The Z/EVES 2.2 Mathematical Toolkit

TR-03-5493-05c

Mark Saaltink

Release date: June 2003

ORA Canada
P.O. Box 46005, 2339 Ogilvie Rd.
Ottawa, Ontario K1J 9M7
CANADA

Contents

1 Introduction

The Z/EVES Mathematical Toolkit[1] includes the declaration of all the constants of the Standard Mathematical Toolkit as described by Spivey [3] or the proposed ISO Standard for Z [1], and presents useful theorems about these constants. The theorems are divided into two groups. The first group contains theorems meant for human consumption, which are presented in the most natural way. The second group contains theorems meant for the prover to use automatically, which are presented in whatever way will work best. Often, however, many theorems in the human consumption group are also suitable for automatic use, and are marked as rewrite rules.

This report is the specification of the standard "toolkit" section distributed with Z/EVES, Version 2.2 [2].

The toolkit defines operations in several categories, which we summarize here. For each operation, we show its LaTeX markup command (needed for users of the command line interface to Z/EVES, but not by users of the graphical interface), give the page of this report containing its definition (or the main theorems about it), and a brief description of its meaning.

General

KnownMember	`KnownMember`	p. 7	membership (for "weakening" rules)
$\mu\, x : S$	`\mu x: S`	p. 9	definite description terms
$x \neq y$	`x \neq y`	p. 11	not equal
$x \notin y$	`x \notin y`	p. 11	not a member

Sets

$\mathbb{P}\, X$	`\power X`	p. 13	powerset
$X \times Y$	`X \cross Y`	p. 14	cross product
$\emptyset$	`\emptyset`	p. 15	empty set
$\mathbb{P}_1\, X$	`\power_1 X`	p. 17	non-empty powerset
$S \subseteq T$	`S \subseteq T`	p. 18	subset relation
$S \subset T$	`S \subset T`	p. 18	proper subset relation
$S \cup T$	`S \cup T`	p. 19	set union
$S \cap T$	`S \cap T`	p. 20	set intersection
$S \setminus T$	`S \setminus T`	p. 22	set difference (relative complement)
$\bigcup S, \bigcap S$	`\bigcup S, \bigcap S`	p. 24	generalized union or intersection

Ordered Pairs

first	`first`	p. 25	first component of a pair
second	`second`	p. 25	second component of a pair
$x \mapsto y$	`x \mapsto y`	p. 27	maplets

[1]This work was funded by the United States Department of Defense under contract MDA904-95-C-2031.

Relations

$X \leftrightarrow Y$	`X \rel Y`	p. 26	relation space
$\operatorname{dom} R, \operatorname{ran} R$	`\dom R, \ran R`	p. 28	domain, range of a relation
$\operatorname{id} S$	`\id S`	p. 30	identity relation
$Q \mathbin{;} R, R \circ Q$	`Q \comp R, R \circ Q`	p. 31	composition
$S \lhd R$	`S \dres R`	p. 33	domain restriction
$R \rhd S$	`R \rres S`	p. 33	range restriction
$S \ntriangleleft R$	`S \ndres R`	p. 36	domain anti-restriction
$R \ntriangleright S$	`R \nrres S`	p. 36	range anti-restriction
$R^{\sim}$	`R \inv`	p. 38	inverse relation
$R(\lvert S \rvert)$	`R \limg S \rimg`	p. 40	relational image
$Q \oplus R$	`Q \oplus R`	p. 42	overriding
R^+, R^*	`R\plus, R\star`	p. 43	transitive closure
R^k	`R \bsup k \esup`	p. 57	iterate of a relation

Functions

$X \pfun Y, X \rightarrow Y$	`X \pfun Y, X \fun Y`	p. 45	function spaces
$X \pinj Y, X \rightarrowtail Y$	`X \pinj Y, X \inj Y`	p. 48	injective (1-1) function spaces
$X \psurj Y, X \twoheadrightarrow Y$	`X \psurj Y, X \surj Y`	p. 50	surjective (onto) function spaces
$X \bij Y$	`X \bij Y`	p. 51	bijections
$X \ffun Y, X \finj Y$	`X \ffun Y, X \finj Y`	p. 62	finite functions
$g \circ f$	`g \circ f`	p. 31	composition
$f(\lvert S \rvert)$	`f \limg S \rimg`	p. 40	image
$f \oplus g$	`f \oplus g`	p. 42	overriding
f^n	`f \bsup n \esup`	p. 57	iterate of a function
$f\ applies\$to\ x$	`f~applies\$to~x`	p. 10	applicability

Numbers and finiteness

$\mathbb{N}, \mathbb{N}_1$	`\nat, \nat_1`	p. 56	natural numbers
$succ(n)$	`succ(n)`	p. 56	successor $(n+1)$
$k \mathrel{..} n$	`k \upto n`	p. 58	ranges
$min\ S, max\ S$	`min~S, max~S`	p. 64	minimum and maximum
$\mathbb{F}\,X$	`\finset X`	p. 59	finite subsets
$\mathbb{F}_1\,X$	`\finset_1 X`	p. 59	non-empty finite subsets
$\#S$	`\# S`	p. 61	cardinality

Sequences

$\mathrm{seq}\, X, \mathrm{seq}_1\, X$	`\seq X, \seq_1 X`	p. 67	sequences
$\mathrm{iseq}\, X$	`\iseq X`	p. 67	injective sequences
$s \frown t$	`s \cat t`	p. 70	concatenation
$\frown/\, s$	`\dcat s`	p. 79	distributed concatenation
$head\, s, last\, s$	`head~s, last~s`	p. 71	first (last) element
$tail\, s, front\, s$	`tail~s, front~s`	p. 71	parts of a sequence
$rev\, s$	`rev~s`	p. 73	reversal
$S \upharpoonleft s$	`S \extract s`	p. 74	selection of a subsequence
$s \upharpoonright S$	`s \filter S`	p. 74	selection of a subsequence
$squash(f)$	`squash(f)`	p. 74	creation of a sequence
$s\ \mathsf{prefix}\ t$	`s \prefix t`	p. 78	subsequence relations
$s\ \mathsf{suffix}\ t$	`s \suffix t`	p. 78	subsequence relations
$s\ \mathsf{in}\ t$	`s \inseq t`	p. 78	subsequence relations
$\mathsf{disjoint}\ s$	`\disjoint s`	p. 80	disjointness
$\mathsf{partition}\ s$	`\partition s`	p. 80	partitions

Bags

$\mathrm{bag}\, X$	`\bag X`	p. 83	bags (multisets)
$count\, B, B \sharp x$	`count~B, B \bcount x`	p. 85	multiplicity in a bag
$x\ \mathrm{in}\ B$	`x \inbag B`	p. 85	membership in a bag
$A \sqsubseteq B$	`A \subbageq B`	p. 86	subbag relationship
$n \otimes B$	`n \otimes B`	p. 87	bag scaling
$A \uplus B$	`A \uplus B`	p. 88	bag union
$A \uminus B$	`A \uminus B`	p. 89	bag difference
$items(s)$	`items(s)`	p. 90	bag of elements from a sequence

1.1 Changes since the Z/EVES Version 2.0 and 2.1 Toolkit

1. Many new theorems have been added. Significant additions are for constant functions, transitive closure ($\_^+$), reflexive transitive closure ($\_^*$), and arithmetic.
2. Some typographical errors were corrected.
3. The three predicates defining prefix, suffix, and in were labelled, so that they can be referred to in proofs.

1.2 Changes since Version 2.2 (for Z/EVES Version 1.5)

There have been several changes from Version 2.2 of the Toolkit:

1. The toolkit no longer has a version number distinct from the Z/EVES version, as that just seems confusing.
2. A number of rewriting rules have been disabled, as they were in general quite inefficient. These rules were capable of causing the prover to do lots of work on subgoals that usually failed. Where possible, simple cases of these rules, that recognize special cases syntactically, have been added.
3. The induction theorems have been rewritten to use $\_ \subseteq \_$ in their conclusions, and have been made disabled rewrite rules. This makes them slightly easier to use, since they can be applied, with the rewriter working out the instantiation.

4. A few theorems were generalized to be applicable in cases where non-maximal generic actuals are used.

5. Several new theorems were added.

6. Five errors were corrected.

2 Automation strategies

Before presenting the specification of the Toolkit, we will discuss some of the technical issues that influence the form of its theorems. This section is rather technical and should be skipped on first reading.

2.1 Weakening

There is a basic rewriting strategy that colours much of the Toolkit theory. It is a bit tricky to automate "weakening" proofs, where membership in a large set is inferred from membership in a small set. For example, $x \in \mathbb{N} \times \mathbb{N}_1$ implies $x \in \mathbb{Z} \times \mathbb{N}$. These sorts of goals arise all the time, and should be trivial to prove.

We adopt the following approach:

- Given a global constant declared as $c : T$, a grule $c \in T$ is automatically generated. (Such theorems must be added by hand for constants declared as abbreviations. We give these theorems names of the form *x_type*.)
- A special "known membership" function is defined, and we add a forward rule $x \in S \Rightarrow x \mathit{knownIn}\, S$ and another $\neg\, x \in S \Rightarrow \neg\, x \mathit{knownIn}\, S$. (Unfortunately, a *knownIn* relation is unsuitable here, as it would need a generic parameter. Therefore, we use a generic schema *KnownMember*, with the set as the generic actual and component *element* as the member.)
- We add the rewrite rule $x \mathit{knownIn}\, T \land T \in \mathbb{P}\, S \Rightarrow x \in S$. We similarly add $\neg\, x \mathit{knownIn}\, T \land S \in \mathbb{P}\, T \Rightarrow \neg\, x \in S$.

Most weakening proofs give rise to subgoals of the form $S \in \mathbb{P}\, T$. If S is itself a global constant, the weakening rule can apply again. We also give rules for such subgoals for interesting cases of S and T below. For example, $A \nrightarrow B \in \mathbb{P}(A' \nrightarrow B') \Leftrightarrow A \in \mathbb{P}\, A' \land B \in \mathbb{P}\, B'$. These theorems have names ending with "*_sub*". Generally, these rules express the monotonicity of set constructors such as $\_ \nrightarrow \_$, $\mathbb{F}\,\_$ and $\mathrm{seq}\,\_$.

