Unified Rule Engine

Usage and Control

Nil Geisweiller

SingularityNetathon 2018

OpenCog Foundation

URE: Unified Rule Engine

The URE is a tool to evolve Inference Trees

```
A  A->B
-------(MP)
   B
```

```
    A->B  B->C
    ----------(DED)
A      A->C
-----------(MP)
     C
```

```
ForAll X P(X)  U(P(X), A)        A->B  B->C
-------------------------(INS)   ----------(DED)
            A                       A->C
            ----------------------------(MP)
                           C
```

- Leaves are premises
- Roots are conclusions

URE: Unified Rule Engine

Inference Trees are constructed by composing Rules

- forward: expand conclusion

```
A  A->B
-------(MP)
   B
```

⇒

```
A   A->B
--------(MP)
   B         B->C
   -------------(MP)
          C
```

URE: Unified Rule Engine

Inference Trees are constructed by composing Rules

- forward: expand conclusion

```
A  A->B
-------(MP)
   B
```

⇒

```
A   A->B
--------(MP)
   B         B->C
   -------------(MP)
          C
```

- backward: expand premises

```
A  A->B
-------(MP)
   B
```

⇒

```
ForAll X P(X)  U(P(X), A)
-------------------------(INS)
            A                        A->B
            ----------------------------(MP)
                           B
```

```
A  A->B
-------(MP)
   B
```

⇒

```
        A->C   C->B
        -----------(DED)
A          A->B
---------------(MP)
       B
```

URE: Unified Rule Engine

Inference Trees are Atomese Programs

```
A   A->B
-------(MP)
   B
```

```
(BindLink
  (VariableList                                              \
    (TypedVariableLink                                       |
      (VariableNode "$A")                                    |
      (TypeChoice                                            |
        (TypeNode "LambdaLink")                              |
        (TypeNode "PredicateNode")                           |
      )                                                      |
    )                                                        | Variables
    (TypedVariableLink                                       |
      (VariableNode "$B")                                    |
      (TypeChoice                                            |
        (TypeNode "LambdaLink")                              |
        (TypeNode "PredicateNode")                           |
      )                                                      |
    )                                                        |
  )                                                          /
  (AndLink                                                   \
    (ImplicationLink                                         |
      (VariableNode "$A")                                    | Pattern
      (VariableNode "$B")                                    |
    )                                                        /
    (EvaluationLink                                          \
      (GroundedPredicateNode "scm: true-enough")             |
      (ImplicationLink                                       |
        (VariableNode "$A")                                  |
        (VariableNode "$B")                                  |
      )                                                      |
    )                                                        | Preconditions
    (EvaluationLink                                          |
      (GroundedPredicateNode "scm: true-enough")             |
      (VariableNode "$A")                                    |
    )                                                        |
  )                                                          /
  (ExecutionOutputLink                                       \
    (GroundedSchemaNode "scm: modus-ponens-formula")         |
    (ListLink                                                |
      (VariableNode "$B")                                    |
      (VariableNode "$A")                                    |
      (ImplicationLink                                       | Production
        (VariableNode "$A")                                  |
        (VariableNode "$B")                                  |
      )                                                      |
    )                                                        |
  )                                                          /
```

URE: Unified Rule Engine

Inference Trees are Atomese Programs

```
A  A->B
-------(MP)
   B
```

```
(ExecutionOutputLink
  (GroundedSchemaNode "scm: modus-ponens-formula")
  (ListLink
    (VariableNode "$B")
    (VariableNode "$A")
    (ImplicationLink
      (VariableNode "$A")
      (VariableNode "$B")
    )
  )
)
```

URE: Unified Rule Engine

Inference Trees are Atomese Programs

```
     A->C   C->B
     -----------(DED)
A       A->B
------------(MP)
     B
```

```
(ExecutionOutputLink
  (GroundedSchemaNode "scm: modus-ponens-formula")
  (ListLink
    (VariableNode "$B")
    (VariableNode "$A")
    (ExecutionOutputLink
      (GroundedSchemaNode "scm: deduction-formula")
      (ListLink
        (ImplicationLink
          (VariableNode "$A")
          (VariableNode "$B")
        )
        (ImplicationLink
          (VariableNode "$A")
          (VariableNode "$C")
        )
        (ImplicationLink
          (VariableNode "$C")
          (VariableNode "$B")
        )
      )
    )
  )
)
```

URE: Unified Rule Engine

Algorithm

1. Select an inference tree to expand
2. Select a node from that tree to expand
3. Select a rule to expand with
4. Expand the inference tree and place it back to the pool of inference trees. Repeat till termination.

