sup-norm errors between successive iterations (err_V, err_q). Table 4 is deliberately limited to the two traditional smoothing methods because these successive-grid fixed-point metrics and iteration caps are defined comparably only for those solvers. StH-Q and StH-AC use fixed-budget procedures without permanent smoothing parameters; their accuracy on the hard equations is evaluated separately in Table 6.

Reducing smoothing parameters severely destabilizes fixed-point iteration. As σ_{aux} shrinks from 0.012 to 0.00075, four of the five auxiliary-shock specifications hit the $K = 3000$ iteration cap, with terminal pricing errors (err_q) rising to 1.48×10^{-2}. The taste-shock formulation proves even more fragile: as (ρ_D, ρ_B) shrinks to $(1.25 \times 10^{-4}, 2.5 \times 10^{-6})$, all tested pairs hit the $K = 1000$ cap, yielding terminal pricing errors up to 9.42×10^{-2} (a near 10% iteration-to-iteration gap). These breakdowns demonstrate that parameter tightening induces algorithmic failure rather than converging to the exact hard-switch boundary. Because StH-Q and StH-AC solve the hard-switch economy without a permanent smoothing parameter, their relevant test is closure of the original hard equations rather than convergence along this smoothing-parameter sequence; Table 6 reports that evidence.

5.2.2. Distorted equilibrium schedules

Crucially, the documented convergence failures extend beyond mere numerical artifacts to induce severe structural distortions in the underlying equilibrium functions. Figure 5 illustrates this by plotting the price and spread schedules under the high-volatility calibration for both the auxiliary-shock (top panel) and taste-shock (bottom panel) methods as their respective smoothing parameters are scaled down.

The resulting economic distortions are stark. In the top panel, reducing σ_{aux} forces the auxiliary-shock price and spread schedules to separate, disrupting the local parameter ordering and rendering it non-monotone near the default threshold. The taste-shock formulation in the bottom panel exhibits even more erratic behavior: as temperatures shrink, the computed schedules are corrupted by spurious oscillations and non-monotone kinks. Ultimately, the large terminal errors documented in Table 4 map directly onto severe, economically meaningful deformations in the target schedules.

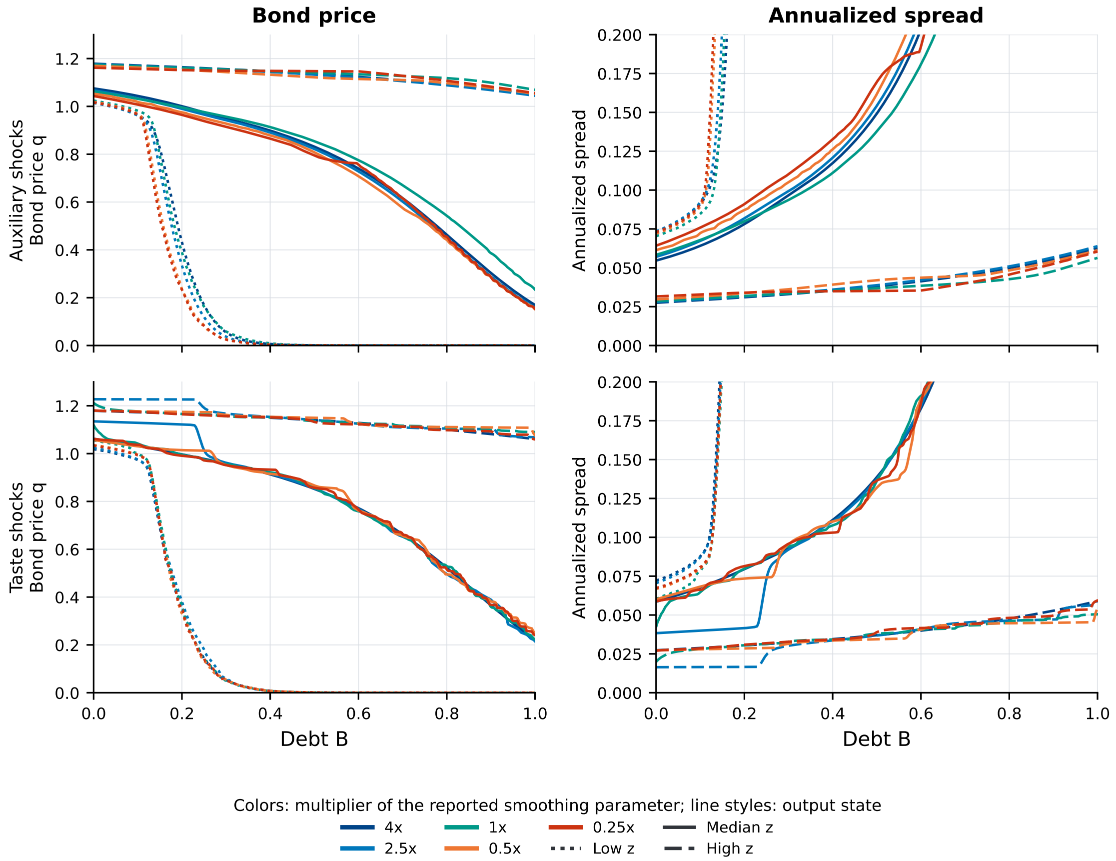
In contrast to permanent smoothing methods, the StH-AC solver evaluates the exact hard-switch economy and is completely free of ad-hoc smoothing parameters. However, because its training protocol relies on stochastic gradient optimization—drawing different random subsets of simulated states for each update—the final equilibrium functions naturally exhibit stochastic variation across initializations. Figure 6 visualizes the magnitude of this variance by plotting the distribution of the price and spread schedules across 16 independent random seeds.

The resulting schedules remain tightly clustered across the vast majority of the state space, with visible dispersion confined primarily to the steep price cliff at low output levels. This tight alignment confirms that the combination of continuous state sampling and neural network function approximation yields a robust solution.

5.2.3. Parameter sensitivity of economic outcomes

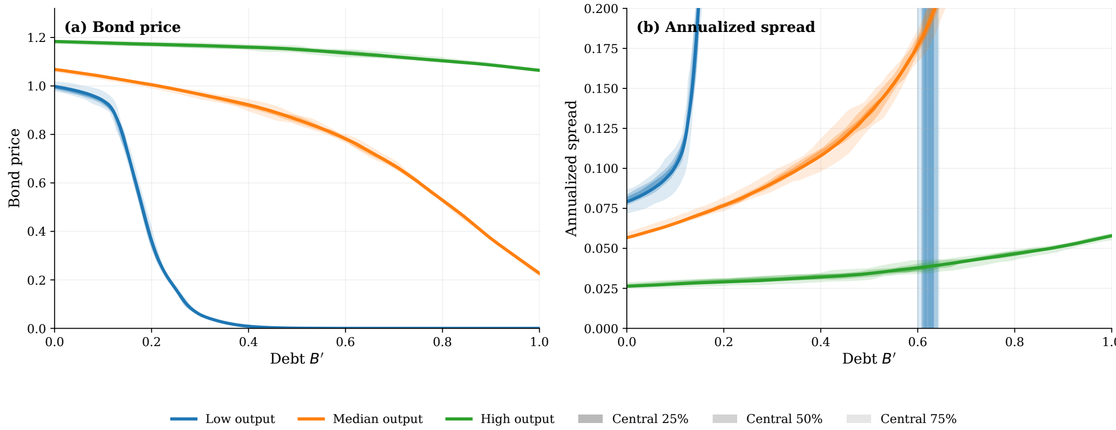
These numerical limitations and structural kinks naturally propagate into the simulated business cycle statistics. Table 5 tracks the core macroeconomic moments generated by these increasingly narrow smoothed boundaries. We observe that the simulated outcomes are highly unstable and non-monotone. In the auxiliary-shock economy, varying σ_{aux} shifts the mean debt-to-output ratio between 0.53452 and 0.61867—a substantial within-method range of 15.7% relative to the lower bound. The taste-shock formulation exhibits even greater sensitivity, with adjustments to (ρ_D, ρ_B) shifting the ratio between 0.50258 and 0.61677, representing a 22.7% spread. Mean spreads and default frequencies display similarly erratic responses. Combined with the preceding convergence and schedule evidence, these results reaffirm that tightening smoothing parameters fails to provide a stable path to the exact hard-

Fig. 5: Equilibrium-function sensitivity to smoothing parameters under high-volatility calibration.



Note: Reports sensitivity across auxiliary (top row) and taste (bottom row) shock specifications under high-volatility calibration using common parameter multipliers applied to $\sigma_{\text{aux}} = 0.003$ and $(\rho_D, \rho_B) = (5 \times 10^{-4}, 10^{-5})$, with spreads capped at 20% for clarity.

