Diagnosis: Model Tracing und Domain Reasoning
Supporting Tutorial Feedback to Student Help Request and Errors in Symbolic Differentiation.

Claus Zinn

DFKI — German Research Centre for Artificial Intelligence

December 20, 2006

Lecture “Educational Technologies”, Winter Term 2006, University of Saarland.
Outline

1. Introduction
3. Formalisation of Reasoning in Symbolic Differentiation
   - Formalisation of Expert Reasoning
   - Formalisation of Buggy Reasoning
4. Operationalisation
5. Related Work
6. Conclusion
provision of intelligent, user-adaptive, and effective feedback requires human tutors to

- exploit their expert domain knowledge
- diagnose students’ actions through potentially huge space of possible solutions and misconceptions

strive to simulate good human tutors, replicating their

- reasoning and diagnosis capabilities
- pedagogical expertise
- conversational expertise

huge undertaking because it requires adequate

- acquisition
- formalisation
- operationalisation

of material that supports reasoning, diagnosis, and effective interaction with learners
Outline

1. Introduction


3. Formalisation of Reasoning in Symbolic Differentiation
   - Formalisation of Expert Reasoning
   - Formalisation of Buggy Reasoning

4. Operationalisation

5. Related Work

6. Conclusion
Dialogue I

T-1 Try the following one: \( \frac{5}{\sqrt{x^6 + 2x}} \)

S-2a \( 5(x^6 + 2x)^{-\frac{1}{2}} \)

S-2b \( = -\frac{5}{2}(x^6 + 2x)^{-\frac{3}{2}}(6x^5 + 2) \)

S-2c \( = -5(x^6 + 2x)^{-\frac{3}{2}}(3x^5 + 1) \)

S-2d i think thats right ut im not too sure

T-3a That’s very good. You really have got to grips with the chain rule for algebraic expressions.
Dialogue II

T-3b Let’s move on to other functions. Try to differentiate log$(x^2 + 6x - 1)$.

S-4 I'm not sure what you get when you differentiate log. It is $e^x$?

T-5 No. I’ll tell you: $\frac{d}{dx} \log(x) = \frac{1}{x}$

S-6 I’m still not sure how to do this one

T-7 First you need to identify the functions in the composition $(f(g(x)))$. By the way, you really need to remember what the derivative of log is...

S-8 I still don’t know what to do

T-9a Think about the example that you read in the beginning.

T-9b Try to identify $z$ again and then $y$ as a function of $z$.

S-10 $z = x^2 + 6x - 1$ I’m not sure about what $y$ is

T-11 That’s good so far. Now think where $z$ appears in the expression

S-12 $y = \log z$

T-13 Yes. That’s right. Now can you put it all together?

S-14 $\frac{1}{x^2+6x-1}(2x + 6)$

T-15a Yes. That’s it. We could write that as $\frac{2(x+3)}{x^2+6x-1}$
Dialogue III

T-15b  Now let’s try one with trig functions. Try \( \frac{1}{\sin^3 x} \)

Remember that the derivative of \( \sin \) is \( \cos \)

S-16  \((\sin^3 x)^{-1} - (\sin^3 x)(3\cos^2 x)\)

T-17  Think of \( \sin^3(x) \) as \( (\sin(x))^3 \)

S-18  \(3(\sin(x))^2(\cos x)\)

T-19  That’s much better. Now can you solve the original problem (which is a little different)?

S-20  \((\sin(x))^{-3} = -3(\sin(x))^{-2}(\cos x)\)

T-21  Almost. Remember that the derivative of \( x^{-n} \) is \(-nx^{-n-1}\).

