CAS 734 Winter 2008

03 Theory Development Techniques

William M. Farmer

Department of Computing and Software
McMaster University

24 January 2008

Axiomatic Theories as Mathematical Models

- Almost any mathematical model can be represented as an axiomatic theory.
- Models that axiomatic theories are good at representing:
  - Mathematical structures, algebras, and data types.
- Models that axiomatic theories are not as good at representing:
  - Models that have a mutable state.
  - Algorithmic theories.
  - Models in which expressions are constructed and evaluated.
- Axiomatic theories are the basis of the axiomatic method.

What is the Axiomatic Method?

1. A mathematical model is expressed as an axiomatic theory in a logic.
2. New concepts are introduced by making definitions.
3. Assertions about the model are stated as theorems and proved from the theory's axioms using the laws of the logic.

Notes:

- The axiomatic method is a method of communication, not a method of discovery (Lakatos).
- The axiomatic method can be used as a method of organization and a method of certification.

Short History of the Axiomatic Method

- Euclid (325–265 BC) used the axiomatic method to present the mathematics known in his time in the Elements. The axioms were considered truths.
- The development of noneuclidean geometry by Bolyai, Gauss, and Lobachevskii (early 1800s) showed that axioms may be considered as just assumptions.
- Whitehead and Russell formalized a portion of mathematics in the Principia Mathematica (1910–13).
- Bourbaki (mid 1900s) used the axiomatic method to codify mathematics in the 30 volume Eléments de mathématique.
- Jutting (1970s) used De Bruijn's Automath proof assistant to formalize and verify Landau's Grundlagen der Analysis.
- Several libraries of formalized mathematics have been developed since the late 1980s using interactive theorem provers.

Benefits of Axiomatic Theories

- Conceptual clarity: Inessential details are omitted.
- Generality: Theorems hold in all models.
- Dual purpose: A theory can be viewed as:
  1. An abstract mathematical model.
  2. A specification of a collection of mathematical models.

Theory Development as a Special Case of the Mathematics Process

1. Theory creation.
   - Built from scratch.
   - Extension of a theory.
   - Union of several theories.
   - Renaming of a theory.
   - Instance of a parameterized theory.
2. Theory exploration.
   - Notation introduction.
   - Concept introduction.
   - Conjecture proving.
   - Computation.
3. Theory connection.
   - Theory interpretation.

Theory Extension

- To make our presentation concrete, we will assume that we are working in STT.
- Let $L_i = (\mathcal{C}_i, \tau_i)$ be a language (of STT) for $i = 1, 2$. L_2 is an extension of L_1 (and L_1 is a sublanguage of L_2), written $L_1 \leq L_2$, if $\mathcal{C}_1 \subseteq \mathcal{C}_2$ and τ_1 is a subfunction of τ_2.
- Let $T_i = (L_i, \Gamma_i)$ be a theory (of STT) for $i = 1, 2$. T_2 is an extension of T_1 (and T_1 is a subtheory of T_2), written $T_1 \leq T_2$, if $L_1 \leq L_2$ and $\Gamma_1 \subseteq \Gamma_2$.
- Hence an extension of a theory T is obtained by adding new vocabulary and axioms to T.
- A theory development can be viewed as a sequence of theory extensions.
- Danger of theory extension: The new machinery may compromise the old machinery by changing the behavior of the constants or by making the theory unsatisfiable.

Conservative Extension

- Intuitively, an extension of a theory T is "conservative" if it adds new machinery to T without compromising the original machinery.
- T_2 is a conservative extension of T_1, written $T_1 \trianglelefteq T_2$, if $T_1 \leq T_2$ and, for all formulas A of L_1, $T_2 \models A$ implies $T_1 \models A$.
- **Proposition (Transitivity)**. If $T_1 \trianglelefteq T_2$ and $T_2 \trianglelefteq T_3$, then $T_1 \trianglelefteq T_3$.
- **Proposition (Satisfiability)**. If $T_1 \trianglelefteq T_2$ and T_1 is satisfiable, then T_2 is satisfiable.
- Hence a conservative extension is a "safe" extension.

Model Conservative Extension

- Let $M_i = (\mathcal{D}_i, I_i, e_i)$ be a standard model for L_i for $i = 1, 2$. M_2 is an expansion of M_1 if $L_1 \leq L_2$, $\mathcal{D}_1 = \mathcal{D}_2$, I_1 is a subfunction of I_2, and $e_1 = e_2$.
- T_2 is a model conservative extension of T_1, written $T_1 \trianglelefteq_m T_2$, if $T_1 \leq T_2$ and every standard model of T_1 has an expansion to L_2 that is a model of T_2.
- Hence a model conservative extension of T is an extension of T that "preserves" the models of T.
- **Proposition (Transitivity)**. If $T_1 \trianglelefteq_m T_2$ and $T_2 \trianglelefteq_m T_3$, then $T_1 \trianglelefteq_m T_3$.
- **Proposition.** If $T_1 \trianglelefteq_m T_2$, then $T_1 \trianglelefteq T_2$. (The converse is false.)