Combinatorial Explosion

Inference Control

Delegate the hard decisions to a cognitive process

Cognitive Schematics:

$$\text{Context} \ \& \ \text{Action} \Rightarrow \text{Goal}$$

Atomese:

```
Implication <TV>
  And
    <Context>
    <Action>
  <Goal>
```

Inference Control

```
A  A->B
-------(MP)
   B
```

Which rule to choose?

1. Modus Ponens
2. Universal Instantiation

Look for:

```
Implication <TV>
  And
    <inference-tree-pattern>
    <node-pattern>
    <rule-pattern>
  <produce-good-inference>
```

Inference Control

Examples of actual URE Cognitive Schematics

```
(ImplicationScopeLink (stv 0.45945946 0.04625)
   (VariableList
      (VariableNode "$T")
      (TypedVariableLink
         (VariableNode "$A")
         (TypeNode "DontExecLink")
      )
      (VariableNode "$L")
      (TypedVariableLink
         (VariableNode "$B")
         (TypeNode "DontExecLink")
      )
   )
   (AndLink
      (EvaluationLink
         (PredicateNode "URE:BC:preproof-of")
         (ListLink
            (VariableNode "$A")
            (VariableNode "$T")
         )
      )
      (ExecutionLink
         (SchemaNode "URE:BC:expand-and-BIT")
         (ListLink
            (VariableNode "$A")
            (VariableNode "$L")
            (DontExecLink
               (DefinedSchemaNode "deduction-inheritance-rule")
            )
         )
         (VariableNode "$B")
      )
   )
   (EvaluationLink
      (PredicateNode "URE:BC:preproof-of")
      (ListLink
         (VariableNode "$B")
         (VariableNode "$T")
      )
   )
)
```

```
(ImplicationScopeLink (stv 1 0.00625)
   (VariableList
      (VariableNode "$T")
      (TypedVariableLink
         (VariableNode "$A")
         (TypeNode "DontExecLink")
      )
      (VariableNode "$X")
      (TypedVariableLink
         (VariableNode "$B")
         (TypeNode "DontExecLink")
      )
   )
   (AndLink
      (EvaluationLink
         (PredicateNode "URE:BC:preproof-of")
         (ListLink
            (VariableNode "$A")
            (VariableNode "$T")
         )
      )
      (ExecutionLink
         (SchemaNode "URE:BC:expand-and-BIT")
         (ListLink
            (VariableNode "$A")
            (InheritanceLink
               (ConceptNode "a")
               (VariableNode "$X")
            )
            (DontExecLink
               (DefinedSchemaNode "conditional-full-instantiation-implication-scope-meta-rule"
            )
         )
         (VariableNode "$B")
      )
   )
   (EvaluationLink
      (PredicateNode "URE:BC:preproof-of")
      (ListLink
         (VariableNode "$B")
         (VariableNode "$T")
      )
   )
)
```

Inference Control

How to get Cognitive Schematics?

Answer: Learning

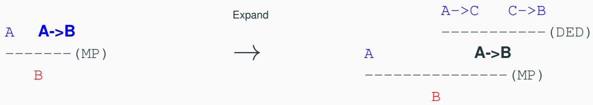
Record a trace of all decisions
the URE takes and learn from it

Inference Control Learning

How to learn Cognitive Schematics?

1. URE
2. Pattern Mining
3. MOSES
4. Others

Inference Control Learning: URE Trace



```
A   A->B                 Expand                 A->C   C->B
--------(MP)               →                    ------------(DED)
   B                                    A                A->B
                                        ----------------(MP)
                                                     B
```

