Supporting Information

Towards Efficient Discovery of Green Synthetic Pathways with Monte Carlo Tree Search and Reinforcement Learning

Xiaoxue Wang[1,3], Yujie Qian[2], Hanyu Gao[1], Connor W. Coley[1], Yiming Mo[1], Regina Barzilay[2], Klavs F. Jensen[*1]

[1]Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts, 02139, USA

[2]Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts, 02139, USA

[3]Department of Chemical and Biomolecular Engineering, The Ohio State University, Columbus, Ohio, 43210, USA

*Email: kfjensen@mit.edu

Keywords: synthesis planning, green chemistry, Monte Carlo Tree Search, Reinforcement Learning, Machine Learning

Part 1 The modified UCB1 without dynamic c tuning maintains a time-bounded regret

In this part, we will demonstrate that by slightly modifying the UCB1[1] algorithm used in original UCT[2-3], we are able to maintain the characteristic of a bounded expected cumulative regret that scales logarithmically with visiting time, while enhance the sampling efficiency of the MCTS based on the modified UCB1. The proof here is based on the orginal proofs for the original UCB1 algorithm proposed by Auer, Cesa-Bianchi and Fischer[1]. The notations used are the same as Auer *et al.*'s original paper[1].

***Problem formulation**:*

MCTS is closely related with the so called multi-armed bandit problem. A K-armed bandit problem is defined as follows:

We have K gambling machines (or actions in Markov Decision Processes) with the payoffs expressed using random variables $X_{i,n}$ for $1 \le i \le K$ and $n \ge 1$, where i denotes the index of a gambling machine, and n is the number of times machine i is visited. The random variables in the series of rewards, $X_{i,1}$, $X_{i,2}$ …, generated by playing machine i successively, are independent and follow the same distribution with expectation μ_i. In addition, across different machines, $X_{i,s}$ and $X_{j,t}$ are also independent but usually not identically distributed, for each $1 \le i < j \le K$ and each $s,t \ge 1$. Here for UCB1 and the modified UCB1 that we will discuss later, $X_{i,n} \in [0,1]$.

We define the policy, or allocation strategy A as an algorithm that chooses the next machine to play based on history. The target of the K-armed bandit problem is to wisely choose correct strategies to visit each machine in order to maximize the gained cumulated rewards. This goal requires exploration of unvisited machines and exploitation of the known best machine. In order to quantify the success of a policy, we will use the expected cumulative regret as the standard. The expected cumulative regret is defined as follows:

Let $T_i(n)$ be the number of times machine i has been played by A during the first n plays. Then the expected cumulative regret of A after n plays is defined by

$$\mu^* n - \mu_j \sum_{j=1}^{K} E(T_j(n)) \text{ where } \mu^* \overset{def}{=} \max_{1 \le i \le K} \mu_i$$

where $E(\bullet)$ denotes expectation. We call optimal the machine with the least index i such that $\mu_i = \mu^*$

The expected cumulative regret after n plays can also be written as:

$\sum_{j:\mu_j < \mu^*} \Delta_j E(T_j(n))$ where $\Delta_i = \mu^* - \mu_i$, where μ_i is the reward expectation for machine i and μ^* is any maximal element in the set $\{\mu_1, ..., \mu_K\}$, $X_{i,s} \in [0,1]$

Thus the expected cumulative regret is the expected loss due to the fact that the policy does not always play the best machine.

UCB1 algorithm[1]:

To deal with the exploration vs. exploitation dilemma in the multi-armed bandit problem, in the classic MCTS algorithm called UCT (upper confidence bound applied to trees) proposed by Kocsis and Szepesvari[2], the algorithm UCB1 is used to actively explore new actions while keep visiting the most promising action so far frequently. The UCB1 algorithm, proposed by Auer, Cesa-Bianchi and Fischer[1] to solve the multi-armed bandit problem is described as follows:

The algorithm is initialized by playing each machine once. Then we define an upper confidence bound (UCB) by $UCB_{j,n_j} = \overline{x}_j + \sqrt{\frac{2 \ln n}{n_j}}$ where $\overline{x}_j$ is the average reward obtained from machine j, n_j is the number of times machine j has been played so far, and n is the overall number of plays done so far. We play machine j that maximizes UCB_{j,n_j}

One can prove that $E(regret) \le [8 \sum_{i:\mu_i < \mu^*} \frac{\ln n}{\Delta_i}] + (1 + \frac{\pi^2}{3})(\sum_{j=1}^{K} \Delta_j)$ when the rewards are bounded at [0,1][1].

