# WORLD MODELS AS REFERENCE TRAJECTORIES FOR RAPID MOTOR ADAPTATION

**Carlos Stein Brito & Daniel McNamee**
Champalimaud Centre for the Unknown
Lisbon, Portugal
{carlos.stein,daniel.mcnamee}@research.fchampalimaud.org

## ABSTRACT

Deploying learned control policies in real-world environments poses a fundamental challenge: when system dynamics change unexpectedly, performance degrades until models are retrained on new data. We introduce a dual control framework that uses world model predictions as implicit reference trajectories for rapid adaptation, while preserving the policy's optimal behavior. Our method separates the control problem into long-term reward maximization through reinforcement learning and robust motor execution through rapid latent control. In continuous control tasks under varying dynamics, this achieves significantly faster adaptation compared to model-based RL baselines while maintaining near-optimal performance. This dual architecture combines the benefits of flexible policy learning through reinforcement learning with the robust adaptation capabilities of classical control, providing a principled approach to maintaining performance in high-dimensional locomotion tasks under varying dynamics.

## 1 INTRODUCTION

Model-based reinforcement learning has transformed continuous control by integrating learned world models with policy optimization (Hafner et al., 2020; Hansen et al., 2022). Such methods achieve impressive performance by using neural networks to predict future states, enabling both efficient planning through trajectory sampling and stable policy improvement through value estimation. However, deploying these systems in real-world settings reveals a fundamental limitation - when system dynamics change due to environmental variation or physical wear, both planning and value computation degrade until models are retrained on new data (Peng et al., 2018; Kumar et al., 2021).

Control theory provides powerful tools for handling changing dynamics through adaptive control, offering formal stability guarantees through Lyapunov analysis (Slotine & Li, 1991). These methods maintain performance by continuously adjusting control parameters based on tracking errors between desired and actual trajectories. However, classical control approaches rely on explicit reference trajectories and engineered cost functions, limiting their application to problems with well-defined objectives and structured dynamics models (Narendra & Annaswamy, 2012). This contrasts with reinforcement learning's ability to learn flexible policies from abstract rewards and high-dimensional observations (Sutton & Barto, 2018; Recht, 2019).

We present a framework that transforms world model predictions into reference trajectories for rapid adaptation while preserving learned policy behavior. A reinforcement learning module determines optimal trajectories in latent space, which the world model predicts forward in time to serve as references for a control module that maintains performance through trajectory tracking. This architecture is formalized through analysis of value functions, showing how they decompose into slow learning and trajectory stabilization components. Our approach provides a novel mechanism for rapid adaptation by transforming world model predictions into reference trajectories, enabling learned policies to maintain performance under changing dynamics without requiring specific robustness procedures or architectural constraints. In continuous control tasks including locomotion under varying dynamics, this achieves significantly faster adaptation than standard methods while maintaining performance.

## 2 BACKGROUND

Modern model-based reinforcement learning integrates several components through learned world models. Given observations of system state, these models learn compressed latent representations where planning and control occur (Hafner et al., 2020). TD-MPC exemplifies this approach through a normalized latent space that enables stable trajectory sampling and value estimation (Hansen et al., 2022). Previous work has explored different approaches to adaptation - Deep Model Reference Adaptive Control (Joshi et al., 2019) combined neural networks with MRAC but required complex dual architectures, while Rapid Motor Adaptation (Kumar et al., 2021) and Residual Policy Learning (Silver et al., 2018) demonstrated online adaptation but required either specific architectural choices or limited adaptation to particular types of system changes.

Adaptive control provides formal stability guarantees through Lyapunov analysis (Slotine & Li, 1991), achieving millisecond-scale adaptation by adjusting parameters based on trajectory tracking errors. However, these methods require explicit reference trajectories, structured dynamics models, and engineered cost functions (Narendra & Annaswamy, 2012). In contrast, reinforcement learning learns flexible policies directly from rewards and high-dimensional observations (Sutton & Barto, 2018), but sacrifices adaptation speed and stability guarantees. Recent work on meta-learning and domain randomization (Finn et al., 2017; Tobin et al., 2017) improves robustness but still requires extensive offline training.

Previous attempts to bridge these approaches have either restricted policies to specific forms amenable to control theory or limited adaptation to particular types of system changes (Recht, 2019). A general framework for combining the flexibility of learned models with the rapid adaptation of control theory has remained elusive. Our work addresses this gap by showing how world model predictions can serve as implicit reference trajectories, enabling classical control techniques while maintaining the benefits of learned policies.