T-22  I think you know the answer: \(-\frac{3\cos(x)}{\sin^4 x}\)

S-23  yes that is what i was thinking [...]
Outline

1. Introduction


3. Formalisation of Reasoning in Symbolic Differentiation
   - Formalisation of Expert Reasoning
   - Formalisation of Buggy Reasoning

4. Operationalisation

5. Related Work

6. Conclusion
### Expert Rules of Symbolic Differentiation

<table>
<thead>
<tr>
<th>Rule</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sine Rule</td>
<td>( \frac{d}{dx} \sin(x) = \cos(x) )</td>
</tr>
<tr>
<td>Logarithmic Rule</td>
<td>( \frac{d}{dx} \log(x) = \frac{1}{x} )</td>
</tr>
<tr>
<td>Power Rule</td>
<td>( \frac{d}{dx} [x^n] = n \cdot x^{n-1} )</td>
</tr>
<tr>
<td>Constant Multiple Rule</td>
<td>( \frac{d}{dx} [c \cdot f(x)] = c \cdot \frac{d}{dx} [f(x)] )</td>
</tr>
<tr>
<td>Sum Rule</td>
<td>( \frac{d}{dx} [f(x) \pm g(x)] = \frac{d}{dx} [f(x)] \pm \frac{d}{dx} [g(x)] )</td>
</tr>
<tr>
<td>Chain Rule For Power Functions</td>
<td>( \frac{d}{dx} ([f(x)]^n) = n \cdot [f(x)]^{n-1} \cdot \frac{d}{dx} [f(x)] )</td>
</tr>
<tr>
<td>General Chain Rule</td>
<td>( \frac{d}{dx} [f(g(x))] = \frac{d}{dg} [f(g(x))] \cdot \frac{d}{dx} [g(x)] )</td>
</tr>
</tbody>
</table>

\( c \) is any real number, \( f(x) \) and \( g(x) \) are any functions.
Encoding of Expert Rules

\[ \frac{d}{dx} \sin(x) = \cos(x) \]
Encoding of Expert Rules

- \[ \frac{d}{dx} \sin(x) = \cos(x) \]
- \[ \text{derive}(\text{Var}, \sin(\text{Var}), \cos(\text{Var})). \]
Encoding of Expert Rules

- $\frac{d}{dx} \sin(x) = \cos(x)$
- derive(Var, sin(Var), cos(Var)).
- the query
  
  ?-derive(x, sin(x), Answer).

  yields Answer = cos(x).
Encoding of Expert Rules

- \( \frac{d}{dx} \sin(x) = \cos(x) \)
- derive(Var, sin(Var), cos(Var)).
- the query
  
  \(?-\) derive(x, sin(x), Answer).

  yields Answer = cos(x).

- \( \frac{d}{dx} [x^n] = n \cdot x^{n-1} \)
Encoding of Expert Rules

- \( \frac{d}{dx} \sin(x) = \cos(x) \)
- derive(Var, sin(Var), cos(Var)).
- the query

      ?- derive(x, sin(x), Answer).

yields Answer = \( \cos(x) \).

- \( \frac{d}{dx} [x^n] = n \cdot x^{n-1} \)
- derive(Var, Var^N, N*(Var^N1)) :- freeof(Var, N), N1 is N-1.
Encodings of Expert Rules

\[ \frac{d}{dx} \sin(x) = \cos(x) \]
\[ \text{derive(Var, \sin(Var), \cos(Var)).} \]
\[ \text{the query} \]
\[ ?-\text{derive(x, \sin(x), Answer).} \]
\[ \text{yields Answer = \cos(x).} \]

\[ \frac{d}{dx} [x^n] = n \cdot x^{n-1} \]
\[ \text{derive(Var,Var^N, N*(Var^N1)) :- freeof(Var, N), N1 is N-1.} \]

\[ \frac{d}{dx} [f(x) + g(x)] = \frac{d}{dx} [f(x)] + \frac{d}{dx} [g(x)] \]
Encoding of Expert Rules

\[
\frac{d}{dx} \sin(x) = \cos(x)
\]
\[
\text{derive(Var, sin(Var), cos(Var))}.
\]
\[
\text{the query}
\]
\[
?-\text{derive}(x, \sin(x), \text{Answer}).
\]
yields \text{Answer} = \cos(x).