In this part, we will demonstrate that by slightly modifying the UCB1 algorithm, we are able to maintain the characteristic of a bounded expectation of regret, while enhance the sampling efficiency of the MCTS based on the modified UCB1. Before we go into the analysis of the expected cumulative regret for modified UCB1, let's first look at the notations used in the proof. We follow the notations in Auer *et al.*[1]

Notations:

We follow the notations used by Auer et al. in their orginal paper about UCB1 algorithm. For each $1 \le i \le K$, $E(X_{i,n}) = \mu_i \ \forall \ n \ge 1$ and $\mu^* = \max_{1 \le i \le K} \mu_i$. Also, for any fixed policy, $T_i(n)$ is the number of times machine i has been played in the first n times. As a result, $\sum_{i=1}^{K} T_i(n) = n$. We also define the random variables $I_1, I_2, \ldots$ where I_t denotes the machine played at time t. For each $1 \le i \le K$ and $n \ge 1$ define:

$$\overline{X}_{i,n} = \frac{1}{n} \sum_{t=1}^{n} X_{i,t} \ .$$

We use * to refer any quantity related to the optimal machine, such as $T^*(n)$ and $\overline{X}_n{}^*$ instead of $T_i(n)$ and $\overline{X}_{i,n}$ where i is the index of the optimal machine. Further, we have the indicator function $\{\Pi(x)\}$ of event $\Pi(x)$: $\{\Pi(x)\} = 1$ when $\Pi(x)$ is true, and $\{\Pi(x)\} = 0$ if $\Pi(x)$ is false.}

Therefore the upper confidence bound can be written as $UCB_{i,t} = \overline{X}_{i,T_i(t-1)} + \sqrt{\frac{2 \ln t}{T_i(t-1)}}$ }

Modified UCB1 algorithm*:*

In the modified UCB1 algorithm, we choose the upper confidence bound (UCB) of machine i at time t as:

$$UCB_{i,t} = \overline{X}_{i,T_i(t-1)} + c_p \sqrt{\frac{2 \ln t}{1 + T_i(t-1)}}$$

let $c_{t,s} = c_p \sqrt{\frac{2 \ln t}{1+s}}$, $\Delta_i = \mu^* - \mu_i$

Then

$$T_i(n) = \sum_{t=1}^{n} \{I_t = i\}$$

$$= \sum_{t=1}^{n} \{I_t = i, T_i(t-1) \le l-1\} + \sum_{t=1}^{n} \{I_t = i, T_i(t-1) \ge l\}$$

$$\le l + \sum_{t=1}^{n} \{I_t = i, T_i(t-1) \ge l\}$$

$$\le l + \sum_{t=1}^{n} \{\min_{0<s<t} \overline{X}_s{}^* + c_{t-1,s} \le \max_{l \le s_i < t} \overline{X}_{i,s_i} + c_{t-1,s_i}\}$$

$$\le l + \sum_{t=1}^{\infty} \sum_{s=1}^{t-1} \sum_{s_i=l}^{t-1} \{\overline{X}_s{}^* + c_{t,s} \le \overline{X}_{i,s_i} + c_{t,s_i}\}$$

If $\{\overline{X}_s^* + c_{t,s} \le \overline{X}_{i,s_i} + c_{t,s_i}\} = 1$, or in other words, $\overline{X}_s^* + c_{t,s} \le \overline{X}_{i,s_i} + c_{t,s_i}$ is True, then at least one of the following must hold:

$\overline{X}_s^* \le \mu^* - c_{t,s}$ (1)

$\overline{X}_{i,s_i} \ge \mu_i + c_{t,s_i}$ (2)

$\mu^* < \mu_i + 2c_{t,s_i}$ (3)