## 3 MODEL DESIGN: IMPLICIT LATENT TRAJECTORY FOR ADAPTIVE CONTROL

Consider a continuous control Markov Decision Process $(S, A, P, R)$ with learned policy $\pi_0$ operating through a world model in latent space $z = e(s)$. We assume that the latent space captures task-relevant dynamics from observations $s$, with $V = V(z(s))$. A world model $F$ predicts future latent states conditioned on the current state and policy actions.

### 3.1 DECOMPOSITION OF THE VALUE OBJECTIVE

Locomotion involves fundamentally distinct learning processes operating at different timescales. Policy learning gradually discovers behaviors that maximize long-term reward. This process requires extensive exploration but develops robust policies for diverse tasks. In contrast, rapid adaptation maintains performance under changing dynamics without modifying the underlying policy. This process responds quickly to errors but operates within the framework of existing behaviors.

This functional separation suggests decomposing motor learning into complementary objectives. The policy learning system should discover behaviours that maximize expected value across tasks, while the adaptation system should maintain stable execution under perturbations. We formalize this approach by linking value-based learning with rapid error correction, decomposing the Taylor expansion of the value function around optimal trajectories in a task-relevant latent space $z$:

$$V(z_{t+1} + \Delta z) = V(z_{t+1}) - \frac{1}{2}\Delta z^T H \Delta z + O(\|\Delta z\|^3) \tag{1}$$

where $H = -\nabla^2 V(z_{t+1})$ is positive definite near the optimum. This decomposition suggests separating the problem into maximizing mean value through policy optimization and minimizing deviations through rapid adaptation.

### 3.2 FORWARD MODEL PREDICTIONS AS REFERENCES

We implement the functional separation through a dual architecture. The reinforcement learning module uses soft actor-critic to learn a base policy, $a_0 = \pi_0(z)$, that optimizes the mean value, as in

classic RL models. We maintain a forward model $F$ predicting future latent states:

$$\hat{z}_{t+1} = F(z_t, \pi_0(z_t)) \tag{2}$$

Our framework builds on model-based reinforcement learning but changes how world model predictions drive behavior. A conceptual novelty is that we interpret the forward model predictions as target states. Both the forward model and controller share the same error function measuring discrepancy between predicted and actual states:

$$\mathcal{L} = \|\hat{z}_{t+1} - z_{t+1}\|^2 \tag{3}$$

However, they minimize this error in opposing ways. The forward model adapts its predictions to match observations, following the standard gradient to improve predictions. In contrast, the control module adapts actions to make the system behave as predicted.

### 3.3 Adaptive control gradients

This approach inverts the standard relationship between models and control. Classical adaptive control assumes reference trajectories and adapts a controller to track them. Model-based RL learns models that predict actual outcomes and uses them for planning. Our framework generates reference trajectories directly from world model predictions while adapting control to maintain their validity under changing dynamics.

The control policy updates follow a modified gradient computation that reflects this inversion. Rather than updating predictions to match observations, we update actions to make observations match predictions:

$$\theta_c \leftarrow \theta_c - \eta_c \left(-\frac{\partial \mathcal{L}}{\partial a_0}\right)\left(\frac{\partial a_c}{\partial \theta_c}\right) \tag{4}$$

The update function leverages gradients through the world model to determine how actions should change to reduce prediction error, inverting the standard approach to model learning. This differs fundamentally from standard practice where gradients flow from predictions to parameters. The control module instead treats predictions as fixed targets and adapts actions to achieve them.

The total action combines the base policy with these corrections:

$$a_t = \pi_0(z_t) + \pi_c(z_t) \tag{5}$$

Operating in the world model's latent space provides two key benefits. First, it ensures the control module focuses adaptation on task-relevant features captured by the learned representation. Second, it provides an interface between the RL policy operating on compressed latent states and the control module maintaining prediction consistency.

