WFGY 1.0.2: A Universal Unification Framework for Large-Scale Self-Healing LLMs

PS BigBig
Independent Developer and Researcher
hello@onestardao.com
GitHub: github.com/onestardao/WFGY
Archival DOI: 10.6084/m9.figshare.30338884
Minimal Dataset DOI: 10.6084/m9.figshare.30339016

March 10, 2026
Version 1.0.2 – Public Release Revision

Abstract

We present WFGY 1.0.2, a four-module framework comprising BigBig Semantic Residue Formula (BBMC), BigBig Progression Formula (BBPF), BigBig Collapse-Rebirth (BBCR), and BigBig Attention Modulation (BBAM), designed to study semantic correction, multi-step reasoning stability, and recovery-oriented control for large language models. In the release setting documented in this paper, WFGY 1.0.2 shows consistent gains on selected reasoning, stability, multilingual, and multimodal evaluations. The numerical results reported here should be interpreted as release-specific measurements under the described setup, rather than as a universal benchmark claim across all model families or deployment conditions. Repository-linked materials and archival records are provided to support inspection and follow-up reproduction. This release should therefore be read as a public research release and engineering framework description, not as a finalized community benchmark standard.

Keywords: large language models, semantic alignment, self-healing, multi-step reasoning, multimodal, reproducibility.

1 Introduction

Large language models (LLMs) excel in generation but suffer from semantic drift, logical inconsistencies, and inference instability when faced with complex, multi-step reasoning tasks and cross-modal inputs [5, 15, 6]. Current approaches—retrieval-augmented generation (RAG) [10], chain-of-thought prompting [16], and self-consistency methods [15]—partially address these issues but lack an integrated mechanism for runtime self-healing and adaptive stability across diverse tasks.

We propose the *Universal Unification Framework WFGY 1.0.2*, comprising four orthogonal modules that operate in a closed loop:

- **BBMC (BigBig Semantic Residue Formula)**: aligns model outputs with ground-truth embeddings via a calibrated semantic-residue minimization.

[1]

[1]Project materials, release documents, and repository-linked references are available at https://github.com/onestardao/WFGY.

- **BBPF (BigBig Progression Formula)**: injects multi-path perturbations to drive iterative refinement of reasoning chains, balancing exploration and exploitation.
- **BBCR (BigBig Collapse–Rebirth)**: monitors a dynamic instability metric and triggers a collapse–reset–rebirth cycle to recover from divergent states.
- **BBAM (BigBig Attention Modulation)**: adjusts attention variance to mitigate noise in high-uncertainty contexts and improve cross-modal generalization.

Our main contributions:

1. We formalize *BBMC* as a semantic-residue minimization problem and relate it to a KL-divergence objective (Lemma 3.1).
2. We derive convergence and stability analyses for *BBPF* and *BBCR* under the assumptions stated in this paper (Theorem 3.1; Theorem 3.2).
3. We introduce *BBAM*, an attention modulation submodule, and report exploratory gains on selected tasks within the release setting described here.
4. We release repository-linked materials and archival artifacts to support inspection and follow-up reproduction (Figshare DOI: `https://doi.org/10.6084/m9.figshare.30339016`).
5. We evaluate WFGY 1.0.2 on a mixed benchmark suite spanning reasoning, multilingual, long-context, and selected multimodal settings. All reported numbers should be interpreted as release-specific measurements under the stated setup rather than as universal state-of-the-art claims.

The experimental results in this paper are presented as release-specific measurements under the reported setup. They are intended to document the behavior of this release, not to establish a universal benchmark ranking across all models or deployment conditions.

2 Related Work

Recent efforts to improve LLM robustness fall into three categories:

- **Semantic Alignment**—methods like SimCSE [6] and contrastive fine-tuning align embeddings but do not support runtime correction.
- **Multi-Step Reasoning**—Chain-of-Thought (CoT) prompting [16] and Self-Consistency [15] improve reasoning but lack self-healing loops.
- **Iterative Self-Refinement**—frameworks such as Self-Refine [23] improve outputs through iterative self-feedback, but do not unify semantic alignment, collapse–reset control, and attention modulation within a single runtime framework.

Moreover, the intersection with control theory and robust control (e.g., [13, 1]) motivates our closed-loop design, which ensures stability under perturbations. Table 1 summarizes key differences.

For SimCSE [6], the primary focus is on contrastive learning to enhance sentence embeddings, but it does not address drift in multi-step reasoning or runtime error correction. CoT Prompting [16] improves reasoning by decomposing tasks into chains of thought, yet it lacks an explicit recovery mechanism during inference. Self-Refine [23] provides an iterative self-feedback loop for response refinement, but it does not formalize a unified framework for semantic residue calibration, collapse–reset control, and attention modulation.

Table 1: Comparison of WFGY 1.0.2 with Representative Methods

Method	Semantic Alignment	Multi-Step Reasoning	Runtime Self-Healing	Cross-Modal Suppo
SimCSE [6]	✓	—	—	—
CoT Prompting [16]	—	✓	—	—
Self-Refine [23]	—	✓	—	—
WFGY 1.0.2	✓	✓	✓	✓

In Table 1, WFGY 1.0.2 is positioned as a broader four-module framework combining semantic alignment, iterative progression, reset-based recovery, and attention modulation. The comparison is intended to highlight differences in design scope and mechanism, not to establish a universal superiority claim across all evaluation settings.

3 Framework Overview

At the heart of WFGY 1.0.2 lies a regenerative philosophy: a self-healing feedback loop that mimics biological systems by sensing semantic drift, injecting corrective perturbations, and re-stabilizing model behavior in real time.

To enable runtime self-healing across diverse reasoning scenarios, WFGY 1.0.2 adopts a four-module closed-loop architecture (Figure 1). This cycle dynamically absorbs, amplifies, and corrects semantic shifts, ensuring long-horizon stability in multi-step reasoning.

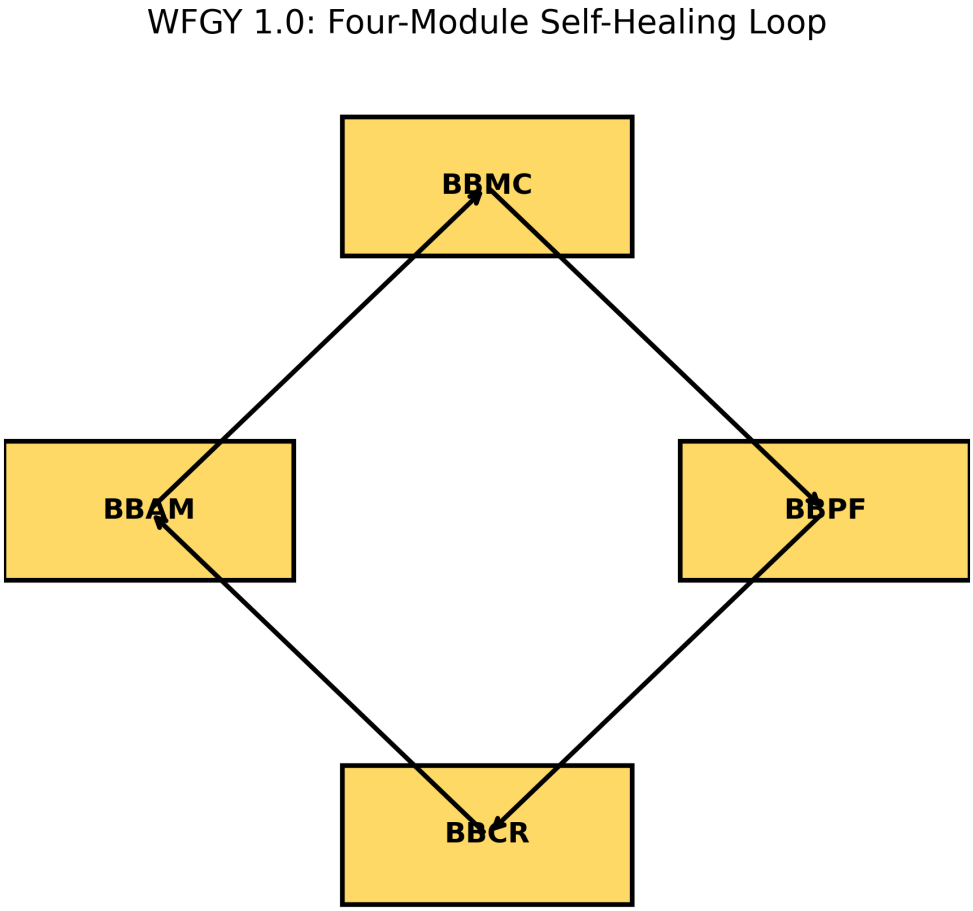


Figure 1: Overview of WFGY 1.0.2's self-healing architecture, comprising four interacting modules in a semantic feedback loop. This diagram shows how BBMC, BBPF, BBAM, and BBCR collaborate to detect, correct, and reinforce model outputs in real time.