2.2 Ideal rules

Theorems expressing properties inherited by subsets are also worth automating. For example, any subset of a relation is a relation, any subset of a partial function is a partial function, and any subset of a finite set is a finite set.

This sort of reasoning plays out as follows: the fact that X is a subset of Y is recorded as $X \in \mathbb{P}\, Y$. Thus, if we are trying to show $X \in I$, the weakening rule will give a subgoal of the form $\mathbb{P}\, Y \in \mathbb{P}\, I$. If I is one of the sets mentioned above (e.g., $I = A \leftrightarrow B$ or ..., or $I = \mathbb{F}\, A$), then $\mathbb{P}\, Y \in \mathbb{P}\, I \Leftrightarrow Y \in I$. Thus, adding this *_ideal* rule for each different set I is enough to allow the automation of these proofs.

2.3 Facts about function results

A general rule gives the fact $f(x) \in R$ if $f \in D \nrightarrow R$ and $x \in \mathrm{dom}\, f$—that is, function applications have values in the range of the function.

In cases where a tighter containing set is available for a function application, it may be useful to give an additional weakening rule. For example, the domain restriction $(S \lhd R)$ is a subset of R, whereas from the declaration of $\_ \lhd \_$ we can conclude only that it is a subset of $X \leftrightarrow Y$ (where $X \leftrightarrow Y$ is the type of R). This fact about domain restriction could be expressed by the predicate

$$\forall\, S : \mathbb{P}\, X;\ R : X \leftrightarrow Y \bullet S \lhd R \subseteq R.$$

It is possible to express this fact in a form that can be used more automatically by EVES, by writing instead the rule

$$\forall S : \mathbb{P}\, X;\; RX \leftrightarrow Y \mid \mathbb{P}\, R \in \mathbb{P}\, Z \bullet S \lhd R \in Z.$$

This form interacts particularly well with the "ideal" rules. For example, if Z is an ideal set, then the subgoal $\mathbb{P}\, R \in \mathbb{P}\, Z$ will be rewritten to $R \in Z$—so, for example, if R is an injection, Z/EVES can conclude that $S \lhd R$ is also an injection.

These theorems are given names ending in *_result*.

2.4 Computation rules

It is useful to be able to use Z/EVES to calculate the value of a Z expression, such as $\mathrm{dom}\{1 \mapsto 1, 2 \mapsto 4, 3 \mapsto 9\}$.

Set constructions, sequence constructions, and bag constructions are in fact formed from three primitives: a constant for the empty value (e.g., $\{\}$, $\langle\rangle$, or $[\![]\!]$); a constructor for singletons; and a "join" operation ($\_ \cup \_$ for sets, $\_ \frown \_$ for sequences, and $\_ \uplus \_$ for bags). For example, $\{1, 2, 3\}$ is really just an abbreviation for $\{1\} \cup \{2\} \cup \{3\}$).

In writing enough rewrite rules to allow for computations of functions applied to constructions, it is therefore necessary to cover the three cases (empty, unit, join). Wherever possible in this Toolkit, we have included enough such rules to allow these computations to be performed. For example, the three rules *domEmpty*, *domSingleton*, and *domCup* are enough to allow domains of explicitly given relations to be computed by rewriting.

3 Weakening

Here is the definition of the "known membership" relation. As explained in Section 2.1, it is necessary to use a schema rather than a relation.

Definitions

$$\begin{array}{|l}
\hline
KnownMember[X] \\
element : X \\
\hline
\end{array}$$

Theorems

theorem frule knownMember $[X]$
$element \in X \Rightarrow KnownMember[X]$

theorem rule weakening
$KnownMember[X] \land X \in \mathbb{P}\ Y \Rightarrow element \in Y$

4 Tuples

Tuples are part of the Z notation. We present the main theorems used in proofs.

4.1 Two-element tuples

theorem grule select_2_1
$(x, y).1 = x$

theorem grule select_2_2
$(x, y).2 = y$

theorem rule eqTuple2
$(x, y) = (x', y') \Leftrightarrow x = x' \land y = y'$

theorem rule select_1_member
$x \in X \times Y \Rightarrow x.1 \in X$

theorem rule select_2_member
$x \in X \times Y \Rightarrow x.2 \in Y$

theorem grule tupleComposition2
$x \in X \times Y \Rightarrow x = (x.1, x.2)$

4.2 Three-element tuples

theorem grule select_3_1
$(x, y, z).1 = x$

theorem grule select_3_2
$(x, y, z).2 = y$

theorem grule select_3_3
$(x, y, z).3 = z$

theorem rule eqTuple3
$(x, y, z) = (x', y', z') \Leftrightarrow x = x' \land y = y' \land z = z'$

5 Mu terms

Mu terms are treated specially by Z/EVES; a term of the form $\mu\, ST \bullet e$ is converted into $\mu\, m : \{ST \bullet e\}$ (unless it already has this form $\mu\, x : S$ for some set S). This latter form is treated as a function of S.

Theorems

We present here some of the theorems needed for dealing with mu terms.

theorem muInSet $[S]$
$(\exists\, a : S \bullet \forall\, b : S \bullet b = a) \Rightarrow (\mu\, x : S) \in S$

theorem muValue $[S]$
$s \in S \land (\forall\, s' : S \bullet s' = s) \Rightarrow (\mu\, x : S) = s$

Automation

We generally do not provide much automation for mu terms. When a mu term appears in an equality, we can do a bit better, since we then have a candidate value for the expression.

theorem rule muValue1 $[S]$
$\forall\, s : S \mid (\forall\, s' : S \bullet s' = s) \bullet (\mu\, x : S) = s \Leftrightarrow \mathit{true}$

theorem rule muValue2 $[S]$
$\forall\, s : S \mid (\forall\, s' : S \bullet s' = s) \bullet s = (\mu\, x : S) \Leftrightarrow \mathit{true}$

theorem rule muSingleton
$(\mu\, x : \{y\}) = y$

6 Applicability

Relation $\_applies\$to\_$ is used in domain checking conditions. It is declared as

$$\_applies\$to\_[X, Y] : (X \leftrightarrow Y) \leftrightarrow X.$$

Most of the rules about $\_applies\$to\_$ appear later in the Toolkit, after "dom" has been introduced.

Theorems

theorem disabled rule appliesToDef $[X, Y]$
$\forall R : X \leftrightarrow Y;\ x : X \bullet R\ applies\$to\ x \Leftrightarrow (\exists y : Y \mid (x, y) \in R \bullet \forall y' : Y \mid (x, y') \in R \bullet y = y')$

7 Negations

Definitions

syntax $\neq$ *inrel* `\neq`
syntax $\notin$ *inrel* `\notin`

$[X]$

$\_ \neq \_ : X \leftrightarrow X$

$\_ \notin \_ : X \leftrightarrow \mathbb{P}\, X$

$\langle\!\langle\, \mathrm{notEqDef} \,\rangle\!\rangle$
$\forall\, x, y : X \bullet x \neq y \Leftrightarrow \neg\, x = y$

$\langle\!\langle\, \mathrm{notinDef} \,\rangle\!\rangle$
$\forall\, x : X;\ S : \mathbb{P}\, X \bullet x \notin S \Leftrightarrow \neg\, x \in S$

Automation

theorem rule notEqRule $[X]$
$x \neq y \Leftrightarrow (x \in X \land y \in X \land \neg\, x = y)$

theorem rule notInRule $[X]$
$x \notin S \Leftrightarrow (x \in X \land S \in \mathbb{P}\, X \land \neg\, x \in S)$

8 Sets

8.1 Extensionality

The extensionality property is disabled, and needs to be enabled or applied manually in those proofs where it is needed.

Additional extensionality properties are defined for relations (theorem *relationExtensionality* in Section 10.1), functions (theorems *pfunExtensionality* and *funExtensionality* in Section 11.1), and bags (theorem *bagExtensionality* in Section 14.1).

Theorems

theorem disabled rule extensionality
$X = Y \Leftrightarrow (\forall x : X \bullet x \in Y) \wedge (\forall y : Y \bullet y \in X)$

theorem disabled rule extensionality2
$X = Y \Leftrightarrow X \in \mathbb{P}\, Y \wedge Y \in \mathbb{P}\, X$

Theorem *extensionality*3 cannot be a rule, because X is not bound in the pattern $S = T$.

theorem extensionality3 $[X]$
$\forall S, T : \mathbb{P}\, X \bullet S = T \Leftrightarrow (\forall x : X \mid x \in S \bullet x \in T) \wedge (\forall x' : X \mid x' \in T \bullet x' \in S)$

Theorem *extensionality*4, expressed using the subset relation, appears in Section 8.7.

8.2 Powersets

The powerset notation is a predefined part of the Z notation. Here are some basic theorems.