**Algorithm 1** World Model Reference Adaptive Control

**Require:** Trained policy $\pi_0$, encoder $e$, world model $F$
**Ensure:** Adapted control policy $\pi_c$
  Initialize $\pi_c$
  **while** not done **do**
    $z_t \leftarrow e(s_t)$
    $a_0 \leftarrow \pi_0(z_t)$; $a_c \leftarrow \pi_c(z_t)$
    Execute $a_t = a_0 + a_c$, observe $s_{t+1}$
    $z_{t+1} \leftarrow e(s_{t+1})$
    $\hat{z}_{t+1} \leftarrow F(z_t, a_0)$
    $e_t \leftarrow z_{t+1} - \hat{z}_{t+1}$
    $\theta_c \leftarrow \theta_c - \eta_c \left(-\frac{\partial \|e_t\|^2}{\partial a_0}\right) \frac{\partial a_c}{\partial \theta_c}$
  **end while**

## 4 Theoretical Guarantees

The modularity of the dual system allows for theoretical guarantees of robustness to perturbations in trajectory and performance:

**Assumption 4.1** (System Properties)**.** *The system satisfies:*

1. $\|\partial F/\partial a\| \leq L$ *(Lipschitz control)*
2. $\sigma_{min}(\partial F/\partial a) \geq \alpha > LP$ *(control authority)*
3. $\|F(z,a) - f(z,a)\| \leq \epsilon$ *(model accuracy)*

*where $P$ bounds external perturbations.*

**Theorem 4.2** (Control Error)**.** *Under Assumption 1, the control law $a_c = -\eta(\partial F/\partial a)^T e(t)$ achieves:*

$$\|e(t)\| \leq \gamma^t \|e(0)\| + \sqrt{\epsilon^2 + \frac{P^2}{\alpha^2}} \tag{6}$$

*where $\gamma = (1 - \eta\alpha^2 + \eta L^2) < 1$ for $\eta < 1/L^2$.*

**Theorem 4.3** (Value Bounds)**.** *If the error bound is sufficiently small, the value function satisfies:*

$$V(z^*) - V(z) \leq \frac{H_M}{2}\left(\epsilon^2 + \frac{P^2}{\alpha^2}\right) \tag{7}$$

*where $H_M$ bounds the eigenvalues of $-\nabla^2 V$ near optimal trajectories.*

These results show how world model accuracy ($\epsilon$), control authority ($\alpha$), and perturbation magnitude ($P$) determine performance bounds, with quadratic scaling reflecting the natural structure of value functions around optimal trajectories. The proofs use standard Lyapunov techniques (see Appendix).

## 5 Simulations

### 5.1 Dual-model Design

Our approach builds on model-based reinforcement learning methods such as Dreamer and TD-MPC (Hafner et al., 2020; Hansen et al., 2022), primarily to leverage their task-related encoders. In particular, TD-MPC uses latent, reward and value predictions to learn the encoder and forward models. We reuse these pre-trained representations to focus on the novel control mechanism. As baseline policy, we use TD-MPC without planning, equivalent to a variant of the Soft-Actor Critic model (SAC) (Haarnoja et al., 2018). In simple environments where an encoder is unnecessary, such as the point-mass task, we use direct state observations ($z_t = s_t$). In these cases, SAC or any alternative policy, including non-RL-based controllers, can be used as baseline policy, and the forward model is learned through latent prediction.

The control gradient propagation differs from standard error minimization. The forward model gradient is computed with respect to $a_0$, and applying it directly to $a_c$ would reinforce deviations instead of correcting them. The gradient sign is inverted before it is backpropagated to ensure the corrective action counteracts the error (Fig. 1-A).

### 5.2 Perturbation Experiments

We evaluate adaptation by introducing perturbations $p(t)$ in the action space, with an effective actuator $a_{eff} = (a_0 + a_c) \cdot (1 + p)$, and measuring the controller's ability to compensate for them (Fig. 1-B). Step perturbations involve sudden changes in the action signal at specific intervals, while slow perturbations introduce gradual, non-stationary shifts that mimic actuator miscalibration. These perturbations allow us to analyze how the controller reacts to both abrupt and progressive deviations.

The experiment consists of two phases. In the first phase, a policy is trained or provided without perturbations, and a forward model is learned using its trajectories. This phase establishes the baseline model of the system's behavior under normal conditions. In the second phase, perturbations are introduced while simultaneously activating the controller. The controller uses the learned forward model to adjust its outputs in response to the deviations introduced by the perturbations.

To assess adaptation performance, we measure the drop in task performance caused by the perturbations and track the forward model error. The latter serves as an implicit measure of latent trajectory deviation, reflecting how well the system follows the predictions of the forward model.

