SimStudent: Building a Cognitive Tutor by Teaching a Simulated Student

Noboru Matsuda
Human-Computer Interaction Institute
Carnegie Mellon University

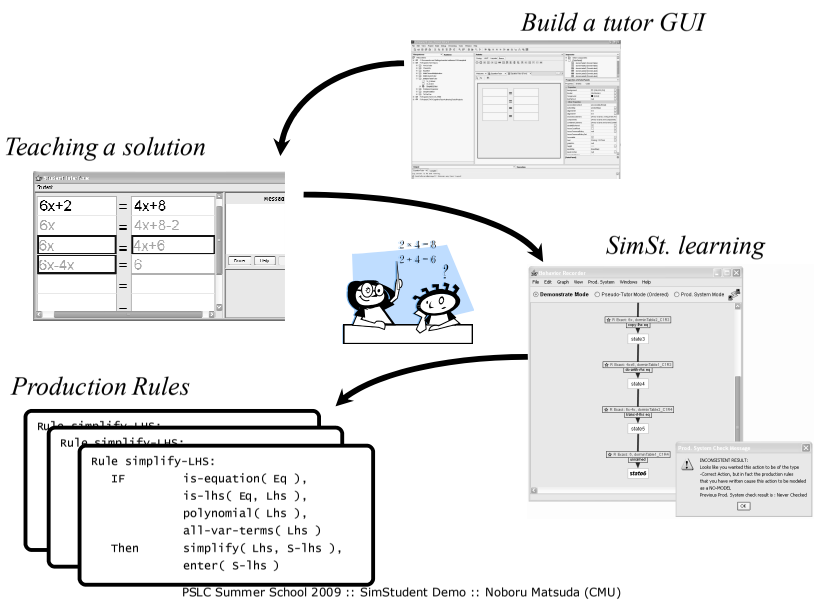
LearnLab
Pittsburgh Science of Learning Center
SimStudent Project

CTAT: Cognitive Tutor Authoring Tools

- *Example-Tracing Tutor* with zero programming
  - A cognitive model specific to a particular problem
  - Some generalization by modifying a behavior graph
- *Model-Tracing Tutor* requires a cognitive model
  - Cognitive task analysis is challenging
  - Writing production rules is even more challenging
  - *Performing the task* is much easier…

LearnLab
Pittsburgh Science of Learning Center
SimStudent Project

Next Generation Authoring



LearnLab
Pittsburgh Science of Learning Center
SimStudent Project

SimStudent

- Machine learning agent
  - Learns problem-solving steps by …
  - Observes model solutions / solving problems, and …
  - Outputs a set of production rules
- Fundamental technology
  - Programming by Demonstration
  - Inductive Logic Programming

Lau, T. A., & Weld, D. S. (1998). Programming by demonstration
Blessing, S. B. (1997). A programming by demonstration authoring tool for model-tracing tutors

Authoring Strategies

- Authoring by demonstration
  - Learning from worked-out examples
  - Demonstrate whole solutions
  - Learning by generalizing examples (when it can't "self-explain")
- Authoring by tutoring
  - Learning by doing (with tutor feedback)
  - Interactively tutor with immediate feedback and hint
  - Learning by generalizing hint with taking feedback into account

Demo!

Learning Production Rules in 3 parts: What-When-How

If

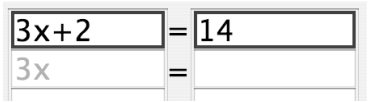
such and such *constraints* hold
When

among this and that *GUI elements*
What

Then

do *actions* with the GUI elements
How

Production Rule in JESS




```
(defrule trans-lr-lhs
  ?problem <- (problem (interface-elements ?table1 ? ? ?))
  ?table1 <- (table (columns ?column1))
  ?column1 <- (column (cells $?m1 ?cell0 $?))
  ?cell0 <- (cell (value ?val0&~nil))

  ?problem <- (problem (interface-elements ? ?table2 ? ?))
  ?table2 <- (table (columns ?column2))
  ?column2 <- (column (cells $?m2 ?cell1 $?))
  ?cell1 <- (cell (value ?val1&~nil))

  ?column1 <- (column (cells $?m3 ?cell2 $?))
  ?cell2 <- (cell (name ?selection) (value ?input))

  (test (consecutive-row ?cell0 ?cell2))
  (test (same-column ?cell0 ?cell2))
  (test (distinctive ?cell0 ?cell2))
  (test (consecutive-row ?cell1 ?cell2))
  (test (same-column ?cell1 ?cell2))
  (test (distinctive ?cell1 ?cell2))
  (test (polynomial ?val0))
  (test (not (has-var-term ?val1)))
=>
  (bind ?input (first-var-term ?val0))
  (modify ?cell2 (value ?input)) )
```

GUI elements
Working Memory Element (WME)
WME path
LHS
Topological constraints
WME conditions
Feature constraints
Constraints
Actions
RHS

LearnLab Pittsburgh Science of Learning Center

SimStudent Project

Background Knowledge

- Domain concepts to "explain" demonstrations
  - Operators
  - Feature predicates
- External Jess function written in Java

```
(defrule multi-lhs
        ...
        ?var22140 <- (column  (cells ? ? ?var22143 ? ? ? ? ?))
        ?var22143 <- (cell  (value ?val0&~nil))

        (test (fraction-term ?val0 ))
   =>
        (bind ?val2 (denominator ?val0))
        (bind ?input (mul-term-by ?val0 ?val2))
        ...
)
```