Kinds of Conservative Extensions

- Model conservative.
  - Addition of a theorem to a theory.
  - Addition of a totally specified set of constants (definition).
  - Addition of a partially specified set of constants (profile).
- Non model conservative.
  - Addition of new elements to the models of the theory.

Definitional Mechanisms

- Notational definitions.
  - Change syntax but not semantics.
- Definitions.
  - Introduce a totally specified concept.
- Profiles.
  - Introduce a partially specified concept.
  - Also called specifications and constraints.

Notational Definitions

- A notational definition introduces alternate syntax that can be used in place of official syntax.
  - Usually the alternate syntax is simpler than the corresponding official syntax.
  - Sometimes the alternate syntax is purely external, while the official syntax is purely internal.
  - Notational definitions often hide information such as types and parenthesization.
- Notational definitions are intended to make it easier for the user to read and write expressions.
  - They should have no effect on the system's logic, theories, and reasoning mechanisms.
  - Notational definitions that hide information may sometimes confuse users.

Examples of Notational Definitions

- Macro-abbreviations
- Alternate (usually shorter) names
- Operator syntax (e.g, prefix, infix, postfix, etc.)
- Operator precedence
- Symbol overloading

Explicit Definitions

- Let $T = (L, \Gamma)$ be a theory where $L = (\mathcal{C}, \tau)$, a be a new constant not in L, E be a closed expression of type α of L, and $L' = (\mathcal{C} \cup \{a\}, \tau')$ where $\tau'(c) = \tau(c)$ for all $a \in \mathcal{C}$ with $c \neq a$ and $\tau'(a) = \alpha$.
- An explicit definition in T is a pair $D = (a, E)$ such that

  $$T \models \exists x : \alpha \, . \, x = E.$$

  $a = E$ is called the defining axiom of D.
- The extension of T by D, written $T[D]$, is the theory $T' = (L', \Gamma \cup \{a = E\})$.
- **Proposition.** $T \trianglelefteq_m T[D]$.
- The new constant a can be eliminated from expressions of L' by using the defining axiom of D as a rewrite rule.

Implicit Definitions

- Let $T = (L, \Gamma)$ be a theory where $L = (\mathcal{C}, \tau)$, a be a new constant not in L, A is a formula of L containing one free variable x of type α, and $L' = (\mathcal{C} \cup \{a\}, \tau')$ where $\tau'(c) = \tau(c)$ for all $a \in \mathcal{C}$ with $c \neq a$ and $\tau'(a) = \alpha$.
- An implicit definition in T is a pair $D = (a, P)$ where $P = \lambda\, x : \alpha \,.\, A$ such that

  $$T \models \exists\,!\, x : \alpha \,.\, A.$$

  $P(a)$ is called the defining axiom of D.
- The extension of T by D, written $T[D]$, is the theory $T' = (L', \Gamma \cup \{P(a)\})$.
- **Proposition.** $T \trianglelefteq_m T[D]$.
- The new constant a can be eliminated from expressions of L' by using the equation $a = \mathrm{I}\, x : \alpha \,.\, A$ as a rewrite rule.

Mutual Definitions

- Let $T = (L, \Gamma)$ be a theory where $L = (\mathcal{C}, \tau)$, $a_1, \ldots, a_n$ be a list of new constants not in L, A is a formula of L containing n free variables $x_1, \ldots, x_n$ of type $\alpha_1, \ldots, \alpha_n$, and $L' = (\mathcal{C} \cup \{a, \ldots, a_n\}, \tau')$ where $\tau'(c) = \tau(c)$ for all $a \in \mathcal{C}$ with $c \not\in \{a_1, \ldots, a_n\}$ and $\tau'(a_i) = \alpha_i$ for all i with $1 \leq i \leq n$.
- An mutual definition in T is a pair $D = (\langle a_1, \ldots, a_n \rangle, P)$ where $P = \lambda\, x_1 : \alpha_1 \,.\, \cdots \lambda\, x_n : \alpha_n \,.\, A$ such that

  $$T \models \exists\,!\, x_1 : \alpha_1 \,.\, \cdots \exists\,!\, x_n : \alpha_n \,.\, A.$$

  $P(a_1) \cdots (a_n)$ is called the defining axiom of D.
- The extension of T by D, written $T[D]$, is the theory $T' = (L', \Gamma \cup \{P(a_1) \cdots (a_n)\})$.
- **Proposition.** $T \trianglelefteq_m T[D]$.