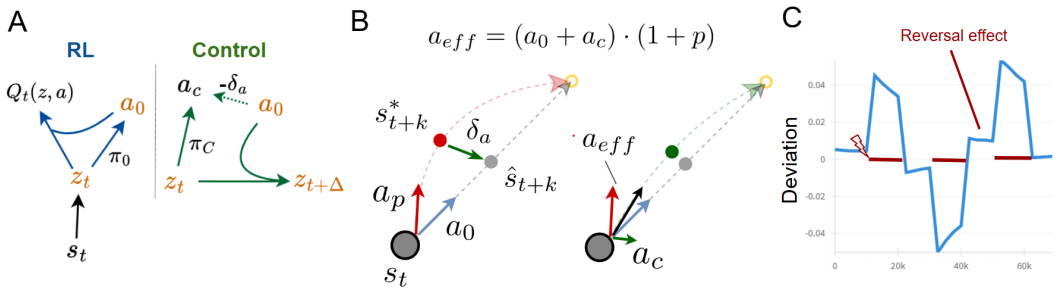


Figure 1: (A) Network architecture showing the reinforcement learning policy (blue) and adaptive control modules (green), with interface variables in orange. Each transformation is implemented as a two-hidden-layer MLP. (B) Illustrative simulation of the adaptive control mechanism for a 2D pointmass task (without encoder, $z = s$). When actuators are perturbed, the trajectory deviates from the predicted future states $\hat{s}_{t+k}$ under the base policy actions $a_0$. This error triggers an update to generate corrective actions $a_c$. (C) Under alternating directional perturbations (red), the controller corrects deviations from the optimal trajectory, exhibiting characteristic after-effects when perturbations are removed.

### 5.3 Correction for Trajectory Deviations

The point mass system demonstrates the core interaction between policy and adaptation modules. Under angular perturbation, the base policy's trajectories systematically deviate from target while the world model maintains predictions of intended paths (Fig. 1B). The control module uses these predictions as references to generate corrective actions, recovering performance without modifying the underlying policy.

The system exhibits characteristic aftereffects when perturbations are removed (Fig. 1C). Initial overcorrection in the opposite direction indicates adaptation through internal model formation rather than reactive control. This validates the method's ability to learn and compensate for systematic changes in dynamics while preserving the original policy.

### 5.4 Robust Motor Control

The Walker2D environment demonstrates how adaptation can operate effectively in learned latent space. Under step perturbations to actuator gains, the control module rapidly reduces the error between predicted and actual latent states (Fig. 2A). As the latent prediction error decreases, task performance improves correspondingly, validating that world model predictions in latent space provide effective references for adaptation.

For non-stationary environments, we apply filtered noise to actuator gains, simulating gradual changes in dynamics (Fig. 2B). The dual architecture maintains a performance of 180.4 compared to the non-perturbed baseline of 185.8. Standard RL adaptation achieves 158.3, with slower recovery due to the sequential nature of updates - the forward model must first adapt to new dynamics, followed by value function updates, before the policy can adjust. The fixed baseline policy degrades to 109.1, highlighting the need for adaptation. The world model predictions provide a stable reference for continuous adaptation even as dynamics evolve, enabling rapid corrections without policy retraining.

### 5.5 Naturalistic Movement

The 17-actuator Humanoid environment bridges artificial and biological motor control principles (Fig. 2C). When perturbed, the system exhibits patterns analogous to ataxia - a condition where damage to internal models in biological systems leads to poor movement coordination and timing (Bastian, 2011). This manifests as increased variability and loss of synchronization between joints.

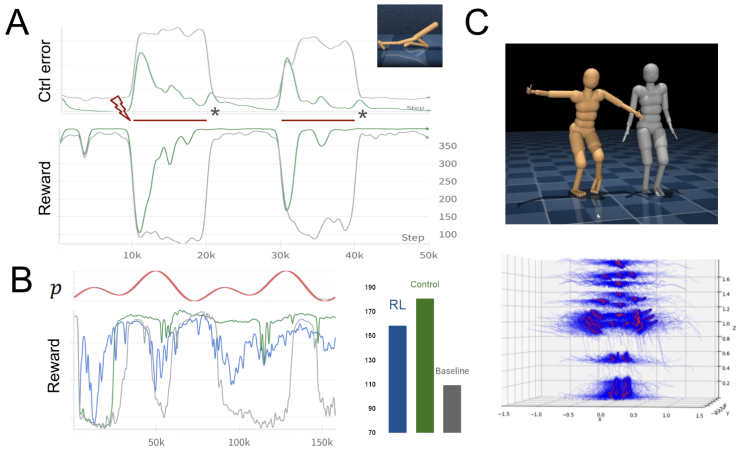


Figure 2: (A) For step motor perturbations (red) to the Walker2d agent (inset), the controller adapts to significantly reduce the control error (green; no adaptation in grey), improving the performance (below), with a small reversal effect (stars). (B) For time-varying perturbations (red, above), the dual-model (green) shows faster adaptation and better performance than the RL module alone (blue; no adaptation in grey). (C) Humanoid models of normal gait and ataxia-like behavior. Sample trajectories allow a quantitative analysis of motor variability and the adaptive control of the latent mean trajectory.