For (1), from the Chernoff-Hoeffding inequality, we know that

$$P(\overline{X}_s^* \le \mu^* - c_{t,s}) = P(\overline{X}_s^* \le \mu^* - c_p\sqrt{\frac{2\ln t}{1+s}}) \le \exp(-2s\cdot\frac{2\ln t}{1+s}\cdot {c_p}^2) = t^{-4\frac{s}{1+s}{c_p}^2}$$

We need to note here that the reason why $UCB_{i,t} = \overline{X}_{i,T_i(t-1)} + \sqrt{\frac{2\ln t}{T_i(t-1)}}$ (the original UCB1) or $UCB_{i,t} = \overline{X}_{i,T_i(t-1)} + c_p\sqrt{\frac{2\ln t}{1+T_i(t-1)}}$ (the modified UCB1 here) are called the "upper confidence bound" is that the probability of the UCB smaller than the true expectation μ_i decays with t in the fashion of t^{-4} (the original UCB1) or $t^{-4\frac{T_i(t-1)}{1+T_i(t-1)}{c_p}^2}$ (the modified UCB1 here) as a result of the Chernoff-Hoeffding inequality.

Similarly, as a result of the Chernoff-Hoeffding inequality, for (2), the probability is bounded as below:

$$P(\overline{X}_{i,s_i} \ge \mu_i + c_{t,s_i}) = P(\overline{X}_{i,s_i} \ge \mu_i + c_p\sqrt{\frac{2\ln t}{1+s_i}}) \le \exp(-2s_i\cdot\frac{2\ln t}{1+s_i}\cdot {c_p}^2) = t^{-4\frac{s_i}{1+s_i}{c_p}^2}$$

By choosing appropriate l, we can make (3) false for all the time.

Let $l = \left\lceil \frac{8\ln n}{{\Delta_i}^2} \right\rceil {c_p}^2$, then since $s_i \ge l \ge \frac{8\ln n}{{\Delta_i}^2}{c_p}^2$ and $t \le n$

$$\mu^* - \mu_i - 2c_{t,s_i} = \Delta_i - 2c_p\sqrt{\frac{2\ln t}{1+s_i}} \ge \Delta_i - 2c_p\sqrt{\frac{2\ln t}{s_i}} \ge \Delta_i - 2c_p\sqrt{\frac{2\ln t}{l}} \ge \Delta_i - 2c_p\sqrt{\frac{2\ln n}{\frac{8\ln n}{{\Delta_i}^2}{c_p}^2}} = 0$$

Therefore (3) is False when $l = \left\lceil \frac{8\ln n}{{\Delta_i}^2} \right\rceil {c_p}^2$.

Hence, the expectation

$$E(T_i(n)) \le \left\lceil \frac{8\ln n}{\Delta_i^{\,2}} \right\rceil c_p^{\,2} + \sum_{t=1}^{\infty}\sum_{s=1}^{t-1} \sum_{s_i=\left\lceil \frac{8\ln n}{\Delta_i^{\,2}} \right\rceil c_p^{\,2}}^{t-1} (P(\bar{X}_s^{\,*} \le \mu^* - c_{t,s}) + P(\bar{X}_{i,s_i} \ge \mu_i + c_{t,s_i}))$$

$$\le \left\lceil \frac{8\ln n}{\Delta_i^{\,2}} \right\rceil c_p^{\,2} + \sum_{t=1}^{\infty}\sum_{s=1}^{t-1} \sum_{s_i=\left\lceil \frac{8\ln n}{\Delta_i^{\,2}} \right\rceil c_p^{\,2}}^{t-1} (t^{-4\frac{s}{1+s}c_p^{\,2}} + t^{-4\frac{s_i}{1+s_i}c_p^{\,2}})$$

$$\le \left\lceil \frac{8\ln n}{\Delta_i^{\,2}} \right\rceil c_p^{\,2} + \sum_{t=1}^{\infty}\sum_{s=1}^{t}\sum_{s_i=1}^{t} (t^{-4\frac{s}{1+s}c_p^{\,2}} + t^{-4\frac{s_i}{1+s_i}c_p^{\,2}})$$

