Goal-directed Procedure Learning

Patrick Hammer (OpenNARS)

Tony Lofthouse (OpenNARS)

The Problem

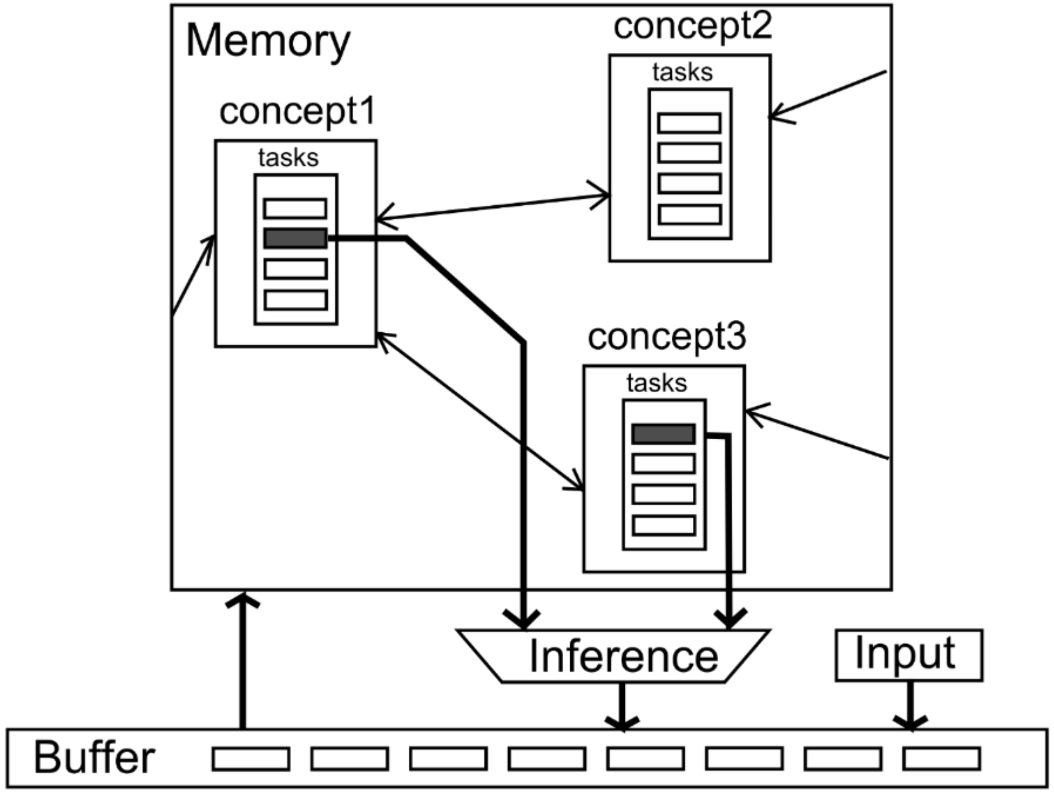
- Given a goal **G!**, which a system wants to make as true as possible, how does it satisfy the goal?

OpenNARS Operating Cycle

A reasoning cycle in NARS is a boundless loop:

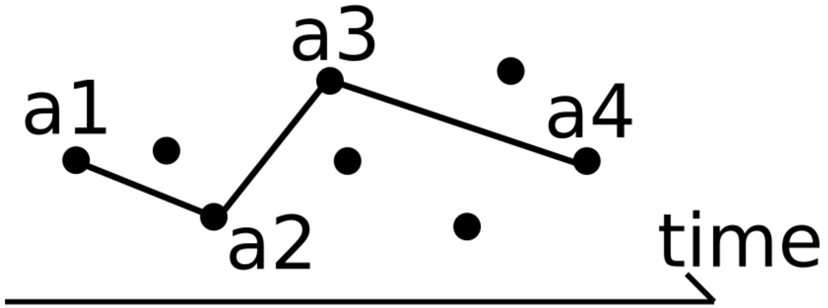
- Select a concept ***A****
- Select premise 1 task from concept ***A****
- Select a neighbour concept ***B***
- Select a premise 2 belief from concept **B***
- Carry out inference on the premises
- Buffer inference results back to memory
- Repeat…

* = under attentional control

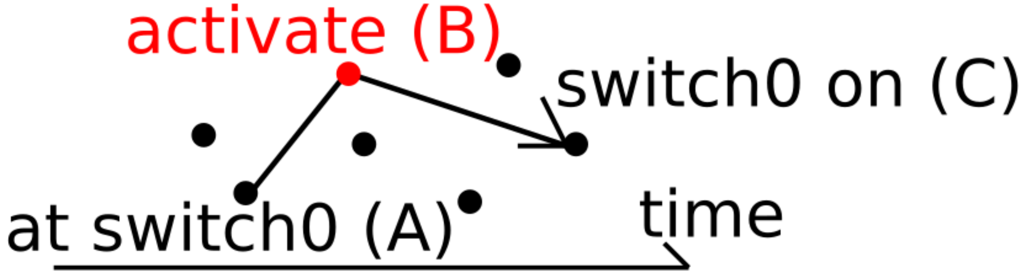


Procedure Learning: Temporal Reasoning

- A temporal pattern can be represented as **(a1, …, an)**



- In general a hypothesis can be defined as **A ==> B**
- With a special case defined as **(A, B) ==> C**
  - Where **A** is the antecedent, **B** an operation and **C** the consequent
- And with temporal constraints: **(A, B) =/> C** where **C** occurs in the future (A prediction)



