Automated Discovery in Pure Mathematics

Simon Colton, Alan Bundy and Toby Walsh,
Institute of Representation and Reasoning
Division of Informatics, University of Edinburgh
80 South Bridge, Edinburgh. EH1 1HN. Scotland.
simonco,bundy,tw@dai.ed.ac.uk

The HR project aims to automate two important discovery processes which occur in mathematics before theorem proving happens, namely the making of the conjecture to be proved and the invention of the definitions in the conjecture statement. Our approach is to:

(1) Represent pure mathematics concepts as data-tables, eg. represent division of integers as a data-table with 3 columns, $[n, a, b]$, where $a, b, n \in \mathbf{N}$ and $a \times b = n$. Definitions for concepts can be generated when needed, using information about how they were constructed.

(2) Use production rules (PRs) to turn old concepts into new ones. HR has 10 production rules and uses parameters to detail exactly what to do. In figure 1, the **match** PR extracts rows where certain columns are the same (the parameters [1,2,2] say that col.1=col.1, col.2=col.2 and col.3=col.2). The **count** PR counts in how many rows the entries in column [1] appear. The **exists** PR removes columns, in this case, only keeping column [1]. Finally, the **split** PR, with parameters [c,=,v], extracts rows where column c has value v. The PRs are general enough to work in number theory, and to also find concepts such as complete graphs in graph theory and Abelian groups in group theory. Note that HR has been developed using group theory as the test domain.

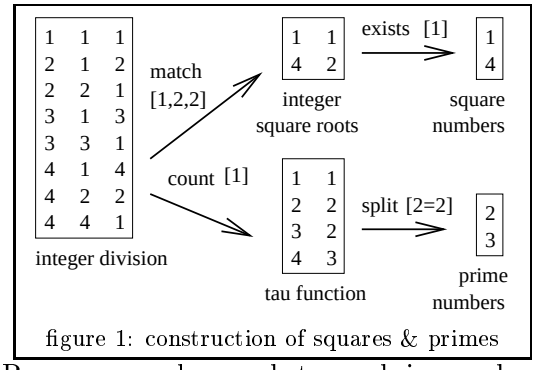

figure 1: construction of squares & primes

(3) Derive measures which estimate how interesting a concept is, and use the heuristic of basing new concepts only on the most interesting old ones. A concept's data-table can be used to describe the models HR is working with, eg. the output from the tau function in figure 1 describes integer 1 as 1, 2 as 2, 3 as 2 and integer 4 as 3. The *parsimony* measure of a concept is inversely proportional to the size of its data table, ie. more parsimonious descriptions are advantageous. These descriptions can be used to categorise HR's models by putting two models in the same category if described the same, eg. the tau function gives integers 1 to 4 this categorisation: [1],[2,3],[4]. The user can supply a 'gold standard' categorisation which they would like the concepts to achieve, eg. into squares and non-squares: [1,4],[2,3]. HR can then use the *invariance/discrimination* measures, which find the proportion of pairs of models categorised together/apart in the gold standard, which are categorised correctly by the concept. Given the isomorphic classification of groups up to order 6, HR found this function which categorises them correctly: $f(G) = |\{(a, b, c) \in G^3 : a * b = c \wedge b * c = a\}|$. A fourth measure is the *complexity* of a concept which counts the number of old concepts

appearing in its construction path, giving a rough idea of how long the definition will be, with more concise definitions being advantageous.

(4) Use empirical evidence to spot conjectures about concepts. The most common way HR does this is to notice that the data-table of a newly formed concept is exactly the same as an old concept. HR then makes the conjecture that the definition of the new concept is equivalent to the definition of the old concept, eg. $\forall$ groups, G, $predicate_1(G) \iff predicate_2(G)$.

(5) Enable HR to use theorem prover OTTER, [McC90], to prove some conjectures it makes. OTTER performs poorly with inductively defined concepts, so only a subset of the conjectures can attempted. However, some of these are non-trivial, eg. in group theory, HR spotted that for any group, G, and $\forall a, b, c \in G$, $(a*b = c \wedge a*c = b \wedge b = id \iff a*b = c \wedge c*a = b \wedge a*a = b)$. Instead of passing this verbatim to OTTER, HR first tries to extract and prove any prime implicates (conjectures where no subset of the premises imply the goal). For example HR extracts this: $a * c = b \wedge b = id \rightarrow c * a = b$, which OTTER proves. HR can use the prime implicates to prove the overall conjecture, and will use them to prove other conjectures later. If a conjecture cannot be proved, HR passes it to model generator MACE, [McC94], which is asked to find a counterexample of size 1, then 2, etc. If a counterexample is found, it is added to the theory, and all future conjectures will be based on the extra data.

(6) Complete a cycle of mathematical activity by assessing the conjectures and using the average interestingness of the conjectures a concept appears in to estimate the interestingness of the concept itself. A conjecture could be interesting if OTTER found it difficult to prove - HR uses OTTER's proof length statistic to determine this. Also, a conjecture could be interesting if it is surprising, ie. the (supposedly) equivalent concepts look quite different - HR calculates the proportion of concepts appearing in one, but not both, construction paths.

HR can bootstrap theory formation, ie. starting with just the axioms of a finite algebra like group theory, it uses MACE to find initial models of the algebra, invents definitions, spots conjectures and uses OTTER to prove them and MACE to disprove them, which introduces counterexamples. HR's biggest success so far was in number theory, [Col99], where it invented the concept of refactorable numbers: $1, 2, 8, 9, 12, 18, 24, 36, 40, 56, 60, 72, 80, 84, 88, 96, \ldots$ These have the appealing definition of being those integers where the number of divisors is itself a divisor. This concept was novel because it was missing from, and subsequently added to, the online-encyclopedia of integer sequences, [Slo]. HR also spotted some conjectures about refactorables, eg. they are congruent to 0, 1, 2 or 4 mod 8, which we have subsequently proved, along with some results of our own, eg. all odd refactorables are squares. For further details about HR, please see [CB] or the web page: `http://dream.dai.ed.ac.uk/group/simonco/hr`.
This work is supported by EPSRC grant GR/L 11724 and EPSRC studentship GR/K/65706.

References

[CB] S Colton and A Bundy. HR: Automatic concept formation in pure mathematics. In *IJCAI '99.*

[Col99] S Colton. Refactorable numbers - a machine invention. *Journal of Integer Sequences*, 2, 1999.

[McC90] W McCune. The OTTER user's guide. Technical Report ANL/90/9, ANL, 1990.

[McC94] W McCune. A Davis-Putnam program and its application to finite first-order model search. Technical Report ANL/MCS-TM-194, ANL, 1994.

[Slo] N Sloane. *Online Encyclopedia of Integer Sequences.* www.research.att.com/~njas/sequences.