Since $s \ge 1$, $s_i \ge 1$ and $t \ge 1$, therefore $t^{-4\frac{s}{1+s}c_p^{\,2}} \le t^{-2c_p^{\,2}}$

$$E(T_i(n)) \le \left\lceil \frac{8\ln n}{\Delta_i^{\,2}} \right\rceil c_p^{\,2} + \sum_{t=1}^{\infty}\sum_{s=1}^{t}\sum_{s_i=1}^{t} (t^{-4\frac{s}{1+s}c_p^{\,2}} + t^{-4\frac{s_i}{1+s_i}c_p^{\,2}})$$

$$\le \left\lceil \frac{8\ln n}{\Delta_i^{\,2}} \right\rceil c_p^{\,2} + \sum_{t=1}^{\infty}\sum_{s=1}^{t}\sum_{s_i=1}^{t} (2t^{-2c_p^{\,2}})$$

Let $c_p^{\,2} = 2$, then we are able to get the upper bound for the expectation of $T_i(n)$

$$E(T_i(n)) \le \left\lceil \frac{8\ln n}{\Delta_i^{\,2}} \right\rceil c_p^{\,2} + \sum_{t=1}^{\infty}\sum_{s=1}^{t}\sum_{s=1}^{t} 2t^{-4}$$

$$\le (\frac{8\ln n}{\Delta_i^{\,2}} + 1)c_p^{\,2} + \sum_{t=1}^{\infty}\sum_{s=1}^{t}\sum_{s=1}^{t} 2t^{-4}$$

$$= (\frac{8\ln n}{\Delta_i^{\,2}} + 1)c_p^{\,2} + \sum_{t=1}^{\infty} 2t^{-2}$$

$$= (\frac{8\ln n}{\Delta_i^{\,2}} + 1)c_p^{\,2} + \frac{\pi^2}{3}$$

$$= (\frac{16\ln n}{\Delta_i^{\,2}} + 2) + \frac{\pi^2}{3}$$

Then the expectation of regret is

$$E(regret) = \sum_{i=1}^{K} \Delta_i E(T_i(n)) \le [16 \sum_{i:\mu_i<\mu^*} \frac{\ln n}{\Delta_i}] + (2 + \frac{\pi^2}{3}) \sum_{i=1}^{K} \Delta_i$$

Compared with the original bound for the UCB1 algorithm, the regret bound of the modified UCB1 only defers by the value of the coefficientss. The essence of a visit time n bounded expectation of regret does not change. When applied to MCTS, the coefficient c_p can be combined with the c_p in UCT to be modulated in order to account the effect of biased sampling[2-3]. In the MCTS search using the modified UCB1, the mandatory initialization step of visiting all available actions once in the original UCT (using

original UCB1) is not necessary, as a result of a non-zero denominator in the modified UCB1 equation even when the action is not visited. Similar strategy is used in the Alpha Go type PUCT MCTS[4-6], which makes the sampling of fruitful actions in the search tree faster, when a policy network is used to give the prior probability distribution of the actions. In the modified UCT, a policy network is also used to give the ranking of the actions based on their prior probabilities(see the main text).

References

1. Auer, P.; Cesa-Bianchi, N.; *Machine Learning* **2002,** *47* (2), 235-256.
2. Kocsis, L.; Szepesvári, C. In *Bandit based Monte-Carlo planning*, European conference on machine learning, Springer: 2006; pp 282-293.
3. James, S.; Konidaris, G.; Rosman, B., An Analysis of Monte Carlo Tree Search, *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, February 2017, 3576–3582
4. Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; Dieleman, S.; Grewe, D.; Nham, J.; Kalchbrenner, N.; Sutskever, I.; Lillicrap, T.; Leach, M.; Kavukcuoglu, K.; Graepel, T.; Hassabis, D., *Nature* **2016,** *529* (7587), 484-489.
5. Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.,. *Nature* **2017,** *550* (7676), 354-359.
6. Rosin, C. D., *Annals of Mathematics and Artificial Intelligence* **2011,** *61* (3), 203-230.

Table S1 The performance of the Round 2 RL value network in MCTS tree expansion for 30s on 1000 compounds (the same test set as in Fig 3 in the main text).

MCTS variants	mUCT-dc-V c initialized as 0.1	mUCT-V		PUCT-V c=1	UCT-V	
Success rate on test set	0.648	c=0.1	0.515	0.656	c=0.1	0.548
		$c=\sqrt{2}$	0.55		c=1	0.51

Table S2 Study of the effect of c values for UCT type MCTS variants, 30s test on 1000 compounds. Here the values of compounds are solely given by the value network without setting the values of the compounds in the buyable catalog to 1. The value net used here is the Round 1 RL value network.

