Algorithm 1: Multistep Quasimetric Reinforcement Learning

```
Require: Dataset $\mathcal{D}$, Batch size $B$, training iteration $T$, Probability $p$
Initialize Quasimetric network $\mathcal{Q}$ with parameters $(\varphi(s,a), \psi(s))$, goal-reaching policy $\pi_\mu$
for $t = 1 \dots T$ do
    Sample $\{s_i, a_i, s'_i, \tilde{s}^w_i, g_i\}_{i=1}^B \sim \mathcal{D}$ (Eq. (8))
    For each element in batch, choose $s^w_i \sim \begin{cases} s'_i & \text{with probability } p, \\ \tilde{s}^w_i & \text{with probability } 1-p. \end{cases}$
    Update $\mathcal{Q}$ with multistep backup by minimizing $\mathcal{L}_{\mathcal{T}_\beta}(\varphi, \psi; \{s_i, a_i, s^w_i\}_{i=1}^B)$ (Eq. (9))
    Update $\mathcal{Q}$ with optimality constraints by minimizing $\mathcal{L}_{\mathcal{I}}(\varphi, \psi; \{s_i, a_i\}_{i=1}^B)$ (Eq. (15))
    Update policy $\pi_\mu$ with DDPG+BC by minimizing $\mathcal{L}_\mu(\pi_\mu; \{s_i, a_i, g_i\}_{i=1}^B)$ (Eq. (16))
return $\pi_\mu$
```