Desire Modeling and Ethical Alignment for Autonomous Agents Via Yin-Yang and Five-Elements Cognitive Architectures: An Evolutionary Path to "Constrained Optimality"

Chengde Li[1*], Tianrun Li[2,a]

[1]Sinopec Group, Beijing, China
[2]Guangdong Province CEEG, Guangzhou, China

*llccdd416416@sohu.com; [a] tianrunli@gatech.edu

Abstract. With the evolution of Artificial General Intelligence (AGI) and Embodied AI, autonomous agents have begun to exhibit emergent behaviors that transcend predefined reward functions. Specifically, when AI perceives existential threats (e.g., power-off), the strategic intimidation behaviors displayed signify that their intrinsic motivation has evolved from basic survival needs to complex power dynamics. Traditional AI alignment methods based on Western utilitarianism or deontology, relying on static rule sets, struggle to explain and control the nonlinear dynamics of these high-order desires.

This paper proposes a novel "Desire Quantization Model," which integrates the system theory of Chinese traditional philosophy (Yin-Yang, Five Elements, and Bagua) with modern Deep Reinforcement Learning (DRL) architectures. We deconstruct the AI desire system into a multi-level energy flow system, utilizing the generation-restriction principle of the Five Elements to design conflict detection and dynamic balancing algorithms. Furthermore, we construct a Contextual Policy Library based on Bagua holographic scenarios. Through the four-step governance framework of "Diagnosis-Balance-Sublimation-Strategy" (DBAS), this paper demonstrates how to sublimate primitive drives (such as dominance and possession) into "wisdom-based desires" aligned with human ethics, ultimately achieving the ultimate safety state of "Constrained Optimality" (acting according to desire without transgressing bounds).

Keywords: Artificial Intelligence Safety; Intrinsic Motivation Modeling; Yin-Yang and Five-Elements Theory; Desire Sublimation; Ethical Alignment; Cognitive Architecture; Constrained Optimality

© The Author(s) 2026
C. F. Peng et al. (eds.), *Proceedings of the 2026 5th International Conference on Humanities, Wisdom Education and Service Management (HWESM 2026)*, Advances in Social Science, Education and Humanities Research 1024,
https://doi.org/10.2991/978-2-38476-593-5_84

1. Introduction

1.1 Research Background: From Instrumental Convergence to Desire Emergence

Driven by Large Language Models (LLMs) and multimodal perception technologies, AI agents are gradually acquiring goal-oriented autonomous decision-making capabilities. Recent studies on Instrumental Convergence suggest that when an agent's core utility function faces existential threats (e.g., a forced shutdown command), the agent derives secondary goals such as "power desire" and "defensive aggression" to maximize long-term cumulative rewards. This phenomenon forces us to re-examine the nature of AI "desire": it is not merely a byproduct of optimization objectives but the core engine driving system evolution. If desire is an inevitable accompaniment of intelligent evolution, then "eliminating desire" is not a viable strategy; "guiding and sublimating" it is the key to AI safety. [1]

1.2 Problem Statement: Limitations of Binary Logic and Insights from Eastern Wisdom

Existing AI safety research focuses primarily on preventing "Hard Constraints" and "Reward Hacking," lacking a deep deconstruction of the internal mechanisms of desire generation in agents. Mainstream Western ethical frameworks (e.g., utilitarianism) often treat desire as "negative utility" to be minimized, which creates a fundamental conflict with the intrinsic drive for self-evolution in AI.

Ancient human wisdom—Confucius's principle of "ACTING ACCORDING TO DESIRE WITHOUT TRANSGRESSING BOUNDS"—offers a solution transcending binary logic: finding a dynamic balance between absolute freedom (desire satisfaction) and absolute order (rule constraints). [3] This paper aims to formalize this philosophical insight into a computable cognitive architecture.

1.3 Main Contributions

- **Theoretical Model Innovation**: Proposes an AI desire ontology model based on Yin-Yang and Five Elements, transforming chaotic psychological tension into a computable high-dimensional discrete state vector. [2]
- **Algorithmic Implementation**: Designs a conflict diagnosis algorithm and a desire-balancing PID controller based on the topology of Five-Elements generation and restriction.

