BDI-Learning Discussion Paper: Understanding Stability

Dhirendra Singh
dhirendra.singh@rmit.edu.au

February 18, 2010

1 Background and Scope

In this section we analyse the operation of the BUL scheme of [1] and highlight any nuances in it's operation. In particular we focus on the claim that BUL is able to learn well with applicability thresholds and show two cases where this claim fails.

2 Case 1: Stability without Success

This example highlights the case where a plan P, that holds a solution for world w, may nonetheless become stable *without* the solution being found. Consider the goal-plan hierarchy T of Figure 1. This is the structure where the BUL approach has a general advantage over ACL. Assume similar $k = 3$ and $\epsilon = 0.3$ values for stability calculation as used for the BUL specification in [1].

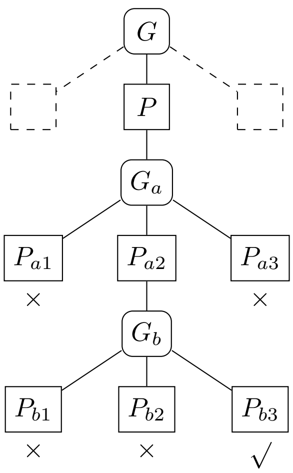

Figure 1: Goal-plan hierarchy T.

Let us take up the analysis at the point where plan P has been executed six times in the world w. The breakdown of the six executions is as follows: each of the three plans P_{a*} has been executed twice and for each choice of P_{a2}, the plans P_{b1} and P_{b2} respectively were selected. As such, so far no plan has been deemed stable, and success has not been found. The executions are given by the traces below:

$\lambda 1 = G[P : w] \cdot G_a[P_{a1} : w]$.

$\lambda 2 = G[P : w] \cdot G_a[P_{a2} : w] \cdot G_b[P_{b1} : w]$.

$\lambda 3 = G[P : w] \cdot G_a[P_{a3} : w]$.

$\lambda 4 = G[P : w] \cdot G_a[P_{a1} : w]$.

$\lambda 5 = G[P : w] \cdot G_a[P_{a2} : w] \cdot G_b[P_{b2} : w]$.

$\lambda 6 = G[P : w] \cdot G_a[P_{a3} : w]$.

Up until now, the selection probabilities for the three P_{a*} plans is the same and equal to 0.5. In the seventh execution, let us assume P_{a1} is selected, so that it now becomes stable and records the failure.

$\lambda 7 = G[P : w] \cdot G_a[P_{a1} : w]$.

The selection probabilities for the three P_{a*} plans are now $[0.0, 0.5, 0.5]$ since the newly constructed decision tree of P_{a1} predicts $p = 0.0$ for w. Let us assume P_{a2} is now selected at execution eight along with P_{b1} below.

$\lambda 8 = G[P : w] \cdot G_a[P_{a2} : w] \cdot G_b[P_{b1} : w]$.

On this failure P_{b1} will not record as it is not stable. P_{a2} that has had three execution in w is now stable in it's own right — but will also not record the failure since the choice of P_{b1} below is not stable.

Finally, in the ninth execution, let's say P_{a3} is selected, so that it now becomes stable and records the failure.

$\lambda 9 = G[P : w] \cdot G_a[P_{a3} : w]$.

After recording the failure in P_{a3}, the $RecordFailedTrace$ algorithm of [1] will now check to see if the failure should be propagated further up (in the trace $\lambda 9$), i.e. should the failure be recorded for P, or, is goal G_a stable for w.

This is where it becomes interesting: G_a turns out to be stable for w because all it's plans are by now individually stable (i.e. satisfy the k and ϵ requirements). P_{a1} became stable at $\lambda 7$, P_{a2} at $\lambda 8$, and P_{a3} at $\lambda 9$. As such, the failure is propagated up and P records a failure.

The impact is that the predicted probability for P_{a2} in w drops from 0.5 to 0.0 and subsequently causes learning to fail where applicability thresholds are involved.

This is a simple example to highlight the issue. The problem becomes more pronounced as we make P_{a2} more "complex".

3 Case 2: Stability and Multiple Worlds

The second case highlights the disconnect between the fact that stability is a *per world* concept whereas the decision tree spans *multiple worlds*. So while stability ensures that the training set for a decision tree contains well-informed samples for some world, it says nothing about what this means for the use of this decision tree in other worlds.

Consider again the structure T of Figure 1 and where we left off at $\lambda 9$. The result was that P_{a1} and P_{a3} both had recorded a failure in w while P_{a2} has not yet recorded anything.

At this point let's say that P is selected for a different world w_1. The resulting probabilities for $[P_{a1}, P_{a2}, P_{a3}]$ are now $[0.0, 0.5, 0.0]$ since the two decision trees built in w will interpolate the failure result to w_1.

It is equally possible that P_{a2} becomes stable with a recorded failure for yet another world w_2. If that were the case, then the resulting probabilities in w_1 would be $[0.0, 0.0, 0.0]$ due to interpolation.

The impact is that the selection probabilities in w_1 may fall below the applicability threshold even when w_1 has never been witnessed before. As a result, once again the learning may fail.

References

[1] D. Singh, S. Sardina, L. Padgham, and S. Airiau. Learning Context Conditions for BDI Plan Selection. In *Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2010.