One stopping condition (c.f. [55, 65]) is to stop when

$$
\begin{equation*}
\hat{\mu}_{h_{t}, T_{h_{t}}(t)}-B_{h_{t}, T_{h_{t}}(t)}>\hat{\mu}_{\ell_{t}, T_{\ell_{t}}(t)}+B_{\ell_{t}, T_{\ell_{t}}(t)} \tag{4.9}
\end{equation*}
$$

and output $h_{t}$. Alternatively, as is proposed and shown to work in this manuscript, one can stop when

$$
\begin{equation*}
\exists i \in[n]: T_{i}(t)>\alpha \sum_{j \neq i} T_{j}(t) \tag{4.10}
\end{equation*}
$$

and output $\arg \max _{i} T_{i}(t)$ for some $\alpha>0$.
While UCB sampling strategies were originally designed for the regret setting to optimize "exploration versus exploitation" [64], it was shown in [65] that UCB strategies were also effective in the pure exploration (find the best) setting. These algorithms are attractive because they are more sequential than the AE algorithms that tend to act more like uniform sampling for the first several epochs.

- LUCB (a variation on UCB) - [60,69] Sample all arms once. For each time $t>n$ sample the arms indexed by $h_{t}$ and $\ell_{t}$ (i.e. at each time $t$ two arms are sampled) and stop when the criterion defined in (4.9) is met.

While the LUCB and UCB sampling strategies appear to be only subtly different, the LUCB strategies appear to be better designed for exploration than UCB sampling strategies. For instance, given just two arms, the most reasonable strategy would be to sample both arms the same number of times until a winner could be confidently proclaimed, which is what LUCB would do. On the other hand, UCB strategies would tend to sample the best arm far more than the second-best arm