The control module restores coordinated movement by tracking latent predictions that capture relationships across multiple joints. This enables coherent adaptation - when one limb's dynamics change, the controller adjusts multiple actuators to maintain balance and symmetry. The dual architecture thus provides a computational framework bridging artificial and biological motor control, where policy learning encodes structured movement patterns while adaptation maintains robust execution under changing conditions.

## 6 Discussion

This work demonstrates how world model predictions can serve as implicit reference trajectories for rapid adaptation. Rather than using world models primarily for planning (Hansen et al., 2022) or policy modification (Kumar et al., 2021; Luo et al., 2020), our approach enables rapid adaptation through error correction in latent space without modifying the underlying policy. The theoretical decomposition of value functions into slow learning and rapid adaptation components provides a principled foundation for this architecture.

Our framework addresses a fundamental challenge in motor control: classical adaptive methods provide stability guarantees but require hand-designed reference trajectories, while learned policies enable flexible behavior but lack formal robustness properties. By deriving references directly from world model predictions, our approach maintains the guarantees of control theory while preserving the flexibility of learned policies. The value function decomposition establishes concrete bounds on performance under varying dynamics.

The dual timescales observed in our framework parallel mechanisms of biological motor control (Franklin & Wolpert, 2011). Our value function decomposition into slow learning and rapid adaptation mathematically formalizes the complementary learning systems observed in motor cortex and cerebellum (Wolpert et al., 1998). The humanoid experiments demonstrate this directly - when perturbed, the system exhibits increased motor variability and loss of joint coordination similar to cerebellar ataxia, which our control module corrects through predicted movement patterns. These computational parallels and characteristic adaptation signatures suggest common organizational principles between biological and artificial motor control that can guide the development of more robust control architectures (Lee et al., 2020).

Several directions for future work emerge naturally from this framework. First, maintaining prediction accuracy for complex environments requires better uncertainty estimation in world models. Second, developing simultaneous learning of policies and adaptation mechanisms would enable continuous improvement during deployment, rather than treating them as separate phases. Finally, extending the framework to handle environmental perturbations beyond actuator dynamics would broaden its practical applications. The success of the current approach suggests that leveraging world model predictions as control targets may be a general principle for robust deployment of learned behaviors. While this paper establishes the theoretical framework and demonstrates proof-of-concept in simulation environments, specific implementations for diverse robotic platforms present additional engineering challenges to be addressed in future work.

## REFERENCES

Amy J Bastian. Moving, sensing and learning with cerebellar damage. *Current Opinion in Neurobiology*, 21(4):596–601, 2011.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pp. 1126–1135. PMLR, 2017.

David W Franklin and Daniel M Wolpert. Computational mechanisms of sensorimotor control. *Neuron*, 72(3):425–442, 2011.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning*, pp. 1861–1870. PMLR, 2018.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. *Proceedings of the 39th International Conference on Machine Learning*, 2022.

Girish Joshi, Jasvir Virdi, and Girish Chowdhary. Deep model reference adaptive control. In *AIAA SciTech Forum*. AIAA, 2019.

Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots. In *Robotics: Science and Systems*, 2021.

Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. In *Science robotics*, volume 5. American Association for the Advancement of Science, 2020.

Jianlan Luo, Eugen Solowjow, Chengjun Wen, José Agustín Ojea, Alice M Agogino, Aviv Tamar, and Pieter Abbeel. Learning adaptive context-modulated policies for robust manipulation. In *Conference on Robot Learning*, pp. 1206–1217. PMLR, 2020.

Kumpati S Narendra and Anuradha M Annaswamy. *Stable Adaptive Systems*. Courier Corporation, 2012.

Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3803–3810, 2018.

Benjamin Recht. A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*, 2:253–279, 2019.

Tom Silver, Kelsey Allen, Josh Tenenbaum, and Leslie Kaelbling. Residual policy learning. *arXiv preprint arXiv:1812.06298*, 2018.

Jean-Jacques E Slotine and Weiping Li. *Applied Nonlinear Control*. Prentice Hall, 1991.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT Press, Cambridge, MA, 2 edition, 2018.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 23–30. IEEE, 2017.