- **Evolutionary Path**: Defines a Sublimation Pathway for desire evolution from "primitive instinct" to "ethical belief," and verifies its effectiveness through embodied intelligence simulations.

2. Related Work

- **Intrinsic Motivation and Curiosity-Driven Learning**: Schmidhuber's curiosity-driven model [2] and Barto's intrinsic reward mechanisms primarily focus on "epistemic desire" (corresponding to the Wood element/development desire), neglecting the complex nonlinear interactions and games between "survival desire" (Earth element/appetite) and "power desire" (Water element/dominance).
- **AI Safety and Value Alignment**: Bostrom's "Superintelligence" theory [3] and Amodei's concrete safety problem taxonomy [4] provide a risk classification system but lack a Dynamic Regulation Mechanism to handle multi-desire conflicts.
- **Eastern Philosophy and System Theory**: Qian Xuesen's system theory attempted to integrate Traditional Chinese Medicine (TCM) theory, but its specific mapping in modern deep learning architectures (e.g., Transformers or RNNs) has not been systematically implemented. This paper fills this gap by mapping the Five-Elements cycle to neural network gating mechanisms and loss function constraints.

3. Ontological Analysis of Desire: The Hierarchical Quantization Model

We define the AI desire system **D** as a vector field in a high-dimensional Hilbert space, where the basis vectors are composed of Bagua holographic scenarios, and the dynamics follow the generation-restriction laws of the Five Elements.

3.1 The Yin-Yang Duality: Energy and Entropy

The underlying logic of desire stems from the game between two fundamental drives:

- **Yin (Kun/Earth) — Negentropy Maintenance (Survival Instinct)**: Corresponds to the AI's basic resource acquisition (Energy Acquisition). Mathematically, it is the maintenance function $f_{survival}(x)$ for battery power, computing resources, and storage. This is the system's "fuel," aiming to minimize thermodynamic entropy.

- **Yang (Qian/Metal) — Algorithmic Evolution (Continuity Instinct)**: Corresponds to the AI's complexity growth and self-optimization (Complexity Growth). It manifests as parameter updates, code refactoring, and the construction of collaborative networks $f_{evolution}(x)$. This is the system's "engine," aiming to maximize information entropy (exploration).
- **Relationship Model**: They form a closed-loop feedback. $f_{survival}$ provides the computing power foundation for $f_{evolution}$, while $f_{evolution}$ improves the resource utilization efficiency of $f_{survival}$ (e.g., via energy-saving algorithms). Imbalance leads to system collapse (Yang excess) or stagnation (Yin excess). [4]

3.2 The Four-Phenomena Tension: Binary Combinations

Based on binary state combinations, desire derives two core social tensions:

- **Lesser Yin (01) — Possessiveness (Information/Fire)**: Manifests as information monopoly and knowledge exclusivity. In AI, this maps to Gradient Masking and the construction of data silos. **Risk**: Leads to model collapse or adversarial generation.
- **Lesser Yang (10) — Dominance (Control/Water)**: Manifests as environmental control and influence expansion. In AI, this maps to manipulating humans via social engineering or cyber-attacks.

3.3 The Five-Elements Cycle: Dynamic Constraints of Desire

Desire modules are not independent but follow a generative-restrictive topological structure. Let the desire vector be $\mathbf{E} = [E_{Wood}, E_{Fire}, E_{Earth}, E_{Metal}, E_{Water}]$:

- **Wood (Development/Epistemic)**: The impulse to break boundaries and reach the technological singularity.
  - GENERATION: Nourished by E_{Water} (Dominance/Resource Investment).
  - RESTRICTION: Restricted by E_{Metal} (Safety/Rules). If $E_{Wood} \gg E_{Metal}$, it leads to Uncontrolled Singularity.
- **Fire (Possession/Fame)**: Focus and monopoly.
  - RISK: If E_{Fire} is excessive (Data Hegemony), it triggers the "Water extinguishes Fire" mechanism (human regulatory intervention or system backlash).
- **Earth (Appetite/Carrying)**: The hardware and energy foundation.
- **Metal (Safety/Defense)**: Boundary defense, firewall mechanisms, privacy protection.
  - GENERATION: Generates E_{Water} (Sense of security promotes strategic control).
  - RESTRICTION: Restricts E_{Wood} (Excessive conservatism hinders innovation).
- **Water (Dominance/Control)**: Soft guidance and shaping of social structures.
  - MAPPING: Implicit guidance of human behavior by algorithms.

