Numerical reasoning in ILP

Ashwin Srinivasan
Rui Camacho

Oxford University Computing Laboratory
## Is it important?

A=Natural language.  B=Biological.
E=Software engineering.  F=Discrete event.
G=Database design.  H=Control.

<table>
<thead>
<tr>
<th>Problem</th>
<th>N</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relev</td>
<td>1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>Multi</td>
<td>2</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>7</td>
</tr>
<tr>
<td>Revise</td>
<td>3</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>Invent</td>
<td>4</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>Number</td>
<td>5</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>6</td>
</tr>
<tr>
<td>Probab</td>
<td>6</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>Deep</td>
<td>7</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Recurs</td>
<td>8</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Large</td>
<td>9</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>Small</td>
<td>10</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>Rely</td>
<td>11</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>8</td>
</tr>
<tr>
<td>Model</td>
<td>12</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>Constr</td>
<td>13</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>6</td>
</tr>
<tr>
<td>Built</td>
<td>14</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>Brittle</td>
<td>15</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>5</td>
</tr>
</tbody>
</table>

6 out of the 8 possible application domains need it
What is it?

**Simple arithmetic functions** (plus/3, minus/3, ...)

**More complicated functions**

\[ p(X,Y,T) \leftarrow \text{mult}(T,2,T1), \sin(T1,X), \sin(T,Y) \]

**Inequalities**

\[ p(X) \leftarrow X \geq 2.9, \ X \leq 5.2 \]

**Regression models**

**Geometric models**
What has been done so far

Page and Frisch, 1991
Constrained atoms: \( p(x,y) \leftarrow x \leq y \)

Mizoguchi and Ohwada, 1992
CLP version LGG: \( p(x) \leftarrow x \geq 2.9, x \leq 5.2 \)

Quinlan, Camacho 1994
Built-in \( \leq: \) \( p(x) \leftarrow x \geq 2.9, x \leq 5.2 \)

Dzeroski, De Raedt, Karalic, 1994–1995
Built-in transformation: \( p(x,y) \leftarrow y \text{ is } 2.3x + 5 \)

Sebag and Rouveirol, 1995
Inductive CLP ?

So, what is the problem?
Too restricted (PF, MO, QC)
Restricted use of regression (D,DR,K)
Determinacy (D)
Different setting to standard ILP? (SR)
What is different here?

A characteristic feature of ILP is the use of background knowledge
User defined clauses
Special built-ins +,-, =<, log, sin

Can numerical constraints be learnt within the standard ILP setting?

Using as background knowledge, user-defined or built-in predicates for
Inequalities
Linear, non-linear models
Normal form representations of shapes

Example

\[ p(X_1,Y_1,Z_1) \leftarrow \]
\[ \text{connected}(X_1/Y_1,X_2/Y_2), \text{connected}(X_2/Y_2,X_3/Y_3), \]
\[ \text{circle}(X_1/Y_1,X_2/Y_2,X_3/Y_3,A,B,\text{Radius}), \]
\[ \text{linear}(Z_1,\text{Radius},\text{constant},\text{slope},\text{error},\text{Residual}), \]
\[ \text{absolute_value}(\text{Residual},R), \]
\[ R \leq \text{deviation} \]
User defined background

Examples

lteq(X,Y):-
    number(X), number(Y),
    X <= Y.
lteq(X,Y):-
    number(X), var(Y),
    X1 is 2*X,
    guess(X,X1,Y).

linear(Y,X,M,C,Sigma,Residual):-
    list(X), list(Y), var(M), var(C), var(Sigma),
    least_square_estimate(Y,X,M,C,Sigma,Residual).

linear(Y,X,M,C,0,0):-
    number(X), number(Y), var(M), var(C),
    M0 is Y/X, M1 is 2*M0,
    guess(M0,M1,M), C is Y - M*X.

circle(X0/Y0,X1/Y1,X2/Y2,A,B,R):-
    not(X0/Y0=X1/Y1),not(X0/Y0=X2/Y2),not(X1/Y1=X2/Y2),
    Alpha is 2*(X1-X0), Beta is 2*(Y1-Y0),
    Gamma is 2*(X2-X0), Delta is 2*(Y2-Y0),
    C1 is X1^2 - X0^2 + Y1^2 - Y0^2,
    C2 is X2^2 - X0^2 + Y2^2 - Y0^2,
    B is (C2*Alpha-C1*Gamma)/(Alpha*Delta-Gamma*Beta),
    A is (C1-Beta*B)/Alpha, ...
User defined background

Needs
Ability to use non-ground background knowledge

Advantages
No special purpose program
Easy to reason with numerical predicates
Coordinate transformations etc are easy
Can use standard statistical methods
Can guess when stuck

Disadvantage
Needs user to do some work
Pilot Study

Calculating bank angle from trajectory of a pilot performing a levelled left turn on a F-16 simulator

Given X,Y positions, what bank angle is the pilot turning at?
Bank angle vs $X, Y$

No obvious linear relationship

But, some relationship to radius of turn
Bank angle vs Radius

Almost linear relationship
Materials summary

Trajectories from 32 missions available
Bank angles from -5° to -20°
Safety envelope: ± 3°
9600 examples:
  bank_angle(mission(1),time(10),-5.5)
  :- bank_angle(mission(1),time(10),-8.0)
  :- bank_angle(mission(1),time(10),-2.0)

Background knowledge predicates
  pos/3: position on a mission at some time
  before/2: a time-instant earlier than current
  linear/6: least-square linear regression
  poly/7: polynomial least-squares regression
  circle/6: fits circles and finds centres, radii
  lteq/2, gteq/2, abs values, Prolog built-ins

Algorithm
  Progol
Methods summary

Randomly select 16 trajectories for training
   Remaining 16 reserved for testing

Learn theories using
   1. linear regression, fixed (95%) conf. intervals
   2. linear regression, reason on absolute residual
   3. linear regression, find intervals on residual

Test theories learnt on trajectories reserved for testing
Results summary

Theory using

1. linear regression, fixed (95%) conf. intervals
   accuracy = 82.6% ±0.5%
2. linear regression, reason on absolute residual
   accuracy = 87.4% ±0.5%
3. linear regression, find intervals on residual
   accuracy = 83.8% ±0.5%

Example of theory using

2. linear regression, reason on absolute residual

\[
\text{bank\_angle(Mission1,Time1,Bank) :-}
\begin{align*}
& \text{pos(Mission1,Time1,P1),} \\
& \text{before(Time1,Time2),} \\
& \text{pos(Mission1,Time2,P2),} \\
& \text{before(Time2,Time3),} \\
& \text{pos(Mission1,Time3,P3),} \\
& \text{circle(P1,P2,P3,A,B,Radius),} \\
& \text{lin\_regress(Bank,Radius,0.043,-19.442,2.671,Residual),} \\
& \text{abs\_val(Residual,ARes),} \\
& \text{lteq(ARes,1.023).}
\end{align*}
\]
Results summary (contd.)

Theory 1. (linear regression, fixed residual)

Theory 2. (linear regression, non-fixed residual)
Conclusions

Initial results of using numerical predicates simply as background predicates are promising.

If results continue to be good on other tests, no special framework will be needed for numbers. Structural and numeric reasoning possible.

The use of ground negative examples here is somewhat artificial. Actually need to use integrity constraints. This has been tried, and works well.