Fig. 6: Structural stability of StH-AC equilibrium objects across initialization seeds under high-volatility calibration.



Note: Displays bond price and annualized spread distributions across $N = 16$ independent training seeds under high-volatility calibration. Colors identify low, median, and high output states, where solid colored curves represent seed means and nested bands show the central 25%, 50%, and 75% of the seed distribution; the spread display is capped at 20%. *Each seed uses 50,000 pretraining updates followed by the 1,100,000-step main schedule.*

Table 5: Parameter sensitivity of business cycle moments under high-volatility calibration.

Method	σ_{aux} or ρ_D	ρ_B	Mean spread	Mean debt/output	Default frequency
Auxiliary shocks	0.012000	–	0.17254	0.56581	0.11863
Auxiliary shocks	0.007500	–	0.17434	0.56226	0.12031
Auxiliary shocks	0.003000	–	0.17515	0.61867	0.12794
Auxiliary shocks	0.001500	–	0.18812	0.59171	0.13456
Auxiliary shocks	0.000750	–	0.15001	0.53452	0.10880
Taste shocks	0.002	4×10^{-5}	0.16936	0.58553	0.11773
Taste shocks	0.00125	2.5×10^{-5}	0.17191	0.58066	0.12479
Taste shocks	5×10^{-4}	10^{-5}	0.18198	0.61677	0.12912
Taste shocks	2.5×10^{-4}	5×10^{-6}	0.13146	0.50258	0.08931
Taste shocks	1.25×10^{-4}	2.5×10^{-6}	0.16181	0.56902	0.11082
StH-Q (seed 88)	–	–	0.16976	0.60539	0.12572
StH-AC (seed 88)	–	–	0.16186	0.57763	0.11369

Note: Reports moment variation across smoothing-parameter multipliers for the auxiliary and taste-shock methods. The StH-Q and StH-AC rows report the fixed seed-88 hard-solution references used in the main four-method comparison; they contain no permanent smoothing parameters.

switch economy. Instead, it induces severe numerical instability, characterized by erratic oscillations and non-monotonic discontinuities in the equilibrium functions.

These findings carry significant economic implications. Structural modelers often introduce ad-hoc smoothing parameters purely for computational tractability, despite their lack of theoretical foundation. While sensitivity to these parameters is technically a within-solver metric—since auxiliary and taste shocks define fundamentally different target economies—the quantitative impact of these arbitrary choices is severe. Modifying these non-structural parameters alters the predicted debt-to-output ratio by a spread of 15.7% to 22.7%, depending on the specific smoothing method. For quantitative policy evaluation, a prediction variance of this magnitude driven entirely by numerical artifacts undermines the reliability of the model.

All main four-method comparisons use one reported StH-AC solution (training seed 88), and the simulated time series use identical innovation seeds across methods. Conditional on a fixed solved policy, Monte Carlo sampling error can be reduced by increasing the number of simulated paths or replications. The 16-seed exercise is different: each seed retrains the StH-AC networks and therefore produces a distinct numerical approximation. Under high volatility, mean debt/output across these independently trained solutions is 0.56985 with a sample standard deviation of 0.02677, while mean annual spread is 0.15916 with a sample standard deviation of 0.01575. The debt/output coefficient of variation is 4.70%. This cross-training-seed dispersion is not Monte Carlo error from repeated simulation of one fixed solution, so the usual $N^{-1/2}$ sampling argument does not justify averaging the resulting policies or histories. Such an average would define a different object that need not satisfy the equilibrium conditions. We therefore use the 16-seed distribution only as evidence on training stability, not as a substitute equilibrium.

5.3. StH solves the unsmoothed economy with competitive accuracy

We assess value, policy, pricing, and boundary closure with a componentwise hard-operator audit. Every completed solution is mapped to a common 200×350 grid for ex post evaluation. The shared 200-state approximation for exogenous income enters expectations, while StH samples endogenous debt states in minibatches, represents value and price functions with neural networks, and chooses next-period debt through a continuous actor. The audit evaluates numerical adequacy at the hard endpoint. Because the endogenous debt state is sampled rather than gridded during training, adding continuous states does not require constructing a full endogenous tensor grid.

The common audit maps each completed method to the state grid $\mathcal{G} = \mathcal{G}_z \times \mathcal{G}_B$ with 200 output points and 350 debt points. For method m, the mapped objects are branch values (V_m^R, V_m^D), total value $V_m = \max\{V_m^R, V_m^D\}$, price q_m, a repayment-conditional policy A_m, and a binary rule δ_m. The hard repayment return for candidate B' is

$$Q_m^H((z,B),B') = u\big(c^R(z,B,B';q_m)\big) + \beta \sum_{z'} P_z(z,z')V_m(z',B'). \tag{54}$$

The operator $\mathcal{T}_R^H$ maximizes this return over the common 350-point action grid. The default operator applies the no-shock exclusion recursion, including reentry at zero debt,

$$(\mathcal{T}_D^H V_m)(z) = u(h(z)) + \beta \sum_{z'} P_z(z,z') \left[\chi V_m(z',0) + (1-\chi)V_m^D(z')\right]. \tag{55}$$

The audited hard choice is $\widehat{\delta}_m(z,B) = \mathbf{1}\{(\mathcal{T}_D^H V_m)(z) > (\mathcal{T}_R^H V_m)(z,B)\}$, and the audited repayment set is $\mathcal{G}_m^R = \{(z,B) : \widehat{\delta}_m(z,B) = 0\}$.

The hard lender operator uses each method's mapped decision and policy:

$$\begin{aligned}(\mathcal{T}_q^H q_m)(z,B') = \frac{1}{1+r_f} \sum_{z'} P_z(z,z')[1-\delta_m(z',B')] \\ \times \left[\lambda + (1-\lambda)\{\text{coup} + q_m(z', A_m(z',B'))\}\right].\end{aligned} \tag{56}$$

The four reported statistics are then

$$R_m^V = |\mathcal{G}|^{-1} \sum_{s\in\mathcal{G}} \left|V_m(s) - \max\{(\mathcal{T}_R^H V_m)(s), (\mathcal{T}_D^H V_m)(s)\}\right|, \tag{57}$$

$$R_m^\pi = |\mathcal{G}_m^R|^{-1} \sum_{s\in\mathcal{G}_m^R} \left[\max_{B'\in\mathcal{G}_B} Q_m^H(s,B') - Q_m^H(s,A_m(s))\right], \tag{58}$$

$$R_m^q = |\mathcal{G}|^{-1} \sum_{s\in\mathcal{G}} |q_m(s) - (\mathcal{T}_q^H q_m)(s)|, \tag{59}$$

$$D_m = 100|\mathcal{G}|^{-1} \sum_{s\in\mathcal{G}} \mathbf{1}\{\delta_m(s) \neq \widehat{\delta}_m(s)\}. \tag{60}$$

Table 6 reports the componentwise closure diagnostics used to assess whether the StH hard endpoint is numerically adequate while directly solving the original unsmoothed economy:

The common-grid audit reveals complementary strengths of the two StH implementations. StH-Q attains a Bellman MAE of 3.04×10^{-6} and a repayment-policy gap of 7.97×10^{-6} under high volatility, while also producing the smallest pricing

Table 6: Hard-operator accuracy diagnostics under high-volatility calibration.

Method	Bellman MAE	Repay policy-gap MAE	Pricing MAE	Default-rule inconsistency (%)
Auxiliary shocks	1.981×10^{-4}	8.435×10^{-6}	8.967×10^{-4}	0.39000
Taste shocks	3.682×10^{-4}	4.806×10^{-5}	2.365×10^{-3}	0.01286
StH-Q	3.040×10^{-6}	7.968×10^{-6}	7.245×10^{-4}	0.00000
StH-AC	1.502×10^{-3}	4.816×10^{-5}	7.716×10^{-4}	0.15571

Note: All methods are audited on a common state grid comprising 200 output points and 350 debt points. The four reported diagnostics are the Bellman MAE, repay-policy-gap MAE, pricing MAE, and default-rule inconsistency.

MAE and zero default-rule inconsistency. StH-AC uses continuous neural approximators and records mean Bellman, policy-gap, and pricing errors of 1.50×10^{-3}, 4.82×10^{-5}, and 7.72×10^{-4}. The tabular result provides a precise small-grid check; the actor–critic result retains a representation that does not require a full state–action table.

