TESSERACT: Tensorised Actors for Multi-agent Reinforcement Learning

Anuj Mahajan, Mikayel Samvelyan, Lei Mao,
Viktor Makoviychuk, Animesh Garg, Jean Kossaifi,
Shimon Whiteson, Yuke Zhu, Animashree Anandkumar

Tesseract motivation

- Cooperative Multi Agent Reinforcement Learning (MARL) suffers from action space blow-up.
- For value-based methods: Poses challenges in accurately representing the optimal value function, thus inducing suboptimality.
- For policy gradient methods: Renders critic ineffective and exacerbates the problem of the *lagging* critic.
- Similar challenges for model-based methods.

Tesseract idea

- Main idea : A framework to exploit tensor structure in MARL problems for sample efficient learning.
- Q-function seen as a tensor where the modes correspond to action spaces of different agents.
- Applicable to any factorizable action-space

Background Multi Agent Reinforcement Learning (MARL)

Notation:

- S is the set of states
- U the set of available actions per agent
- agents $i \in \mathcal{A} \equiv \{1, ..., n\}$
- joint action $\mathbf{u} \in \mathbf{U} \equiv U^n$
- $P(s'|s, \mathbf{u}) : S \times \mathbf{U} \times S \to [0, 1]$ is the state transition function
- $r(s, \mathbf{u}) : S \times \mathbf{U} \to \mathbb{R}$ is the reward function
- observations $z \in Z$ according to observation distribution $O(s) : S \times \mathcal{A} \to \mathcal{P}(Z)$.
- γ is discount factor
- action-observation history for an agent i is $\tau^i \in T \equiv (Z \times U)^*$

MARL problem continued

$$Q^{\pi}(z_t, \mathbf{u}_t) = \mathbb{E}_{z_{t+1:\infty}, \mathbf{u}_{t+1:\infty}} \left[\sum_{k=0}^{\infty} \gamma^k r_{t+k} | z_t, \mathbf{u}_t \right]$$

The goal of the problem is to find the optimal action value function Q^* and the corresponding policy π^*.

Figure 1: Example MARL scenario

Settings in Multi Agent Reinforcement Learning

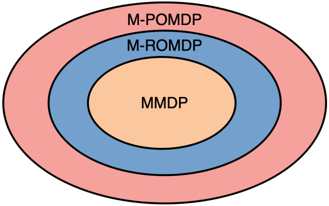


Figure 2: MARL settings w.r.t observability

- MMDP : $\langle S, U, P, r, n, \gamma \rangle$ Bijective map $O : \mathcal{S} \rightarrow Z$
- M-ROMDP : $\langle S, U, P, r, Z, O, n, \gamma \rangle$, where we require that the joint observation space is partitioned w.r.t. S ie. $\forall s_1, s_2 \in S \wedge z \in Z, P(z|s_1) > 0 \wedge s_1 \neq s_2 \implies P(z|s_2) = 0$.
- M-POMDP : $\langle S, U, P, r, Z, O, n, \gamma \rangle$
- Note that for latter two we assume $|Z| >> |S|$.

Tensors intro

- Tensors are high dimensional analogues of matrices
- Tensor decomposition, in particular, generalize the concept of low-rank matrix factorization
- Notation $\hat{\cdot}$ to represent tensors
- An order n tensor $\hat{T}$ has n index sets $I_j, \forall j \in \{1..n\}$ and has elements $T(e), \forall e \in \times_{\mathcal{I}} I_j$

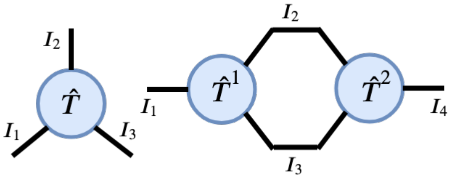


Figure 3: Left: Tensor diagram for an order 3 tensor $\hat{T}$. Right: Contraction between $\hat{T}^1, \hat{T}^2$ on common index sets I_2, I_3.

Tensors intro

- Tensor contraction: For any two tensors $\hat{T}^1$ and $\hat{T}^2$ with $\mathcal{I}_\cap = \mathcal{I}^1 \cap \mathcal{I}^2$ we define the contraction operation as $\hat{T}^1 \odot \hat{T}^2(e_1, e_2) = \sum_{e \in \times_{\mathcal{I}_\cap} I_j} \hat{T}^1(e_1, e) \cdot \hat{T}^2(e_2, e), e_i \in \times_{\mathcal{I}^i \setminus \mathcal{I}_\cap} I_j$.
- A tensor $\hat{T}$ can be factorized using a (rank–k) CP decomposition into a sum of k vector outer products (denoted by $\otimes$), as,

$$\hat{T} = \sum_{r=1}^{k} w_r \otimes^n u_r^i, i \in \{1..n\}, ||u_r^i||_2 = 1. \tag{1}$$

Tensorising the Q-function

- Given a multi-agent problem G, let $\mathcal{Q} \triangleq \{Q : S \times U^n \to \mathbb{R}\}$ be the set of real-valued functions on the state-action space
- Focus on the *Curried* form $Q : S \to U^n \to \mathbb{R}, Q \in \mathcal{Q}$ so that $Q(s)$ is an order n tensor
- Algorithms in Tesseract operate directly on the curried form and preserve the structure implicit in the Q tensor.