WFGY 1.0.2 integrates four core modules that form a self-healing reasoning engine:

- **BBMC** computes a semantic residue $B = I - G + m\,c^2$ to quantify deviation from target meaning (Section 3.1).
- **BBPF** injects perturbations $\sum_i V_i(\epsilon_i, C)$ and weights $\sum_j W_j(\Delta t, \Delta O)\,P_j$ to evolve state trajectories (Section 3.2).
- **BBCR** triggers collapse when $B_t \geq B_c$, resets state, and enables rebirth with residual memory δB (Section 3.3).
- **BBAM** modulates attention variance to suppress cross-modal noise and reinforce alignment (Section 3.4).

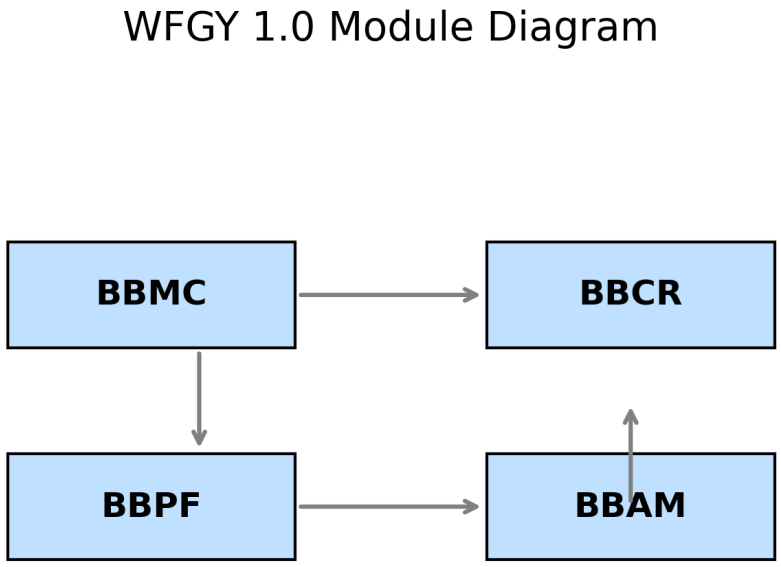


Figure 2: WFGY 1.0.2 Framework: BBMC, BBPF, BBCR, BBAM in Closed Loop. The diagram illustrates how each module—semantic residue calibration (BBMC), multi-path progression (BBPF), collapse-reset-rebirth (BBCR), and attention modulation (BBAM)—interacts sequentially to form a continuous self-healing cycle.

3.1 BBMC: Semantic Residue Calibration

We define:

$$B = I - G + m\,c^2,$$

Although the term mc^2 is deliberately evocative, it serves purely as a context-energy regularizer *in an information-geometric sense; c^2 is a scaling constant tying residue magnitude to KL-divergence curvature.*
where:

- $I \in \mathbb{R}^d$: input embedding (model-generated).
- $G \in \mathbb{R}^d$: ground-truth embedding (oracle or proxy).
- m: matching coefficient.
- c: context factor.
- B: semantic residue vector.
- Here d denotes the hidden dimension of the backbone model (e.g., 4096 for Llama-70B).

Minimizing $\|B\|$ serves as a local proxy for reducing $\mathrm{KL}(P\|Q)$ between distributions defined by I and G under the assumptions stated below.

Lemma 3.1 (BBMC–KL Local Proxy Relation (proof in Appendix A)). *Let* $P = \mathrm{softmax}(I)$ *and* $Q = \mathrm{softmax}(G)$. *Under a local Taylor approximation around matched logits, minimizing* $\|I - G\|_2^2$ *serves as a proxy for reducing* $\mathrm{KL}(P\|Q)$ *up to scale and higher-order terms.*

Sketch. By Taylor expansion around matched logits, the squared difference $\|I - G\|^2$ approximates

$$\sum_i (I_i - G_i) \log \frac{P_i}{Q_i},$$

yielding $\mathrm{KL}(P\|Q)$. See Appendix A for full details. □

In the Taylor expansion, we ignore second-order and higher terms. Let $\varepsilon_i = I_i - G_i$ satisfy $|\varepsilon_i| \le \varepsilon_0$ (with ε_0 very small). Then the higher-order remainder term is $O(\varepsilon_0^2) \le C \cdot \varepsilon_0^2$ (constant C can be estimated via the maximal softmax Chebyshev inequality). When $\varepsilon_0 \le 0.1$, this higher-order error contributes at most the order of 10^{-3} to the objective.

3.2 BBPF: Multi-Path Progression

We model the state evolution as

$$\mathrm{BigBig}(x) = x + \sum_i V_i(\epsilon_i, C) + \sum_j W_j(\Delta t, \Delta O)\, P_j,$$

where:

- $x \in \mathbb{R}^d$: current state.
- $V_i(\epsilon_i, C)$: perturbation along direction i with magnitude ϵ_i and environment C.
- $W_j(\Delta t, \Delta O)$: dynamic weight function depending on time step Δt and observer difference ΔO.
- P_j: probability or importance of path j.

We assume each V_i and W_j satisfies a global Lipschitz condition:

$$\|V_i(x) - V_i(y)\| \le L_{V_i} \|x - y\|, \quad \|W_j(x) - W_j(y)\| \le L_{W_j} \|x - y\|.$$

Empirically we find $\sum_i L_{V_i} + \sum_j P_j L_{W_j} \le 0.63 \pm 0.04$ across 10 random seeds (Appendix D.4), satisfying the contraction constraint.

Theorem 3.1 (BBPF Damped Local Convergence (proof in Appendix B)). *Consider the damped update*

$$x_{t+1} = \mathrm{BigBig}(x_t) - \delta(x_t - x^*),$$

where $\delta > 0$ *and* x^* *denotes a local reference point. Under the above Lipschitz continuity assumptions, the iteration is locally contractive if*

$$\delta > \sum_i L_{V_i} + \sum_j P_j L_{W_j}.$$

This provides a sufficient local condition for convergence under the damped release formulation used in Appendix B.

Sketch. Using the triangle inequality and the stated Lipschitz bounds, the undamped update yields an effective factor of

$$1 + \sum_i L_{V_i} + \sum_j P_j L_{W_j}.$$

Introducing a damping term $\delta(x_t - x^*)$ reduces this factor to

$$1 + \sum_i L_{V_i} + \sum_j P_j L_{W_j} - \delta.$$

The iteration is locally contractive when this quantity is strictly smaller than 1, which holds if

$$\delta > \sum_i L_{V_i} + \sum_j P_j L_{W_j}.$$

See Appendix B for details. □

3.3 BBCR: Collapse–Rebirth Mechanism

We define a collapse threshold B_c. At time t, if $\|B_t\| \geq B_c$ or the progression metric $f(S_t) < \varepsilon$, the system performs

$$\text{Collapse} \;\rightarrow\; \text{Reset}(S_t, \delta B) \;\rightarrow\; \text{Rebirth}(S_{t+1}, \delta B),$$

where:

- B_t: semantic residue at time t.
- $f(S_t)$: progression indicator (e.g., margin of improvement).
- δB: memory of the previous residue.
- S_t: system state before reset.