Theorems

theorem disabled rule inPower
$X \in \mathbb{P}\, Y \Leftrightarrow (\forall e : X \bullet e \in Y)$

theorem rule inPowerSelf
$X \in \mathbb{P}\, X$

theorem rule power_sub
$\mathbb{P}\, X \in \mathbb{P}(\mathbb{P}\, Y) \Leftrightarrow X \in \mathbb{P}\, Y$

The following two facts are automated by the "weakening" rules in Section 3.

theorem inPowerTransitive
$X \in \mathbb{P}\, Y \wedge Y \in \mathbb{P}\, Z \Rightarrow X \in \mathbb{P}\, Z$

theorem inSubset
$x \in Y \wedge Y \in \mathbb{P}\, Z \Rightarrow x \in Z$

8.3 Cross products

Cross products are part of the Z notation. Here are some basic theorems about two and three-element cross products.

theorem disabled rule inCross2
$p \in X \times Y \Leftrightarrow (\exists x : X;\ y : Y \bullet p = (x, y))$

theorem rule tupleInCross2
$(x, y) \in X \times Y \Leftrightarrow x \in X \land y \in Y$

theorem rule crossSubsetCross2
$A \times B \in \mathbb{P}(X \times Y) \Leftrightarrow A = \{\} \lor B = \{\} \lor A \in \mathbb{P}\, X \land B \in \mathbb{P}\, Y$

theorem rule crossNull_2_1
$\{\} \times Y = \{\}$

theorem rule crossNull_2_2
$X \times \{\} = \{\}$

theorem rule crossEqualNull2
$X \times Y = \{\} \Leftrightarrow X = \{\} \lor Y = \{\}$

theorem disabled rule inCross3
$p \in X \times Y \times Z \Leftrightarrow (\exists x : X;\ y : Y;\ z : Z \bullet p = (x, y, z))$

theorem rule tupleInCross3
$(x, y, z) \in X \times Y \times Z \Leftrightarrow x \in X \land y \in Y \land z \in Z$

theorem rule crossSubsetCross3
$A \times B \times C \in \mathbb{P}(X \times Y \times Z) \Leftrightarrow A = \{\} \lor B = \{\} \lor C = \{\} \lor A \in \mathbb{P}\, X \land B \in \mathbb{P}\, Y \land C \in \mathbb{P}\, Z$

theorem rule crossNull_3_1
$\{\} \times Y \times Z = \{\}$

theorem rule crossNull_3_2
$X \times \{\} \times Z = \{\}$

theorem rule crossNull_3_3
$X \times Y \times \{\} = \{\}$

theorem rule crossEqualNull3
$X \times Y \times Z = \{\} \Leftrightarrow X = \{\} \lor Y = \{\} \lor Z = \{\}$

8.4 Empty set

Definition

syntax *word* `\empty`
syntax $\emptyset$ *word* `\emptyset`

The name `\empty` is a synonym for `\emptyset` for backward compatibility with some early versions of LaTeX markup for Z. The name `\emptyset` is preferred, especially for printing, as newer versions of LaTeX markup will not display `\empty` correctly.

$$\emptyset[X] == \{ x : X \mid \mathit{false} \}$$

Theorems

It is convenient in proofs to use the empty set extension instead of the empty set, as the extension is simpler (since it does not use a generic actual).

theorem rule emptyDefinition $[X]$
$\emptyset[X] = \{\}$

theorem rule inNull
$\neg\, x \in \{\}$

theorem rule nullSubset
$\{\} \in \mathbb{P}\, X$

theorem rule powerNull
$\mathbb{P}\{\} = \{\{\}\}$

theorem nonEmptySetHasMember
$S = \{\} \lor (\exists\, x : S \bullet \mathit{true})$

8.5 Unit sets

Unit sets can be denoted by set displays.

Theorems

theorem rule inUnit
$x \in \{y\} \Leftrightarrow x = y$

theorem rule unitSubset
$\{x\} \in \mathbb{P}\, X \Leftrightarrow x \in X$

theorem rule unitEqualUnit
$\{x\} = \{y\} \Leftrightarrow x = y$

theorem rule nullEqualUnit
$\neg\, (\{\} = \{x\})$

theorem rule unitEqualNull
$\neg\, (\{x\} = \{\})$

theorem rule inPowerUnit
$x \in \mathbb{P}\{y\} \Leftrightarrow x = \{y\} \lor x = \{\}$

We cannot directly state the theorem $\mathbb{P}\{x\} = \{\{\}, \{x\}\}$ because of the way Z/EVES represents set displays (as unions of unit sets). We need to have some containing type as a generic actual for the union. Thus, the best we can do is the following, which unfortunately cannot be used as a rewrite rule because X does not appear in the left hand side.

theorem powerUnit $[X]$
$\forall\, x : X \bullet \mathbb{P}\{x\} = \{\{\}, \{x\}\}.$

8.6 Non-empty powerset

Definition

$$\mathbb{P}_1\, X == \{\, S : \mathbb{P}\, X \mid S \neq \emptyset \,\}$$

Theorems

theorem grule power1_type $[X]$
$\mathbb{P}_1\, X \in \mathbb{P}(\mathbb{P}\, X)$

theorem rule inPower1
$x \in \mathbb{P}_1\, X \Leftrightarrow x \in \mathbb{P}\, X \land \neg\, x = \{\}$

theorem rule power1Empty
$\mathbb{P}_1\{\} = \{\}$

theorem rule power1Unit
$\mathbb{P}_1\{x\} = \{\{x\}\}$

Automation

theorem rule power1_strong_type
$\mathbb{P}_1\, X \in \mathbb{P}(\mathbb{P}\, Y) \Leftrightarrow X \in \mathbb{P}\, Y$

theorem rule power1_sub
$\mathbb{P}_1\, X \in \mathbb{P}(\mathbb{P}_1\, Y) \Leftrightarrow X \in \mathbb{P}\, Y$

8.7 Subsets

Definition

syntax $\subseteq$ *inrel* `\subseteq`
syntax $\subset$ *inrel* `\subset`

$[X]$

$\_ \subseteq \_, \_ \subset \_ : \mathbb{P}\, X \leftrightarrow \mathbb{P}\, X$

$\langle\!\langle$ disabled rule subDef $\rangle\!\rangle$
$\forall A, B : \mathbb{P}\, X \bullet A \subseteq B \Leftrightarrow (\forall x : A \bullet x \in B)$

$\langle\!\langle$ disabled rule psubDef $\rangle\!\rangle$
$\forall A, B : \mathbb{P}\, X \bullet A \subset B \Leftrightarrow A \subseteq B \land A \neq B$

Theorems

theorem disabled rule subsetSelf $[X]$
$\forall S : \mathbb{P}\, X \bullet S \subseteq S$

theorem disabled rule nullsetSubset $[X]$
$\forall S : \mathbb{P}\, X \bullet \{\} \subseteq S$

theorem disabled rule subsetTransitive $[X]$
$\forall A, B, C : \mathbb{P}\, X \mid A \subseteq B \subseteq C \bullet A \subseteq C$

theorem rule psubsetSelf $[X]$
$\forall S : \mathbb{P}\, X \bullet \neg\, S \subset S$

theorem rule nullsetPsubset $[X]$
$\forall S : \mathbb{P}\, X \bullet \{\} \subset S \Leftrightarrow \neg\, S = \{\}$

theorem extensionality4 $[X]$
$\forall S, T : \mathbb{P}\, X \bullet S = T \Leftrightarrow S \subseteq T \land T \subseteq S$

Automation

The subset notation is convenient, but is awkward in expressing theorems because of the generic actual. We cannot express the simple fact that A is a subset of B without at the same time constraining them to be subsets of something else (the generic actual). A different notation, $A \in \mathbb{P}\, B$, expresses exactly what we mean, although it is uglier than the subset notation.

theorem rule subsetDef $[X]$
$A \subseteq B \Leftrightarrow A \in \mathbb{P}\, B \land B \in \mathbb{P}\, X$

theorem rule psubsetDef $[X]$
$A \subset B \Leftrightarrow A \in \mathbb{P}\, B \land B \in \mathbb{P}\, X \land \neg\, A = B$

8.8 Union

Definitions

Function $\_ \cup \_$ is predefined, as it is used in the Z/EVES representation of set extensions—$\{a, b, \ldots\}$ is represented as if it were $\{a\} \cup \{b\} \cup \cdots$.

Theorems

theorem rule inCup $[X]$
$\forall A, B : \mathbb{P}\, X \bullet x \in A \cup B \Leftrightarrow x \in A \lor x \in B$

theorem disabled rule cupSubsetLeft $[X]$
$S \subseteq [X] T \Rightarrow S \cup T = T$

theorem disabled rule cupSubsetRight $[X]$
$T \subseteq [X] S \Rightarrow S \cup T = S$

theorem rule cupNullLeft $[X]$
$\forall S : \mathbb{P}\, X \bullet \{\} \cup S = S$

theorem rule cupNullRight $[X]$
$\forall S : \mathbb{P}\, X \bullet S \cup \{\} = S$

theorem rule cupCommutes $[X]$
$\forall S, T : \mathbb{P}\, X \bullet S \cup T = T \cup S$

theorem rule cupAssociates $[X]$
$\forall S, T, V : \mathbb{P}\, X \bullet (S \cup T) \cup V = S \cup (T \cup V)$

theorem rule cupSubset $[X]$
$\forall S, T : \mathbb{P}\, X \bullet (S \cup T) \in \mathbb{P}\, U \Leftrightarrow S \in \mathbb{P}\, U \land T \in \mathbb{P}\, U$

Automation

The following two rules are needed to compute equalities between set extensions, for example, $\{1, 2\} = \{\}$. However, they are not enough, we would need additional rules to show $\neg \{1, 2\} = \{2, 3\}$. These additional facts do not appear to make good rewrite rules.

theorem rule cupEqualNullLeft $[X]$
$\forall S, T : \mathbb{P}\, X \bullet S \cup T = \{\} \Leftrightarrow S = \{\} \land T = \{\}$

theorem rule cupEqualNullRight $[X]$
$\forall S, T : \mathbb{P}\, X \bullet \{\} = S \cup T \Leftrightarrow S = \{\} \land T = \{\}$

Rule *cupPermutes* is needed to complement the associative and commutative laws, because of the way "permutative" rewrite rules are used in rewriting.

theorem rule cupPermutes $[X]$
$\forall S, T, V : \mathbb{P}\, X \bullet S \cup (T \cup V) = T \cup (S \cup V)$

theorem rule subsetCup $[X]$
$\forall T, U : \mathbb{P}\, X \bullet (S \in \mathbb{P}\, T \lor S \in \mathbb{P}\, U) \Rightarrow S \in \mathbb{P}(T \cup U)$

8.9 Intersection

Definitions

syntax $\cap$ *infun*4 `\cap`

$[X]$

$\_ \cap \_ : \mathbb{P}\, X \times \mathbb{P}\, X \rightarrow \mathbb{P}\, X$

$\langle\!\langle$ capDefinition $\rangle\!\rangle$
$\forall x : X;\ A, B : \mathbb{P}\, X \bullet x \in A \cap B \Leftrightarrow x \in A \wedge x \in B$