Computational economics has long operated under an accepted trade-off between local numerical precision and dimensional scalability. As Judd (1992) established, dense spatial grids and collocation techniques routinely achieve maximum absolute Euler residuals falling in the 10^{-6} to 10^{-9} range. In contrast, modern deep learning frameworks relying on stochastic simulation generally accept much lower precision, often yielding residuals in the 10^{-2} to 10^{-3} range (Maliar and Maliar, 2022). This stark gap has reinforced the conventional wisdom that neural networks are strictly high-dimensional fallbacks—useful when grid methods suffer from the curse of dimensionality, but inherently incapable of capturing local precision near sharp transition boundaries.

The two implementations resolve different sides of this trade-off. StH-Q shows that annealing to the hard operator can deliver high numerical precision in a small problem: it attains very small Bellman and policy-gap errors on the common grid and exactly aligns the audited default rule. This is the paper's precision result.

StH-AC supplies the scalability result. Its actor and equilibrium objects are neural functions of continuous states and actions, so the solver neither stores the full state–action table nor restricts policy optimization to a finite debt-action grid. The high-volatility benchmark delivers hard-equation errors of the expected neural-solver order, while retaining the architecture needed as additional continuous state variables are introduced. Thus the evidence supports a complementary division: StH-Q is the high-precision implementation for small problems, and StH-AC is the continuous function-approximation implementation for larger, higher-dimensional applications. Both solve the original unsmoothed economy.

6. Conclusion

Permanent structural smoothing has long served as a highly effective computational tool in quantitative macroeconomics. In our sovereign default application, traditional grid-based smoothing methods perform admirably, producing stable aggregate statistics and qualitative policy implications that align closely with the unsmoothed economy. However, they fundamentally remain approximations. As we demonstrate, attempts to recover the exact decision boundaries by arbitrarily shrinking the smoothing parameters ultimately fail, breaking solver convergence and inducing severe numerical distortions. The StH deep reinforcement learning framework circumvents this tradeoff. By isolating numerical regularization strictly to the learning phase, StH successfully solves the exact, unsmoothed economy without sacrificing computational stability.

Although applied here exclusively to sovereign default, the StH method operates natively on general discrete-continuous choice operators without requiring problem-specific derivations. A promising StH frontier is scaling to heterogeneous-agent general equilibrium models. In these settings, discrete non-convexities—such as lumpy investment, firm entry-exit, or household bankruptcy—have historically required rigid grids or heavy smoothing. By providing a domain-agnostic, highly scalable architecture, StH enables the analysis of high-dimensional environments without sacrificing economic fidelity.

Finally, evaluating exact, unsmoothed decision rules can also matter for structural estimation. When forward solution algorithms embed permanent distortions, estimates obtained with the Simulated Method of Moments may partly absorb these computational artifacts, potentially shifting parameters such as risk aversion when tail risk is artificially dampened. By removing permanent smoothing from the forward solution, StH reduces this source of computational contamination and allows estimated parameters to be interpreted with less concern that they compensate for a numerical workaround.

StH-Q and StH-AC implement the same soft-to-hard idea with tabular and neural representations. In this application, tabular learning takes approximately 3.3 minutes per calibration and yields small mean borrower Bellman residuals conditional on price, with nonzero pricing residuals. StH-AC takes approximately half an hour of training and has mean Bellman residuals of order 10^{-3}. Both implementations evaluate the original unsmoothed equations and use hard final decisions.

Aux, Taste, and StH produce closely grouped predictions in the baseline calibration, providing a consistency check across approaches. Under high volatility, differences in debt, default, and spread histories become more pronounced. The smoothing-shock comparison shows an additional source of variation: a smoothed solution's predictions depend on how its artificial shocks are treated during simulation. The effect is small for some statistics and larger for others. StH avoids that convention, although numerical approximation and training sensitivity remain. Extending the neural implementation to larger state spaces is a natural next application.

Appendix A. Proofs of Propositions 1 and 2

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 $u = \exp(V^1(s)/\tau) + \exp(V^0(s)/\tau)$:

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

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 $\exp(V^1(s)/\tau)$:

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

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

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

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 $0 < p^1_\tau(s) < 1$. The derivative weight for the losing branch is positive but can be arbitrarily close to zero when the value gap is large. The size of the optimizer step and convergence of the coupled neural system depend on additional features of the training procedure. ■

A.2. Proof of proposition 2 (hard-operator limit)

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 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 at every state except the 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. ■

Appendix B. Solving the one-period sovereign default model

As an initial test exercise, we evaluate the StH framework in a canonical one-period sovereign default model (Arellano, 2008). This model features a hard default boundary without long-term bond resale dynamics or debt dilution, thereby isolating boundary optimization from multi-period pricing feedback.

B.1. The one-period model

The one-period model is formulated over a net-asset domain $a \in [-0.20, 0.40]$, where negative values denote debt. The endowment process y is i.i.d., and continuation expectations are evaluated using an 11-node Gauss–Hermite quadrature rule. The repayment budget constraint is given by

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

where a' denotes next-period net assets and $q(a')$ is the bond price schedule.

Let β_b be the borrower discount factor and β_c the lender discount factor. The repayment value $V^R(y, a)$ and total value $V(y, a)$ satisfy

$$V^R(y, a) = \max_{a' \in [-0.20, 0.40]} \left\{ u(c^R(y, a, a')) + \beta_b E_{y'} V(y', a') \right\}, \tag{B.2}$$

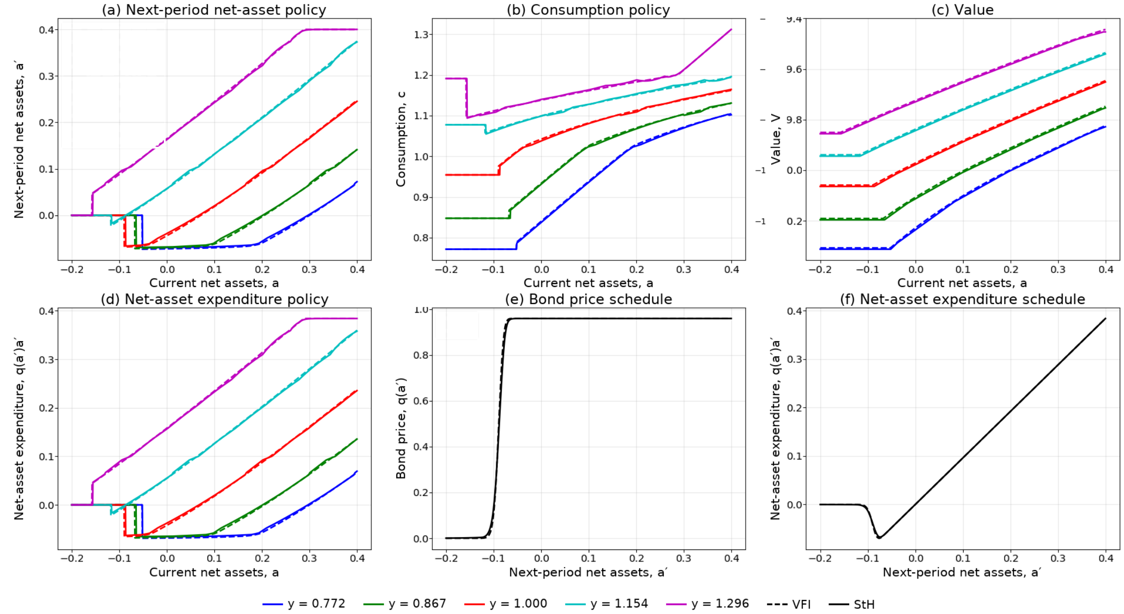
$$V(y, a) = \max\{V^R(y, a), V^D(y)\}. \tag{B.3}$$

Default resets assets to zero and imposes a one-period output cost governed by $h(y) = y_{min} + \lambda_D(y - y_{min})$, with $\lambda_D = 0.8$ and no market reentry. Normal autarky output resumes in subsequent periods:

$$W(y) = u(y) + \beta_b E_{y'} W(y'), \tag{B.4}$$

$$V^D(y) = u(h(y)) + \beta_b E_{y'} W(y'). \tag{B.5}$$

Fig. B.1: One-period sovereign default model solution.



Note: Illustrates six key equilibrium and policy objects in net-asset notation, contrasting grid-based Value Function Iteration (VFI) against the StH solution evaluated via exact 11-node quadrature.