LearnLab Pittsburgh Science of Learning Center

SimStudent Project

Example : Algebra domain

- 16 Feature predicates & 28 operators

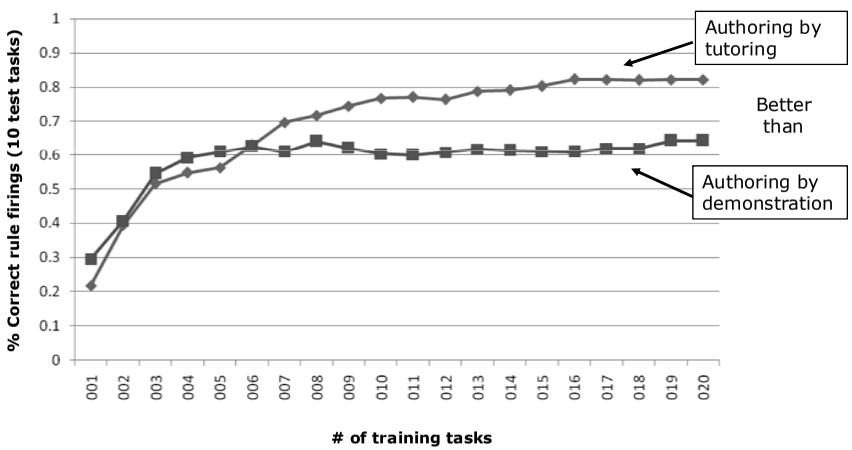
Feature Predicates for LHS conditions	Operators for RHS actions	
HasCoefficient	AddTerm	LastVarTerm
HasConstTerm	AddTermBy	MulTerm
HasVarTerm	Coefficient	MulTermBy
Homogeneous	CopyTerm	Numerator
IsFractionTerm	Denominator	ReverseSign
IsConstant	DivTerm	RipCoefficient
IsDenominatorOf	DivTermBy	SkillAdd
IsNumeratorOf	EvalArithmetic	SkillClt
IsPolynomial	FirstTerm	SkillDivide
Monomial	FirstVarTerm	SkillMultiply
NotNull	GetOperand	SkillRf
VarTerm	InverseTerm	SkillMt
IsSkillAdd	LastConstTerm	SkillSubtract
IsSkillSubtract	LastTerm	VarName
IsSkillDivide		
IsSkillMultiply		

LearnLab Pittsburgh Science of Learning Center

SimStudent Project

Learning Results

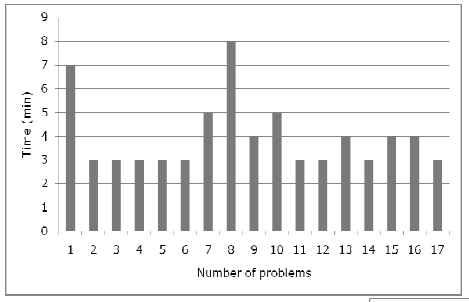
$$StepScore = \begin{cases} 0, \text{ if } C = 0 \\ \dfrac{C}{C+I} \end{cases}$$



LearnLab Pittsburgh Science of Learning Center

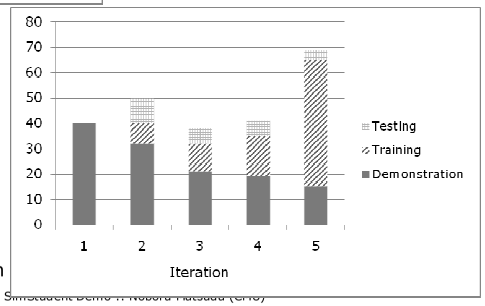
SimStudent Project

Authoring Time



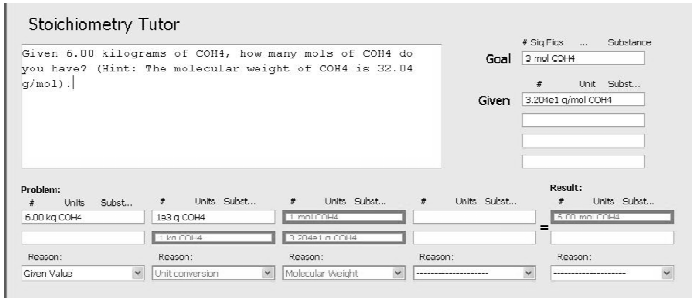
Authoring by Tutoring

- Authoring by tutoring took 86 minutes
- Authoring by demonstration took 238 minutes
- A 2.8x speed-up!

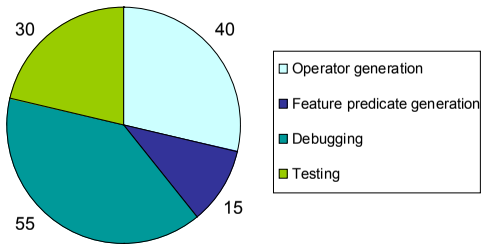


Authoring by Demonstration

Example: Stoichiometry Tutor



Approximate Time Distribution (hours)



Learn more about SimStudents

- Project Web
  - www.cs.cmu.edu/~mazda/SimStudent
- Download & Tutorial
  - http://ctat.pact.cs.cmu.edu (linked from project web)
- Contact Noboru
  - mazda@cs.cmu.edu

SimStudent Projects

- Intelligent Authoring
  - Building a Cognitive Tutor as a CTAT Plug-in
- Student Modeling and Simulation
  - Controlled educational studies
  - Error formation study
  - Prerequisite conceptual knowledge study
- Teachable Peer Learner
  - Learning by teaching

Learning by Teaching SimStudent