Theorems

theorem rule inCap $[X]$
$\forall A, B : \mathbb{P}\, X \bullet x \in A \cap B \Leftrightarrow x \in A \wedge x \in B$

theorem disabled rule capSubsetLeft $[X]$
$S \subseteq [X] T \Rightarrow S \cap T = S$

theorem disabled rule capSubsetRight $[X]$
$T \subseteq [X] S \Rightarrow S \cap T = T$

theorem rule capNullLeft $[X]$
$\forall S : \mathbb{P}\, X \bullet \{\} \cap S = \{\}$

theorem rule capNullRight $[X]$
$\forall S : \mathbb{P}\, X \bullet S \cap \{\} = \{\}$

theorem rule unitCap $[X]$
$\forall x : X;\ S : \mathbb{P}\, X \bullet \{x\} \cap S = \mathbf{if}\ x \in S\ \mathbf{then}\ \{x\}\ \mathbf{else}\ \{\}$

theorem rule capUnit $[X]$
$\forall x : X;\ S : \mathbb{P}\, X \bullet S \cap \{x\} = \mathbf{if}\ x \in S\ \mathbf{then}\ \{x\}\ \mathbf{else}\ \{\}$

theorem rule capCommutes $[X]$
$\forall S, T : \mathbb{P}\, X \bullet S \cap T = T \cap S$

theorem rule capAssociates $[X]$
$\forall S, T, V : \mathbb{P}\, X \bullet (S \cap T) \cap V = S \cap (T \cap V)$

theorem disabled rule capSubset $[X]$
$\forall S, T, U : \mathbb{P}\, X \bullet (S \subseteq U \vee T \subseteq U) \Rightarrow S \cap T \subseteq U$

theorem rule subsetCap $[X]$
$\forall T, U : \mathbb{P}\, X \bullet S \in \mathbb{P}(T \cap U) \Leftrightarrow S \in \mathbb{P}\, T \wedge S \in \mathbb{P}\, U$

Automation

Rule *capPermutes* is needed to complement the associative and commutative laws.

theorem rule capPermutes $[X]$
$\forall S, T, V : \mathbb{P}\, X \bullet S \cap (T \cap V) = T \cap (S \cap V)$

theorem rule cap_result $[X]$
$\forall S, T : \mathbb{P}\, X \mid \mathbb{P}\, S \in \mathbb{P}\, Z \lor \mathbb{P}\, T \in \mathbb{P}\, Z \bullet S \cap T \in Z$

In order to compute intersections of literals, we need the following two rules.

theorem rule computeCap1 $[X]$
$\forall x : X;\ S, T : \mathbb{P}\, X \mid x \in T \bullet (\{x\} \cup S) \cap T = \{x\} \cup (S \cap T)$

theorem rule computeCap2 $[X]$
$\forall x : X;\ S, T : \mathbb{P}\, X \mid \lnot\, x \in T \bullet (\{x\} \cup S) \cap T = S \cap T$

8.10 Set difference

Definitions

syntax $\setminus$ *infun*3 `\setminus`

$$
\begin{array}{l}
[X] \\
\hline
\_ \setminus \_ : \mathbb{P}\, X \times \mathbb{P}\, X \rightarrow \mathbb{P}\, X \\
\hline
\langle\!\langle \text{diffDefinition} \rangle\!\rangle \\
\forall x : X;\ A, B : \mathbb{P}\, X \bullet x \in A \setminus B \Leftrightarrow x \in A \land \lnot x \in B
\end{array}
$$

Theorems

theorem rule inDiff $[X]$
$\forall S, T : \mathbb{P}\, X \bullet x \in S \setminus T \Leftrightarrow x \in S \land \lnot x \in T$

theorem rule diffDiff $[X]$
$\forall S, T, U : \mathbb{P}\, X \bullet (S \setminus T) \setminus U = S \setminus (T \cup U)$

theorem rule diffSubset $[X]$
$\forall S, T, U : \mathbb{P}\, X \bullet S \setminus T \subseteq U \Leftrightarrow S \subseteq U \cup T$

theorem disabled rule diffSuperset $[X]$
$\forall S, T : \mathbb{P}\, X \mid S \in \mathbb{P}\, T \bullet S \setminus T = \{\}$

theorem rule diffEmptyLeft $[X]$
$\forall S : \mathbb{P}\, X \bullet \{\} \setminus S = \{\}$

theorem rule diffEmptyRight $[X]$
$\forall S : \mathbb{P}\, X \bullet S \setminus \{\} = S$

theorem rule unitDiff $[X]$
$\forall x : X;\ S : \mathbb{P}\, X \bullet \{x\} \setminus S = \mathbf{if}\ x \in S\ \mathbf{then}\ \{\}\ \mathbf{else}\ \{x\}$

Automation

The following is derived from the fact $S \setminus T \subseteq S$.

theorem rule diff_result $[X]$
$\forall S, T : \mathbb{P}\, X \mid \mathbb{P}\, S \in \mathbb{P}\, Z \bullet S \setminus T \in Z$

In order to compute differences of literals, we need the following two rules.

theorem rule computeDiff1 $[X]$
$\forall x : X;\ S, T : \mathbb{P}\, X \mid x \in T \bullet (\{x\} \cup S) \setminus T = S \setminus T$

theorem rule computeDiff2 $[X]$
$\forall x : X;\ S, T : \mathbb{P}\, X \mid \lnot x \in T \bullet (\{x\} \cup S) \setminus T = \{x\} \cup (S \setminus T)$

8.11 Distribution laws

There are a number of distributivity properties for the set operators. These are expressed as disabled rules, because there is no obvious reason to prefer one form over another.

Theorems

theorem disabled rule distributeCupOverCapRight $[X]$
$\forall A, B, C : \mathbb{P} X \bullet A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

theorem disabled rule distributeCupOverCapLeft $[X]$
$\forall A, B, C : \mathbb{P} X \bullet (A \cap B) \cup C = (A \cup C) \cap (B \cup C)$

theorem disabled rule distributeCapOverCupRight $[X]$
$\forall A, B, C : \mathbb{P} X \bullet A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

theorem disabled rule distributeCapOverCupLeft $[X]$
$\forall A, B, C : \mathbb{P} X \bullet (A \cup B) \cap C = (A \cap C) \cup (B \cap C)$

theorem disabled rule distributeDiffOverCupRight $[X]$
$\forall A, B, C : \mathbb{P} X \bullet A \setminus (B \cup C) = (A \setminus B) \cap (A \setminus C)$

theorem disabled rule distributeDiffOverCupLeft $[X]$
$\forall A, B, C : \mathbb{P} X \bullet (A \cup B) \setminus C = (A \setminus C) \cup (B \setminus C)$

theorem disabled rule distributeDiffOverCapRight $[X]$
$\forall A, B, C : \mathbb{P} X \bullet A \setminus (B \cap C) = (A \setminus B) \cup (A \setminus C)$

theorem disabled rule distributeDiffOverCapLeft $[X]$
$\forall A, B, C : \mathbb{P} X \bullet (A \cap B) \setminus C = (A \setminus C) \cap (B \setminus C)$

8.12 Generalized union and intersection

Definitions

$[X]$

$\bigcup, \bigcap : \mathbb{P}(\mathbb{P}\, X) \rightarrow \mathbb{P}\, X$

$\langle\!\langle$ rule inBigcup $\rangle\!\rangle$
$\forall\, x : X;\ A : \mathbb{P}(\mathbb{P}\, X) \bullet x \in \bigcup A \Leftrightarrow (\exists\, B : A \bullet x \in B)$

$\langle\!\langle$ rule inBigcap $\rangle\!\rangle$
$\forall\, x : X;\ A : \mathbb{P}(\mathbb{P}\, X) \bullet x \in \bigcap A \Leftrightarrow (\forall\, B : A \bullet x \in B)$

Theorems

theorem rule bigcupEmpty $[X]$
$\bigcup[X]\{\} = \{\}$

theorem rule bigcupUnit $[X]$
$S \in \mathbb{P}\, X \Rightarrow \bigcup\{S\} = S$

theorem rule bigcupUnion $[X]$
$\forall\, S, T : \mathbb{P}(\mathbb{P}\, X) \bullet \bigcup(S \cup T) = (\bigcup S) \cup (\bigcup T)$

theorem rule inPowerBigcup $[X]$
$\forall\, S : \mathbb{P}(\mathbb{P}\, X) \bullet x \in S \Rightarrow x \in \mathbb{P}(\bigcup S)$

theorem rule bigcupInPower $[X]$
$\forall\, S : \mathbb{P}(\mathbb{P}\, X) \bullet \bigcup S \in \mathbb{P}\, T \Leftrightarrow S \in \mathbb{P}(\mathbb{P}\, T)$

theorem disabled rule bigcupSubsetBigcup $[X]$
$\forall\, S, T : \mathbb{P}(\mathbb{P}\, X) \mid S \subseteq T \bullet \bigcup S \subseteq \bigcup T$

theorem rule bigcapEmpty $[X]$
$\bigcap[X]\{\} = X$

theorem rule bigcapUnit $[X]$
$S \in \mathbb{P}\, X \Rightarrow \bigcap\{S\} = S$

theorem rule bigcapUnion $[X]$
$\forall\, S, T : \mathbb{P}(\mathbb{P}\, X) \bullet \bigcap(S \cup T) = (\bigcap S) \cap (\bigcap T)$

theorem rule bigcapInPower $[X]$
$\forall\, S : \mathbb{P}(\mathbb{P}\, X) \bullet x \in S \Rightarrow \bigcap S \in \mathbb{P}\, x$

theorem disabled rule inPowerBigcap $[X]$
$\forall\, T : \mathbb{P}(\mathbb{P}\, X) \bullet S \in \mathbb{P}(\bigcap T) \Leftrightarrow (\forall\, U : T \bullet S \in \mathbb{P}\, U)$

theorem disabled rule bigcapSubsetBigcap $[X]$
$\forall\, S, T : \mathbb{P}(\mathbb{P}\, X) \mid T \subseteq S \bullet \bigcap S \subseteq \bigcap T$

9 Ordered pairs

The definitions of *first* and *second* are here for compatibility with the original toolkit. It is usually more convenient to use the numeric projection functions (i.e., write $p.1$ instead of $\mathit{first}\ p$). This is better in proofs because there are no generic actuals needed.