Procedure Learning: Hypothesis Creation

- The crucial insight was to separate the incoming experience stream into events and operations (as operations also generate input events as feedback).
- With this separation the task of forming meaningful preconditions was a simpler problem to solve: operations simply become the context under which certain events cause others to occur
- Now when a new event **E*** enters the system and an operation **Op** is sampled from Operations, two steps occur:
  - Sample a second premise $\mathbf{E_{past}}$ from Events to form more complex precondition events, such as temporal sequences $\mathbf{(E_{past},E+)}$
  - Retrieve one of the **(E, Op)** preconditons and form **(E, Op) =/> E***

Procedure Learning: Hypothesis Selection

- Assuming an incoming or derived goal **G!** and a recently experienced event **E**, the task of hypothesis selection is to choose the most relevant hypothesis that can satisfy **G!** with **E** as precondition.

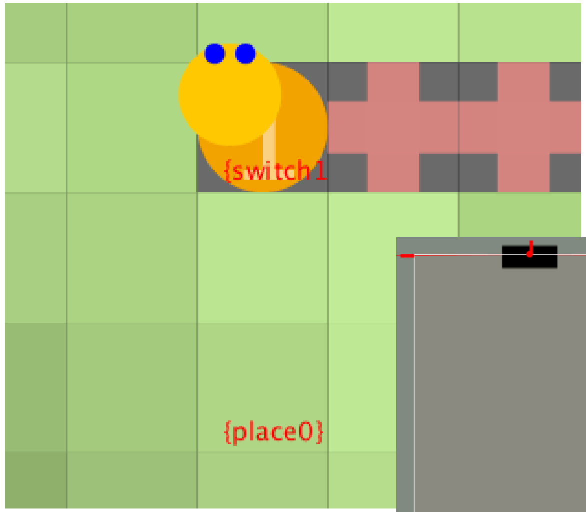
To be relevant, the hypothesis needs to be applicable to the current situation:

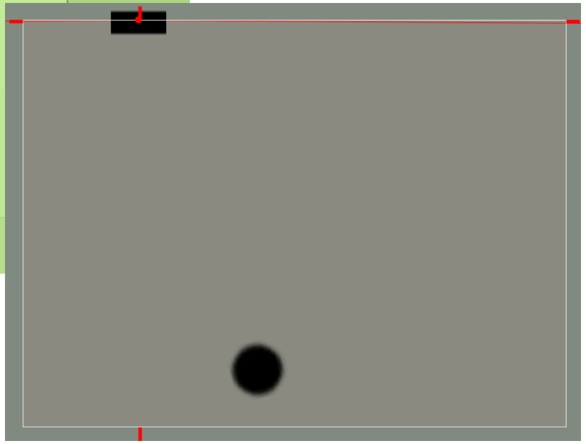
- The consequent is desired and has sufficient priority
- The hypothesis has high truth expectation, predicts successfully
- The antedecent incorporates recent events in its preconditions.

Procedure Learning: Hypothesis Testing

- Given the uncertain nature of input experience, it is not possible, in advance, to identify what will be relevant and useful.
- Learning is necessary and demands hypotheses to be tested: given a predictive hypothesis of the form: **antecedent =/> consequent**, we define Anticipation as a predicted event corresponding to the antecedent, on which the system forms an expectation that it will happen. With Anticipation a system is able to find negative evidence for such hypotheses too.

Results





- Test Chamber
  - A possible event stream generated from observing a user:
    - 1. Event **E1**: Reached start place0
    - 2. Operation **Op1**: Go to switch1
    - 3. Event **E2**: Reached switch1
    - 4. Operation **Op2**: Activate switch1
    - 5. Event **E3**: Switch1 activated
  - OpenNARS easily creates hypothesis **((E1,Op1) =/> E3)** by making use of the explained mechanisms. The same is true for **((E2,Op2) =/> E3)**.
- Reinforcement Learning
  - Learning Pong:
    - assuming a goal **G!**, an input event **G.** can directly act as "reward signal",
    - which in Pong basically is an event encoding "the ball collided with the bat".
    - Additionally, the system is given events about the ball position relative to the bat.
    - This allows it to invoke different operations dependent on whether the ball is left or right of the bat.
  - This turned out to be sufficient to let the system learn to play Pong in a very short time with high reliability

Conclusion

- The Reasoning-Learning Mechanism employed by NARS has been shown to be capable of goal-directed Procedure Learning:
  - the separation of operations from other events has enabled the system to form more successful and useful hypotheses with little resource effort.
  - These are subsequently used to enable fast hypothesis selection through the precondition memorization mechanism.
  - This mechanism allows the system to make effective use of procedural knowledge and have fast response times when the relevant knowledge already exists.
- Collectively our methods allow the system to self-program and automatize itself to become gradually more competent over time.
- We have shown that the system can learn goal-oriented procedures involving multiple operations, without building explicit plans.
- Furthermore, we have demonstrated, that the system can perform well in Reinforcement-Learning style tasks as a special case, and that the “reward signal” can naturally be represented