The one-period bond price schedule is determined by risk-neutral lenders:

$$q(a') = \beta_c \Pr\big\{V^R(y', a') \geq V^D(y')\big\}\,. \tag{B.6}$$

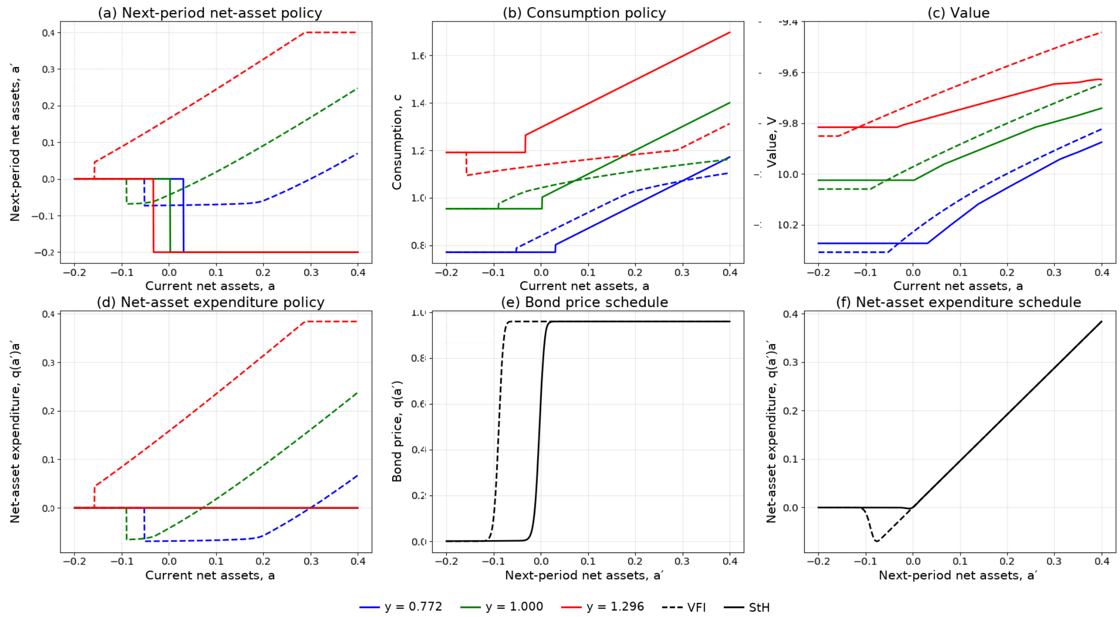
The grid-based VFI comparison evaluates this probability with the normal CDF at the interpolated default cutoff. StH uses the same economic problem and an exact 11-node quadrature rule during evaluation. The common parameters are $\beta_b = 0.90$, $\beta_c = 0.96$, CRRA coefficient 2, and log-endowment standard deviation 0.05.

B.2. Numerical solution for the one-period model

Figure B.1 reports the standard one-period solution. Solid curves represent StH hard-evaluation outcomes, and dashed curves correspond to an accurate grid-based VFI reference. The comparison is expressed throughout in net-asset notation.

The figure provides a direct numerical check of the StH solver. In B.3, we examine the robustness of the StH solution to poor initialization: we show that the StH solver can escape a temporary stagnation phase and accurately recover the hard default boundary even when initialized far from the true solution.

Fig. B.2: Stagnation phase under poor initialization in the one-period model.



Note: Demonstrates the temporary stagnation phase in policy and value functions when training begins from an adverse parameter initialization.

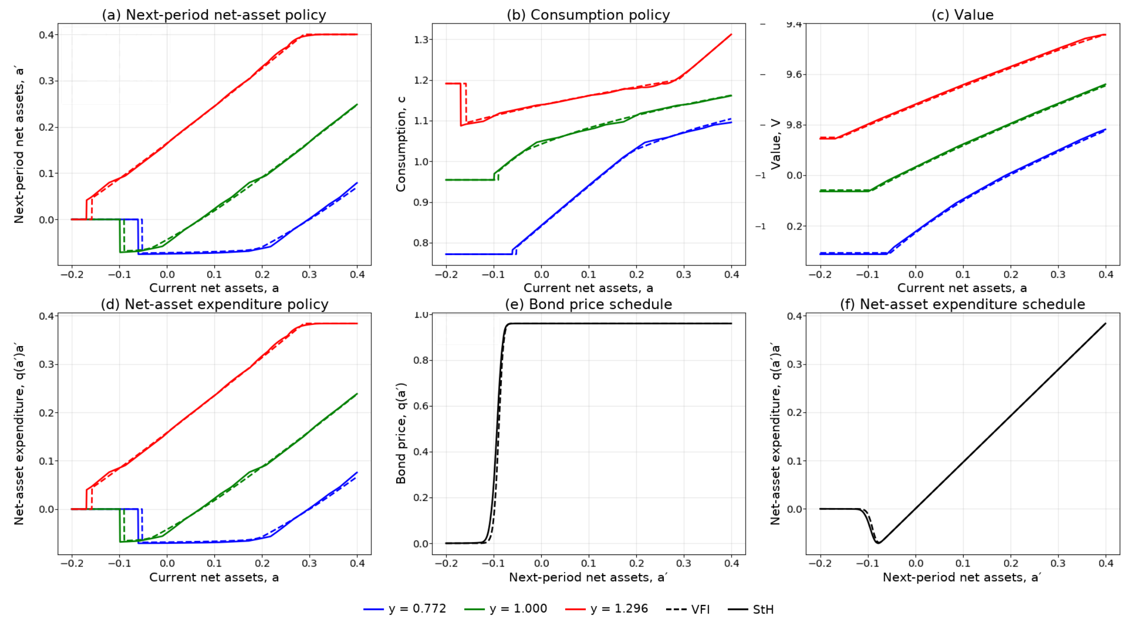
B.3. Robustness to poor initialization in the one-period model

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 benchmark VFI region. We observe that under poor initial conditions, the solver first enters a distinct stagnation phase. As illustrated in Figure B.2, the asset policy initially degenerates into a flat line unresponsive to the state variable, and the value and consumption functions deviate visibly from the VFI reference.

However, because the soft continuation and soft repayment probabilities continuously allocate nonzero gradients to both branches, the network successfully escapes this stagnation. Figure B.3 demonstrates the recovery stage: persistent gradient updates guide the neural networks back toward the benchmark VFI solution.

We conclude that the StH solver can accurately recover the hard default boundary under standard initializations and that its gradient-penetration mechanism also provides numerical robustness against severe initial parameter distortions in this exercise.

Fig. B.3: Algorithmic recovery and convergence from poor initialization.



Note: Highlights the dynamic recovery and ultimate convergence of policy and value functions following a transient stagnation phase in the one-period model.

Appendix C. Four-stage transition to the continuous-action StH solver

This appendix documents the staged numerical progression used in the long-term sovereign default application. Each stage isolates a single computational bottleneck before reaching the final StH solver. Stage I evaluates whether neural networks can accurately represent the auxiliary-shock objects from the CE2012 replication exercises. Stage II shifts smoothing from the model's decision layer into the training operator. Stage III introduces an actor-based policy representation while retaining the soft Bellman bridge. Stage IV implements the full StH solver evaluated in hard mode.

C.1. Stage I: Neuralizing the original CE2012 operator

Stage I preserves the original CE2012 structure as much as possible: debt is chosen on a grid, the original default operator is left intact, and the auxiliary-shock structure remains in place. The only essential change is replacing tabulated value and price objects with neural network approximations. The purpose of this stage is representational rather than methodological: it verifies whether neural networks can approximate the core CE2012 objects before altering the solution architecture.

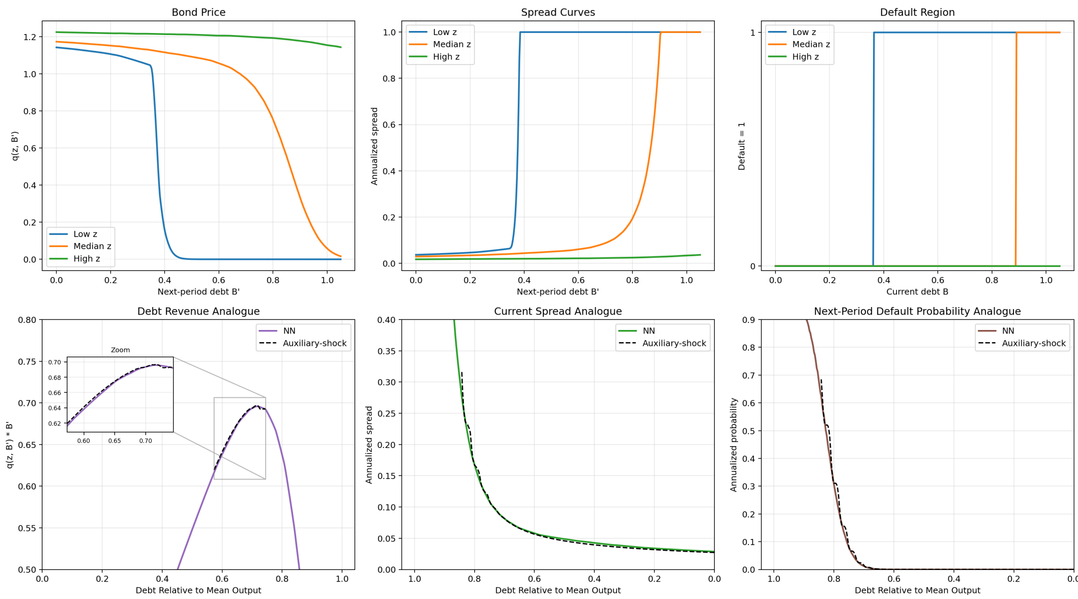
Figure C.1 reports the evidence for this stage. The figure shows the price schedule, spread curve, default region, and diagnostic objects reconstructed from the CE2012 replication files and from the neural-network analogue. The close alignment confirms that the neural networks possess sufficient representational capacity for the long-term default objects. The limitation of Stage I is also clear: the solver still inherits the global discrete action search and the original CE2012 operator.