Definitions

$[X, Y]$

$\mathit{first} : X \times Y \rightarrow X$
$\mathit{second} : X \times Y \rightarrow Y$

$\langle\!\langle$ rule firstDefinition $\rangle\!\rangle$
$\forall x : X;\ y : Y \bullet \mathit{first}(x, y) = x$

$\langle\!\langle$ rule secondDefinition $\rangle\!\rangle$
$\forall x : X;\ y : Y \bullet \mathit{second}(x, y) = y$

$\langle\!\langle$ pairComposition $\rangle\!\rangle$
$\forall p : X \times Y \bullet p = (\mathit{first}\ p, \mathit{second}\ p)$

Theorems

theorem rule firstIsDot1 $[X, Y]$
$\forall p : X \times Y \bullet \mathit{first}[X, Y]p = p.1$

theorem rule secondIsDot2 $[X, Y]$
$\forall p : X \times Y \bullet \mathit{second}[X, Y]p = p.2$

10 Relations

10.1 Relation space

The function $\_ \leftrightarrow \_$ is predefined by the equation $X \leftrightarrow Y = \mathbb{P}(X \times Y)$.

Theorems

theorem grule relDefinition $[X, Y]$
$X \leftrightarrow Y = \mathbb{P}(X \times Y)$

theorem rule nullInRel
$\{\} \in X \leftrightarrow Y$

theorem rule unitInRel
$\{p\} \in X \leftrightarrow Y \Leftrightarrow p \in X \times Y$

theorem rule cupInRel $[X, Y]$
$\forall Q, R : \mathbb{P}(X \times Y) \bullet$
$Q \cup R \in A \leftrightarrow B \Leftrightarrow Q \in A \leftrightarrow B \land R \in A \leftrightarrow B$

theorem subsetOfRelIsRel $[X, Y]$
$R \in X \leftrightarrow Y \land S \subseteq R \Rightarrow S \in X \leftrightarrow Y$

theorem rule crossIsRel $[X, Y]$
$\forall A : \mathbb{P}\, X;\; B : \mathbb{P}\, Y \bullet A \times B \in X \leftrightarrow Y$

theorem rule relEqualNull
$\lnot\, X \leftrightarrow Y = \{\}$

theorem relationExtensionality $[X, Y]$
$\forall Q, R : X \leftrightarrow Y \bullet Q = R \Leftrightarrow (\forall x : X;\; y : Y \bullet x \,\underline{R}\, y \Leftrightarrow x \,\underline{Q}\, y)$

Automation

theorem rule rel_type $[X, Y]$
$\mathbb{P}(X \times Y) \in Z \Rightarrow X \leftrightarrow Y \in Z$

theorem rule rel_ideal $[X, Y]$
$\mathbb{P}\, S \in \mathbb{P}(X \leftrightarrow Y) \Leftrightarrow S \in X \leftrightarrow Y$

theorem rule rel_sub $[X, Y]$
$\forall A : \mathbb{P}\, X;\; B : \mathbb{P}\, Y \bullet A \leftrightarrow B \in \mathbb{P}(X \leftrightarrow Y)$

10.2 Maplets

Maplets provide an alternative notation for ordered pairs. They are usually used in defining functions or relations. The defining axiom is phrased as a rewrite rule to eliminate maplets in favour of pairs.

Definitions

syntax $\mapsto$ *infun*1 `\mapsto`

$[X, Y]$

$\_ \mapsto \_ : X \times Y \to X \times Y$

$\langle\!\langle$ rule mapDef $\rangle\!\rangle$
$\forall x : X;\ y : Y \bullet x \mapsto y = (x, y)$

10.3 Domain and range

syntax dom *word* `\dom`
syntax ran *word* `\ran`

Definitions

$[X, Y]$

$\mathrm{dom} : (X \leftrightarrow Y) \rightarrow \mathbb{P}\, X$
$\mathrm{ran} : (X \leftrightarrow Y) \rightarrow \mathbb{P}\, Y$

$\langle\!\langle$ disabled rule domDefinition $\rangle\!\rangle$
$\forall R : X \leftrightarrow Y \bullet \mathrm{dom}\, R = \{x : X;\ y : Y \mid (x, y) \in R \bullet x\}$

$\langle\!\langle$ disabled rule ranDefinition $\rangle\!\rangle$
$\forall R : X \leftrightarrow Y \bullet \mathrm{ran}\, R = \{x : X;\ y : Y \mid (x, y) \in R \bullet y\}$

Theorems

theorem disabled rule inDom $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet x \in \mathrm{dom}\ R \Leftrightarrow (\exists y : Y \bullet (x, y) \in R)$

theorem memberFirstInDom $[X, Y]$
$\forall R : X \leftrightarrow Y \mid (x, y) \in R \bullet x \in \mathrm{dom}\, R$

theorem rule domEmpty $[X, Y]$
$\mathrm{dom}[X, Y]\{\} = \{\}$

theorem rule domSingleton $[X, Y]$
$\forall p : X \times Y \bullet \mathrm{dom}\{p\} = \{p.1\}$

theorem rule domCup $[X, Y]$
$\forall Q, R : X \leftrightarrow Y \bullet \mathrm{dom}(Q \cup R) = (\mathrm{dom}\, Q) \cup (\mathrm{dom}\, R)$

theorem rule domCross $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \mid \neg\ B = \{\} \bullet \mathrm{dom}(A \times B) = A$

theorem disabled rule domSubset $[X, Y]$
$\forall S : X \leftrightarrow Y \bullet \forall R : \mathbb{P}\, S \bullet \mathrm{dom}\, R \in \mathbb{P}(\mathrm{dom}\, S)$

theorem disabled rule inRan $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet y \in \mathrm{ran}\ R \Leftrightarrow (\exists x : X \bullet (x, y) \in R)$

theorem memberSecondInRan $[X, Y]$
$\forall R : X \leftrightarrow Y \mid (x, y) \in R \bullet y \in \mathrm{ran}\, R$

theorem disabled rule inRanFunction $[X, Y]$
$\forall f : X \nrightarrow Y \bullet y \in \mathrm{ran}\ f \Leftrightarrow (\exists x : \mathrm{dom}\, f \bullet y = f(x))$

theorem rule ranEmpty $[X, Y]$
$\mathrm{ran}[X, Y]\{\} = \{\}$

theorem rule ranSingleton $[X, Y]$
$\forall p : X \times Y \bullet \mathrm{ran}\{p\} = \{p.2\}$

theorem rule ranCup $[X, Y]$
$\forall Q, R : X \leftrightarrow Y \bullet \mathrm{ran}(Q \cup R) = (\mathrm{ran}\, Q) \cup (\mathrm{ran}\, R)$

theorem rule ranCross $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \mid \neg\, A = \{\} \bullet \mathrm{ran}(A \times B) = B$

theorem disabled rule ranSubset $[X, Y]$
$\forall S : X \leftrightarrow Y \bullet \forall R : \mathbb{P}\, S \bullet \mathrm{ran}\, R \in \mathbb{P}(\mathrm{ran}\, S)$

Automation

theorem rule domInPower $[X, Y]$
$S \in \mathbb{P}\, X \wedge R \in S \leftrightarrow Y \Rightarrow \mathrm{dom}[X, Y]R \in \mathbb{P}\, S$

theorem rule ranInPower $[X, Y]$
$S \in \mathbb{P}\, Y \wedge R \in X \leftrightarrow S \Rightarrow \mathrm{ran}[X, Y]R \in \mathbb{P}\, S$

10.4 Identity relation

Definitions

syntax id *pregen* `\id`

$$\mathrm{id}\, X == \{ x : X \bullet x \mapsto x \}$$

Theorems

theorem disabled rule inId $[X]$
$p \in \mathrm{id}\, X \Leftrightarrow p \in X \times X \land p.1 = p.2$

theorem rule pairInId $[X]$
$(x, y) \in \mathrm{id}\, X \Leftrightarrow x = y \land x \in X$

theorem rule applyId $[X]$
$\forall x : X \bullet (\mathrm{id}\, X)(x) = x$

theorem rule domId $[X]$
$\forall S : \mathbb{P}\, X \bullet \mathrm{dom}(\mathrm{id}\, S) = S$

theorem rule ranId $[X]$
$\forall S : \mathbb{P}\, X \bullet \mathrm{ran}(\mathrm{id}\, S) = S$

theorem rule idNull
$\mathrm{id}\{\} = \{\}$

theorem rule idUnit
$\mathrm{id}\{x\} = \{(x, x)\}$

theorem rule idCup $[X]$
$\forall A, B : \mathbb{P}\, X \bullet \mathrm{id}(A \cup B) = \mathrm{id}\, A \cup \mathrm{id}\, B$

theorem rule idSubsetId
$\mathrm{id}\, X \in \mathbb{P}(\mathrm{id}\, Y) \Leftrightarrow X \in \mathbb{P}\, Y$

Automation

The next four rules could be replaced by a *_type* rule (see Section 2.1). Better, though, would be to use $X \rightarrowtail\!\!\!\!\twoheadrightarrow X$ as the declared set, if bijections were declared yet.

theorem rule idType $[X]$
$\mathrm{id}\, X \in \mathbb{P}(A \times B) \Leftrightarrow X \in \mathbb{P}\, A \land X \in \mathbb{P}\, B$

theorem rule idInRel $[X]$
$\mathrm{id}\, X \in A \leftrightarrow B \Leftrightarrow X \in \mathbb{P}\, A \land X \in \mathbb{P}\, B$

theorem rule idInPfun $[X]$
$\mathrm{id}\, X \in (A \nrightarrow B) \Leftrightarrow X \in \mathbb{P}\, A \land X \in \mathbb{P}\, B$

theorem rule idInFun $[X]$
$\mathrm{id}\, X \in (A \rightarrow B) \Leftrightarrow X = A \land X \in \mathbb{P}\, B$

10.5 Composition

Two composition operators are defined; they are identical except for the order of their arguments. Rather than have two sets of rules, one for each composition operator, we use rule *circDef* to replace $g \circ f$ by $f \mathbin{⨾} g$.