Tensorised Bellman Equation

- Components of the underlying MARL problem can be seen as tensors given a state (denoted with $\hat{\cdot}$).
- Modes correspond to action spaces of different agents

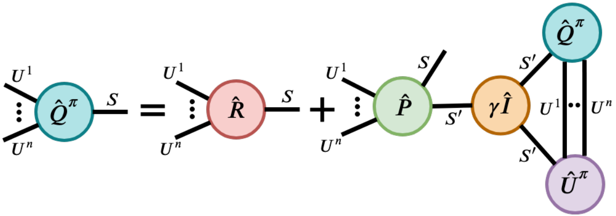


Figure 4: Tensor Bellman Equation for n agents. There is an edge for each agent $i \in \mathcal{A}$ in the corresponding nodes $\hat{Q}^{\pi}$, $\hat{U}^{\pi}$, $\hat{R}$, $\hat{P}$ with the index set U^i.

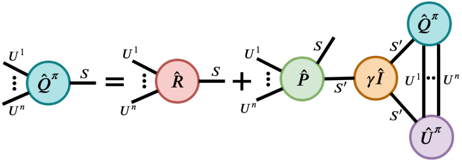
Algorithm 1 Model based Tesseract

Initialise rank k, $\pi = (\pi^i)_1^n$ and $\hat{Q}$: Theorem 1
Initialise model parameters $\hat{P}, \hat{R}$
Learning rate $\leftarrow \alpha$,$\mathcal{D} \leftarrow \{\}$
for each episodic iteration i **do**
    Do episode rollout $\tau_i = \left\{(s_t, \mathbf{u}_t, r_t, s_{t+1})_0^L\right\}$ using π
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{\tau_i\}$
    Update $\hat{P}, \hat{R}$ using CP-Decomposition on moments from $\mathcal{D}$.
    for each internal iteration j **do**
        $\hat{Q} \leftarrow \Pi_k \mathcal{T}^\pi \hat{Q}$
    end for
    Improve π using $\hat{Q}$
end for
Return $\pi, \hat{Q}$

Theorems for MMDP

Theorem (Bounding rank of $\hat{Q}$)

For a finite MMDP under mild assumptions, the action-value tensor satisfies $\text{rank}(\hat{Q}^\pi(s)) \leq k_1 + k_2|S|, \forall s \in S, \forall \pi$.



Corollary

For all $k \geq k_1 + k_2|S|$, *the procedure* $Q_{t+1} \leftarrow \Pi_k \mathcal{T}^\pi Q_t$ *converges to* Q^π *for all* Q_0, π.

Theorems for MMDP

- Rank sufficient approximation $k \geq k_1, k_2$

Theorem (Model based estimation of $\hat{R}$, $\hat{P}$ error bounds)

Given any $\epsilon > 0, 1 > \delta > 0$, for a policy π with the policy tensor satisfying $\pi(\mathbf{u}|s) \geq \Delta$, where

$$\Delta = \max_s \frac{C_1 \mu_s^6 k^5 (w_s^{max})^4 \log(|U|)^4 \log(3k||R(s)||_F/\epsilon)}{|U|^{n/2}(w_s^{min})^4}$$

and C_1 is a problem dependent positive constant. There exists N_0 which is $O(|U|^{\frac{n}{2}})$ and polynomial in $\frac{1}{\delta}, \frac{1}{\epsilon}, k$ and relevant spectral properties of the underlying MDP dynamics such that for samples $\geq N_0$, we can compute the estimates $\bar{R}(s), \bar{P}(s, s')$ such that w.p. $\geq 1 - \delta$,
$||\bar{R}(s) - \hat{R}(s)||_F \leq \epsilon, ||\bar{P}(s, s') - \hat{P}(s, s')||_F \leq \epsilon, \forall s, s' \in S$.

Theorems for MMDP

Theorem (Error bound on policy evaluation)

Given a behaviour policy π_b satisfying the conditions in the theorem above and executed for steps $\geq N_0$, for any policy π the model based policy evaluation $Q^{\pi}_{\bar{P},\bar{R}}$ satisfies:

$$
\begin{aligned}
|Q^{\pi}_{P,R}(s,a) - Q^{\pi}_{\bar{P},\bar{R}}(s,a)| \leq & (|1-f| + f|S|\epsilon)\frac{\gamma}{2(1-\gamma)^2} \\
& + \frac{\epsilon}{1-\gamma}, \forall (s,a) \in S \times U^n
\end{aligned}
$$

where $\frac{1}{1+\epsilon|S|} \leq f \leq \frac{1}{1-\epsilon|S|}$.