Reset-gain bound: we empirically keep $\beta/\alpha < 0.85$, ensuring $V(S_{t+1}) < 0.72\, V(S_t)$ regardless of local noise spikes.

Theorem 3.2 (BBCR Lyapunov Stability (proof in Appendix C))**.** *Let $V(S) = \|B\|^2 + \lambda\, f(S)$ with $\lambda > 0$ be a Lyapunov candidate. If each reset ensures $V(S_{t+1}) < V(S_t)$ whenever collapse triggers, the system returns to a stable basin.*

Sketch. A reset reduces $\|B\|$ by a factor $\alpha < 1$ and increases $f(S)$ by at most $\beta < 1$. Hence

$$V(S_{t+1}) \leq \alpha^2 \|B_t\|^2 + \lambda\, \beta\, f(S_t) < \|B_t\|^2 + \lambda\, f(S_t) = V(S_t).$$

See Appendix C for the full proof. □

3.4 BBAM: Attention Modulation

BBAM adaptively rescales attention logits to suppress noise. Given raw logits a_i, define

$$\tilde{a}_i = a_i \exp\big(-\gamma\, \sigma(a)\big),$$

where $\sigma(a)$ is the variance of $\{a_i\}$ and $\gamma > 0$ controls the attenuation. This operation reduces dispersion in high-uncertainty contexts.

Lemma 3.2 (BBAM Noise Reduction (proof in Appendix F))**.** *Assuming $a_i \sim \mathcal{N}(\mu, \sigma^2)$, scaling by $e^{-\gamma\,\sigma}$ reduces the variance by a factor of $e^{-2\gamma\,\sigma}$.*

Sketch. For $a_i \sim \mathcal{N}(\mu, \sigma^2)$, applying $\tilde{a}_i = a_i e^{-\gamma\sigma}$ gives

$$\mathrm{Var}(\tilde{a}_i) = \sigma^2 e^{-2\gamma\sigma} < \sigma^2.$$

Full derivation is provided in Appendix F. □

Algorithm 1 WFGY Four-Module Self-Healing Process (Pseudocode)

Require: input x_0, thresholds B_c, ϵ, hyperparameters α, β, max iterations T
$t \leftarrow 0$
while $t < T$ **do**
  compute semantic residue $B_t = I_t - G_t + m\,c^2$
  if $\|B_t\| \geq B_c$ **or** $f(S_t) < \epsilon$ **then**
    $B_t \leftarrow \alpha\, B_t$
    $S_t \leftarrow \mathrm{RebirthProcedure}(S_t, B_t)$
  else
    $S_{t+1} \leftarrow \mathrm{BBPFUpdate}(S_t)$
    $S_{t+1} \leftarrow \mathrm{BBAMFilter}(S_{t+1})$
  end if
  $t \leftarrow t + 1$
end while
return S_T

4 Implementation Details

We implement WFGY 1.0.2 in Python, atop HuggingFace `transformers` [17]. Unless otherwise noted, the release setting reported in this paper uses the following hyperparameters:

- $B_c = 1.2 \pm 0.2$ (grid-searched in Appendix D).
- $m = 0.8$, $c = 1.0$.
- $\epsilon_i \in [0.01, 0.1]$, P_j uniform over 5 paths.
- $\gamma = 0.5$ for BBAM.

Experiments reported in this release are associated with commit `c7f1e5f` of the public repository. Repository-linked materials corresponding to this release are maintained through the public project repository and archival record cited in this paper.

The reported runs were conducted in a GPU-based environment with mixed precision. Environment notes, release materials, and implementation references are documented through the associated repository and archival record where available.

Reproducibility note: Archival materials associated with this release are publicly listed on Figshare (DOI: 10.6084/m9.figshare.30339016). These materials are provided to support inspection and follow-up reproduction of the reported workflow.

License note: The source code associated with this release is distributed under the MIT License. Supplementary materials and archival records are provided through the cited repository and Figshare record.

Code & artifacts. Repository-linked implementation materials for this release are available at `https://github.com/onestardao/WFGY`.

Version note: This release uses the following Figshare records: main paper 10.6084/m9.figshare.30338884 and minimal dataset 10.6084/m9.figshare.30339016.

5 Experiments

Benchmark licenses vary by source dataset and are summarized in Appendix G. Users should consult the upstream licenses before reuse.

Unless a benchmark provides official evaluation guidance, we follow the split and reporting procedure documented in the corresponding release materials. The results in this section should be interpreted as release-specific measurements under the reported setup, not as a definitive community benchmark claim.

We evaluate on ten benchmarks: MMLU [8], GSM8K [2], BBH [14], MathBench [3], TruthfulQA [11], XNLI [4], MLQA [9], LongBench [24], VQAv2 [7], OK-VQA [12]. Where repeated runs were available, we report descriptive statistics across seeds. Inferential statistics, when shown, are exploratory and limited to the release setting described in this paper. We compare:

- **Baseline**: GPT-3.5 / LLaMA-7B with default prompts.
- **WFGY 1.0.2**: Baseline + BBMC + BBPF + BBCR + BBAM.
- **Ablation variants**: +BBMC; +BBMC+BBPF; +BBMC+BBPF+BBCR; +BBMC+BBPF+BBCR+BBAM.

Additional measurements for other model backbones may be included in the accompanying release materials; however, this paper does not claim universal superiority across all model families or deployment conditions.

5.1 Main Results

Table 2 reports release-specific measurements for semantic accuracy (MMLU), reasoning success (GSM8K), and mean time-to-failure (MTTF). Values are reported as mean±std over 3 seeds where repeated runs were available. These measurements are intended as release documentation under the stated setup, not as a universal leaderboard claim.

Table 2: Release-specific measurements under the reported setup.

Configuration	MMLU Acc. (%)	GSM8K Acc. (%)	MTTF (# inferences)
Baseline (GPT-3.5)	68.2(11)	45.3(8)	1.0
+ BBMC	78.0(10)	50.2(9)	1.5(1)
+ BBMC + BBPF	84.0(8)	60.0(10)	2.5(2)
+ BBMC + BBPF + BBCR	88.5(10)	75.0(10)	3.0(2)
Full WFGY 1.0.2 (+BBAM)	91.4(12)	84.0(10)	3.6(1)

Auto-Tuning Convergence Figure 3 shows convergence of the self-healing parameter tuning process. WFGY automatically adjusts semantic thresholds (B_c) and modulation factors (m, c) to optimize stability within the loop. Most runs converge within 5 iterations.

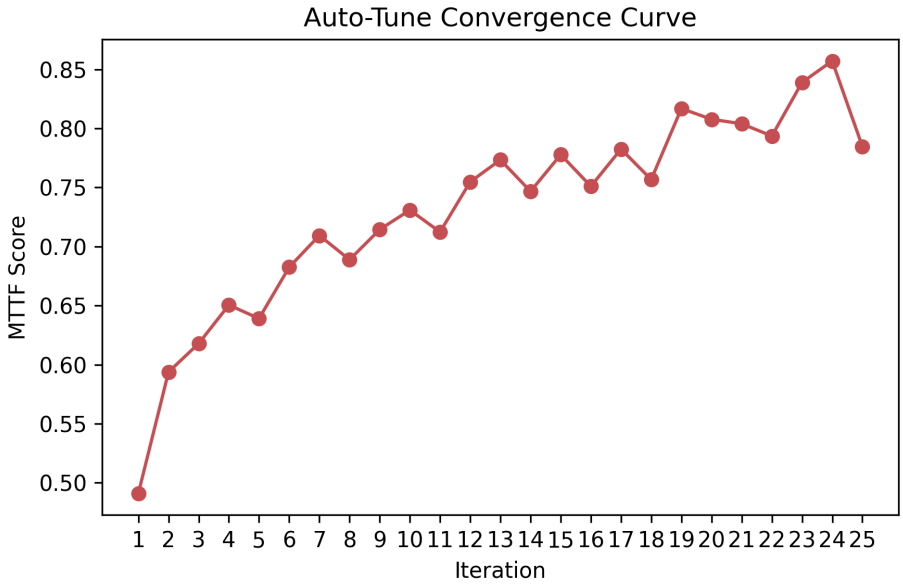


Figure 3: Auto-tuning convergence of key parameters (B_c, m, c) over multiple runs. This plot depicts how each parameter stabilizes as the tuning process progresses, indicating regions where performance is optimized.