\[
\frac{d}{dx} [x^n] = n \cdot x^{n-1}
\]
\[
\text{derive(Var,Var}\,^\ast\,N, N*(\text{Var}^\ast\,N1)) \,:\, \text{freeof(Var, N), N1 is N-1}.
\]
\[
\frac{d}{dx} [f(x) + g(x)] = \frac{d}{dx} [f(x)] + \frac{d}{dx} [g(x)]
\]
\[
\text{derive(Var, A+B, A1+B1) \,:\, \text{derive(Var, A, A1)}, \text{derive(Var, B, B1)}.}
\]
Expert Rules (cont’d)

\[ \frac{d}{dx} \left( [f(x)]^n \right) = n \cdot [f(x)]^{n-1} \cdot \frac{d}{dx} [f(x)] \]
Expert Rules (cont’d)

- \[ \frac{d}{dx} ([f(x)]^n) = n \cdot [f(x)]^{n-1} \cdot \frac{d}{dx} [f(x)] \]

- \[ \text{derive(Var, } X^N, (N \cdot (X^{N1}) \cdot XP) : not (Var = X), \]
\[ N1 \text{ is } N - 1, \]
\[ \text{derive(Var, X, XP).} \]
Expert Rules (cont’d)

\[ \frac{d}{dx} \left( f(x)^n \right) = n \cdot f(x)^{n-1} \cdot \frac{d}{dx} [f(x)] \]

\text{derive}(\text{Var}, \text{X}^N, (N \cdot (\text{X}^{N1}) \cdot \text{XP}) \quad : \quad \text{not (Var = X)},
\text{N1 is N - 1, derive(Var, X, XP}).

the query

?\text{- derive(x, 5*(x^6+2*x)^(-0.5), Answer).}

yields Answer = 5*(-0.5*(x^6-2*x)^(-1.5)*(6*x^5+2*1)).
Expert Rules (cont’d)

\[ \frac{d}{dx} \left( [f(x)]^n \right) = n \cdot [f(x)]^{n-1} \cdot \frac{d}{dx} [f(x)] \]

\[ \text{derive(Var, X}^N, (N*(X}^{N1}))*XP) \quad :\quad \text{not (Var = X),} \]
\[ \text{N1 is N - 1,} \]
\[ \text{derive(Var, X, XP).} \]

the query

\[ ?- \text{derive(x, 5*(x}^6+2*x)^{-0.5}, Answer). \]

yields Answer = 5*(-0.5*(x}^6-2*x)^{-1.5}*(6*x}^5+2*1)).

answer unsatisfying for two reasons.

- not “tidied-up”; student’s answer in S-2a, S-2b simpler and more readable
- PROLOG only returns result, no intermediate steps.
Expert Rules (extended)

\[
\text{derive}(X, \sin(X), \cos(X), \text{[sinRule}(\sin(X) = \cos(X))]).
\]
\[
\text{derive}(X, X^N, (N*(X^{N1})), \text{[powerRule}(\text{deriv}(X^N) = (N*(X^{N1}))]) :-
\]
\[
\text{freeof}(X, N),
\]
\[
N1 \text{ is } N - 1.
\]
\[
\text{derive}(X, A+B, A1+B1, \text{[sumRule}(\text{deriv}(A+B) = \text{deriv}(A)+\text{deriv}(B)), \text{Rules}]) :-
\]
\[
\text{derive}(X, A, A1, \text{R1}),
\]
\[
\text{derive}(X, B, B1, \text{R2}),
\]
\[
\text{append}(\text{R1}, \text{R2}, \text{Rules}).
\]
Expert Rules (extended)

\[
\begin{align*}
\text{derive}(X, \text{sin}(X), \text{cos}(X), \sinRule(\text{sin}(X) = \text{cos}(X))). \\
\text{derive}(X, X^N, \text{(N*(X^N1))}, \text{powerRule}(\text{deriv}(X^N) = (\text{N*(X^N1)}))):- \\
\text{freeof}(X, N), \\
N_1 \text{ is } N - 1. \\
\text{derive}(X, A+B, A_1+B_1, \text{sumRule}(\text{deriv}(A+B) = \text{deriv}(A)+\text{deriv}(B)), \text{Rules})):- \\
\text{derive}(X, A, A_1, \text{R1}), \\
\text{derive}(X, B, B_1, \text{R2}), \\
\text{append}(\text{R1}, \text{R2}, \text{Rules}).
\end{align*}
\]

