Notes: PowerPlay [1]

Xingdong Zuo

IDSIA, Switzerland

February 26, 2018

1 Key idea

Incrementally search the simplest pair of (task, solver) without forgetting old skills. Suppose old solver can solve a set of tasks, we search for new task under following conditions

1. New task cannot be solved by old solver
2. New task can be solved by new solver (updated from old solver)
3. New solver can also solve old skills (Prevent forgetting/Reuse old skills)

2 Algorithm

Let $\mathcal{B}$ be a set of bit strings encoding programs such as scalar, vector, matrices etc. We define $\mathcal{S} \subset \mathcal{B}$ as a set of possible solvers and $\mathcal{T} \subset \mathcal{B}$ as a set of possible tasks. The PowerPlay algorithm is shown in the following

Algorithm 1: PowerPlay

```
Initialize a solver $s_0$
for $i = 1, 2, \dots$ do
    repeat
        Invent new task $T \in \mathcal{T}$ and update a new solver candidate $s_i^{'}$ from $s_{i-1}$
    until $s_{i-1}$ cannot solve $T$ and $s_i^{'}$ can solve both $T$ and $T_{k<i}$;
    Set $T_i \leftarrow T$ and $s_i \leftarrow s_i^{'}$
end
```

References

[1] Jürgen Schmidhuber. “Powerplay: Training an increasingly general problem solver by continually searching for the simplest still unsolvable problem”. In: *Frontiers in psychology* 4 (2013), p. 313.