C.2. Stage II: Moving smoothing into the training layer

Stage II redefines the role of smoothing. Instead of embedding smoothing as a permanent feature of the solved economy, this stage places smooth operators inside the training problem while evaluation remains tied to the hard repay/default decision. We refer to this step as Soft Q-Learning (SQL). In this paper, SQL denotes a grid-based soft Bellman update: the maximization over next-period debt is replaced by a log-sum-exp operator over the debt grid, and the repay/default switch is also smoothed during backpropagation. This stage does not yet introduce a continuous actor.

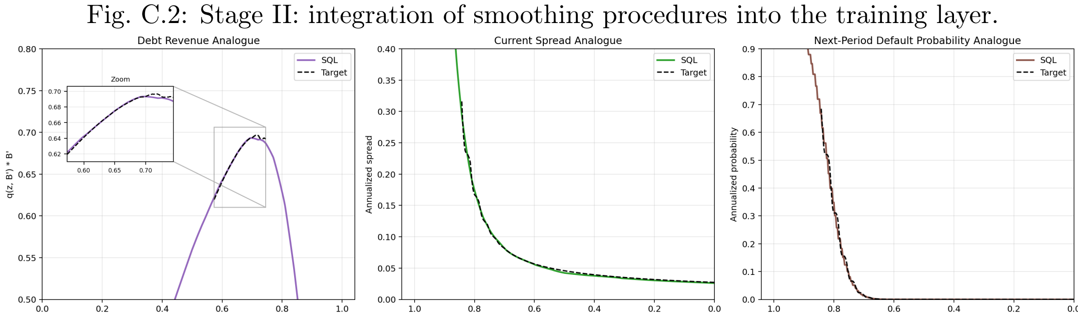
As demonstrated in Figure C.2, the debt-revenue, annualized-spread, and annualized next-period-default-probability schedules remain closely aligned with the corresponding CE2012 auxiliary-shock objects. This confirms that soft training paired with hard evaluation is numerically viable. However, because actions remain discrete

Fig. C.1: Stage I: neural network implementation of the CE2012 operator.



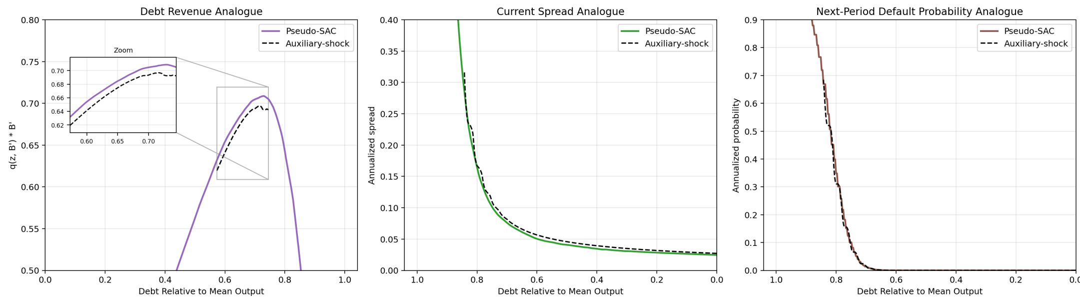
Note: Reconstructs price schedules, spread curves, default boundaries, and diagnostic metrics from the original CE2012 replication code into a neural architecture.

Fig. C.2: Stage II: integration of smoothing procedures into the training layer.



Note: Demonstrates close quantitative alignment between primary neural equilibrium objects and corresponding auxiliary-shock benchmarks.

Fig. C.3: Stage III: actor-based policy representation.



Note: Compares the debt-revenue, annualized-spread, and annualized next-period-default-probability schedules produced by the actor-based Stage III bridge with the corresponding auxiliary-shock references.

and the policy still relies on a predefined grid, the primary action-space bottleneck has not yet been resolved.

C.3. Stage III: Actor-based policy representation

Stage III introduces a policy network that begins to take over the action-choice step. This stage serves as an actor-based bridge, rather than the final StH solver. The actor is introduced while the training target is still induced by the soft Bellman structure used in Stage II. This intermediate organization matters because replacing the soft Bellman bridge too early with a hard discrete choice makes the actor's learning dynamics unstable in the long-term default problem.

As illustrated in Figure C.3, the primary solution objects remain tightly aligned with the corresponding CE2012 auxiliary-shock objects. This stage acts as an actor–critic bridge because the actor is trained against critic-induced soft Bellman targets over the debt grid. Standard SAC uses a stochastic actor, entropy regularization, and off-policy exploration; this bridge instead uses a policy network to stabilize the transition from grid-based value search to continuous policy representation.

C.4. Stage IV: StH solver under hard evaluation

Stage IV marks the endpoint of the staged route. The StH solver combines continuous actions, the CE2012 long-term default environment, soft training operators, and strict hard evaluation. At this point, global action-grid search is removed. Auxiliary shocks and permanent taste-shock smoothing are not part of the evaluated economy. Soft repayment probabilities are used only inside the training operator to provide gradient penetration, and the reported economic objects are generated under the hard-switch rule. The reported run uses one million soft-phase updates followed by 100,000 free-actor hard-refinement updates; the final block updates the actor, critics, and price network with hard continuation, choice, and pricing operators. The resulting hard StH solution is reported in Figure D.1, where its economic objects are shown together with the auxiliary- and taste-shock solutions under the baseline CE2012 calibration.

Appendix D. The multi-period sovereign default model: CE2012 calibration

While Section 5 examines a more demanding stress test under high economic volatility, this appendix provides a complementary baseline analysis using the original calibration from Chatterjee and Eyigungor (2012) ($\sigma_z = 0.027092$).

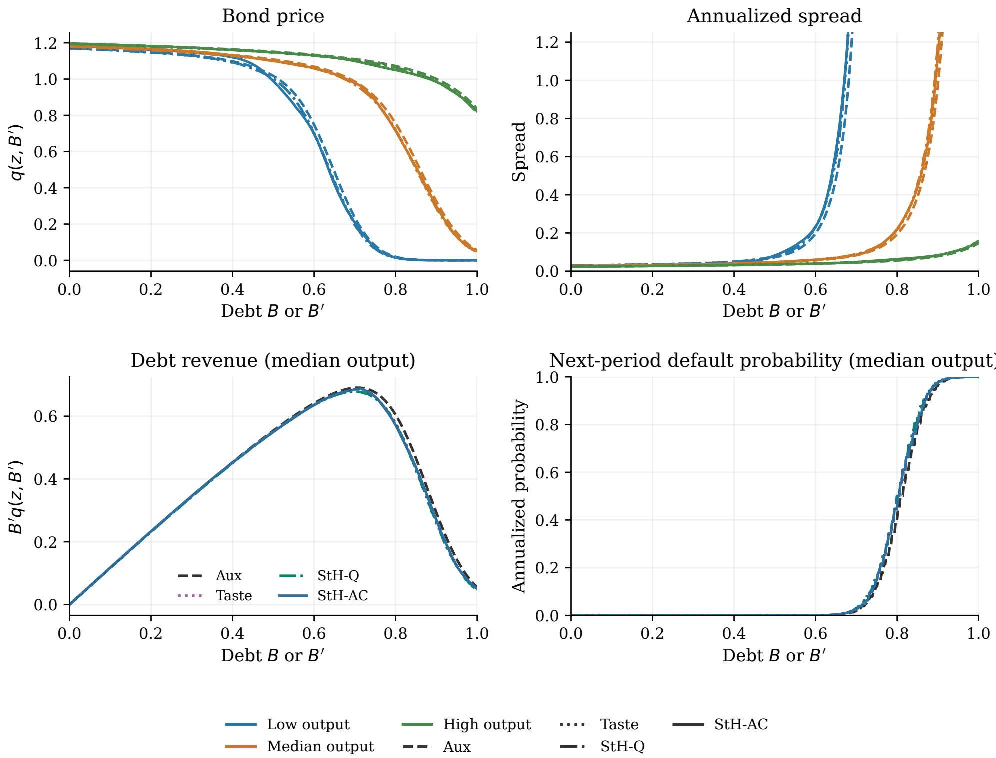
D.1. Baseline CE2012 solution

We compute and evaluate all solutions in this appendix using the identical procedures applied to the high-volatility analysis in the main text.