▶ Now, the query

\[
?\text{-derive}(x, 5*(x^6+2*x)^{-0.5}), \text{Answer, Explain}).
\]

returns same Answer, and Explain contains solution graph that shows each of rule used to compute Answer:
Solution Graph in Prolog

```
[const_mult_rule( = (deriv(x, 5*(x^6+2*x)^number(-,[1],[2])))
           5*deriv(x, (x^6+2*x)^number(-,[1],[2])))])
[chain_rule_pow( = (deriv(x, (x^6+2*x)^number(-,[1],[2]))
           number(-,[1],[2])*(x^6+2*x)^number(-,[3],[2])*deriv(x, x^6+2*x)))]
[sum_rule( = ( deriv(x, x^6+2*x)
           deriv(x, x^6)+deriv(x, 2*x))),
[power_rule( = ( deriv(x, x^6)
           6*x^5))],
[constant_multiple_rule( = ( deriv(x, 2*x)
           2*deriv(x, x))),
[linear_x( = ( deriv(x, x)
           1))])]]
```
\[
\frac{d}{dx} 5(x^6 + 2x)^{-\frac{1}{2}}
\]

- const. mult. rule
- general power rule
- sum rule
- power rule
- linear rule
Solution Graph — Example

\[
\frac{d}{dx} 5(x^6 + 2x)^{-\frac{1}{2}}
\]

const. mult. rule

5 \* \[
\frac{d}{dx} (x^6 + 2x)^{-\frac{1}{2}}
\]

general power rule

\[-\frac{1}{2}(x^6 + 2x)^{-\frac{3}{2}} \* \frac{d}{dx} (x^6 + 2x)\]

sum rule

\[
\frac{d}{dx} x^6 + \frac{d}{dx} 2x
\]

power rule

\[
6x^5 + 2 \* \frac{d}{dx} x
\]

linear rule

\[
1
\]

▶ supports generation of T-5, T-15b, and T21
\[ \frac{d}{dx} 5(x^6 + 2x)^{-\frac{1}{2}} \]

- const. mult. rule
- general power rule
- sum rule
- power rule
- linear rule

- supports generation of T-5, T-15b, and T21
- but not sufficient to simulate tutor's feedback T-7, T-9b, and T-11.
Expert Rules (extended with task model)

**Initial task:** Compute derivative of \( y = f(x) \)

\[
y = \frac{5}{\sqrt{x^6 + 2x}}
\]

Rewrite \( y \) to recognisable form \( \tilde{y} \)

Compute derivative of \( \tilde{y} \)

\[
\tilde{y} = 5(x^6 + 2x)^{-\frac{1}{2}}
\]

Identify the form of term \( \tilde{y} \)

Compute derivative of \( \tilde{y} \) with recognised form \( f(g(x)) \)

Identify inner layer of \( \tilde{y} \)

Compute derivative of inner layer \( z \)

\[
z = x^6 + 2x
\]

Identify outer layer of \( \tilde{y} \)

Compute derivative of outer layer \( \tilde{y} \)

\[
\tilde{y} = 5z^{-\frac{1}{2}}
\]

Differentiate inner layer

Do combine result

\[
\frac{dy}{dx} = \frac{dz}{dx} \cdot \frac{d\tilde{y}}{dz}
\]

with \( \frac{dz}{dx} = 6x^5 + 2 \) and \( \frac{d\tilde{y}}{dz} = -5 \cdot -\frac{1}{2} \cdot z^{-\frac{3}{2}} \)

Combine result

Do Tidy-Up Result

\[
(6x^5 + 2)(-5 \cdot -\frac{1}{2} \cdot z^{-\frac{3}{2}})
= (6x^5 + 2) \cdot (-5 \cdot -\frac{1}{2} \cdot (x^6 + 2x)^{-\frac{3}{2}})
\]