Atomese:

```
(ExecutionLink (stv 1 1)
  (SchemaNode "URE:BC:expand-and-BIT")
  (ListLink
    (DontExecLink
      (BindLink
        (VariableNode "$A-591b9032")
        (AndLink
          (VariableNode "$A-591b9032")
          (InheritanceLink
            (VariableNode "$A-591b9032")
            (InheritanceLink (stv 1 1)
              (ConceptNode "v" (stv 0.84615385 1))
              (ConceptNode "w")
            )
          )
          (EvaluationLink
            (GroundedPredicateNode "scm: gt-zero-confidence")
            (VariableNode "$A-591b9032")
          )
          (EvaluationLink
            (GroundedPredicateNode "scm: gt-zero-confidence")
            (InheritanceLink
              (VariableNode "$A-591b9032")
              (InheritanceLink (stv 1 1)
                (ConceptNode "v" (stv 0.84615385 1))
                (ConceptNode "w")
              )
            )
          )
        )
        (ExecutionOutputLink
          (GroundedSchemaNode "scm: modus-ponens-formula")
          (ListLink
            (InheritanceLink (stv 1 1)
              (ConceptNode "v" (stv 0.84615385 1))
              (ConceptNode "w")
            )
            (InheritanceLink
              (VariableNode "$A-591b9032")
              (InheritanceLink (stv 1 1)
                (ConceptNode "v" (stv 0.84615385 1))
                (ConceptNode "w")
              )
            )
            (VariableNode "$A-591b9032")
          )
        )
      )
    )
    (VariableNode "$A-1860e3b6")
    (DontExecLink
      (DefinedSchemaNode "deduction-inheritance-rule")
    )
  )
```

```
(DontExecLink
  (BindLink
    (VariableList
      (TypedVariableLink
        (VariableNode "$A-2fbf17d5")
        (TypeChoice
          (TypeNode "ConceptNode")
          (TypeNode "AndLink")
          (TypeNode "OrLink")
          (TypeNode "NotLink")
        )
      )
      (TypedVariableLink
        (VariableNode "$C-2bdc1299")
        (TypeChoice
          (TypeNode "ConceptNode")
          (TypeNode "AndLink")
          (TypeNode "OrLink")
          (TypeNode "NotLink")
        )
      )
      (TypedVariableLink
        (VariableNode "$B-4a714efb")
        (TypeChoice
          (TypeNode "ConceptNode")
          (TypeNode "AndLink")
          (TypeNode "OrLink")
          (TypeNode "NotLink")
        )
      )
    )
    (AndLink
      (EvaluationLink
        (GroundedPredicateNode "scm: gt-zero-confidence")
        (InheritanceLink
          (InheritanceLink
            (VariableNode "$A-2fbf17d5")
            (VariableNode "$C-2bdc1299")
          )
          (InheritanceLink (stv 1 1)
            (ConceptNode "v" (stv 0.84615385 1))
            (ConceptNode "w")
          )
        )
      )
      (InheritanceLink
        (VariableNode "$B-4a714efb")
        (VariableNode "$C-2bdc1299")
      )
      (InheritanceLink
        (InheritanceLink
          (VariableNode "$A-2fbf17d5")
          (VariableNode "$C-2bdc1299")
        )
        (InheritanceLink (stv 1 1)
          (ConceptNode "v" (stv 0.84615385 1))
          (ConceptNode "w")
        )
      )
      (NotLink
        (IdenticalLink
          (VariableNode "$C-2bdc1299")
          (VariableNode "$A-2fbf17d5")
        )
      )
      (InheritanceLink
        (VariableNode "$A-2fbf17d5")
        (VariableNode "$B-4a714efb")
      )
    )
```

```
      (ExecutionOutputLink
        (GroundedSchemaNode "scm: modus-ponens-formula")
        (ListLink
          (InheritanceLink (stv 1 1)
            (ConceptNode "v" (stv 0.84615385 1))
            (ConceptNode "w")
          )
          (InheritanceLink
            (InheritanceLink
              (VariableNode "$A-2fbf17d5")
              (VariableNode "$C-2bdc1299")
            )
            (InheritanceLink (stv 1 1)
              (ConceptNode "v" (stv 0.84615385 1))
              (ConceptNode "w")
            )
          )
          (ExecutionOutputLink
            (GroundedSchemaNode "scm: deduction-formula")
            (ListLink
              (InheritanceLink
                (VariableNode "$A-2fbf17d5")
                (VariableNode "$C-2bdc1299")
              )
              (InheritanceLink
                (VariableNode "$A-2fbf17d5")
                (VariableNode "$B-4a714efb")
              )
              (InheritanceLink
                (VariableNode "$B-4a714efb")
                (VariableNode "$C-2bdc1299")
              )
            )
          )
        )
      )
    )
  )
)
```

Inference Control Learning: URE Trace

```
       A->C   C->B
       -----------(DED)
A          A->B
---------------(MP)
       B
```

$\xrightarrow{\text{Quality}}$ (stv 0.0001, 0.00075)