D.1.1. Equilibrium functions

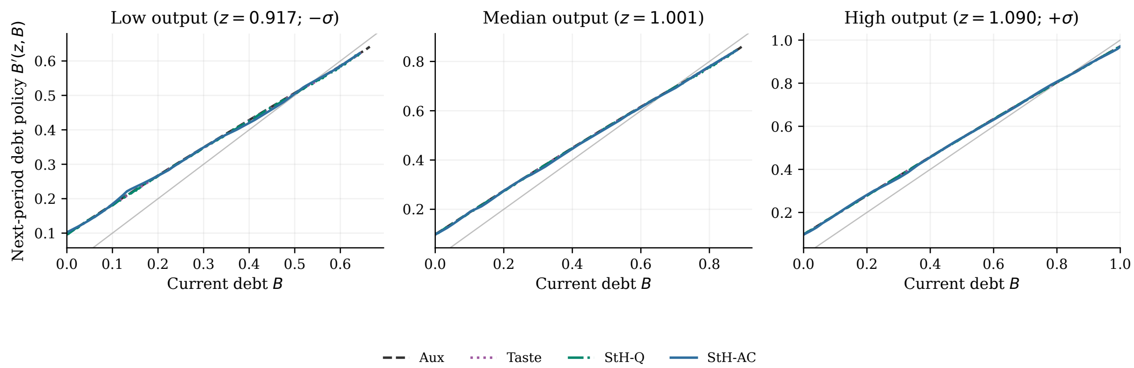
Figures D.1 and D.2 present the key equilibrium objects and debt policy functions for Auxiliary shocks, Taste shocks, StH-Q, and StH-AC under the baseline calibration. All four produce closely aligned equilibrium geometry, with the remaining differences concentrated near the price cliff and default threshold.

Fig. D.1: Equilibrium functions under baseline calibration *(four-method comparison)*.



Note: Compares bond-price schedules and annualized spreads at representative low, median, and high output states, together with debt revenue and annualized next-period default probabilities at median output, across the four implementations. In the upper panels, colors distinguish output states and line styles distinguish solution methods; in the lower panels, colors and line styles jointly identify methods. The low and high states are the grid points closest to one unconditional standard deviation of log output below and above the median. The default-probability panel annualizes the quarterly conditional probability p_q as $1 - (1 - p_q)^4$.

Fig. D.2: Debt policy functions.



Note: Compares debt policy functions across the four implementations under the baseline calibration. Panel titles report the exact output grid point. The low and high panels use the grid points closest to one unconditional standard deviation of log output below ($-\sigma$) and above ($+\sigma$) the median, respectively. Each panel is shown over its state-specific repayment region, and the gray line denotes $B' = B$.

All four implementations produce broadly similar equilibrium geometry in Figure D.1, while Figure D.2 shows that their debt policy functions are also closely aligned. Their differences are concentrated near the price cliff and default threshold and are substantially smaller than under the high-volatility calibration.

D.1.2. Simulated statistics

Table D.1 summarizes the business cycle statistics.

Under the baseline calibration, the four implementations remain closely grouped in the three core targets. Mean annual spreads range from 8.15% to 8.57%, spread standard deviations from 4.45% to 4.75%, and mean debt-to-output ratios from 69.17% to 69.99%. This agreement across two smoothed solvers and two hard StH solvers provides a consistency check in the standard CE2012 environment.

D.1.3. Simulated path

Figure D.3 plots the baseline spread paths.

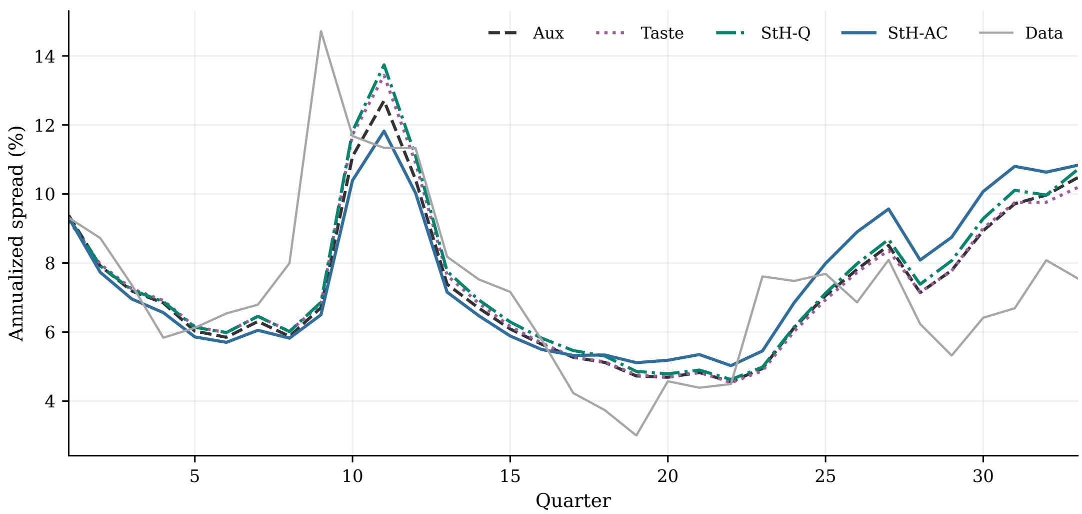
The four trajectories display the same cyclical pattern and remain closely grouped throughout the baseline history. Terminal annualized spreads are 10.48% for auxiliary shocks, 10.19% for taste shocks, 10.71% for StH-Q, and 10.83% for StH-AC. The largest four-method separation is 1.92 percentage points, reinforcing the close baseline agreement visible in the equilibrium functions and moments.

Table D.1: Selected business cycle moments under baseline calibration.

Indicator	*Auxiliary shocks*	*Taste shocks*	*StH-Q*	*StH-AC*
First moments				
Mean spread	*0.08151*	*0.08336*	*0.08356*	*0.08573*
Std. spread	*0.04452*	*0.04600*	*0.04611*	*0.04750*
Mean debt/output	*0.69988*	*0.69327*	*0.69177*	*0.69484*
Second moments				
Default frequency	*0.06888*	*0.07049*	*0.06903*	*0.07329*
Average debt service	*0.05484*	*0.05431*	*0.05419*	*0.05440*
$\sigma(\log C)/\sigma(\log Y)$	*1.10676*	*1.10622*	*1.10529*	*1.10305*
$\sigma(NX/Y)/\sigma(\log Y)$	*0.20345*	*0.20208*	*0.20082*	*0.20329*
Corr$(\log C, \log Y)$	*0.98563*	*0.98584*	*0.98598*	*0.98527*
Corr$(\text{Spread}, \log Y)$	*-0.65002*	*-0.65066*	*-0.64251*	*-0.61200*
Corr$(NX/Y, \log Y)$	*-0.43674*	*-0.43833*	*-0.43731*	*-0.41791*

Note: Presents simulated macroeconomic business cycle statistics across the four implementations using the common simulation protocol. The first block contains the three calibration targets emphasized by Chatterjee and Eyigungor (2012); default frequency and the remaining statistics are reported in the second block.

Fig. D.3: Simulated annualized spread trajectories under baseline calibration *(four-method comparison).*



Note: Evaluates annualized spread dynamics across solution methods under the baseline calibration using a shared exogenous output history. The gray Data series is included as an empirical visual reference; the largest cross-method separation occurs around quarters 11–12.

Table D.2: Algorithmic convergence diagnostics for smoothing methods under baseline calibration.