Definitions

syntax $⨾$ *infun*4 `\comp`
syntax $\circ$ *infun*4 `\circ`

$$[X, Y, Z]$$
$$\_ ⨾ \_ : (X \leftrightarrow Y) \times (Y \leftrightarrow Z) \rightarrow (X \leftrightarrow Z)$$
$$\_ \circ \_ : (Y \leftrightarrow Z) \times (X \leftrightarrow Y) \rightarrow (X \leftrightarrow Z)$$

$\langle\!\langle$ disabled rule compDef $\rangle\!\rangle$
$\forall Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z \bullet$
$\quad Q ⨾ R = \{ x : X;\ y : Y;\ z : Z \mid x \underline{Q} y \underline{R} z \bullet (x, z) \}$

$\langle\!\langle$ rule circDef $\rangle\!\rangle$
$\forall Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z \bullet$
$\quad R \circ Q = Q ⨾ R$

Theorems

theorem rule pairInComp $[X, Y, Z]$
$\forall Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z \bullet (x, z) \in Q ⨾ R \Leftrightarrow (\exists y : Y \bullet x \underline{Q} y \underline{R} z)$

theorem rule compAssociates $[W, X, Y, Z]$
$\forall P : W \leftrightarrow X;\ Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z \bullet (P ⨾ Q) ⨾ R = P ⨾ (Q ⨾ R)$

theorem rule nullComp $[X, Y, Z]$
$\forall R : Y \leftrightarrow Z \bullet \{\} ⨾ [X, Y, Z] R = \{\}$

theorem rule compNull $[X, Y, Z]$
$\forall R : X \leftrightarrow Y \bullet R ⨾ [X, Y, Z] \{\} = \{\}$

The domain of a composition $Q ⨾ R$ is $Q^{\sim}(\!|\ \mathrm{dom}\, R\ |\!)$, but this fact cannot be legally stated this early in the Toolkit. A similar fact (and problem) applies to the range of a composition. See Section 10.9, where these theorems appear.

theorem domCompSmaller $[X, Y, Z]$
$\forall Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z \bullet \mathrm{dom}(Q ⨾ R) \subseteq \mathrm{dom}\, Q$

theorem disabled rule easyDomComp $[X, Y, Z]$
$\forall Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z \mid \mathrm{ran}\, Q \in \mathbb{P}(\mathrm{dom}\, R) \bullet \mathrm{dom}(Q ⨾ R) = \mathrm{dom}\, Q$

theorem ranCompSmaller $[X, Y, Z]$
$\forall Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z \bullet \mathrm{ran}(Q ⨾ R) \subseteq \mathrm{ran}\, R$

theorem disabled rule easyRanComp $[X, Y, Z]$
$\forall Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z \mid \mathrm{dom}\, R \in \mathbb{P}(\mathrm{ran}\, Q) \bullet \mathrm{ran}(Q \mathbin{⨾} R) = \mathrm{ran}\, R$

theorem rule applyComp $[X, Y, Z]$
$f \in X \mathbin{⇸} Y \land g \in Y \mathbin{⇸} Z \land x \in \mathrm{dom}\, f \land f(x) \in \mathrm{dom}\, g \Rightarrow (f \mathbin{⨾} g)(x) = g(f(x))$

It is hard to make a stronger rule than the following, because a composition may apply to a value in cases where its first member does not. For example, if $f = \mathbb{Z} \times \mathbb{Z}$ and $g = \{0 \mapsto 0\}$, then f does not apply to anything, while $f \mathbin{⨾} g$ is a function with domain $\mathbb{Z}$.

theorem rule compAppliesTo $[X, Y, Z]$
$\forall f : X \leftrightarrow Y;\ g : Y \leftrightarrow Z \bullet f\ \mathit{applies\$to}\ x \Rightarrow ((f \mathbin{⨾} g)\ \mathit{applies\$to}\ x \Leftrightarrow g\ \mathit{applies\$to}\ f(x))$

theorem disabled rule compMonotone $[X, Y, Z]$
$\forall Q, Q' : X \leftrightarrow Y;\ R, R' : Y \leftrightarrow Z \mid Q' \subseteq Q \land R' \subseteq R \bullet Q' \mathbin{⨾} R' \subseteq (Q \mathbin{⨾} R)$

theorem rule compInRel $[X, Y, Z]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Z \mid Q \in A \leftrightarrow Y \land R \in Y \leftrightarrow B \bullet Q \mathbin{⨾} R \in A \leftrightarrow B$

theorem rule compInPfun $[X, Y, Z]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Z \mid f \in A \mathbin{⇸} Y \land g \in Y \mathbin{⇸} B \bullet f \mathbin{⨾} g \in A \mathbin{⇸} B$

theorem rule compInFun $[X, Y, Z]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Z \mid f \in X \mathbin{⇸} Y \land g \in Y \mathbin{⇸} Z \bullet$
$f \mathbin{⨾} g \in A \rightarrow B \Leftrightarrow (\mathrm{dom}(f \mathbin{⨾} g) = A \land \mathrm{ran}(f \mathbin{⨾} g) \subseteq B)$

Automation

theorem rule applyCompKnownFunctions $[X, Y, Z]$
$\mathit{KnownMember}[A \rightarrow B][f/\mathit{element}] \land \mathit{KnownMember}[C \rightarrow D][g/\mathit{element}]$
$\land\, x \in A \land A \in \mathbb{P}\, X \land B \in \mathbb{P}\, C \land C \in \mathbb{P}\, Y \land D \in \mathbb{P}\, Z$
$\Rightarrow (f \mathbin{⨾} g)(x) = g(f(x))$

We could add *domCompResult* and *RanCompResult* here.

10.6 Domain and range restriction

Definitions

syntax $\lhd$ *infun*6 `\dres`
syntax $\rhd$ *infun*6 `\rres`

$[X, Y]$

$$\_ \lhd \_ : \mathbb{P}\, X \times (X \leftrightarrow Y) \rightarrow (X \leftrightarrow Y)$$
$$\_ \rhd \_ : (X \leftrightarrow Y) \times \mathbb{P}\, Y \rightarrow (X \leftrightarrow Y)$$

$\langle\!\langle$ disabled rule dresDef $\rangle\!\rangle$
$$\forall S : \mathbb{P}\, X;\ R : X \leftrightarrow Y \bullet S \lhd R = \{ p : R \mid p.1 \in S \}$$

$\langle\!\langle$ disabled rule rresDef $\rangle\!\rangle$
$$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet R \rhd S = \{ p : R \mid p.2 \in S \}$$

Theorems

theorem rule inDres $[X, Y]$
$$\forall S : \mathbb{P}\, X;\ R : X \leftrightarrow Y \bullet x \in S \lhd R \Leftrightarrow x \in R \land x.1 \in S$$

theorem rule inRres $[X, Y]$
$$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet x \in R \rhd S \Leftrightarrow x \in R \land x.2 \in S$$

theorem rule domDres $[X, Y]$
$$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, X \bullet \mathrm{dom}(S \lhd R) = S \cap \mathrm{dom}\, R$$

theorem rule ranRres $[X, Y]$
$$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet \mathrm{ran}(R \rhd S) = (\mathrm{ran}\, R) \cap S$$

theorem dresIsSubset $[X, Y]$
$$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, X \bullet S \lhd R \subseteq R$$

theorem rresIsSubset $[X, Y]$
$$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet R \rhd S \subseteq R$$

theorem rule compIdLeft $[X, Y]$
$$\forall S : \mathbb{P}\, X;\ R : X \leftrightarrow Y \bullet (\mathrm{id}\, S) \mathbin{;} R = S \lhd R$$

theorem rule compIdRight $[X, Y]$
$$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet R \mathbin{;} (\mathrm{id}\, S) = R \rhd S$$

theorem rule dresId $[X]$
$$\forall S, T : \mathbb{P}\, X \bullet S \lhd (\mathrm{id}\, T) = \mathrm{id}(S \cap T)$$

theorem rule rresId $[X]$
$$\forall S, T : \mathbb{P}\, X \bullet (\mathrm{id}\, S) \rhd T = \mathrm{id}(S \cap T)$$

theorem rule dresDres $[X, Y]$
$\forall S, T : \mathbb{P}\,X;\; R : X \leftrightarrow Y \bullet S \lhd (T \lhd R) = (S \cap T) \lhd R$

theorem rule rresRres $[X, Y]$
$\forall S, T : \mathbb{P}\,Y;\; R : X \leftrightarrow Y \bullet (R \rhd S) \rhd T = R \rhd (S \cap T)$

We should normalize $S \lhd R \rhd T$, as the order of association does not matter.