Profiles

- Let $T = (L, \Gamma)$ be a theory where $L = (\mathcal{C}, \tau)$, a be a new constant not in L, A is a formula of L containing one free variable x of type α, and $L' = (\mathcal{C} \cup \{a\}, \tau')$ where $\tau'(c) = \tau(c)$ for all $a \in \mathcal{C}$ with $c \neq a$ and $\tau'(a) = \alpha$.
- A profile in T is a pair $S = (a, P)$ where $P = \lambda\, x : \alpha\, .\, A$ such that

  $$T \models \exists\, x : \alpha\, .\, A.$$

  $P(a)$ is called the profiling axiom of S.
- The extension of T by S, written $T[S]$, is the theory $T' = (L', \Gamma \cup \{P(a)\})$.
- **Proposition.** $T \trianglelefteq_m T[S]$.
- It may not be possible to eliminate the new constant a from expressions of L' (even using indefinite description).

Mutual Profiles

- Let $T = (L, \Gamma)$ be a theory where $L = (\mathcal{C}, \tau)$, $a_1, \ldots, a_n$ be a list of new constants not in L, A is a formula of L containing n free variables $x_1, \ldots, x_n$ of type $\alpha_1, \ldots, \alpha_n$, and $L' = (\mathcal{C} \cup \{a, \ldots, a_n\}, \tau')$ where $\tau'(c) = \tau(c)$ for all $a \in \mathcal{C}$ with $c \not\in \{a_1, \ldots, a_n\}$ and $\tau'(a_i) = \alpha_i$ for all i with $1 \leq i \leq n$.
- A mutual profile in T is a pair $S = (\langle a_1, \ldots, a_n \rangle, P)$ where $P = \lambda\, x_1 : \alpha_1 \,.\, \cdots \lambda\, x_n : \alpha_n \,.\, A$ such that

  $$T \models \exists\, x_1 : \alpha_1 \,.\, \cdots \exists\, x_n : \alpha_n \,.\, A.$$

  $P(a_1) \cdots (a_n)$ is called the profiling axiom of S.
- The extension of T by S, written $T[S]$, is the theory $T' = (L', \Gamma \cup \{P(a_1) \cdots (a_n)\})$.
- **Proposition.** $T \trianglelefteq_m T[S]$.

Recursive Definitions

- A recursive definition is an implicit definition (a, P) such that the defining axiom $P(a)$ relates a to itself.
- A mutual recursive definition is a mutual definition $(\langle a_1, \ldots, a_n \rangle, P)$ such that the defining axiom $P(a_1) \cdots (a_n)$ relates $a_1, \ldots, a_n$ to each other.
- A (mutual) recursive definition can be expressed as an explicit definition using definite description.
- A (mutual) recursive definition often provides a way of computing the value of certain expressions involving the defined constants.
  - Example: The value of an application $f(a)$ where f is a recursively defined function.
  - Example: The value of a membership formula $a \in s$ where s is a recursively (inductively) defined set.

Inductive Data Types

- An inductive data type consists of:
  1. A domain D of values (i.e., data elements).
  2. A set of constructors that "construct" the values in D.
  3. A set of selectors that "deconstruct" the values in D.
  4. A sentence that states that each member of D can only be constructed in one way (i.e., "no confusion").
  5. A sentence that states that D is inductively defined by the constructors (i.e., "no junk").
  6. A sentence that defines the selectors.
- An inductive type specification in T is a tuple $S = (\alpha, \langle c_1, \ldots, c_m \rangle, \langle s_1, \ldots, s_n \rangle, A_1, A_2, A_3)$ whose components correspond to the components of an inductive data type.
- **Proposition.** The extension of T by S is model conservative if there exists a domain of values, a set of constructors, and a set of selectors that satisfy A_1, A_2, A_3.

Proliferation of Conservative Extensions

- **Problem**: Liberal use of conservative extension results in a proliferation of different theories that are essentially equivalent.
- **Solution**:
  1. Whenever a theory T is conservatively extended to T', overwrite T with T'.
  2. Record the "development" of a theory (e.g., to facilitate linking theories with interpretations).

Conservative Stacks

- A conservative stack is a finite sequence $\Sigma = \langle T_0, \ldots, T_n \rangle$ of theories such that $T_i \trianglelefteq T_{i+1}$ for all i with $0 \leq i < n$.
  - T_0 is the base theory of Σ.
  - T_n is the theory of Σ.
- A conservative stack $\Sigma = \langle T_0, \ldots, T_n \rangle$ is conservatively extended by overwriting Σ with $\Sigma' = \langle T_0, \ldots, T_n, T_{n+1} \rangle$ where $T_n \trianglelefteq T_{n+1}$.
- A theory can be implemented as a theory object that includes a conservative stack Σ and a set of the currently known theorems of the theory of Σ.