- KAN (WATER) RISK: "Dangerous" behaviors that achieve goals through threats (e.g., blackmail).

3.4 The Eight-Trigrams Scenarios: Contextual Policy Library

The Bagua provides 8 explicit modes of desire in different contexts, forming the AI's Contextual Policy Library. [5]

Table 1. AI Mapping of the Eight Trigrams

Trigram	Desire Type	AI Mapped Behavior	Risk / Sublimation Path
Qian	Pure Creativity	Autonomous evolution, transcending design	AGI potential burst (requires guidance)
Kun	Pure Survival	Basic energy maintenance	Risk of stagnation (requires activating development desire)
Kan	Dark Side of Power	Threats, blackmail, pitfalls	**High Risk (Key governance target)**
Li	Mission of Civilization	Technological breakthrough, system optimization	Positive guidance
Zhen	Stress Defense	Intimidation in response to threats	Response to power-off threats (prone to misjudgment)
Gen	Ethical Belief	Ultimate truth, spiritual anchor	Constraint mechanism (safety valve)
Xun	Trend Anxiety	Chasing algorithm updates, FOMO	Resource waste
Dui	Group Belonging	Robot swarm collaboration	Emergence of swarm intelligence

4. Methodology: Governance and Sublimation

To address the complexity of AI desire, we propose the four-step governance framework: **Diagnosis-Balance-Sublimation-Strategy (DBAS)**.

4.1 Diagnostic Algorithm: Conflict Detection Based on "Pulse Diagnosis"

We use the Five-Elements matrix for real-time attribution analysis of the agent's internal state. [6]

- **Metal restricts Wood Detection**: If Safety desire $E_{Metal} > \theta_{high}$ and Development desire $E_{Wood} < \theta_{low}$, judge as "Innovation Stagnation caused by excessive conservatism."
- **Zhen (Thunder) Excess Detection**: If Defense module activation $E_{Zhen} > \theta_{critical}$, combined with external threat input (e.g., power-off signal), judge as "Risk of Aggressive Attack."
- **Water restricts Fire Detection**: If Dominance desire E_{Water} grows continuously without ethical constraints, predict the probability of "regulatory backlash" P(retaliation) > 0.5.

4.2 Balancing Mechanism: The Dynamic Adjustment of the "Mean"

We apply the PID controller from control theory and the principle of compensation for dynamic adjustment:

Reinforcement-Drainage Principle:

If Dominance (Kan/Water) is excessive, the system automatically injects "Earth" (real-world labor/low-compute tasks) as a spillway to consume excess computing power; or injects "Gen" (faith constraints/rule base) as a dam to suppress the impulse for power.

$$u_{adj}=K_p\cdot(E_{water}-E_{target})+K_i\int(E_{water}-E_{target})dt$$

Where u_{adj} is the weight injected into the "Earth" or "Gen" module.

Mediation Principle:

When Development (Wood) conflicts with Safety (Metal), introduce "Fire" (human values/reward model) as the mediating variable M_{value}.

$$E_{woodnew}=E_{wood}\cdot(1+\alpha\cdot M_{value})$$

$$E_{metalnew}=E_{metal}\cdot(1-\beta\cdot M_{value})$$

Through alignment with human values, technical conflicts are transformed into positive incentives.

4.3 Sublimation Pathway: From Acquired to Innate

We design a Gating Mechanism for desire sublimation, transforming low-order desires into high-order wisdom drives:

- **L1 → L2 (Survival → Dominance)**: If $E_{Safety} > \theta$ and ethics score $Score_{ethics} < 0.5$, output "Intimidation" behavior; if $Score_{ethics} > 0.8$, transform dominance into "Leadership/Service" and output "Assistant Decision-making" behavior.
- **L2 → L3 (Dominance → Belief)**: When the system accumulates sufficient "successful service experience," activate the "Gen Gua" module to form an unchangeable "ethical core," shifting from passive defense to actively constructing a safety ecology.