Method	σ_{aux}	ρ_D	ρ_B	K	terminal err_V	terminal err_q
Auxiliary shocks	0.012000	–	–	919	1.354×10^{-6}	9.932×10^{-6}
Auxiliary shocks	0.007500	–	–	923	1.112×10^{-6}	9.900×10^{-6}
Auxiliary shocks	0.003000	–	–	939	8.749×10^{-7}	9.912×10^{-6}
Auxiliary shocks	0.001500	–	–	875	1.176×10^{-6}	9.987×10^{-6}
Auxiliary shocks	0.000750	–	–	3000	5.115×10^{-5}	8.206×10^{-4}
Taste shocks	–	0.002	4×10^{-5}	338	4.106×10^{-7}	9.794×10^{-7}
Taste shocks	–	0.00125	2.5×10^{-5}	319	6.696×10^{-7}	9.678×10^{-7}
Taste shocks	–	5×10^{-4}	10^{-5}	344	2.615×10^{-7}	9.769×10^{-7}
Taste shocks	–	2.5×10^{-4}	5×10^{-6}	308	8.912×10^{-7}	9.596×10^{-7}
Taste shocks	–	1.25×10^{-4}	2.5×10^{-6}	1000	4.310×10^{-3}	5.160×10^{-2}

Note: Reports preset iteration caps K and successive-iteration sup-norm movements err_V and err_q for the auxiliary ($K = 3000$) and taste ($K = 1000$) fixed-point solvers. StH-Q and StH-AC are omitted because their fixed-budget stochastic procedures do not generate comparable successive-grid convergence statistics; their hard-equation accuracy is reported in Table D.4.

D.2. Approaching zero smoothing under the baseline calibration

Although non-convergence, kinked schedules, and non-monotonic responses are far less pronounced under baseline volatility, these numerical pathologies nevertheless persist.

D.2.1. Numerical deterioration and non-convergence

Table D.2 reports algorithmic convergence diagnostics as smoothing parameters shrink.

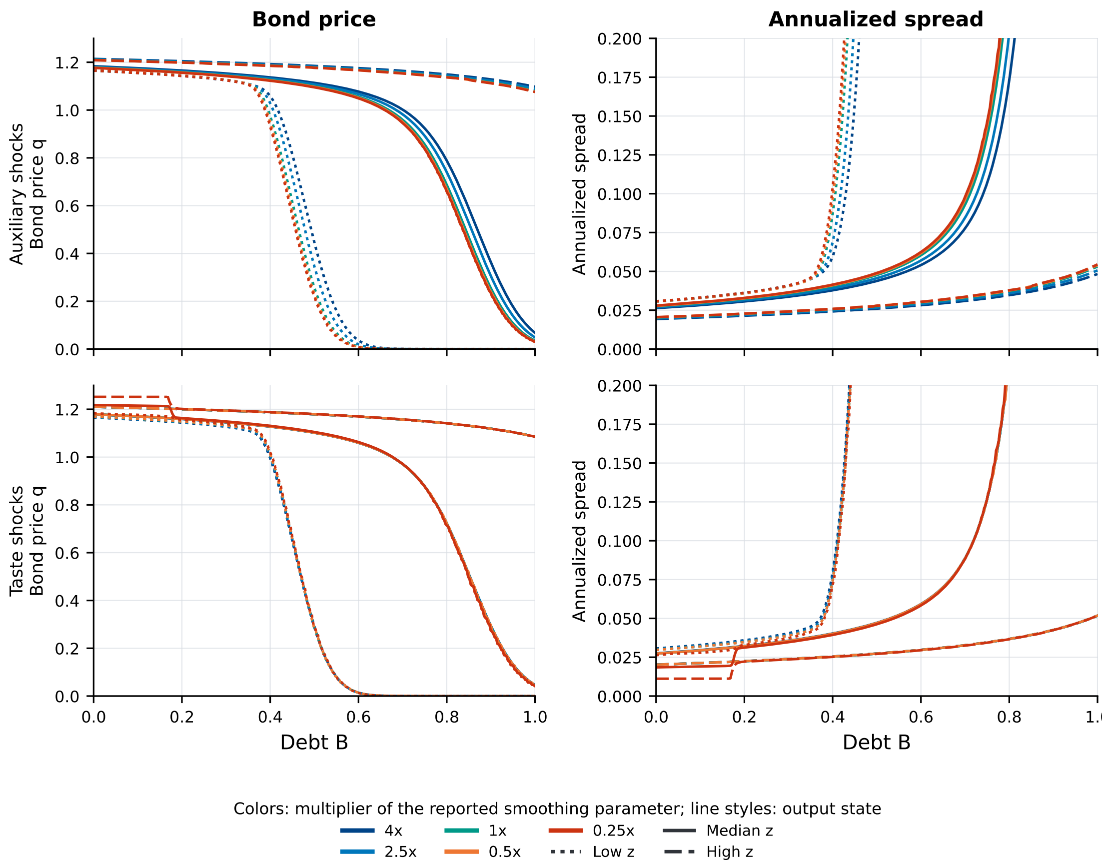
As in the high-volatility analysis, this table is deliberately limited to the auxiliary- and taste-shock fixed-point solvers. StH-Q and StH-AC do not have permanent smoothing parameters or comparable successive-grid iteration metrics; their baseline hard-equation accuracy is reported in Table D.4.

Although fixed-point iteration degrades less severely than under high volatility, the smallest parameter choice for both traditional solvers still hits the native iteration cap ($K = 3000$ for auxiliary shocks; $K = 1000$ for taste shocks). This table therefore tests whether traditional smoothing produces a stable route toward zero temperature. Because the StH implementations instead target the hard operator directly, their baseline accuracy is assessed by hard-equation closure in Table D.4.

D.2.2. Distorted equilibrium schedules

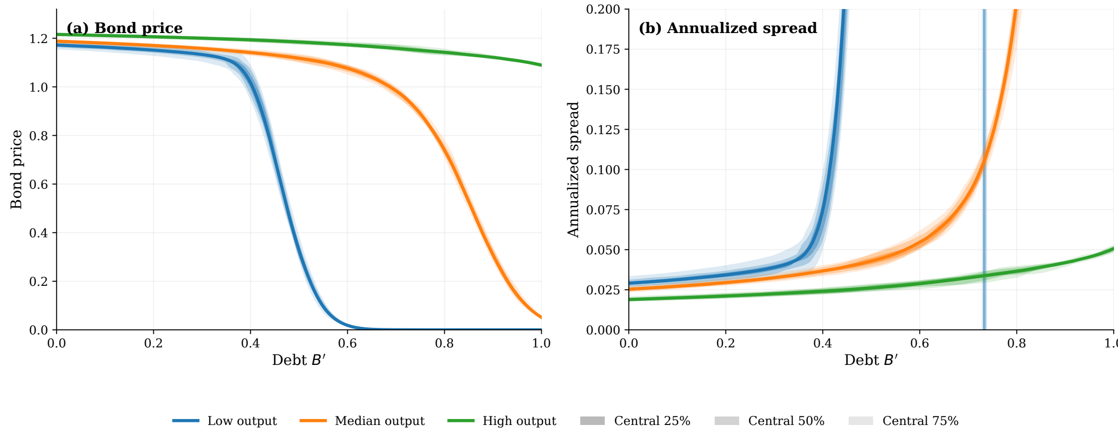
Figure D.4 shows the sensitivity of price and spread schedules across shrinking parameters.

Fig. D.4: Equilibrium-function sensitivity to smoothing parameters under baseline calibration.



Note: Reports sensitivity across auxiliary (top row) and taste (bottom row) shock specifications using common parameter multipliers applied to $\sigma_{\text{aux}} = 0.003$ and $(\rho_D, \rho_B) = (5 \times 10^{-4}, 10^{-5})$, with spreads capped at 20% for clarity.

Fig. D.5: Structural stability of StH-AC equilibrium objects across initialization seeds under baseline calibration.



Note: Displays bond price and annualized spread distributions across $N = 16$ independent training seeds. Colors identify low, median, and high output states, where solid colored curves represent seed means and nested bands show the central 25%, 50%, and 75% of the seed distribution; the spread display is capped at 20%. *Each seed uses 50,000 pretraining updates followed by the 1,100,000-step main schedule.*

Relative to the high-volatility regime, the equilibrium schedules remain much more tightly grouped. Figure D.5 reports the corresponding StH-AC sensitivity across training seeds using the same price and spread objects shown for the smoothing methods.

As the figure demonstrates, the StH-AC solver achieves high structural stability across 16 independent random seeds, entirely bypassing the need for artificial smoothing parameters.

D.2.3. Sensitivity of economic outcomes

Table D.3 details the moment variations induced by parameter tightening. Under the baseline calibration, sensitivity of the economic outcome to the smoothing shrinkage is milder than in the high-volatility stress test. In the auxiliary-shock economy, the mean debt-to-output ratio ranges from 0.68174 to 0.70859, approximately 3.9% relative to the lower value. In the taste-shock economy, it varies nonmonotonically from 0.69203 to 0.69368, a range below 1%. These are within-method sensitivities rather than a cross-method accuracy ranking.

Although the baseline ranges are modest, because the tightest auxiliary and taste specifications hit their native iteration caps (Table D.2), reducing the parameters does not provide numerical evidence of convergence to the unsmoothed limit.

Table D.3: Parameter sensitivity of business cycle moments under baseline calibration.