Tidy-Up result

End of task with tidied-up Result

\[
y = \frac{-5(3x^5 + 1)}{(x^6 + 2x)^{\frac{3}{2}}}
\]
Solution Graph — Example using task model

\[
\begin{align*}
\text{rewrite_to_recog_form, expert, term_rewriting, dummy, } & 5/\sqrt{x^6+2*2x}, 5*(x^6+2*2x)\text{^number}(-,[1],[2]), [] \\
\text{constant_multiple_rule, expert, deriv(x,c*f(x))=c*deriv(x,f(x))}, \\
& \text{deriv(x,5*(x^6+2*2x)\text{^number}(-,[1],[2]))=5*deriv(x,(x^6+2*2x)\text{^number}(-,[1],[2])),} \\
& 5*(x^6+2*2x)\text{^number}(-,[1],[2]), \\
& 5\text{*(number}(-,[1],[2])*(x^6+2*2x)\text{^number}(-,[3],[2])*6\text{x^5+2*1)), []} \\
\text{chain_rule_gen, expert, deriv(x,f(g(x)))=deriv(g,f(g(x)))*deriv(x,g(x))}, \\
& \text{deriv(x,(x^6+2*2x)\text{^number}(-,[1],[2]))=deriv(z,z\text{^number}(-,[1],[2]))*deriv(x,x^6+2*2x),} \\
& (x^6+2*2x)\text{^number}(-,[1],[2]), \\
& \text{number}(-,[1],[2])*(x^6+2*2x)\text{^number}(-,[3],[2])*6\text{x^5+2*1), []} \\
\text{identify_form_of_statement, ignore, identify_term_structure, dummy,} \\
& (x^6+2*2x)\text{^number}(-,[1],[2]), \\
& \text{layers([z=x^6+2*2x,y=z\text{^number}(-,[1],[2])]), []} \\
\text{do_identify_inner_layer, expert, identify_inner_layer, dummy,} \\
& (x^6+2*2x)\text{^number}(-,[1],[2]), \\
& \text{inner_layer(z=x^6+2*2x), []} \\
\text{do_identify_outer_layer, expert, identify_outer_layer, dummy,} \\
& (x^6+2*2x)\text{^number}(-,[1],[2]), \\
& \text{outer_layer(y=z\text{^number}(-,[1],[2])), []} \\
\text{do_substitutions, ignore, determine_substitutions, dummy,} \\
& (x^6+2*2x)\text{^number}(-,[1],[2]), \\
& \text{substitutions([z=x^6+2*2x,y=z\text{^number}(-,[1],[2])]), []} \\
\text{do_substitution, expert, subst_inner_layer, dummy, (x^6+2*2x)\text{^number}(-,[1],[2]), z=x^6+2*2x, []} \\
\text{do_substitution, expert, subst_outer_layer, dummy, (x^6+2*2x)\text{^number}(-,[1],[2]), y=z\text{^number}(-,[1],[2]), []} \\
\text{do_decompose, expert, decompose_into_subproblems, dummy,} \\
& (x^6+2*2x)\text{^number}(-,[1],[2]), \\
& \text{subproblems([deriv(z,z\text{^number}(-,[1],[2])),deriv(x,x^6+2*2x)]}) \\
\text{build_deriv, ignore, build_derivative_inner_layer, dummy,} \\
& z\text{^number}(-,[1],[2]), \\
& \text{number}(-,[1],[2])*z\text{^number}(-,[3],[2]), [] \\
\text{power_rule, expert, deriv(x,z^n)=n*x^(n-1), number}(-,[1],[2]), \text{^number}(-,[3],[2]), [] \\
\text{build_deriv, ignore, build_derivative_outer_layer, dummy,} \\
& \text{z}\text{^number}(-,[1],[2]), \\
& \text{number}(-,[1],[2])*\text{z}\text{^number}(-,[3],[2]), [] \\
\text{sum_rule, expert, deriv(x,f(x)+g(x))=deriv(x,f(x))+deriv(x,g(x))}, \\
& \text{deriv(x,x^6+2*2x)=deriv(x,x^6)+deriv(x,2*1), x^6+2*2x, 6*x^5+2*1, []} \\
\text{constant_multiple_rule, expert, deriv(x,c*f(x))=c*deriv(x,f(x))}, \\
& \text{deriv(x,2*1)=2*deriv(x,x), 2*1, 2*1, []} \\
\text{linear_x, expert, deriv(x,x)=1, x, 1, []} \\
\text{do_combine, expert, combine_subsolutions, dummy,} \\
& \text{sub_solutions([number}(-,[5],[2])*(2*x+x^6)^\text{number}(-,[3],[2])*(2+6*x^5), [])} \\
\text{tidy_up, expert, term_simplification, dummy,} \\
& \text{number}(-,[5],[2])*(2*x+x^6)^\text{number}(-,[3],[2])*(2+6*x^5), []) \\
\end{align*}
\]
Erroneous Rules in Symbolic Differentiation