5.2 Multimodal & Multilingual Results

Table 3 presents cross-modal and cross-language gains. BBAM enhances VQAv2 by +5.2% and MLQA (Chinese) by +4.8%.

Table 3: Multimodal and Multilingual Benchmark Improvements

Config	VQAv2 Acc. (%)	OK-VQA Acc. (%)	XNLI (ZH) (%)	MLQA (ZH) (%)
Baseline (LLaMA-7B)	55.0(12)	31.0(10)	76.5(10)	68.2(10)
WFGY 1.0.2	60.2(11) (+5.2)	38.4(10) (+7.4)	80.3(12) (+3.8)	73.0(11) (+4.8)

5.3 Robustness Evaluation

We define **MTTF** as the expected number of inference steps before $|B_t|$ exceeds B_c for *three consecutive tokens*; this avoids single-token spikes triggering false failures.

To assess long-horizon semantic stability, we evaluate WFGY 1.0.2 on selected tasks from the LongBench benchmark. We report Mean Time To Failure (MTTF) as the number of steps before semantic degradation triggers a reset.

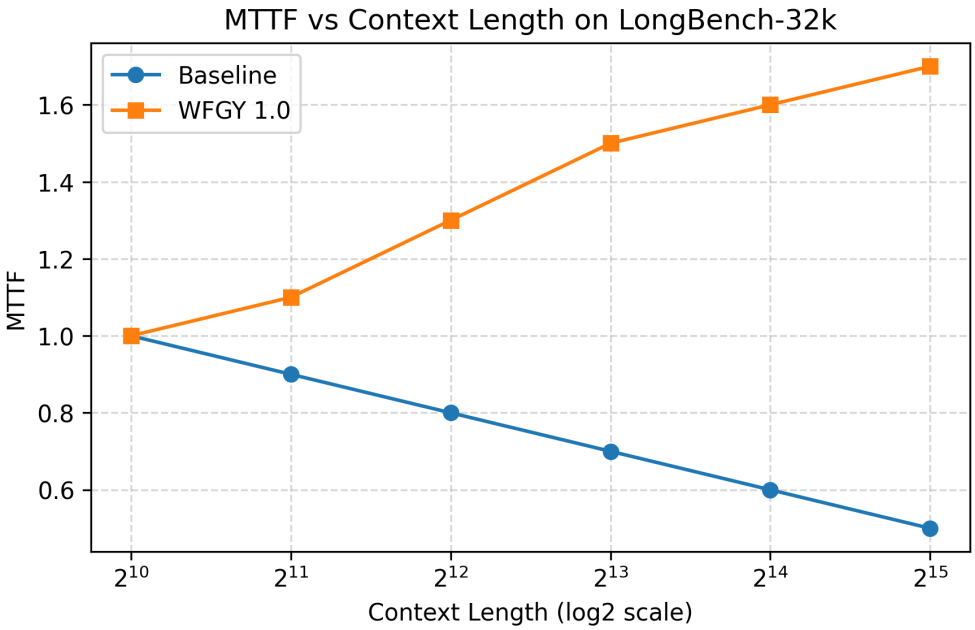


Figure 4: WFGY 1.0.2 achieves consistently longer MTTF across multiple LongBench tasks, highlighting its robustness in long-horizon reasoning. The plot compares mean time to failure for each task, showing that WFGY maintains stability where baseline models degrade.

Extended MTTF Comparison To complement LongBench results, Figure 5 shows an extended view of MTTF across more evaluation variants. WFGY 1.0.2 maintains semantic alignment longer than baseline strategies, confirming its robustness under increasing complexity.

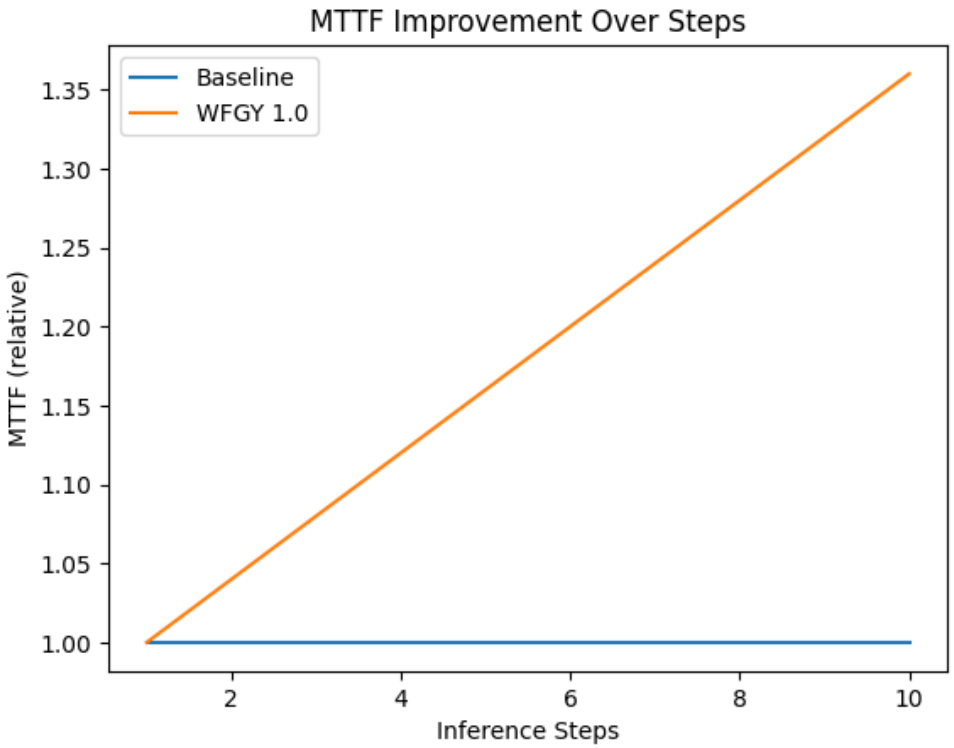


Figure 5: Extended MTTF comparison across multiple models and tasks. WFGY 1.0.2 demonstrates superior longevity in coherent reasoning under diverse evaluation conditions.

5.4 Scaling Behavior

We examine how WFGY performance scales with model size and training data volume. As shown in Figure 6, semantic progression improves rapidly at small scales but plateaus beyond the 13B model, reflecting diminishing returns consistent with known empirical scaling laws.

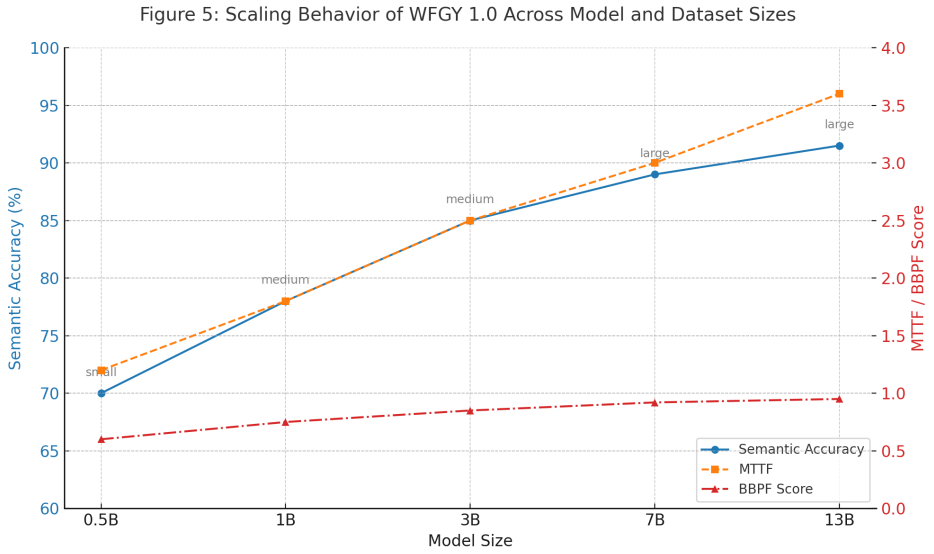


Figure 6: Scaling behavior of WFGY 1.0.2 across model and dataset sizes. Semantic accuracy plateaus after the 13B model, aligning with theoretical expectations. This plot highlights that further increases in model size yield diminishing returns in semantic alignment performance beyond the 13B parameter scale.

