7 Appendix

7.1 Proofs of Theoretical Results

Theorem 5. *Suppose that the representation* $\varphi : \mathcal{O} \to \mathcal{Z}$ *satisfies:*

1. φ *is equivariant (Definition 1),*
2. φ *is injective,*
3. *for all* $o \in \mathcal{O}$ *and* $a \in \mathcal{A}$ *it holds that* $z'_{\text{ext}} \neq z_{\text{ext}}$ *if and only if* $z_{\text{ext}} \in \lfloor z_{\text{int}}, z_{\text{int}} + a \rfloor$ *where* $(z_{\text{int}}, z_{\text{ext}}) = \varphi(o)$ *and* $(z'_{\text{int}}, z'_{\text{ext}}) = \varphi(a \cdot o)$.

Then $\varphi \circ \omega$ *is a translation i.e., there is a constant vector* $h \in \mathbb{R}^n$ *such that for all* $s \in \mathcal{S}$ *it holds that* $\varphi(\omega(s)) = s + h$*. In particular,* φ *is an isometry w.r.t. the Euclidean metric on both* $\mathcal{S}$ *and* $\mathcal{Z}$*.*

Fig. 7: Graphical depiction of the proof of Theorem 4.

Proof. Pick an arbitrary state $s^0 \in \mathcal{S}$ together with its represented internal state z^0_{int} and set $h = z^0_{\text{int}} - s^0_{\text{int}}$. For any state s, consider the action $a = s_{\text{int}} - s^0_{\text{int}}$. Equivariance then implies that $z_{\text{int}} = z^0_{\text{int}} + a = s_{\text{int}} + h$. This shows that the claim holds for internal states.

To prove that the same happens for external states, suppose by contradiction that there is a state s such that $z_{\text{ext}} \neq s_{\text{ext}} + h$. Consider any path during which the agent interacts with the object without passing through z_{ext}. Formally, this means considering a sequence of actions $a_1, \cdots, a_r$ such that (see Figure 7, left):

- z_{ext} and $s_{\text{ext}} + h$ do not belong to $\lfloor z_{\text{int}} + a_1 + \cdots + a_{i-1}, z_{\text{int}} + a_1 + \cdots + a_i \rfloor$ for every $i = 1, \cdots, r-1$,
- z_{ext} does not belong to $\lfloor z_{\text{int}} + a_1 + \cdots + a_{r-1}, z_{\text{int}} + a_1 + \cdots + a_r \rfloor$ but $s_{\text{ext}} + h$ does.

The existence of such a path follows from Assumptions 1 and 3. After interaction the state becomes $s' = a_r \cdot (a_{r-1} \cdots (a_1 \cdot s))$ with $s'_{\text{ext}} \neq s_{\text{ext}}$ because of Assumption 2. One can then consider a path back to the initial agent's position z_{int} i.e., another sequence of actions $a_{r+1}, \cdots, a_R$ such that (see Figure 7, right):

- $s'_{\text{ext}} + h$ and z_{ext} do not belong to $\lfloor z_{\text{int}} + a_1 + \cdots + a_{i-1}, z_{\text{int}} + a_1 + \cdots + a_i \rfloor$ for every $i = r+1, \cdots, R$,

– $a_1 + \cdots + a_R = 0$.

All the conditions imply together that the representation of the object remains equal to z_{ext} during the execution of the actions $a_1, \cdots, a_R$. Since the actions sum to 0, the representation of the agent does not change as well. But then $\varphi(\omega(s)) = \varphi(\omega(s_{\text{int}}, s'_{\text{ext}}))$ while $s_{\text{ext}} \neq s'_{\text{ext}}$, contraddicting injectivity. We conclude that $z_{\text{ext}} = s_{\text{ext}} + h$ and thus $z = s + h$ as desired.

7.2 Pseudocode for Loss Computation

Algorithm 1 Loss Computation

Input: Batch $\mathcal{B} \subseteq \mathcal{D}$, models $\varphi_{\text{int}}, \varphi_{\text{ext}}, \varphi_{\text{cont}}$
Output: Loss $\mathcal{L}$

```
ℒ = 0
for all (o, a, o′) ∈ ℬ do
    Compute z_int = φ_int(o), z_ext = φ_ext(o), z′_int = φ_int(o′), z′_ext = φ_ext(o′), w = φ_cont(o), w′ = φ_cont(o′)
end for
Compute the classes C₋, C₊ via Otsu's algorithm based on {d_𝒲(w, w′)}
for all (o, a, o′) ∈ ℬ do
    Compute ℒ_int(o, a, o′) via Equation 3
    Compute A = {d_𝒲(w′, w″), d(z′_int, z″_int)} for o″ marginalized from ℬ
    Based on A compute ℒ_cont(o, o′) via Equation 5
    if d_𝒲(w, w′) ∈ C₋ then
        Compute ℒ_ext(o, a, o′) = ℒ₋(o, a, o′) via Equation 4 (left)
    else
        Compute ℒ_ext(o, a, o′) = ℒ₊(o, a, o′) via Equation 4 (right)
    end if
    ℒ ← ℒ + ℒ_int + ℒ_ext + ℒ_cont
end for
```