theorem disabled rule dresEverything $[X, Y]$
$\forall R : X \leftrightarrow Y;\; S : \mathbb{P}\,X \bullet S \cap \operatorname{dom} R = \{\} \Rightarrow S \lhd R = \{\}$

theorem disabled rule rresEverything $[X, Y]$
$\forall R : X \leftrightarrow Y;\; S : \mathbb{P}\,Y \bullet S \cap \operatorname{ran} R = \{\} \Rightarrow R \rhd S = \{\}$

theorem rule nullDres $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet \{\} \lhd R = \{\}$

theorem rule rresNull $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet R \rhd \{\} = \{\}$

theorem rule dresNull $[X, Y]$
$\forall S : \mathbb{P}\,X \bullet S \lhd [X, Y]\{\} = \{\}$

theorem rule nullRres $[X, Y]$
$\forall S : \mathbb{P}\,Y \bullet \{\} \rhd [X, Y]S = \{\}$

theorem disabled rule dresElimination $[X, Y]$
$\forall R : X \leftrightarrow Y;\; S : \mathbb{P}\,X \bullet \operatorname{dom} R \in \mathbb{P}\,S \Rightarrow S \lhd R = R$

theorem disabled rule rresElimination $[X, Y]$
$\forall R : X \leftrightarrow Y;\; S : \mathbb{P}\,Y \bullet \operatorname{ran} R \in \mathbb{P}\,S \Rightarrow R \rhd S = R$

theorem rule dresUnit $[X, Y]$
$\forall x : X;\; y : Y;\; S : \mathbb{P}\,X \bullet S \lhd \{(x, y)\} = \textbf{if } x \in S \textbf{ then } \{(x, y)\} \textbf{ else } \{\}$

theorem rule unitRres $[X, Y]$
$\forall x : X;\; y : Y;\; S : \mathbb{P}\,Y \bullet \{(x, y)\} \rhd S = \textbf{if } y \in S \textbf{ then } \{(x, y)\} \textbf{ else } \{\}$

theorem rule unitDres $[X, Y]$
$\forall R : X \leftrightarrow Y;\; x : X \bullet (\neg\, x \in \operatorname{dom} R) \Rightarrow \{x\} \lhd R = \{\}$

theorem rule rresUnit $[X, Y]$
$\forall R : X \leftrightarrow Y;\; y : Y \bullet (\neg\, y \in \operatorname{ran} R) \Rightarrow R \rhd \{y\} = \{\}$

theorem rule dresCup $[X, Y]$
$\forall S : \mathbb{P}\, X;\ Q, R : X \leftrightarrow Y \bullet S \lhd (Q \cup R) = (S \lhd Q) \cup (S \lhd R)$

theorem rule rresCup $[X, Y]$
$\forall Q, R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet (Q \cup R) \rhd S = (Q \rhd S) \cup (R \rhd S)$

There should be theorems about restricting compositions.

theorem rule applyDres $[X, Y]$
$\forall f : X \pfun Y;\ S : \mathbb{P}\, X \bullet x \in S \land x \in \operatorname{dom} f \Rightarrow (S \lhd f)(x) = f(x)$

theorem rule applyRres $[X, Y]$
$\forall f : X \pfun Y;\ S : \mathbb{P}\, Y \bullet x \in \operatorname{dom} f \land f(x) \in S \Rightarrow (f \rhd S)(x) = f(x)$

Automation

theorem rule dres_result $[X, Y]$
$\forall S : \mathbb{P}\, X;\ R : X \leftrightarrow Y \bullet \mathbb{P}\, R \in \mathbb{P}\, Z \Rightarrow S \lhd R \in Z$

theorem rule rres_result $[X, Y]$
$\forall S : \mathbb{P}\, Y;\ R : X \leftrightarrow Y \bullet \mathbb{P}\, R \in \mathbb{P}\, Z \Rightarrow R \rhd S \in Z$

10.7 Domain and range anti-restriction

Definitions

syntax $\ndres$ *infun*6 `\ndres`
syntax $\nrres$ *infun*6 `\nrres`

$[X, Y]$

$\_ \ndres \_ : \mathbb{P}\, X \times (X \leftrightarrow Y) \rightarrow (X \leftrightarrow Y)$
$\_ \nrres \_ : (X \leftrightarrow Y) \times \mathbb{P}\, Y \rightarrow (X \leftrightarrow Y)$

$\langle\!\langle$ disabled rule ndresDef $\rangle\!\rangle$
$\forall S : \mathbb{P}\, X;\ R : X \leftrightarrow Y \bullet S \ndres R = \{ p : R \mid \neg\, p.1 \in S \}$

$\langle\!\langle$ disabled rule nrresDef $\rangle\!\rangle$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet R \nrres S = \{ p : R \mid \neg\, p.2 \in S \}$

Theorems

theorem rule inNdres $[X, Y]$
$\forall S : \mathbb{P}\, X;\ R : X \leftrightarrow Y \bullet x \in S \ndres R \Leftrightarrow x \in R \land \neg\, x.1 \in S$

theorem rule inNrres $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet x \in R \nrres S \Leftrightarrow x \in R \land \neg\, x.2 \in S$

theorem rule domNdres $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, X \bullet \mathrm{dom}(S \ndres R) = (\mathrm{dom}\, R) \setminus S$

theorem rule ranNrres $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet \mathrm{ran}(R \nrres S) = (\mathrm{ran}\, R) \setminus S$

theorem ndresIsSubset $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, X \bullet S \ndres R \subseteq R$

theorem nrresIsSubset $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet R \nrres S \subseteq R$

theorem rule ndresId $[X]$
$\forall S, T : \mathbb{P}\, X \bullet S \ndres (\mathrm{id}\, T) = \mathrm{id}(T \setminus S)$

theorem rule nrresId $[X]$
$\forall S, T : \mathbb{P}\, X \bullet (\mathrm{id}\, S) \nrres T = \mathrm{id}(S \setminus T)$

theorem rule ndresNdres $[X, Y]$
$\forall S, T : \mathbb{P}\, X;\ R : X \leftrightarrow Y \bullet S \ndres (T \ndres R) = (S \cup T) \ndres R$

theorem rule nrresNrres $[X, Y]$
$\forall S, T : \mathbb{P}\, Y;\ R : X \leftrightarrow Y \bullet (R \nrres S) \nrres T = R \nrres (S \cup T)$

More similar rules are possible, for various combinations of $\lhd$, $\ndres$, $\rhd$, and $\nrres$.

theorem disabled rule ndresNothing $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, X \bullet S \cap \operatorname{dom} R = \{\} \Rightarrow S \ndres R = R$

theorem disabled rule nrresNothing $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet S \cap \operatorname{ran} R = \{\} \Rightarrow R \nrres S = R$

theorem rule nullNdres $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet \{\} \ndres R = R$

theorem rule nrresNull $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet R \nrres \{\} = R$

theorem disabled rule ndresEverything $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, X \bullet \operatorname{dom} R \in \mathbb{P}\, S \Rightarrow S \ndres R = \{\}$

theorem disabled rule nrresEverything $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet \operatorname{ran} R \in \mathbb{P}\, S \Rightarrow R \nrres S = \{\}$

theorem rule ndresNull $[X, Y]$
$\forall S : \mathbb{P}\, X \bullet S \ndres [X, Y]\{\} = \{\}$

theorem rule nullNrres $[X, Y]$
$\forall S : \mathbb{P}\, Y \bullet \{\} \nrres [X, Y] S = \{\}$

theorem rule ndresUnit $[X, Y]$
$\forall x : X;\ y : Y;\ S : \mathbb{P}\, X \bullet S \ndres \{(x, y)\} = \mathbf{if}\ x \in S\ \mathbf{then}\ \{\}\ \mathbf{else}\ \{(x, y)\}$

theorem rule nrresUnit $[X, Y]$
$\forall x : X;\ y : Y;\ S : \mathbb{P}\, Y \bullet \{(x, y)\} \nrres S = \mathbf{if}\ y \in S\ \mathbf{then}\ \{\}\ \mathbf{else}\ \{(x, y)\}$

theorem rule ndresCup $[X, Y]$
$\forall S : \mathbb{P}\, X;\ Q, R : X \leftrightarrow Y \bullet S \ndres (Q \cup R) = (S \ndres Q) \cup (S \ndres R)$

theorem rule nrresCup $[X, Y]$
$\forall Q, R : X \leftrightarrow Y;\ S : \mathbb{P}\, Y \bullet (Q \cup R) \nrres S = (Q \nrres S) \cup (R \nrres S)$

There should be theorems about anti-restricting compositions.

theorem rule applyNdres $[X, Y]$
$\forall f : X \pfun Y;\ S : \mathbb{P}\, X \bullet \neg\, x \in S \land x \in \operatorname{dom} f \Rightarrow (S \ndres f)(x) = f(x)$

theorem rule applyNrres $[X, Y]$
$\forall f : X \pfun Y;\ S : \mathbb{P}\, Y \bullet x \in \operatorname{dom} f \land \neg\, f(x) \in S \Rightarrow (f \nrres S)(x) = f(x)$

Automation

theorem rule ndres_result $[X, Y]$
$\forall S : \mathbb{P}\, X;\ R : X \leftrightarrow Y \bullet \mathbb{P}\, R \in \mathbb{P}\, Z \Rightarrow S \ndres R \in Z$

theorem rule nrres_result $[X, Y]$
$\forall S : \mathbb{P}\, Y;\ R : X \leftrightarrow Y \bullet \mathbb{P}\, R \in \mathbb{P}\, Z \Rightarrow R \nrres S \in Z$

10.8 Relational inversion

Definitions

syntax $\sim$ *postfun* `\inv`

$[X, Y]$

$\_^{\sim} : (X \leftrightarrow Y) \rightarrow (Y \leftrightarrow X)$

$\langle\!\langle$ disabled rule invDef $\rangle\!\rangle$
$\forall R : X \leftrightarrow Y \bullet R^{\sim} = \{p : R \bullet (p.2, p.1)\}$

