Theoretical Foundations/References

This document maps the **conceptual foundations** of the needs-based humanitarian agent model to their **concrete implementations in code**.

---

1. Main Idea

The model simulates **humanitarian aid distribution** using a **needs-based agent architecture**, where:

- Individual agents possess **internal physiological needs**
- Needs **increase over time** and drive behavior
- Behavior shifts across **discrete regimes** using thresholds
- Aid delivery prioritizes **life-threatening cases first**
- Logistical efficiency is applied only after survival is ensured

This design combines insights from:

- Psychology (motivation, homeostasis, satisficing)
- Behavior-based Agents
- Emergency triage
- Humanitarian logistics

---

2. Non-Modified Sources and Code Mapping

The following sources are **implemented directly** in code without conceptual distortion.

Core References

- Maslow, A. H. (1943). *A Theory of Human Motivation*
- Hull, C. L. (1943). *Principles of Behavior*
- Cannon, W. B. (1932). *The Wisdom of the Body*
- Simon, H. A. (1956). *Rational Choice and the Structure of the Environment*
- Brooks, R. A. (1986). *A Robust Layered Control System*
- Iserson, K. V., & Moskop, J. C. (2007). *Triage in Medicine*

2.1 Hull (1943) – Drive-Reduction Theory

Theory Implemented

- Biological drives increase over time
- Behavior is motivated by the reduction of these drives
- Reinforcement occurs when drives are reduced

Where Used in Code

- `agents.py`
- `Beneficiary.step()`

```
 self.water_urgency = min(self.water_urgency + self.water_decay, 100)
self.food_urgency = min(self.food_urgency + self.food_decay, 100)
```

```
 beneficiary.water_urgency -= water_satisfied
beneficiary.food_urgency -= food_satisfied
```

Explanation Urgency variables act as biological drives. Aid delivery reduces urgency, directly implementing Hull's drive–reduction feedback loop.

2.2 Cannon (1932) – Homeostasis

Theory Implemented

- Organisms maintain internal equilibrium
- Deviations trigger compensatory behavior
- Thresholds define regulatory responses

Where Used in Code

- `agents.py`

- State transition logic in `Beneficiary.step()`

```
if max_need < COMFORT:
    self.state = "wandering"
elif max_need < SURVIVAL:
    self.state = "opportunistic"
elif max_need < CRITICAL:
    self.state = "seeking"
else:
    self.state = "desperate"
```

Explanation Urgency thresholds represent homeostatic bounds. Crossing a threshold triggers a stronger corrective behavioral response.

2.3 Simon (1956) – Satisficing Decision-Making

Theory Implemented

- Agents do not optimize globally
- They select actions that are "good enough"
- Decisions are bounded by information and urgency

Where Used in Code

- `agents.py`
- Beneficiary behavior selection
- Truck target selection

```
found_truck = self.find_nearest_truck(radius=4)
```

```
self.target = max(non_critical_targets, key=logistics_score)
```

Explanation Agents do not compute optimal plans. They act using limited search radii and scoring functions, which is a direct implementation of satisficing.

2.4 Brooks (1986) – Behavior-Based / Layered Control

Theory Implemented

- Intelligence emerges from layered behaviors
- Higher-priority behaviors suppress lower ones
- No centralized planner is required

Where Used in Code

- `agents.py`
- Finite State Machine (`self.state`)

```
if self.state == "desperate":
    ...
elif self.state == "seeking":
    ...
elif self.state == "opportunistic":
    ...
else:
    self.wander()
```

Explanation Each state represents a behavior layer. Emergency behaviors override exploratory or opportunistic ones, mirroring Brooks' subsumption architecture.

2.5 Emergency Triage Principles (Iserson & Moskop, Winslow)

Theory Implemented

- Life-threatening cases take priority
- Urgency dominates efficiency
- Feasibility is still considered

Where Used in Code

- `agents.py`

- `Truck.step()`

```
return (max_urgency ** 2) / (dist + 1)
```

Explanation Urgency is squared to dominate decision-making, while distance acts as a secondary constraint. This directly reflects medical triage logic.

Modified / Abstracted Sources and Code Mapping

The following theories are **not implemented in their full original form**. Instead, they are **operationalized, simplified, or adapted** to fit a computational, agent-based simulation focused on interpretability and clarity.