MCTS variants	c value	Success rate	
		test set	training set
UCT-V	c = 0.1	0.53	0.569
	c = 1	0.509	0.55
mUCT-V	c = 0.1	0.548	0.576
	$c = \sqrt{2}$	0.56	0.597

Table S3 The greenness of the synthetic routes generated by mUCT-dc-V using CSS value network when compared with mUCT-dc-V using Round 1 RL value network as the baseline method.

Number of improved compounds	131
Root CSS average improved by	0.052
Number of unchanged compounds	15
Number of worsened compounds	165
Root CSS average worsened by	0.11
Average root CSS score changed by	-0.036

Table S4 . The greenness of the synthetic routes generated by mUCT-dc-V using Round 1 RL value network when compared with PUCT-bootstrapping as the baseline method.

Number of improved compounds	206
Root CSS average improved by	0.41
Number of unchanged compounds	15
Number of worsened compounds	79
Root CSS average worsened by	0.22
Average root CSS score changed by	0.22

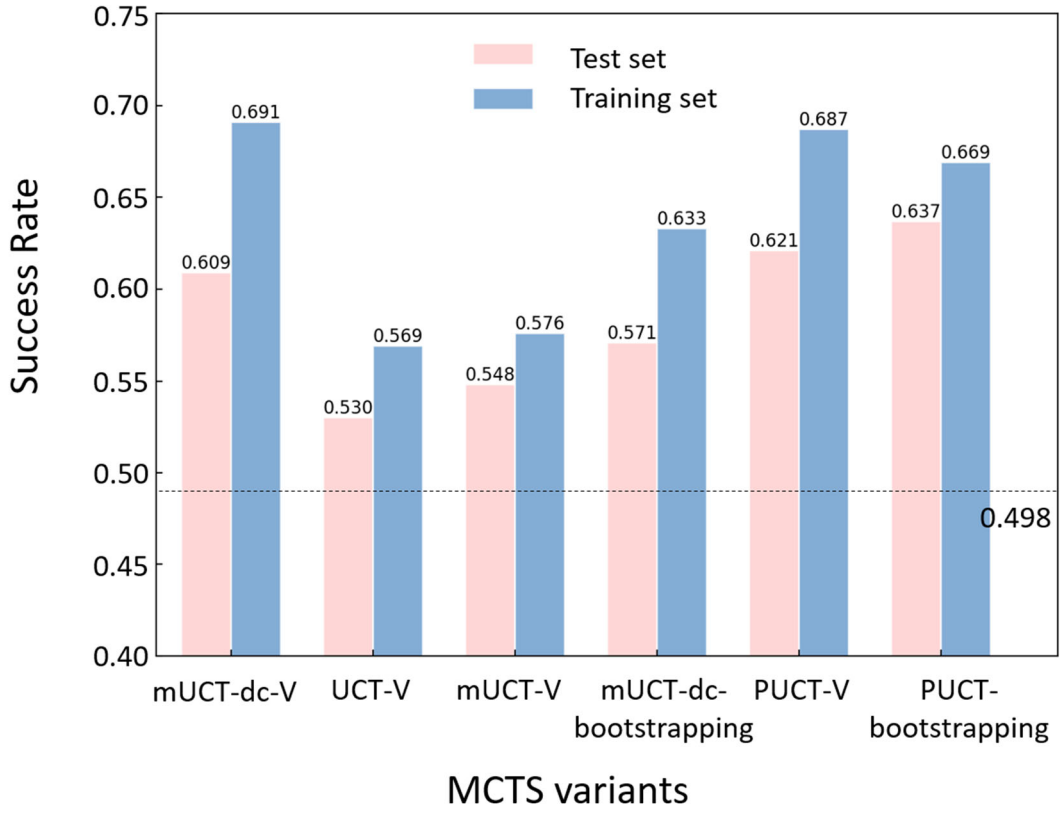

Figure S1 The success rate of finding buyable synthesis pathways by MCTS variants for 30s on 1000 compounds from test set and training set respectively, where the values of compounds are solely given by the Round 1 RL value network (in the MCTS variants requiring a value network) without setting the values of the buyable compounds to 1. The line of 0.498 indicates the performance of a random value network with mUCT-dc-V method on the test set. The *c* values are the same as in Fig. 3 in the main text.