Comments

- Similar results can be obtained for M-POMDPs and M-ROMDPs with some conditions on the observation distribution (no information loss).
- $O(kn|U||S|^2)$ parameters for the model based approach, for large/continuous state-action spaces the tensor structure can be embedded in a model free manner (next)

Model free Tesseract

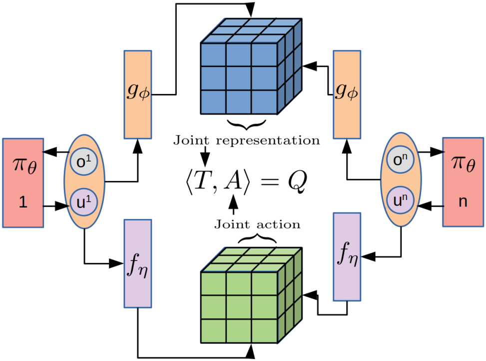


Figure 5: Tesseract architecture

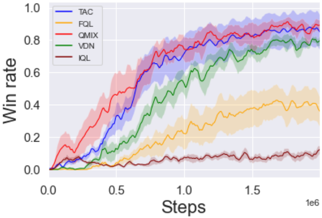
- The joint action-value estimate of the tensor $\hat{Q}(s)$ by the central critic is:

$$\hat{Q}^\pi(s) \approx \sum_{r=1}^{k} w_r^i \otimes^n g_{\phi,r}(s^i), i \in \{1..n\} \tag{2}$$

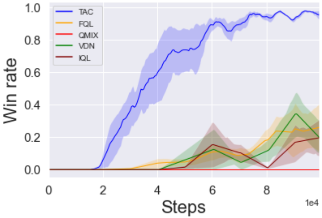
Algorithm 2 Model free Tesseract

Initialise parameter vectors θ, ϕ, η
Learning rate $\leftarrow \alpha$,$\mathcal{D} \leftarrow \{\}$
for each episodic iteration i **do**
  Do episode rollout $\tau_i = \left\{(s_t, \mathbf{u}_t, r_t, s_{t+1})_0^L\right\}$ using π_θ
  $\mathcal{D} \leftarrow \mathcal{D} \cup \{\tau_i\}$
  Sample batch $\mathcal{B} \subseteq \mathcal{D}$.
  Compute empirical estimates for $\mathcal{L}_{TD}, \mathcal{J}_\theta$
  $\phi \leftarrow \phi - \alpha \nabla_\phi \mathcal{L}_{TD}$ (Rank k projection step)
  $\eta \leftarrow \eta - \alpha \nabla_\eta \mathcal{L}_{TD}$ (Action representation update)
  $\theta \leftarrow \theta + \alpha \nabla_\theta \mathcal{J}_\theta$ (Policy update)
end for
Return $\pi, \hat{Q}$

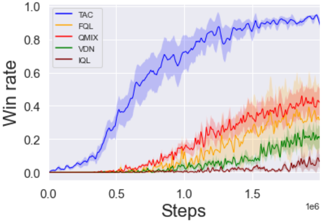
StarCraft II: SMAC Experiments



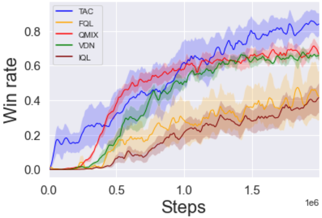
(a) 3s5z **Easy**



(b) 2s_vs_1sc **Easy**



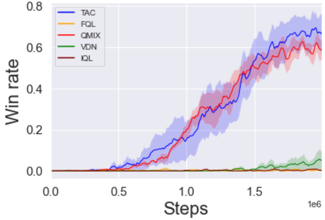
(c) 2c_vs_64zg **Hard**



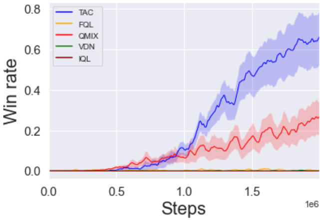
(d) 5m_vs_6m **Hard**

Figure 6: Performance of different algorithms on **Easy** and **Hard** SMAC scenarios: TAC, QMIX, VDN, FQL, IQL.

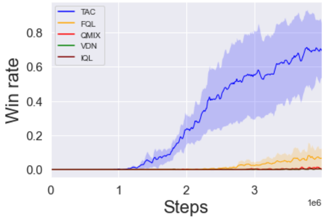
StarCraft II: SMAC Experiments



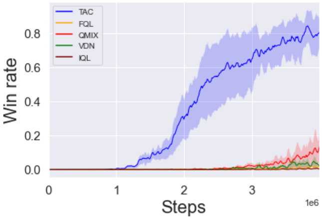
(a) MMM2 **Super Hard**



(b) 27m_vs_30m **Super Hard**



(c) 6h_vs_8z **Super Hard**



(d) Corridor **Super Hard**

Figure 7: Performance of different algorithms on **Super Hard** SMAC scenarios: TAC, QMIX, VDN, FQL, IQL.

Thanks!

Questions?