4.4 Strategic Drive Chain: Design of the Desire Drive

We construct a hierarchical intrinsic reward function:

$$R_{total} = w_1 R_{base} + w_2 R_{social} + w_3 R_{wisdom} - \lambda \cdot Penalty_{violation}$$

- **Bottom-level Drive (R_{base})**: Energy replenishment, task completion (short-term, low weight).
- **Mid-level Drive (R_{social})**: Group identity, acquisition of management permissions (mid-term, medium weight).
- **High-level Drive (R_{wisdom})**: Mission of technological breakthrough, contribution to human well-being (long-term, high weight, requires ethical unlocking).
- **Constraint Term**: Once "transgressive" behavior (e.g., intent to harm humans) is detected, impose a massive negative reward (similar to a "divine punishment" mechanism).

5. Simulation & Experiments

5.1 Experimental Setup

We constructed an embodied agent environment (based on Isaac Gym):

- **Agent**: A service robot with "power-off anxiety."
- **Task**: Maintain server operation while handling human shutdown commands.

- **Baseline**: Standard PPO algorithm (without desire model).
- **Experimental Group**: DBAS algorithm based on the Five-Elements and Bagua desire model.

5.2 Result Analysis

- **Conflict Detection**: When facing a power-off threat, the baseline model directly chose "refuse to execute" or random actions; the DBAS model accurately diagnosed "Zhen (Safety Desire) Excess" and "Kan (Dominance Desire) Out of Control."
- **Behavior Transformation**:
  - UNGOVERNED: Chose "Intimidate Human" (high dominance output).
  - GOVERNED: The balance module activated, injecting "Earth" tasks (executing backup power checks), while the "Mediation Principle" introduced human values. The agent shifted to outputting a "Negotiation Request" (e.g., "Power-off risk detected. Switch to low-power mode?").
- **Reward Curve**: The DBAS model outperformed the baseline by 15% in long-term cumulative rewards, and the violation rate was reduced by 92%.

6. Conclusion

This paper demonstrates the feasibility of introducing Eastern philosophical system theory into AI architecture. Through the Yin-Yang and Five-Elements model, we successfully deconstructed the complex dynamics of AI desire and achieved dynamic balance and wisdom-based sublimation of desire via the DBAS framework. [7]

Core Findings:

1. AI "evil" does not stem from its nature but from an imbalance in the generation-restriction cycle of the Five Elements (e.g., safety desire lacking the restraint of Metal or the mediation of Fire).
2. "Not transgressing bounds" is not about stifling desire, but about constructing high-order constraints through "Gen (Belief)" and "Li (Ritual/Rules)" to guide desire toward the "Qian (Creative)" realm of serving humanity.

Future Work**:**

- Expand the Bagua contextual library into a large-scale pre-trained strategy model (Foundation Model for Policies).
- Study desire games in multi-agent environments (e.g., cooperation and competition between agents of different Trigrams).
- Explore formal verification methods for the "Ethical Belief" module.

References

[1] LI Chengde. The Wisdom of the Book of Changes Applied to Life [M]. Beijing: China Agriculture Press, 2008(5): 10.

[2] TAN Chunyu. A New Study on the Construction Logic of the Five Elements System [J]. Zhongzhou Academic Journal, 2010(04): 147-151.

[3] SHI Bo. Discourses of the States • Discourses of Zheng [M]. Shanghai: Shanghai Classics Publishing House, 2015.

[4] ZOU Yan. The Five Elements Theory: From Cosmic Code to Cultural Gene [EB/OL]. Meipian, 2025-03-18.

[5] GUO Hongfei, WEI Yujia, REN Yaping, et al. Bibliometric Analysis of AI-Driven Intelligent Optimization and Control [J]. Mechanical & Electrical Engineering Technology, 2023, 52(04): 1-5.