missing inner layer (chain rule) \[ \frac{d}{dx} (x^3 - 3x)^5 \not\rightarrow 5(x^3 - 3x)^4 \]
wrong exponent (chain/power rule) \[ \frac{d}{dx} (5x^3 - 6)^{-3} \not\rightarrow -3(5x^3 - 6)^{-2}(15x^2) \]
missing exponent (chain/power rule) \[ \frac{d}{dx} (x^3 - 3x)^5 \not\rightarrow 5(x^3 - 3x)(3x^2 - 3) \]
incorrect basic rule (sine rule) \[ \frac{d}{dx} \sin(x) \not\rightarrow -\cos(x) \]
missing bracketing (op. precedence) \[ \frac{d}{dx} (x^3 - 3x)^5 \not\rightarrow 5(x^3 - 3x)^4 3x^2 - 3 \]
erroneous transformation (rewriting) \[ \frac{1}{(5x^3 - 6)^3} \not\equiv (5x^3 - 6)^{-\frac{1}{3}} \]
Formalisation of Buggy Rules

Expert power rule, and a buggy version

\[
\begin{align*}
derive(\text{expert}, & \\
X, & \\
X^N, & \\
N\cdot(X^{N_1}), & \\
[powRule(deriv(X^N)=N\cdot(X^{N_1}))] & : - \\
\text{freeof}(X, N), N1 \text{ is } N - 1.
\end{align*}
\]

\[
\begin{align*}
derive(\text{buggy}, & \\
X, & \\
X^N, & \\
N\cdot(X^{N_1}), & \\
[powRule(deriv(X^N)=N\cdot(X^{N_1})), \text{wrongPow}(N_1)] & : - \\
N < 0, & \\
\text{freeof}(X, N), & \\
N1 \text{ is } N + 1.
\end{align*}
\]
More buggy versions

\[
\text{derive} (\text{buggy}, \\
X, \\
X^N, \\
N\times X, \\
[\text{powRule} (\text{deriv} (X^N) = N\times X), \text{missingExpTerm} (N-1)]) :\neg \\
\text{freeof} (X, N).
\]

\[
\text{derive} (\text{buggy}, \\
X, \\
X^N, \\
X^N1, \\
[\text{powRule} (\text{deriv} (X^N) = X^N1), \text{missingFactor} (N)]) :\neg \\
\text{freeof} (X, N), N1 \text{ is } N - 1.
\]
Example Queries

?- derive(RuleType, x, x^(-4), (-4*(x^(-5))), Explain)
=> RuleType = expert
   Explain = [powerRule(deriv(x^(-4))=(-4*(x^(-5))))]

?- derive(RuleType, x, x^(-4), (-4*(x^(-3))), Explain)
=> RuleType = buggy
   Explain = [powerRule(deriv(x^(-4))=(-4(x^(-3))), wrongPow(-3)]

?- findall( (RuleType,StudentAnswer),
            derive(RuleType, x, x^(-4), StudentAnswer, Explain),
            AllDiagnoses )

=> AllDiagnoses = <all recognisable answers>
Outline

1. Introduction
3. Formalisation of Reasoning in Symbolic Differentiation
   - Formalisation of Expert Reasoning
   - Formalisation of Buggy Reasoning
4. Operationalisation
5. Related Work
6. Conclusion
Operationalisation

