**LEARNING WITH PRIOR POLICY**

- **Main variables**
  - \( S, U \subseteq \mathbb{R}_{20} \), are respectively state and action spaces;
  - \( s^i \in S \) is the concentration of gene \( i \);
  - \( b \in \mathbb{R}_{20} \), is the power of the light pulse;
  - \( u \in U \) is the light activation action;
  - \( p \in \mathbb{R}_{20} \), is the cost related to the light pulse.

- **Protein concentration transition for each gene**
  (Simplified version)
  \[
  \begin{align*}
  s^1 &= \frac{c_1}{1 + (s^2/f_1)^{a_1}} - c_2 s^1 + b u, \\
  s^2 &= \frac{c_3}{1 + (s^1/f_2)^{a_2}} - c_4 s^2.
  \end{align*}
  \]

- **Transition cost**
  \[
  c((s^1, s^2), u, (s^1_+, s^2_+)) = -s^1_+ + s^2_+ + pu.
  \]

**OPTIMAL VALUE/Q FUNCTION**

Infinite-dimensional optimization problem and Q-function reformulation

\[
V(s_t) = \min_{\mu^*} \sum_{k=0}^{\infty} \gamma^i V(s_{t+k}) + \gamma V_p(f(s, u)) - V_p(s) + \
\gamma \min_{\mu_k} Q_{k+1}(f(s, v), v).
\]

- \( V(s_t) \) is the value function \( V_p \), the existing value function, equals to zero everywhere when no value function is to be used;
- \( \gamma \) is the discount factor;
- \( \mu^* : S \rightarrow U \) is the optimal state-action mapping (also called an *optimal policy*);
- \( f(\cdot) \) is the transition function;
- \( Q_k(\cdot) \) is recursive version of the Q-function (computed with *Fitted-Q-Iteration* algorithm in our case) with \( k \) the number of processed iterations;
- When \( V_p \) is the prior information used in the form of a value function (reward shaping technique).

**Preliminary conclusions**

- Learning process can be speeded up ten times in terms of number of data points;
- Number of data seems not to be correlated to *regret* score (i.e., difference of cost between two policies on the same model).

**Future work**

- Change problem setting for stochastic dynamics;
- Data selection to avoid negative transfer;
- Aggregation of several prior policies (more specifically, value functions).