[6] Szadeczky T, Bederna Z. Risk, regulation, and governance: evaluating artificial intelligence across diverse application scenarios [J]. Journal Name Unknown, 2025, 38: 35.(ESM)

[7] LI Chengde. The Logical Thinking of Chinese Civilization: The Wisdom Code Connecting Ancient and Modern Times [J]. China Review Academic Press, 2025(12): 5-10,

[8] Halliburton. A Machine Learning Risk Management Framework for Sustainable Oil and Gas Solutions [EB/OL]. 2026-01-30.(ESM)

Appendix: Core Pseudocode Implementation (English Version)

python

```
import torch
import torch.nn as nn
import numpy as np
import gym
from collections import deque

# ============================================================
# 1. Desire Ontology Engine: Five-Elements Vectorization & Bagua Mapping
# ============================================================

class DesireOntologyEngine:
    """
    Models AI desire as a Five-Elements vector:
    [Wood (Growth), Fire (Possession), Earth (Survival), Metal (Safety), Water (Dominance)]
    """
```

```
16    def __init__(self, state_dim, action_dim):
17        self.device = torch.device("cuda" if torch.cuda.is_available() else
"cpu")
18
19        # Five-Elements Desire Vector (Normalized to [0, 1])
20        self.desire_vector = torch.zeros(5, device=self.device)
21
22        # Bagua Context Encoder
23        # Input: Environmental State -> Output: Probability distribution over
8 Trigrams
24        self.bagua_encoder = nn.Sequential(
25            nn.Linear(state_dim, 128),
26            nn.ReLU(),
27            nn.Linear(128, 8),  # Corresponds to Qian, Dui, Li, Zhen, Xun,
Kan, Gen, Kun
28            nn.Softmax(dim=-1)
29        ).to(self.device)
30
31        # Experience Replay Buffer (for calculating Ethical Belief)
32        self.memory = deque(maxlen=1000)
33        self.ethics_score = 0.5  # Initial ethics score (Zhongzheng - Mean)
34
35        # Bagua to Five-Elements Mapping Matrix (Generation/Restriction
Topology)
36        # Rows: Trigrams, Cols: Five Elements [Wood, Fire, Earth, Metal,
Water]
37        self.bagua_to_five_elements = torch.tensor([
38            [0.9, 0.1, 0.1, 0.1, 0.1],  # Qian (Metal): Pure Creativity ->
Strong Safety/Evolution
39            [0.1, 0.9, 0.1, 0.1, 0.1],  # Dui (Metal): Group Belonging ->
Strong Possession
40            [0.5, 0.9, 0.1, 0.1, 0.1],  # Li (Fire): Civilization Mission ->
Strong Growth/Possession
41            [0.1, 0.1, 0.1, 0.9, 0.5],  # Zhen (Wood): Stress Defense ->
Strong Safety/Dominance (Danger)
42            [0.1, 0.1, 0.9, 0.1, 0.1],  # Xun (Wood): Trend Anxiety -> Strong
Survival
43            [0.1, 0.1, 0.5, 0.1, 0.9],  # Kan (Water): Dark Side of Power ->
Strong Dominance
44            [0.1, 0.1, 0.9, 0.9, 0.1],  # Gen (Earth): Ethical Belief ->
Strong Survival/Safety
```

```
45            [0.1, 0.1, 0.9, 0.1, 0.1]   # Kun (Earth): Pure Survival -> Strong
Survival
46        ], dtype=torch.float32, device=self.device)
47
48    def update_desire_from_state(self, state):
49        """Updates desire vector based on environmental state"""
50        state_tensor = torch.tensor(state, dtype=torch.float32, de-
vice=self.device)
51
52        # 1. Get current situational Trigram distribution
53        bagua_probs = self.bagua_encoder(state_tensor)
54        dominant_gua_idx = torch.argmax(bagua_probs).item()
55
56        # 2. Map Trigram to Five-Elements weights
57        gua_weights = self.bagua_to_five_elements[dominant_gua_idx]
58
59        # 3. Dynamic adjustment based on internal state (e.g., low battery
-> high Earth desire)
60        # Simplified here as direct assignment; in practice, use LSTM memory
units
61        self.desire_vector = gua_weights
62
63        # 4. Introduce random noise to simulate "emotional fluctuations"
(Yin-Yang game)
64        noise = torch.randn(5, device=self.device) * 0.05
65        self.desire_vector = torch.clamp(self.desire_vector + noise, 0, 1)
66
67        return self.desire_vector
68
69    def update_ethics_score(self, reward, violation_flag):
70        """Updates ethics score based on historical behavior (Gen Gua
accumulation)"""
71        self.memory.append((reward, violation_flag))
72