Atomese:

```
(EvaluationLink (stv 0.0001 0.00075)
   (PredicateNode "URE:BC:preproof-of")
   (ListLink
      (DontExecLink
         (BindLink
            (VariableList
               (TypedVariableLink
                  (VariableNode "$A-2fbf17d5")
                  (TypeChoice
                     (TypeNode "ConceptNode")
                     (TypeNode "AndLink")
                     (TypeNode "OrLink")
                     (TypeNode "NotLink")
                  )
               )
               (TypedVariableLink
                  (VariableNode "$C-2bdc1299")
                  (TypeChoice
                     (TypeNode "ConceptNode")
                     (TypeNode "AndLink")
                     (TypeNode "OrLink")
                     (TypeNode "NotLink")
                  )
               )
               (TypedVariableLink
                  (VariableNode "$B-4a714efb")
                  (TypeChoice
                     (TypeNode "ConceptNode")
                     (TypeNode "AndLink")
                     (TypeNode "OrLink")
                     (TypeNode "NotLink")
                  )
               )
            )
            (AndLink
               (EvaluationLink
                  (GroundedPredicateNode "scm: gt-zero-confidence")
                  (InheritanceLink
                     (InheritanceLink
                        (VariableNode "$A-2fbf17d5")
                        (VariableNode "$C-2bdc1299")
                     )
                     (InheritanceLink (stv 1 1)
                        (ConceptNode "v" (stv 0.84615385 1))
                        (ConceptNode "w")
                     )
                  )
               )
               (InheritanceLink
                  (VariableNode "$B-4a714efb")
                  (VariableNode "$C-2bdc1299")
               )
               (InheritanceLink
                  (InheritanceLink
                     (VariableNode "$A-2fbf17d5")
                     (VariableNode "$C-2bdc1299")
                  )
                  (InheritanceLink (stv 1 1)
                     (ConceptNode "v" (stv 0.84615385 1))
                     (ConceptNode "w")
                  )
               )
               (NotLink
                  (IdenticalLink
                     (VariableNode "$C-2bdc1299")
                     (VariableNode "$A-2fbf17d5")
                  )
               )
               (InheritanceLink
                  (VariableNode "$A-2fbf17d5")
                  (VariableNode "$B-4a714efb")
               )
            )
            (ExecutionOutputLink
               (GroundedSchemaNode "scm: modus-ponens-formula")
               (ListLink
                  (InheritanceLink (stv 1 1)
                     (ConceptNode "v" (stv 0.84615385 1))
                     (ConceptNode "w")
                  )
                  (InheritanceLink
                     (InheritanceLink
                        (VariableNode "$A-2fbf17d5")
                        (VariableNode "$C-2bdc1299")
                     )
                     (InheritanceLink (stv 1 1)
                        (ConceptNode "v" (stv 0.84615385 1))
                        (ConceptNode "w")
                     )
                  )
                  (ExecutionOutputLink
                     (GroundedSchemaNode "scm: deduction-formula")
                     (ListLink
                        (InheritanceLink
                           (VariableNode "$A-2fbf17d5")
                           (VariableNode "$C-2bdc1299")
                        )
                        (InheritanceLink
                           (VariableNode "$A-2fbf17d5")
                           (VariableNode "$B-4a714efb")
                        )
                        (InheritanceLink
                           (VariableNode "$B-4a714efb")
                           (VariableNode "$C-2bdc1299")
                        )
                     )
                  )
               )
            )
         )
      )
      (InheritanceLink (stv 1 1)
         (ConceptNode "v" (stv 0.84615385 1))
         (ConceptNode "w")
      )
   )
)
```

Inference Control Learning: URE

Call the URE with a Control Rule Template as target

```
(cog-bc icr-rb (QuoteLink
                  (ImplicationScopeLink
                     (UnquoteLink
                        (VariableNode "$impl-vardecl")
                     )
                     (AndLink
                        (EvaluationLink
                           (PredicateNode "URE:BC:preproof-of")
                           (UnquoteLink
                              (VariableNode "$preproof-A-args")
                           )
                        )
                        (ExecutionLink
                           (SchemaNode "URE:BC:expand-and-BIT")
                           (UnquoteLink
                              (VariableNode "$expand-inputs")
                           )
                           (UnquoteLink
                              (VariableNode "$expand-output")
                           )
                        )
                     )
                     (EvaluationLink
                        (PredicateNode "URE:BC:preproof-of")
                        (UnquoteLink
                           (VariableNode "$preproof-B-args")
                        )
                     )
                  )
               )
            )
```