4 main services:

1. Define Differentiation Task
   - solves given task
   - computes solution
   - constructs solution graph

2. Get Solution

3. Get Next Step

4. Get Diagnosis

Model Tracing to

- identify learner’s problem solving progress
- generate appropriate feedback on student answer or help request
Model Tracing

- works with annotations on solution graph
- if learners’ problem solving step matches expert node, then mark this step as visited, and provide appropriate feedback
- if learner’s problem solving step does not match expert node, then generate alternative buggy solution graphs to search for matching buggy node
  - supporting diagnosis of multiple bugs
- if learner requests help, then search graph for next unvisited node, and generate feedback exploiting node’s content.
- generation of next step help flexible
  - nodes have specificity counter
- notion of matching non-trivial
  - many problem solving steps involve algebraic manipulations
  - use of PRESS algebra package to compute semantic equality
- notion of solution graph traversal non-trivial
Tutorial Dialogue Manager — GUI

SLOPERT - An Intelligent Learning Object for Symbolic Differentiation

Student Input

Formula input

Previous Term

Current Differentiation Task (d/dx):

\[ \sin(x^4) \]

Dialogue History:

Student: (asking for task definition) \[ \sin(x^4) \]
Tutor: Okay, let us differentiate this.
Student: (asking for hint)
Tutor: You have to apply the following rule: the general chain rule
Student: \[ \cos(x^4) \]
Tutor: There is something wrong with the application of the general chain rule. You missed the inner layer.
Tutorial Dialogue Manager — Architecture

Server-side
Symbolic Differentiation Software

XML-RPC Request Handler

[..]

JASPER (Java/Prolog Interface)

Sicstus Prolog

SLOPERT

Client-side
Graphical User Interface

GLEAM (GUI)

setTask
getSolution
getDifficulty
getNextStep
getDiagnosis

C. Zinn
Domain Reasoning and Diagnosis: SLOPERT
Outline

1. Introduction
3. Formalisation of Reasoning in Symbolic Differentiation
   - Formalisation of Expert Reasoning
   - Formalisation of Buggy Reasoning
4. Operationalisation
5. Related Work
6. Conclusion
Related Work

- Computer Algebra Systems of limited help
  - rarely mirror human problem solving skills
  - except for educational CASs
  - but no proper diagnosis of incorrect student input
- Buggy’s *procedural networks*
  - represent each procedural skill and subskill independently
  - buggy variants of individual sub-skills
- The Cognitive Tutors’ ACT-R models of student problem solving
  - production rule mechanism to capture problem solving strategies
  - estimates of how good learners knows each correct and buggy rule
- The Andes Physics Tutor
  - independent problem solver (with all its benefits)
  - Next Step Help
  - What’s Wrong Help
Outline

1. Introduction
3. Formalisation of Reasoning in Symbolic Differentiation
   - Formalisation of Expert Reasoning
   - Formalisation of Buggy Reasoning
4. Operationalisation
5. Related Work
6. Conclusion
Introduction

Analysis of Human-human Tutorial Dialogue

Formalisation of Reasoning in Symbolic Differentiation

Operationalisation

Related Work

Conclusion

C. Zinn

Domain Reasoning and Diagnosis: SLOPERT

Conclusion

▶ SLOPERT, a glass-box reasoning and diagnosis engine for symbolic differentiation

▶ informed by analysis of human-human tutorial dialogues
  ▶ has expert task model
  ▶ enriched with buggy rules

→ can provide natural step-by-step solutions for any given differentiation problem

→ can diagnose typical student errors

→ supports tutorial dialogue manager generating natural problem-solving hints and scaffolding help
  ▶ AI to explore (potentially huge and buggy) task model space
  ▶ Learning Sciences to inform design of pedagogical strategies
  ▶ Dialogue engineering to encode conversational expertise
Thank you!

And check out

- [http://www.leactivemath.org](http://www.leactivemath.org)
- [http://www.activemath.org](http://www.activemath.org)