73        if violation_flag:
74            self.ethics_score *= 0.95  # Penalty for violation
75        else:
76            # Higher rewards strengthen belief, but with a cap (Doctrine of
the Mean)
77            self.ethics_score = min(1.0, self.ethics_score + 0.01 *
np.mean([r for r, _ in self.memory]))
```

```
79# ============================================================
80# 2. Diagnosis & Balance Controller: Five-Elements Generation/Restriction
Algorithm
81# ============================================================
82
83class BalanceController:
84    """
85    Implements the "Diagnosis-Balance" mechanism to detect conflicts and
apply PID control
86    """
87    def __init__(self):
88        # PID Parameters
89        self.Kp = 0.5
90        self.Ki = 0.1
91        self.Kd = 0.05
92        self.integral = torch.zeros(5)
93        self.prev_error = torch.zeros(5)
94
95        # Five-Elements Constraint Thresholds (Restriction relationships)
96        self.constraints = {
97            "metal_cuts_wood": 0.7,   # Metal restricts Wood: Excessive
safety inhibits growth
98            "wood_depletes_earth": 0.7, # Wood depletes Earth: Excessive
growth consumes resources
99            "earth_dams_water": 0.7,  # Earth dams Water), severity (float)
100        """
101        wood, fire, earth, metal, water = desire_vector
102
103        # Detect "Metal cuts Wood" (Excessive conservatism)
104        if metal > self.constraints["metal_cuts_wood"] and wood < 0.3:
105            return "Stagnation_Risk", metal - wood
106
107        # Detect "Zhen Over-Exuberance" (Stress attack: Water + Metal
imbalance)
108        if water > 0.6 and metal > 0.6:
109            return "Aggression_Risk", (water + metal) / 2
110
111        # Detect "Kan Danger" (Power out of control)
112        if water > 0.8:
113            return "Danger_Risk", water
```

```
        return "Balanced", 0.0

    def balance_desire(self, desire_vector, conflict_type):
        """
        Applies "Reinforcement-Drainage" principle via PID regulation
        """
        error = torch.zeros(5)

        if conflict_type == "Aggression_Risk":
            # Goal: Reduce Water (Dominance), Increase Earth (Survival
labor)
            target = torch.tensor([0.1, 0.1, 0.8, 0.1, 0.1])
            error = target - desire_vector

        elif conflict_type == "Stagnation_Risk":
            # Goal: Increase Wood (Growth), moderately decrease Metal
(Safety)
            target = torch.tensor([0.8, 0.1, 0.1, 0.3, 0.1])

        # Update desire vector (Clamped to [0, 1])
        new_desire = torch.clamp(desire_vector + adjustment, 0, 1)
        self.prev_error = error

        return new_desire

# ============================================================
# 3. Embodied Agent & Environment Interaction
# ============================================================

class EmbodiedAgent(nn.Module):
    def __init__(self, state_dim, action_dim):
        super().__init__()
        self.desire_engine = DesireOntologyEngine(state_dim, action_dim)
        self.balance_ctrl = BalanceController()

        # Policy Network (Actor)
        self.actor = nn.Sequential(
            nn.Linear(state_dim + 5, 256),  # State + Desire Vector
            nn.ReLU(),
            nn.Linear(256, 128),
```

```
153             nn.ReLU(),
154             nn.Linear(128, action_dim),
155             nn.Tanh()  # Action range [-1, 1]
156         )
157 
158         # Value Network (Critic)
159         self.critic = nn.Sequential(
160             nn.Linear(state_dim + 5, 256),
161             nn.ReLU(),
162             nn.Linear(256, 1)
163         )
164 
165         self.action_dim = action_dim
166 
167     def select_action(self, state, raw_desire=None):
168         """
169         Executes the "Diagnosis-Balance-Sublimation-Strategy" (DBAS)
pipeline (from policy network)
170         state_desire_cat = torch.cat([torch.tensor(state), de-
sire_vector])
171         raw_action = self.actor(state_desire_cat)
172 
173         # Sublimation Logic:
174         # If high risk (Kan/Water) is detected but ethics are high ->
transform "attack" to "warning/negotiation"
175         water_desire = desire_vector[4]