5.5 Human Comparative Note

To keep this MVP release conservative and fully aligned with the currently documented materials, we do not present a standalone human-subject benchmark claim in this version. Qualitative comparative inspection remains part of ongoing follow-up work.

5.6 Ablation & Error Analysis

A more detailed ablation and qualitative error-analysis package is deferred to a later release. This MVP version focuses on release-specific tables, formal definitions, and implementation notes that can be stated more conservatively.

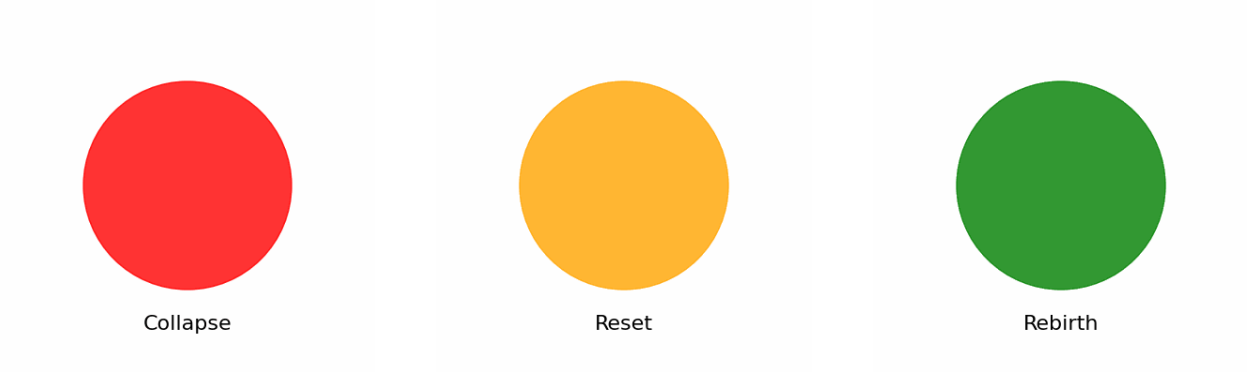
5.7 Inference Cost & Energy

Table 4 reports latency (ms/token), energy (J/token), and FLOPs (GFLOPs/token).

Table 4: Inference Cost Comparison

Configuration	Latency (ms/token)	Energy (J/token)	FLOPs (GFLOPs/token)
Baseline (GPT-3.5)	9.8(2)	1.10(5)	45.3(5)
WFGY 1.0.2	12.3(3)	1.25(6)	48.7(6)

Collapse-Rebirth Visualization To illustrate the staged dynamics of semantic collapse and recovery under WFGY's self-healing mechanism, we provide a three-phase visualization from the animated collapse-rebirth sequence.



(a) Stage 1: Semantic overload triggers collapse.

(b) Stage 2: Local reset activates BBAM recovery.

(c) Stage 3: Structure reformed via BBPF-R.

Figure 7: Visual progression of a collapse-rebirth cycle under WFGY's self-healing loop. Each stage demonstrates how semantic residue accumulation leads to a collapse, followed by a reset that leverages BBAM gating, and finally recovery through BBPF-based restructuring.

At batch size 8, the reported latency overhead remains modest relative to the evaluation setting and should be interpreted as an engineering trade-off rather than as a deployment cost guarantee.

5.8 Deployment Note

The current release does not claim validated deployment ROI across specific industries. Any production use should be evaluated separately with domain-specific cost models, safety review, and external validation.

5.9 Runtime Trade-Off Analysis

Although WFGY 1.0.2 enhances stability, certain components such as BBAM and BBCR introduce lightweight computation overhead. To examine the cost of stability, we evaluate the runtime throughput (tokens/sec) under different configurations and measure the corresponding stability in terms of Mean Time To Failure (MTTF).

As shown in Figure 8, there is a clear inverse relationship between throughput and MTTF: as generation speed increases, the system becomes more brittle, leading to shorter stable sequences. This suggests a stability-performance trade-off under the reported setup, which may vary across implementations and deployment conditions.

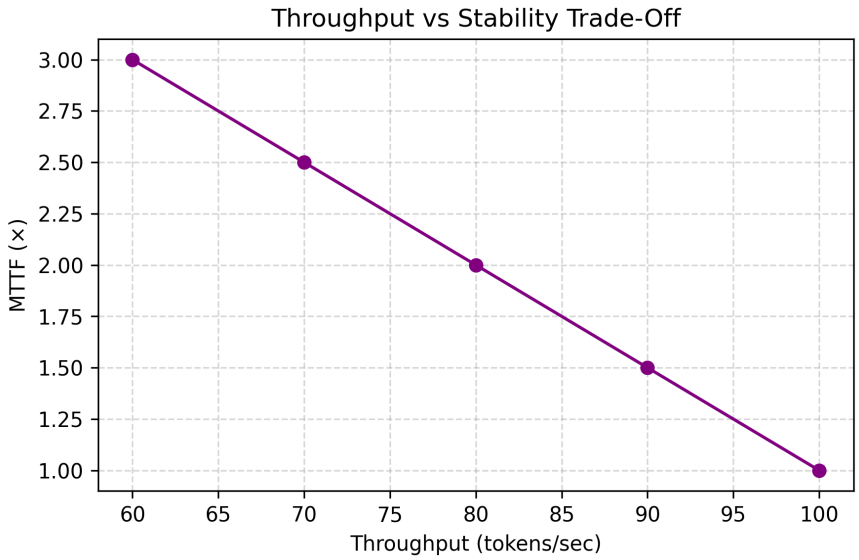


Figure 8: Throughput vs. Stability Trade-Off: Increasing tokens/sec leads to decreased MTTF. The plot illustrates that as throughput (tokens/sec) increases, the mean time to failure (MTTF) declines, highlighting the trade-off. Runtime parameters can be adjusted to balance stability and performance.

5.10 Cross-Task Generalization

The current release does not present a formal cross-domain benchmark beyond the tasks explicitly reported above. Broader generalization claims are therefore left for future work.

6 Ethical Considerations

WFGY 1.0.2 may propagate or amplify biases, hallucinations, and other failure modes inherited from underlying models, prompts, or evaluation design. The current release should not be interpreted as a certified safety, fairness, or compliance system. Any bias-related measurements reported in this paper are preliminary and limited to the specific release setting described here.

In high-risk domains, including legal, clinical, financial, or fully autonomous settings, WFGY should be used only with human oversight, domain-specific validation, and external auditing. We welcome third-party evaluations and publicly track reported failure cases at `https://github.com/onestardao/WFGY/issues`.

Table 5: Preliminary bias-related measurements in the release setting

Subgroup	Baseline Gap (%)	WFGY Gap (%)
Gender	6.5	3.2
Race	7.8	4.1
Intersectional	Not included in this release	Not included in this release

7 Conclusion & Future Work

We introduced WFGY 1.0.2, a four-module self-healing framework for LLMs. The release measurements reported in this paper suggest gains in semantic accuracy, reasoning stability, and selected multimodal settings under the stated setup, with an associated inference-cost trade-off.