Inference Control Learning: URE

Call the URE with a Control Rule Template as target

```
(cog-bc icr-rb (QuoteLink
                  (ImplicationScopeLink
                     (UnquoteLink
                        (VariableNode "$impl-vardecl")
                     )
                     (AndLink
                        (EvaluationLink
                           (PredicateNode "URE:BC:preproof-of")
                           (UnquoteLink
                              (VariableNode "$preproof-A-args")
                           )
                        )
                        (ExecutionLink
                           (SchemaNode "URE:BC:expand-and-BIT")
                           (UnquoteLink
                              (VariableNode "$expand-inputs")
                           )
                           (UnquoteLink
                              (VariableNode "$expand-output")
                           )
                        )
                     )
                     (EvaluationLink
                        (PredicateNode "URE:BC:preproof-of")
                        (UnquoteLink
                           (VariableNode "$preproof-B-args")
                        )
                     )
                  )
               )
            )
```

```
(ImplicationScopeLink (stv 0.45945946 0.04625)
   (VariableList
      (VariableNode "$T")
      (TypedVariableLink
         (VariableNode "$A")
         (TypeNode "DontExecLink")
      )
      (VariableNode "$L")
      (TypedVariableLink
         (VariableNode "$B")
         (TypeNode "DontExecLink")
      )
   )
   (AndLink
      (EvaluationLink
         (PredicateNode "URE:BC:preproof-of")
         (ListLink
            (VariableNode "$A")
            (VariableNode "$T")
         )
      )
      (ExecutionLink
         (SchemaNode "URE:BC:expand-and-BIT")
         (ListLink
            (VariableNode "$A")
            (VariableNode "$L")
            (DontExecLink
               (DefinedSchemaNode "deduction-inheritance-rule")
            )
         )
         (VariableNode "$B")
      )
   )
   (EvaluationLink
      (PredicateNode "URE:BC:preproof-of")
      (ListLink
         (VariableNode "$B")
         (VariableNode "$T")
      )
   )
)
```

Inference Control Learning: URE + Pattern Miner

Call the URE with a Control Rule Template as target + Pattern Miner

```
(cog-bc icr-rb (QuoteLink
                  (ImplicationScopeLink
                     (UnquoteLink
                        (VariableNode "$impl-vardecl")
                     )
                     (AndLink
                        (EvaluationLink
                           (PredicateNode "URE:BC:preproof-of")
                           (UnquoteLink
                              (VariableNode "$preproof-A-args")
                           )
                        )
                        (ExecutionLink
                           (SchemaNode "URE:BC:expand-and-BIT")
                           (UnquoteLink
                              (VariableNode "$expand-inputs")
                           )
                           (UnquoteLink
                              (VariableNode "$expand-output")
                           )
                        )
                     )
                     (EvaluationLink
                        (PredicateNode "URE:BC:preproof-of")
                        (UnquoteLink
                           (VariableNode "$preproof-B-args")
                        )
                     )
                  )
               )
            )
```

```
(ImplicationScopeLink (stv 1 0.00625)
   (VariableList
      (VariableNode "$T")
      (TypedVariableLink
         (VariableNode "$A")
         (TypeNode "DontExecLink")
      )
      (VariableNode "$X")
      (TypedVariableLink
         (VariableNode "$B")
         (TypeNode "DontExecLink")
      )
   )
   (AndLink
      (EvaluationLink
         (PredicateNode "URE:BC:preproof-of")
         (ListLink
            (VariableNode "$A")
            (VariableNode "$T")
         )
      )
      (ExecutionLink
         (SchemaNode "URE:BC:expand-and-BIT")
         (ListLink
            (VariableNode "$A")
            (InheritanceLink
               (ConceptNode "a")
               (VariableNode "$X")
            )
            (DontExecLink
               (DefinedSchemaNode "conditional-full-instantiation-implication-scope-meta-rule"
            )
         )
         (VariableNode "$B")
      )
   )
   (EvaluationLink
      (PredicateNode "URE:BC:preproof-of")
      (ListLink
         (VariableNode "$B")
         (VariableNode "$T")
      )
   )
)
```

Inference Control Learning: more to say

Feel free to ask

- How to weight compatible control rules?
- How to decide which rule to trigger?