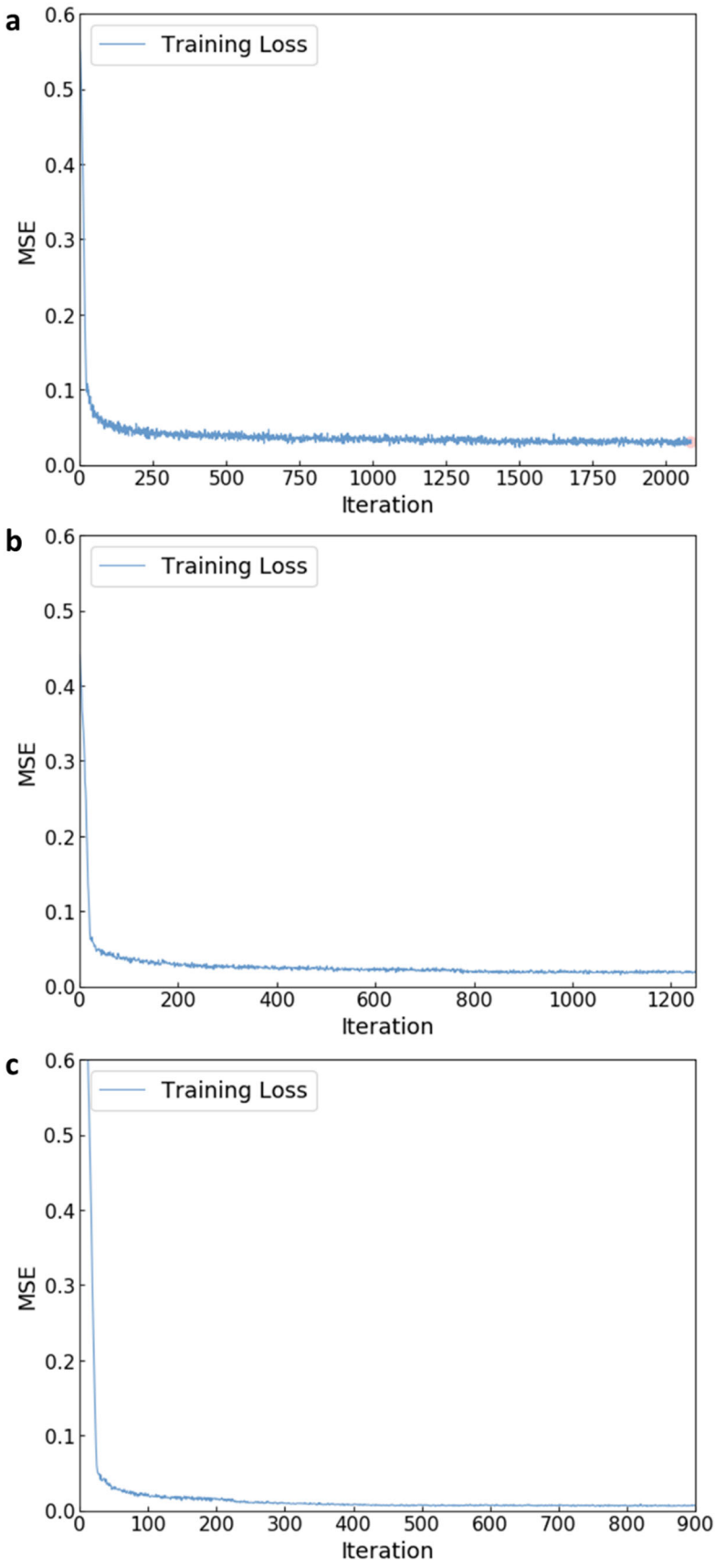

Figure S2 The mean squared error (MSE) vs. iteration curve during the training phase of **(a)**Round 1 RL value network, **(b)**Round 2 RL value network, and **(c)** the compound solvent score value network. The batch size for each iteration is 1000 compounds.