Simulation experiments for hide-and-seek with different seeker distribution update strategies

Tirtharaj Dash

January 18, 2020

1 Setup

We perform some simulation experiments for Hide-and-Seek with three different **seeker distribution update strategies**:

(1) **No update:** No update of the seeker distribution (leads to hide-and-seek with replacement results).

(2) **Uniform update:** Open a box, distribute its probability mass to every other unopened boxes, make its probability 0 (hide-and-seek without replacement).

(3) **Hot-cold update:** The seeker updates its probability distribution based on whether it opened a cold box or a hot box. A cold box is a box for which the performance of a box ($perf$) is less than the cold threshold (θ_c) and a hot box is a box with performance greater than a hot threshold (θ_h). We devise the following procedure for update:

1. Open a box i
2. If $perf(i) \geq \theta_h$: distribute its probability mass to all the unopened boxes in its neighbors.
3. If $perf(i) \leq \theta_c$: distribute its probability mass to all the unopened boxes except its neighbors.
4. If none of 2 or 3: distribute its probability mass to all the unopened boxes.
5. Repeat 1–4 until the hider is found.

The seeker update strategy (3) requires that the hider distribution falls into some continuity assumption. That is: the probability mass of a neighborhood of a box are in monotonic relationship to the probability of that box. This is a realistic demand and clauses which are related with each other are monotonic in their performance in some fashion. We construct such a hider distribution (H) with the following code[1].

2 Experiments

Parameter setting The experiments are performed for number of boxes $n = \{1000, 2000, 3000\}$. The maximum hiding trials is set at 1000. We call it a **failure**, if the hider is not found within n searches by using the seeker distribution. The neighborhood size (nbd) is varied as {1,2,3}. For all the experiments reported here, we define performance of a box by: $perf(i) = \frac{h_i}{\max(h_1,\ldots,h_n)}$, where $H = \{h_1, \ldots, h_n\}$ is the hider distribution. The thresholds are fixed at $\theta_h = 0.80$ and $\theta_c = 0.4$. The proportion of boxes that have high probability mass (spikes in H) is fixed at 10%.

[1] `https://github.com/tirtharajdash/multimodalGaussianDistro`

Results The mean and standard deviations of misses are calculated only for successful runs i.e. the hider was found by the seeker within maximum of n look-ups. Otherwise, it was treated as a failure and this result was not included for statistics. Below, we report results for each seeker update strategies.

choiceUpdS	SuccessRate	mean(misses)	sd(misses)
$n = 1000$			
1	0.622	439.738	279.218
2	1.000	499.379	279.060
3 ($nbd = 1$)	1.000	486.041	271.251
3 ($nbd = 2$)	1.000	458.027	269.987
3 ($nbd = 3$)	1.000	479.270	279.110
$n = 2000$			
1	0.635	859.123	571.302
2	1.000	1022.204	595.486
3 ($nbd = 1$)	1.000	963.457	550.133
3 ($nbd = 2$)	1.000	961.946	551.028
3 ($nbd = 3$)	1.000	968.335	564.515
$n = 3000$			
1	0.652	1303.462	854.176
2	1.000	1473.717	865.756
3 ($nbd = 1$)	1.000	1436.214	801.100
3 ($nbd = 2$)	1.000	1396.195	840.066
3 ($nbd = 2$)	1.000	1464.990	840.695

Figure 1: Average number of misses: H is a multimodal distribution with 10% spikes, and S is updated using three different update strategies (1: no update, 2: without replacement, 3: update using θ_h and θ_c) using three different neighborhood structures {1,2,3}.

Interpretation The results suggest that the threshold based seeker update strategy (3):

- Reduces the average number of misses in comparison with search without replacement for which the expected number of misses is $\frac{n-1}{2}$ (i.e. choiceUpdS= 2 in tables).
- For the seeker updat type (3), best neighborhood size is found to be 2.
- Increasing the neighborhood size of a box increases the average number of misses in almost all the cases.