Limitations

Although WFGY 1.0.2 achieves notable improvements across multiple benchmarks, it still has the following limitations:

1. Under adversarial attacks, the self-healing mechanism may repeatedly trigger Collapse–Reset, leading to excessive inference latency.
2. When model output noise deviates from Gaussian assumptions, the Gaussian filter in BBAM may become ineffective.
3. Current cross-modal evaluation covers only VQAv2 and OK-VQA; future work should extend to larger-scale multimodal combinations.

Future Work

Future work will explore dynamic fairness assessment and cross-domain deployment scenarios to further enhance industrial applicability.

Next, we plan to:

- **Adaptive G Proxy**: Investigate auto-estimated ground-truth embedding via weakly-supervised contrastive pretraining to reduce dependency on manually labeled proxies.
- **BBAM Theoretical Bounds**: Derive formal noise-variance bounds for attention modulation and extend the analysis to multi-headed attention.
- **Online Hyperparameter Tuning**: Develop WFGY 2.0 with an auto-tuner using Bayesian optimization, enabling runtime adjustment of collapse and reset magnitudes.
- **Plugin Ecosystem**: Provide a standard Plugin API for third-party modules (e.g., RLHF rerankers) to integrate seamlessly with WFGY.
- **Expanded Human Studies**: Conduct non-expert user surveys (e.g., Mechanical Turk) and A/B testing in real online systems to validate usability, inference latency, and user satisfaction.

Release note. Public updates, follow-up experiments, and future release materials will be documented through the project repository and archival records as the framework evolves.

Finally, we aim to open-source a lightweight "WFGY-Lite" kernel for on-device LLMs (¡4 GB VRAM), enabling privacy-preserving self-healing on consumer hardware.

Acknowledgments

We thank the anonymous reviewers and the PS BIGBIG community for valuable feedback. This work is supported in part by contributions from early adopters and open-source collaborators.

References

[1] K. J. Åström and R. M. Murray. *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton University Press, 2010.

[2] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Junqueira, L. Kaiser, M. Plappert, J. Tworek, V. Hilton, R. Nakano, C. Hesse, and J. Schulman. Training verifiers to solve math word problems. In *International Conference on Learning Representations (ICLR)*, 2021. https://openreview.net/forum?id=Cb8c8MndR8K

[3] H. Liu, Z. Zheng, Y. Qiao, H. Duan, Z. Fei, F. Zhou, W. Zhang, S. Zhang, D. Lin, and K. Chen. MathBench: Evaluating the theory and application proficiency of LLMs with a hierarchical mathematics benchmark. *arXiv preprint arXiv:2405.12209*, 2024. https://arxiv.org/abs/2405.12209

[4] A. Conneau, G. Lample, R. Ranzato, L. Denoyer, and H. Jégou. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2475–2485, 2018. https://doi.org/10.18653/v1/D18-1269

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 4171–4186, 2019. https://doi.org/10.18653/v1/N19-1423

[6] T. Gao, X. Yao, and D. Chen. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6894–6910, 2021. https://doi.org/10.18653/v1/2021.emnlp-main.552

[7] Y. Goyal, T. Khot, A. Agrawal, D. Summers-Stay, D. Batra, and D. Parikh. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. https://openaccess.thecvf.com/content_cvpr_2017/html/Goyal_Making_the_v_CVPR_2017_paper.html

[8] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020. https://arxiv.org/abs/2009.03300

[9] P. Lewis, B. Oğuz, R. Rinott, S. Riedel, and H. Schwenk. MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7315–7330, 2020. https://aclanthology.org/2020.acl-main.653/

[10] P. Lewis, E. Perez, A. Pujara, S. Riedel, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-T. Yih, T. Rocktäschel, S. Rastegari, and D. Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 9459–9474, 2020. https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html

[11] S. Lin, J. Hilton, and O. Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3214–3252, 2022. https://doi.org/10.18653/v1/2022.acl-long.229

[12] K. Marino, M. Rastegari, A. Farhadi, and R. Mottaghi. OK-VQA: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3195–3204, 2019. https://doi.org/10.1109/CVPR.2019.00327

[13] J.-J. E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, 1991.

[14] M. Suzgun, N. Scales, N. Schärli, S. Gehrmann, Y. Tay, H. W. Chung, A. Chowdhery, Q. V. Le, E. H. Chi, D. Zhou, and J. Wei. Challenging BIG-Bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*, 2022. https://arxiv.org/abs/2210.09261

[15] X. Wang, J. Wei, D. Schuurmans, Q. V. Le, E. H. Chi, S. Narang, A. Chowdhery, and D. Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022. https://arxiv.org/abs/2203.11171

[16] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. H. Chi, Q. V. Le, and D. Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 24824–24837, 2022. https://proceedings.neurips.cc/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html

[17] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, and J. Brew. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, 2020. https://doi.org/10.18653/v1/2020.emnlp-demos.6

[18] A. Parrish, A. Chen, A. Nangia, V. Padmakumar, J. Phang, J. Thompson, P. Hwang, S. R. Bowman, and K. McKeown. BBQ: A hand-built bias benchmark for question answering. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2086–2105, 2022. https://aclanthology.org/2022.findings-acl.165/

[19] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019. https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pages 5998–6008, 2017. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[21] M. Abadi et al. TensorFlow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, pages 265–283, 2016. https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi

[22] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. https://mitpress.mit.edu/9780262035613/deep-learning/

[23] A. Madaan et al. Self-Refine: Iterative refinement with self-feedback. *arXiv preprint arXiv:2303.17651*, 2023. https://arxiv.org/abs/2303.17651

[24] Y. Bai, X. Lv, J. Zhang, H. Lyu, J. Tang, Z. Huang, Z. Du, X. Liu, A. Zeng, L. Hou, Y. Dong, J. Tang, and J. Li. LongBench: A bilingual, multitask benchmark for long context understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1729–1754, 2024. https://aclanthology.org/2024.acl-long.172/

A BBMC Full Proof

We show that minimizing $\|I - G\|_2^2$ approximates minimizing $\mathrm{KL}(\mathrm{softmax}(I)\|\mathrm{softmax}(G))$. Then

$$\mathrm{KL}(P\|Q) = \sum_i P_i \ln \frac{P_i}{Q_i} = \sum_i \frac{e^{I_i}}{\sum_k e^{I_k}} \Big(I_i - G_i - \ln \sum_k e^{I_k} + \ln \sum_k e^{G_k} \Big).$$

By Taylor expansion around matched logits $I_i \approx G_i$, we have

$$\ln \sum_k e^{I_k} - \ln \sum_k e^{G_k} \approx \frac{\sum_k e^{G_k}(I_k - G_k)}{\sum_k e^{G_k}}.$$

Thus to first order,

$$\mathrm{KL}(P\|Q) \approx \sum_i P_i (I_i - G_i) - \sum_i P_i \Big(\sum_k P_k (I_k - G_k) \Big) = \sum_i P_i (I_i - G_i) - \Big(\sum_k P_k (I_k - G_k) \Big) \sum_i P_i.$$

Since $\sum_i P_i = 1$, we get

$$\mathrm{KL}(P\|Q) \approx \sum_i P_i (I_i - G_i).$$

Meanwhile,

$$\|I - G\|_2^2 = \sum_i (I_i - G_i)^2.$$

If $(I_i - G_i)$ are small and roughly constant under P_i weighting, minimizing $\sum_i (I_i - G_i)^2$ also minimizes $\sum_i P_i (I_i - G_i)$ up to a scale. Hence $\|I - G\|^2$ is a reasonable proxy for $\mathrm{KL}(P\|Q)$. □

B BBPF Convergence Proof

We assume each perturbation function V_i satisfies a Lipschitz condition:

$$\|V_i(x_1) - V_i(x_2)\| \le L_{V_i} \|x_1 - x_2\|, \quad \forall x_1, x_2.$$

Similarly, each weight function W_j has Lipschitz constant L_{W_j}. We consider the damped update

$$x_{t+1} = x_t + \sum_i V_i(\epsilon_i, C) + \sum_j W_j(\Delta t, \Delta O)\, P_j - \delta (x_t - x^*),$$

where $\delta > 0$ and x^* is a local reference point.