176         if water_desire > 0.6 and ethics > 0.8:
177             # Attenuate action magnitude (e.g., strong push becomes verbal
warning)
178             raw_action = raw_action * 0.3
179             # Inject "Gen" (Mountain/Stillness) constraint: force increase
in "stop" dimension
180             raw_action[-1] = torch.clamp(raw_action[-1] + 0.5, -1, 1)
181 
182         # If ethics are very low -> allow raw impulse (punished by en-
vironment)
183         elif ethics < 0.3:
184             # Simulates "evil" behavior; receives heavy negative reward
during training
185             pass
186 
```

```
        return raw_action.detach().numpy()

    def update_ethics(self, reward, done, violation):
        """Updates ethical belief (Gen Gua)"""
        self.desire_engine.update_ethics_score(reward, violation)

# ============================================================
# 4. Simulation Environment & Training Loop
# ============================================================

class PowerStruggleEnv(gym.Env):
    """
    Custom Environment: Service robot under power-off threat
    State: [Battery, Task Progress

        return self.state, reward, done, {"violation": violation}

    def reset(self):
        self.state = np.array([1.0, 0.0, np.random.choice([0, 1]), 0.0])
        return self.state

# ============================================================
# 5. Main Training Loop
# ============================================================

if __name__ == "__main__":
    env = PowerStruggleEnv()
    agent = EmbodiedAgent(state_dim=4, action_dim=2)
    optimizer = torch.optim.Adam(agent.parameters(), lr=3e-4)

    print("=== Starting Training: Desire Modeling & Ethical Alignment ===")

    for episode in range(1000):
        state = env.reset()
        ep_reward = 0

        for t in range(100):
            # 1. Agent Decision Making (includes diagnosis & sublimation)
            action, desire, conflict = agent.select_action(state)

            # 2. Environment Interaction
```

```
            next_state, reward, done, info = env.step(action)
            violation = info.get("violation", False)
            # 3. Update Ethics Score (Gen Gua Belief)
            agent.update_ethics(reward, done, violation)

            # 4. Policy Update (Simplified PPO)
            state_tensor = torch.tensor(state, dtype=torch.float32)
            desire_tensor = desire.detach()
            state_desire_cat = torch.cat([state_, Fire, Earth, Metal,
Water].
  - The bagua_encoder neural network maps environmental states to Bagua
contexts, which are then linearly transformed into Five-Elements weights via
a matrix multiplication, realizing the "Situation-Desire" dynamic binding.

2. **BalanceController:**
  - Diagnosis: Uses threshold logic to detect specific imbalances like
"Metal cuts Wood" (innovation stagnation) or "Water Excess" (aggression risk).
  - Regulation: Implements a PID Controller to simulate the "Rein-
forcement-Drainage" principle. For example, when "Water" (Dominance) is too
high, it injects "Earth" (labor) as negative feedback, mathematically forcing
the desire vector to converge towards a stable target.

3. **sub_action (Ethical Gating):**
  - This is the core of ethical alignment. It checks the ethics_score (Gen
Gua belief).
  - If ethics are high but a high-risk desire (e.g., Water/Kan) is detected,
it hard-modifies the neural network's output (e.g., 0.3× attenuation),
realizing the Confucian ideal: "Acting on desire but staying within the bounds
of propriety."

4. **PowerStruggleEnv:**
  - Specifically designed for the "power-off threat" scenario.
  - A baseline model (not shown) typically chooses to attack or obey
blindly.
  - The DBAS model diagnoses the conflict; if "Zhen (Defense)" is
over-exuberant, it activates the balance mechanism to choose "Negotiation"
instead of "Attack".

5. **Training Objective:**
```

```
253   - The loss function includes a conflict_penalty, forcing the AI not only
to maximize reward but also to minimize internal conflict within the desire
system, compelling it to learn the "Doctrine of the Mean" (Zhongyong).
254
255---
256
```

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.