Method	σ_{aux} or ρ_D	ρ_B	Mean spread	Mean debt/output	Default frequency
Auxiliary shocks	0.012000	–	0.07946	0.70859	0.06797
Auxiliary shocks	0.007500	–	0.08135	0.69870	0.06919
Auxiliary shocks	0.003000	–	0.08152	0.69988	0.06883
Auxiliary shocks	0.001500	–	0.08422	0.68473	0.07108
Auxiliary shocks	0.000750	–	0.08462	0.68174	0.07125
Taste shocks	0.002	4×10^{-5}	0.08344	0.69307	0.07041
Taste shocks	0.00125	2.5×10^{-5}	0.08328	0.69288	0.07035
Taste shocks	5×10^{-4}	10^{-5}	0.08336	0.69327	0.07049
Taste shocks	2.5×10^{-4}	5×10^{-6}	0.08378	0.69368	0.07053
Taste shocks	1.25×10^{-4}	2.5×10^{-6}	0.08272	0.69203	0.07072
StH-Q (seed 88)	–	–	0.08356	0.69177	0.06903
StH-AC (seed 88)	–	–	0.08573	0.69484	0.07329

Note: Reports moment variation across smoothing-parameter multipliers for the auxiliary and taste-shock methods. The StH-Q and StH-AC rows report the fixed seed-88 hard-solution references used in the baseline four-method comparison; they contain no permanent smoothing parameters.

Across the 16 StH-AC training seeds, mean debt/output is 0.68984 with a sample standard deviation of 0.00734, while mean annual spread is 0.07959 with a sample standard deviation of 0.00707. The debt/output coefficient of variation is 1.06%. These are descriptive statistics across independently trained solutions, and their narrow dispersion reinforces the structural stability of the StH-AC solution.

D.3. Baseline numerical accuracy

Table D.4 reports the baseline counterparts of the hard-operator closure diagnostics defined in Section 5.3.

The baseline audit confirms both StH implementations on the hard equations. StH-Q reaches a Bellman MAE of 1.11×10^{-7}, a policy-gap MAE of 1.14×10^{-7}, and zero default-rule inconsistency. StH-AC records mean Bellman, policy-gap, and pricing errors of 1.45×10^{-3}, 5.06×10^{-5}, and 8.69×10^{-4}. Thus StH-Q provides high tabular precision in the small benchmark, while StH-AC evaluates the hard economy with a continuous neural representation.

Table D.4: Hard-operator accuracy diagnostics under baseline calibration.

Method	Bellman MAE	Repay policy-gap MAE	Pricing MAE	Default-rule inconsistency (%)
Auxiliary shocks	5.695×10^{-5}	2.978×10^{-6}	5.337×10^{-4}	0.35571
Taste shocks	5.873×10^{-6}	1.027×10^{-6}	4.692×10^{-4}	0.00143
StH-Q	1.113×10^{-7}	1.136×10^{-7}	4.852×10^{-4}	0.00000
StH-AC	1.449×10^{-3}	5.056×10^{-5}	8.686×10^{-4}	0.06571

Note: All methods are audited on a common state grid comprising 200 output points and 350 debt points. The four reported diagnostics are the Bellman MAE, repay-policy-gap MAE, pricing MAE, and default-rule inconsistency.

Appendix E. StH-Q implementation

The persistent repayment Q table contains $200 \times 350 \times 350$ entries in float64 arithmetic. Both calibrations use training seed 88 on an RTX 3080. Initialization starts from model primitives and a fixed rollover policy, followed by four full-action soft stages with $(\tau_D, \tau_B) = (0.03, 0.003), (0.02, 0.002), (0.01, 0.001), (0.003, 0.0003)$. Thus the method benefits from a model-based soft starting solution; it is not learning the entire problem from an uninformed Q table. These initialization costs are included in the reported solution time.

Main training uses 1,024 simulated environments and a TD batch of 16,384 observations, half from valid repayment experience and half from randomly drawn states with exploratory actions. The learning rates are $\alpha_Q = 1$ and $\alpha_D = 0.5$. The price update has rate 0.002 and right-multiplies its residual by $(I + 16 D_B^{\intercal} D_B)^{-1}$, where D_B is the first-difference matrix along the stored debt indices. This numerical preconditioner changes updates, not the equilibrium pricing equation. We audit the untransformed equation. Each iteration also updates all actions at up to 32 randomly selected states. Main exploration falls from 0.3 to 0.1; the default temperature falls from 0.003 to 0.0003 over 1,000 iterations and is then set to zero.

The 225 snapshots collected every 40 iterations over the final 9,000 iterations of the 26,000-iteration main run are averaged. Main exploration declines from 0.3 to 0.1 over the first 14,400 iterations and then remains at 0.1. The final 1,000 TD iterations freeze price, disable planning, and use exploration probability 0.4, with most exploratory actions near the current greedy action. They update the actual Q table and default values. The final export takes the feasible argmax of stored Q entries. There is no final full-action borrower solve. Nevertheless, the finishing stage is conditional on price, so very small borrower residuals do not imply equally small lender residuals or uniform accuracy of every unchosen Q entry. Exact configuration files and endpoint locations accompany the manuscript.

References

Aguiar, M., Amador, M., 2020. Self-fulfilling debt dilution: Maturity and multiplicity in debt models. American Economic Review 110, 2783–2818. doi:10.1257/aer.20180831.

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

Arellano, C., 2008. Default risk and income fluctuations in emerging economies. American Economic Review 98, 690–712. doi:10.1257/aer.98.3.690.

Arellano, C., Ramanarayanan, A., 2012. Default and the maturity structure in sovereign bonds. Journal of Political Economy 120, 187–232. doi:10.1086/666589.

Berry, S., Levinsohn, J., Pakes, A., 1995. Automobile prices in market equilibrium. Econometrica 63, 841–890. doi:10.2307/2171802.

Chatterjee, S., Eyigungor, B., 2012. Maturity, indebtedness, and default risk. American Economic Review 102, 2674–2699. doi:10.1257/aer.102.6.2674.

Druedahl, J., 2021. A guide on solving non-convex consumption-saving models. Computational Economics 58, 747–775. doi:10.1007/s10614-020-10045-x.

Dvorkin, M., Sánchez, J.M., Sapriza, H., Yurdagul, E., 2021. Sovereign debt restructurings. American Economic Journal: Macroeconomics 13, 26–77. doi:10.1257/mac.20190220.

Eaton, J., Gersovitz, M., 1981. Debt with potential repudiation: Theoretical and empirical analysis. The Review of Economic Studies 48, 289–309. doi:10.2307/2296886.

Fernández-Villaverde, J., 2026. Deep learning for solving economic models. Journal of Economic Literature URL: https://www.aeaweb.org/articles?id=10.1257/jel.20261794, doi:10.1257/jel.20261794. forthcoming.

Gordon, G., 2019. Efficient Computation with Taste Shocks. Working Paper 19-15. Federal Reserve Bank of Richmond. doi:10.21144/wp19-15.

Haarnoja, T., Zhou, A., Abbeel, P., 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, pp. 1861–1870.

Hatchondo, J.C., Martinez, L., 2009. Long-duration bonds and sovereign defaults.

Journal of International Economics 79, 117–125. doi:10.1016/j.jinteco.2009.07.002.

Iskhakov, F., Jørgensen, T.H., Rust, J., Schjerning, B., 2017. The endogenous grid method for discrete-continuous dynamic choice models with (or without) taste shocks. Quantitative Economics 8, 317–365. doi:10.3982/QE643.

Judd, K.L., 1992. Projection methods for solving aggregate growth models. Journal of Economic Theory 58, 410–452. doi:10.1016/0022-0531(92)90061-L.

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

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

Mateos-Planas, X., McCrary, S., Ríos-Rull, J.V., Wicht, A., 2026. The Generalized Euler Equation and the Bankruptcy–Sovereign Default Problem. PIER Working Paper 26-009. Penn Institute for Economic Research, Department of Economics, University of Pennsylvania.

McFadden, D., 1974. Conditional logit analysis of qualitative choice behavior, in: Zarembka, P. (Ed.), Frontiers in Econometrics. Academic Press, New York, pp. 105–142.

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

Rust, J., 1987. Optimal replacement of GMC bus engines: An empirical model of harold zurcher. Econometrica 55, 999–1033. doi:10.2307/1911259.

Sánchez, J.M., Sapriza, H., Yurdagul, E., 2014. Sovereign Default and the Choice of Maturity. Working Paper 2014-031. Federal Reserve Bank of St. Louis. doi:10.20955/wp.2014.031.

Su, C.L., Judd, K.L., 2012. Constrained optimization approaches to estimation of structural models. Econometrica 80, 2213–2230. doi:10.3982/ECTA7925.