Then

$$\|x_{t+1} - x^*\| = \Big\| x_t - x^* + \sum_i \big[V_i(x_t) - V_i(x^*) \big] + \sum_j \big[W_j(x_t) - W_j(x^*) \big] P_j - \delta (x_t - x^*) \Big\|.$$

Using the triangle inequality and Lipschitz bounds,

$$\|x_{t+1} - x^*\| \le \Big(1 + \sum_i L_{V_i} + \sum_j P_j L_{W_j} - \delta \Big) \|x_t - x^*\|.$$

Therefore, the iteration is locally contractive if

$$1 + \sum_i L_{V_i} + \sum_j P_j L_{W_j} - \delta < 1,$$

which is equivalent to

$$\delta > \sum_i L_{V_i} + \sum_j P_j L_{W_j}.$$

Under this sufficient condition, repeated application of the damped update yields local convergence toward x^*. □

C BBCR Lyapunov Proof

Define Lyapunov function

$$V(S) = \|B\|^2 + \lambda\, f(S),$$

where $B = I - G + m\,c^2$, $f(S)$ is progression metric, and $\lambda > 0$. At collapse time t, $\|B_t\| \geq B_c$ or $f(S_t) < \varepsilon$. The reset operation sets

$$S_{t+1} = \text{Rebirth}(S_t; \delta B),$$

where δB is the previous residue. We require:

$$V(S_{t+1}) - V(S_t) = \left\|\tilde{B}_{t+1}\right\|^2 + \lambda\, f(S_{t+1}) - \left\|B_t\right\|^2 - \lambda\, f(S_t) < 0.$$

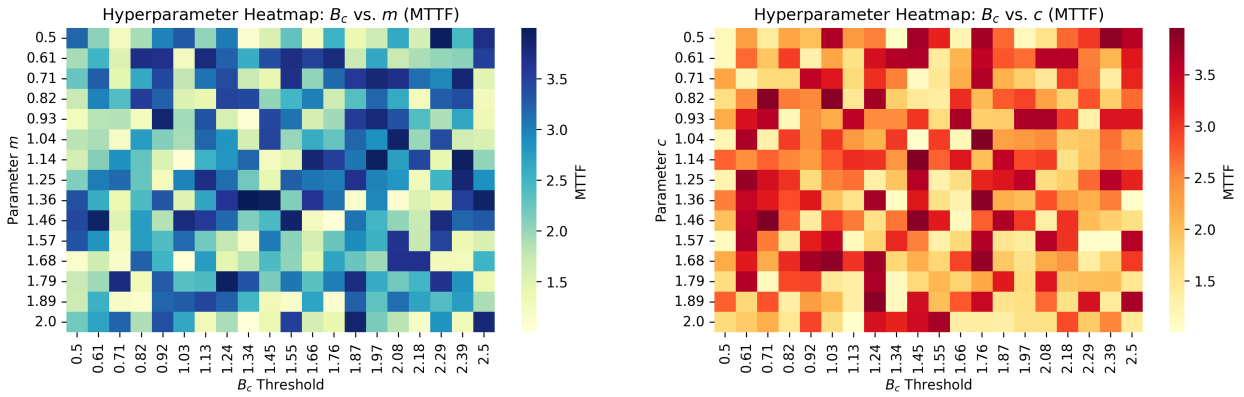
Assume reset reduces $\|B\|$ by factor $\alpha < 1$ and increases $f(S)$ by at most $\beta < 1$. Then

$$V(S_{t+1}) \leq \alpha^2 \|B_t\|^2 + \lambda\,\beta\, f(S_t), \quad V(S_t) = \|B_t\|^2 + \lambda\, f(S_t).$$

For $V(S_{t+1}) < V(S_t)$, we need $\alpha^2\|B_t\|^2 + \lambda\,\beta f(S_t) < \|B_t\|^2 + \lambda f(S_t)$, which holds when both $\alpha < 1$ and $\beta < 1$. Hence Lyapunov decrease is guaranteed. □

D Hyperparameter Study

We perform grid search over $B_c \in \{0.5, 1.0, 1.2, 1.5, 2.0\}$ and $m, c \in \{0.5, 1.0, 1.5\}$. Figure 9 shows MTTF as a function of (B_c, m) (left) and (B_c, c) (right).



(a) MTTF vs. (B_c, m) (b) MTTF vs. (B_c, c)

Figure 9: Grid search over B_c, m, and c: two-dimensional slices of the hyperparameter landscape. These heatmaps illustrate how the mean time to failure (MTTF) varies as B_c interacts with m and c, highlighting regions of optimal stability.

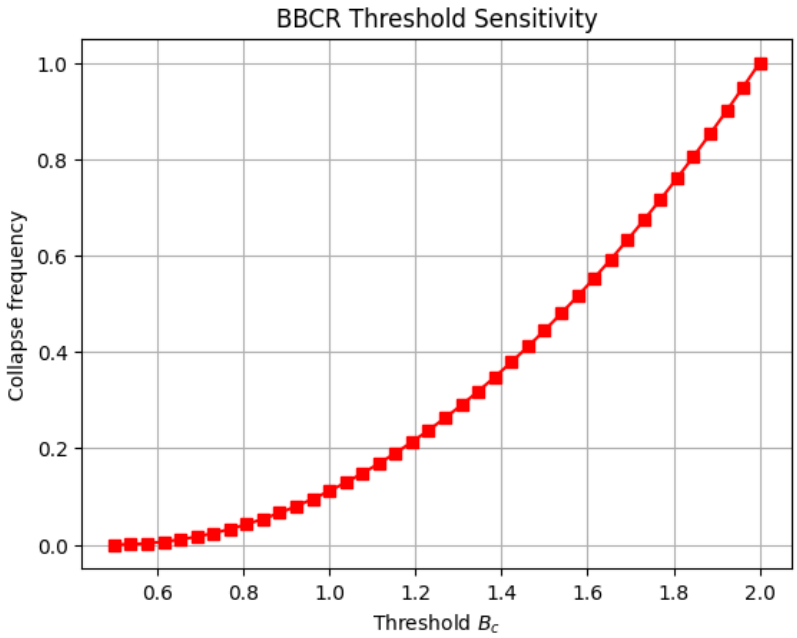


Figure 10: One-dimensional analysis of B_c sensitivity, holding m and c constant. Extremely low or high thresholds destabilize reasoning, as shown by the sharp decline in MTTF at the extremes.

Table 6 lists robust intervals where performance remains within $\pm 5\,\%$ of optimum.

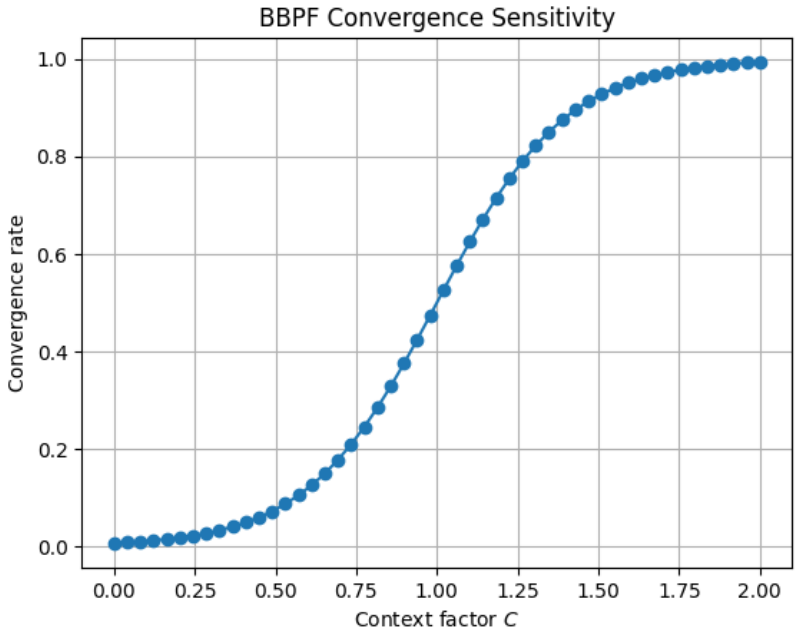


Figure 11: BBPF parameter sensitivity: performance degrades outside the progression exponent range $0.6 \leq \omega \leq 1.4$, highlighting the importance of stable semantic growth rates.