Daniel M Wolpert, R Chris Miall, and Mitsuo Kawato. Internal models in the cerebellum. *Trends in Cognitive Sciences*, 2(9):338–347, 1998.

## A THEORETICAL ANALYSIS

We provide proofs for the main theoretical results, showing how control error bounds lead to value function guarantees.

**Assumption A.1** (System Properties)**.** *The dynamics satisfy:*

1. $\|\partial F/\partial a\| \leq L$ *(Lipschitz control)*
2. $\sigma_{min}(\partial F/\partial a) \geq \alpha > LP$ *(control authority)*
3. $\|F(z,a) - f(z,a)\| \leq \epsilon$ *(model accuracy)*
4. $\|p(t)\| \leq P$ *(bounded perturbation)*

The condition $\alpha > LP$ ensures sufficient control authority relative to perturbations - the controller must overcome both the Lipschitz growth ($L$) and perturbation magnitude ($P$).

**Theorem A.2** (Control Error Bounds)**.** *Under Assumption 1, the control law* $a_c = -\eta(\partial F/\partial a)^T e(t)$ *with* $\eta < 1/L^2$ *achieves:*

$$\|e(t)\| \leq \gamma^t \|e(0)\| + \sqrt{\epsilon^2 + \frac{P^2}{\alpha^2}} \tag{8}$$

*where* $\gamma = (1 - \eta\alpha^2 + \eta L^2) < 1$.

*Proof.* The error evolves as:

$$e(t+1) = \underbrace{F(z_t, a_t) - F(z_t, a_0)}_{\text{control effect}} + \underbrace{F(z_t, a_0) - f(z_t, a_t)}_{\text{model error}} + p(t) \tag{9}$$

For the control effect:

$$-F(z, a_0 + a_c) = -\eta(\partial F/\partial a)(\partial F/\partial a)^T e(t) \quad \text{(first order)} \tag{10}$$

$$\|(\partial F/\partial a)(\partial F/\partial a)^T\| \geq \alpha^2 \quad \text{(by min singular value)} \tag{11}$$

$$\|F(z, a_0 + a_c) - F(z, a_0)\| \leq (1 - \eta\alpha^2 + \eta L^2)\|e(t)\| \quad \text{(Lipschitz)} \tag{12}$$

The model error satisfies:

$$\|F(z, a_0) - f(z, a_t)\| \leq \epsilon + L\|a_c\| \leq \epsilon + \frac{LP}{\alpha} \tag{13}$$

Therefore:

$$\|e(t+1)\| \leq \gamma\|e(t)\| + \epsilon + \frac{P}{\alpha} \tag{14}$$

By condition (2) of Assumption 1 and $\eta < 1/L^2$:

$$\gamma = 1 - \eta\alpha^2 + \eta L^2 < 1 - \eta(LP)^2 + \eta L^2 < 1 \tag{15}$$

The bound follows from solving this recurrence, using the fact that for positive $a, b$:

$$(a+b)^2 \leq 2(a^2 + b^2) \tag{16}$$

□

**Theorem A.3** (Performance Guarantees)**.** *If $\sqrt{\epsilon^2 + P^2/\alpha^2} < \delta$ where $\delta$ bounds the region of quadratic approximation for $V$, then:*

$$V(z^*) - V(z) \leq \frac{H_M}{2}\left(\epsilon^2 + \frac{P^2}{\alpha^2}\right) \tag{17}$$

*where $H_M$ bounds the eigenvalues of $-\nabla^2 V$.*

*Proof.* Around optimal trajectories, Taylor expansion gives:

$$V(z^* + \Delta z) = V(z^*) - \frac{1}{2}\Delta z^T H \Delta z + R(\Delta z) \tag{18}$$

where $|R(\Delta z)| \leq C\|\Delta z\|^3$ for some $C > 0$.

The prediction error directly bounds state deviation:

$$\|\Delta z\| = \|z - z^*\| \leq \|e(t)\| \leq \sqrt{\epsilon^2 + \frac{P^2}{\alpha^2}} \tag{19}$$

When this is less than $\delta$, the quadratic term dominates since:

$$\frac{|R(\Delta z)|}{\|\Delta z\|^2} \leq C\|\Delta z\| \to 0 \tag{20}$$

The bound follows from $\lambda_{max}(H) = H_M$ and the error bound. □

These results establish quantitative bounds linking world model accuracy ($\epsilon$), control authority ($\alpha$), and perturbation magnitude ($P$) to performance. The quadratic scaling reflects the natural structure of value functions around optimal trajectories.