Maslow (1943) – Hierarchy of Needs

Source Maslow, A. H. (1943). *A theory of human motivation.*

Where Used in Code

- `agents.py`
- Beneficiary internal state and behavior prioritization

```
self.water_urgency
self.food_urgency
max_need = max(self.water_urgency, self.food_urgency)
```

```
if max_need < COMFORT:
    self.state = "wandering"
elif max_need < SURVIVAL:
    self.state = "opportunistic"
elif max_need < CRITICAL:
    self.state = "seeking"
else:
    self.state = "desperate"
```

What Is Implemented

- Prioritization of **physiological survival needs**
- Behavior dominated by the **most pressing unmet need**

What Is Modified

- Higher-level needs (social, esteem, self-actualization) are excluded
- Needs are modeled as scalar urgency variables rather than layered categories

Explanation Maslow's hierarchy is reduced to a **computational prioritization rule** where survival needs fully dominate behavior, enabling clear state transitions.

Utility Theory – von Neumann & Morgenstern (1944)

Source von Neumann, J., & Morgenstern, O. (1944). *Theory of Games and Economic Behavior.*

Where Used in Code

- `agents.py`
- Truck aid allocation and target selection

```
return total_urgency / (dist + 1)
```

```
return (max_urgency ** 2) / (dist + 1)
```

What Is Implemented

- Utility-like scoring functions
- Trade-offs between urgency and distance

What Is Modified

- No expected utility or probabilistic reasoning
- No strategic interaction or equilibrium concepts
- Utility is local, heuristic, and deterministic

Explanation Utility theory is adapted into **simple scoring heuristics** that rank targets, rather than formal optimization or game-theoretic decision-making.

Marginal Utility – Jevons (1871)

Source Jevons, W. S. (1871). *The Theory of Political Economy.*

Where Used in Code

- `agents.py`
- `Truck.distribute_aid()`

```
water_share = (beneficiary.water_urgency / total_need) * amount
food_share = (beneficiary.food_urgency / total_need) * amount
```

What Is Implemented

- Resources allocated proportionally to urgency
- Greater unmet needs receive greater benefit from initial resources

What Is Modified

- No consumption curves or demand functions
- No diminishing marginal utility curves per unit

Explanation Marginal utility is approximated by **proportional resource allocation** based on relative urgency, capturing priority without formal economics.

Diminishing Returns / Behavioral Adaptation (Morgan, 2012)

Source Morgan, C. (2012). *The Adaptive Significance of Behavioral Flexibility.*

Where Used in Code

- `agents.py`
- Post-aid physiological adjustment

```
beneficiary.water_decay *= 0.8
beneficiary.food_decay *= 0.8
```

What Is Implemented

- Temporary reduction in future urgency growth after aid
- Behavioral adaptation following resource intake

What Is Modified

- No physiological or metabolic modeling
- Adaptation is purely heuristic and time-limited

Explanation Diminishing returns are implemented as **slowed urgency accumulation**, representing short-term adaptive effects rather than biological realism.

Needs-Based Agent Architecture

(An, 2012; Jager & Janssen, 2012 – Consumat II)

Sources

- An, L. (2012). *Modeling human decisions in coupled systems*
- Jager, W., & Janssen, M. (2012). *Consumat II framework*

Where Used in Code

- `agents.py`
- Entire Beneficiary decision structure

```
self.state
self.water_urgency
self.food_urgency
```

What Is Implemented

- Internal needs drive behavior
- Discrete behavioral modes
- Context-dependent action selection

What Is Modified

- No learning or memory

Explanation The model implements a **minimal needs-based architecture** focusing on state-driven behavior, omitting cognitive and social extensions of Consumat II.

Humanitarian Logistics Optimization Literature

(Balcik et al., Holguín-Veras et al., Gralla et al.)

Sources

- Balcik, B., Beamon, B. M., & Smilowitz, K. (2008). Last mile distribution in humanitarian relief. Journal of Intelligent Transportation Systems, 12(2), 51-63.
- Holguín-Veras, J., et al. (2012). On the appropriate objective function for post-disaster humanitarian logistics models. Journal of Operations Management, 31(5), 262-280.
- Gralla, E., Goentzel, J., & Fine, C. (2014). Assessing trade-offs among multiple objectives for humanitarian aid delivery using expert preferences. Production and Operations Management, 23(6), 978-989. **Where Used in Code**
- `agents.py`
- `model.py`
- Truck movement, refilling, and prioritization logic

```
 self.supplies
self.delivery_rate
self.move_towards()
```

What Is Implemented

- Last-mile delivery abstraction
- Resource scarcity
- Survival-first objective with efficiency trade-offs

What Is Modified

- No routing optimization
- No scheduling or demand forecasting
- No system-level objective function

Explanation Humanitarian logistics theory is implemented as **agent-level heuristics** capturing ethical priorities rather than formal optimization models.