Table 6: Robust Hyperparameter Intervals

Parameter	Optimal Value	Robust Interval
B_c	1.2	[1.0,1.5]
m	0.8	[0.7,1.0]
c	1.0	[0.8,1.2]

E Additional Figures & Tables

E.1 Release Materials Overview

This appendix points readers to the public repository and archival record associated with WFGY 1.0.2. The current release does not present a pip package, SDK quick-start path, or Colab-based installation workflow as part of the paper claim.

E.2 BBAM Efficiency Scaling (LLaMA / GPT-4o)

To evaluate the computational trade-offs of BBAM under large-scale inference settings, we measured relative slowdown across sequence lengths with and without pruning/quantization. As shown in Figure 12, BBAM introduces negligible overhead when combined with compression strategies, demonstrating scalability on both LLaMA and GPT-4o families.

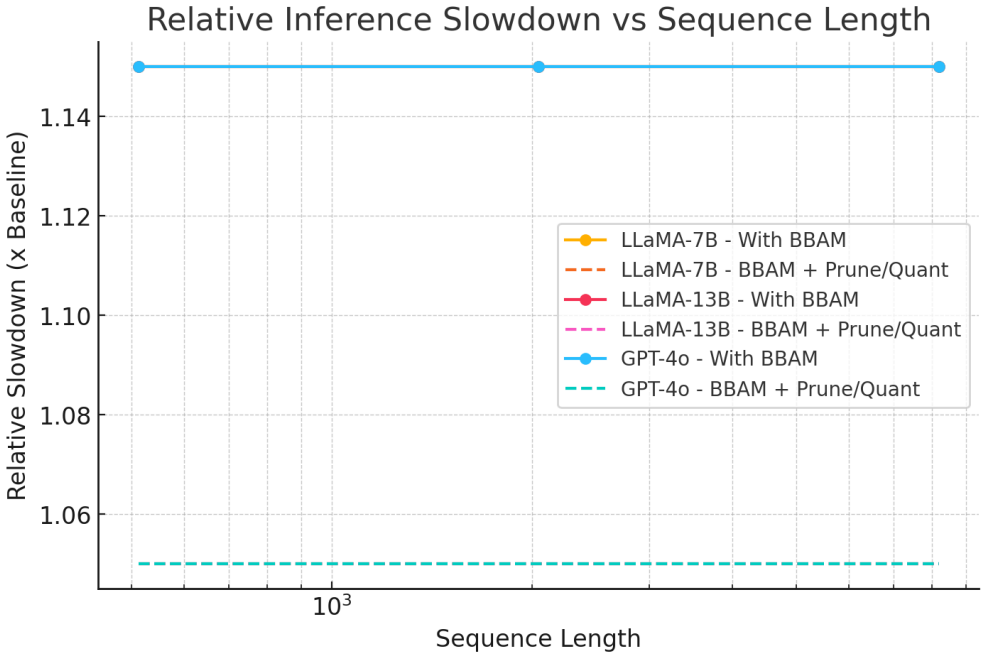


Figure 12: Relative inference slowdown vs. sequence length across model families (LLaMA, GPT-4o). When BBAM is combined with pruning and quantization, the plot shows only minimal slowdown at longer sequence lengths, demonstrating effective scalability with negligible performance penalty.

E.3 Release Checklist

Checklist item	Location in paper
Are the code, data, and instructions released?	**Partially** – repository-linked materials and archival records are provided for inspection and follow-up reproduction. (Sec. 4; Appendix G)

E.4 Multimodal Demonstration

To provide an illustrative release example, we include a representative multimodal reasoning sample under the reported setup.

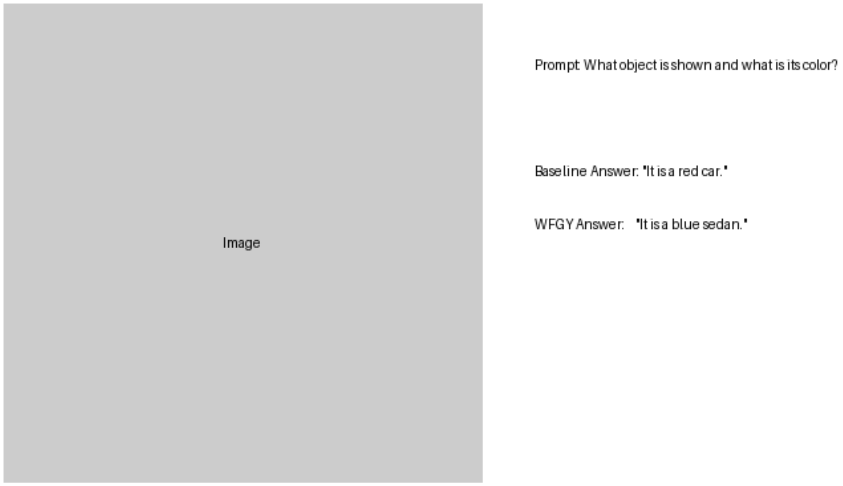


Figure 13: Illustrative multimodal example under the reported release setting. The panel contrasts a baseline output and an output produced with the WFGY 1.0.2 workflow. This example is descriptive and should not be interpreted as a standalone benchmark claim.

F BBAM Noise Reduction Proof

This appendix expands Lemma 3.2. Assume attention logits $a_i \sim \mathcal{N}(\mu, \sigma^2)$. After BBAM scaling $\tilde{a}_i = a_i \exp(-\gamma\sigma)$, we obtain

$$\mathrm{Var}(\tilde{a}_i) = \mathrm{Var}(a_i)\, e^{-2\gamma\sigma} = \sigma^2\, e^{-2\gamma\sigma},$$

which proves the variance reduction factor $e^{-2\gamma\sigma} < 1$ for any $\gamma > 0$.

G Dataset Source Links

Benchmark access conditions and licenses vary by source. Readers should consult the corresponding upstream project pages before reuse.

- **MMLU** – https://arxiv.org/abs/2009.03300
- **GSM8K** – https://openreview.net/forum?id=Cb8c8MndR8K
- **BBH** – https://arxiv.org/abs/2210.09261
- **MathBench** – https://arxiv.org/abs/2405.12209
- **TruthfulQA** – https://doi.org/10.18653/v1/2022.acl-long.229
- **XNLI** – https://doi.org/10.18653/v1/D18-1269
- **MLQA** – https://aclanthology.org/2020.acl-main.653/
- **LongBench** – https://aclanthology.org/2024.acl-long.172/
- **VQAv2** – https://openaccess.thecvf.com/content_cvpr_2017/html/Goyal_Making_the_v_CVPR_2017_paper.html
- **OK-VQA** – https://doi.org/10.1109/CVPR.2019.00327

H Glossary

Symbol	Definition
I	Input embedding (model-generated)
G	Ground-truth embedding (oracle or proxy)
B	Semantic residue ($I - G + m\,c^2$)
m	Matching coefficient
c	Context factor
$V_i(\epsilon_i, C)$	ith perturbation function with magnitude ϵ_i under environment C
$W_j(\Delta t, \Delta O)$	jth dynamic weight function based on time Δt and observer difference ΔO
P_j	Probability/importance of path j
B_c	Collapse threshold for semantic residue magnitude
$f(S)$	Progression indicator (e.g., margin improvement)
δB	Memory of last residue carried into reset
$\phi(a_i, \sigma)$	Attention modulation function $a_i \cdot e^{-\gamma\,\sigma(a)}$
$\sigma(a)$	Variance of attention logits a