Theorems

theorem rule invInPowerCross $[X, Y]$
$\forall A : \mathbb{P}\, Y;\; B : \mathbb{P}\, X;\; R : X \leftrightarrow Y \bullet R^{\sim} \in \mathbb{P}(A \times B) \Leftrightarrow R \in \mathbb{P}(B \times A)$

theorem rule invInRel $[X, Y]$
$\forall A : \mathbb{P}\, Y;\; B : \mathbb{P}\, X;\; R : X \leftrightarrow Y \bullet R^{\sim} \in A \leftrightarrow B \Leftrightarrow R \in B \leftrightarrow A$

theorem rule pairInInv $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet (x, y) \in R^{\sim} \Leftrightarrow (y, x) \in R$

theorem disabled rule inversesEqual $[X, Y]$
$\forall Q, R : X \leftrightarrow Y \bullet Q^{\sim} = R^{\sim} \Leftrightarrow Q = R$

theorem disabled rule inversesSubseteq $[X, Y]$
$\forall Q, R : X \leftrightarrow Y \bullet Q^{\sim} \subseteq R^{\sim} \Leftrightarrow Q \subseteq R$

theorem rule invCross $[X, Y]$
$\forall A : \mathbb{P}\, X;\; B : \mathbb{P}\, Y \bullet (A \times B)^{\sim} = B \times A$

theorem rule invEmpty $[X, Y]$
$(\_^{\sim})[X, Y]\{\} = \{\}$

theorem rule invUnit $[X, Y]$
$\forall x : X;\; y : Y \bullet \{(x, y)\}^{\sim} = \{(y, x)\}$

theorem rule invCup $[X, Y]$
$\forall Q, R : X \leftrightarrow Y \bullet (Q \cup R)^{\sim} = (Q^{\sim}) \cup (R^{\sim})$

theorem rule invCap $[X, Y]$
$\forall Q, R : X \leftrightarrow Y \bullet (Q \cap R)^{\sim} = (Q^{\sim}) \cap (R^{\sim})$

theorem rule invSetminus $[X, Y]$
$\forall Q, R : X \leftrightarrow Y \bullet (Q \setminus R)^{\sim} = (Q^{\sim}) \setminus (R^{\sim})$

theorem rule invInv $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet (R^{\sim})^{\sim} = R$

theorem rule invComp $[X, Y, Z]$
$\forall Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z \bullet (Q \mathbin{;} R)^{\sim} = (R^{\sim}) \mathbin{;} (Q^{\sim})$

theorem rule invId $[X]$
$\forall S : \mathbb{P}\, X \bullet (\mathrm{id}\, S)^{\sim} = \mathrm{id}\, S$

theorem rule invDres $[X, Y]$
$\forall S : \mathbb{P}\, X;\ R : X \leftrightarrow Y \bullet (S \lhd R)^{\sim} = (R^{\sim}) \rhd S$

theorem rule invRres $[X, Y]$
$\forall S : \mathbb{P}\, Y;\ R : X \leftrightarrow Y \bullet (R \rhd S)^{\sim} = S \lhd (R^{\sim})$

theorem rule invNdres $[X, Y]$
$\forall S : \mathbb{P}\, X;\ R : X \leftrightarrow Y \bullet (S \ntriangleleft R)^{\sim} = (R^{\sim}) \ntriangleright S$

theorem rule invNrres $[X, Y]$
$\forall S : \mathbb{P}\, Y;\ R : X \leftrightarrow Y \bullet (R \ntriangleright S)^{\sim} = S \ntriangleleft (R^{\sim})$

theorem rule domInv $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet \mathrm{dom}(R^{\sim}) = \mathrm{ran}\, R$

theorem rule ranInv $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet \mathrm{ran}(R^{\sim}) = \mathrm{dom}\, R$

Rules about applying inverses appear in Section 11.3.

10.9 Relational image

Definitions

$[X, Y]$

$\_(\!|\ \_\ |\!) : (X \leftrightarrow Y) \times \mathbb{P}\, X \rightarrow \mathbb{P}\, Y$

$\langle\!\langle$ disabled rule imageDef $\rangle\!\rangle$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, X \bullet R(\!|\ S\ |\!) = \mathrm{ran}(S \lhd R)$

Theorems

theorem disabled rule inImage $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, X \bullet y \in R(\!|\ S\ |\!) \Leftrightarrow (\exists x : S \bullet x\ \underline{R}\ y)$

theorem imageSubsetRange $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P}\, X \bullet R(\!|\ S\ |\!) \subseteq \mathrm{ran}\, R$

theorem disabled rule imageMonotonic $[X, Y]$
$\forall Q, R : X \leftrightarrow Y;\ S, T : \mathbb{P}\, X \mid S \subseteq T \land Q \subseteq R \bullet Q(\!|\ S\ |\!) \in \mathbb{P}(R(\!|\ T\ |\!))$

theorem imageMonotonic1 $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S, T : \mathbb{P}\, X \mid S \subseteq T \bullet R(\!|\ S\ |\!) \subseteq R(\!|\ T\ |\!)$

theorem imageMonotonic2 $[X, Y]$
$\forall Q, R : X \leftrightarrow Y;\ S : \mathbb{P}\, X \mid Q \subseteq R \bullet Q(\!|\ S\ |\!) \subseteq R(\!|\ S\ |\!)$

theorem rule imageNull $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet R(\!|\ \{\}\ |\!) = \{\}$

theorem rule nullImage $[X, Y]$
$\forall S : \mathbb{P}\, X \bullet (\_(\!|\ \_\ |\!))[X, Y](\{\}, S) = \{\}$

The following rule should perhaps be disabled, as it can lead to ugly formulas (e.g., $R(\!|\ \{1, 2\}\ |\!)$ will be greatly expanded). On the other hand, it is needed for calculating functional images (e.g., $succ(\!|\ \{1, 2\}\ |\!)$).

theorem rule imageCup $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S, T : \mathbb{P}\, X \bullet R(\!|\ S \cup T\ |\!) = R(\!|\ S\ |\!) \cup R(\!|\ T\ |\!)$

theorem rule crossImage $[X, Y]$
$\forall A, S : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet (A \times B)(\!|\ S\ |\!) = \mathbf{if}\ A \cap S = \{\}\ \mathbf{then}\ \{\}\ \mathbf{else}\ B$

theorem disabled rule fullImage $[X, Y]$
$\forall R : X \leftrightarrow Y \mid \mathrm{dom}\, R \in \mathbb{P}\, S \bullet R(\!|\ S\ |\!) = \mathrm{ran}\, R$

theorem rule firstImage $[X, Y]$
$\forall S : X \leftrightarrow Y \bullet \mathit{first}(\!|\ S\ |\!) = \mathrm{dom}\, S$

theorem rule secondImage $[X, Y]$
$\forall S : X \leftrightarrow Y \bullet \mathit{second}(\!| S |\!) = \operatorname{ran} S$

theorem rule idImage $[X]$
$\forall S, T : \mathbb{P} X \bullet (\operatorname{id} S)(\!| T |\!) = S \cap T$

theorem rule imageDres $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S, T : \mathbb{P} X \bullet (S \lhd R)(\!| T |\!) = R(\!| S \cap T |\!)$

theorem rule imageRres $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P} Y;\ T : \mathbb{P} T \bullet (R \rhd S)(\!| T |\!) = R(\!| T |\!) \cap S$

theorem rule imageNdres $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S, T : \mathbb{P} X \bullet (S \ntriangleleft R)(\!| T |\!) = R(\!| T \setminus S |\!)$

theorem rule imageNrres $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P} Y;\ T : \mathbb{P} T \bullet (R \ntriangleright S)(\!| T |\!) = R(\!| T |\!) \setminus S$

theorem rule imageComp $[X, Y, Z]$
$\forall Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z;\ S : \mathbb{P} X \bullet (Q \mathbin{;} R)(\!| S |\!) = R(\!| Q(\!| S |\!) |\!)$

theorem rule inImageInv $[X, Y]$
$\forall f : Y \rightarrowtail X;\ S : \mathbb{P} X \bullet x \in f^{\sim}(\!| S |\!) \Leftrightarrow x \in \operatorname{dom} f \land f(x) \in S$

theorem disabled rule domComp $[X, Y, Z]$
$\forall Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z \bullet \operatorname{dom}(Q \mathbin{;} R) = Q^{\sim}(\!| \operatorname{dom} R |\!)$

theorem disabled rule ranComp $[X, Y, Z]$
$\forall Q : X \leftrightarrow Y;\ R : Y \leftrightarrow Z \bullet \operatorname{ran}(Q \mathbin{;} R) = R(\!| \operatorname{ran} Q |\!)$

theorem rule applicationInImage $[X, Y]$
$\forall f : X \rightarrowtail Y;\ S : \mathbb{P} X \bullet \forall x : S \mid x \in \operatorname{dom} f \bullet f(x) \in f(\!| S |\!)$

Automation

The following rules allow simple relational images to be calculated, e.g., $\mathit{succ}(\!| \{1, 2, 3\} |\!)$. The first rule is disabled because of the if-form it introduces.

theorem disabled rule functionImageUnit $[X, Y]$
$\forall f : X \rightarrowtail Y \bullet f(\!| \{x\} |\!) = \textbf{if } x \in \operatorname{dom} f \textbf{ then } \{f(x)\} \textbf{ else } \{\}$

theorem rule functionImageUnitOnDom $[X, Y]$
$\forall f : X \rightarrowtail Y \mid x \in \operatorname{dom} f \bullet f(\!| \{x\} |\!) = \{f(x)\}$

theorem rule functionImageUnitOffDom $[X, Y]$
$\forall f : X \rightarrowtail Y \mid \neg x \in \operatorname{dom} f \bullet f(\!| \{x\} |\!) = \{\}$

theorem rule image_result $[X, Y]$
$\forall R : X \leftrightarrow Y;\ S : \mathbb{P} X \bullet \mathbb{P}(\operatorname{ran} R) \in \mathbb{P} Z \Rightarrow R(\!| S |\!) \in Z$

10.10 Overriding

Definitions

syntax $\oplus$ *infun*5 `\oplus`

$[X, Y]$

$\_ \oplus \_ : (X \leftrightarrow Y) \times (X \leftrightarrow Y) \to (X \leftrightarrow Y)$

$\langle\!\langle$ disabled rule oplusDef $\rangle\!\rangle$
$\forall Q, R : X \leftrightarrow Y \bullet Q \oplus R = ((\mathrm{dom}\, R) \ntriangleleft Q) \cup R$

Theorems

theorem rule overrideInPowerCross $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet \forall Q, R : A \leftrightarrow B \bullet Q \oplus R \in \mathbb{P}(A \times B)$

theorem rule overrideInRel $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet \forall Q, R : A \leftrightarrow B \bullet Q \oplus R \in A \leftrightarrow B$

theorem rule overrideInPfun $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet \forall f, g : A \nrightarrow B \bullet f \oplus g \in A \nrightarrow B$

theorem rule overrideInFun $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet \forall f : A \to B;\ g : A \nrightarrow B \bullet f \oplus g \in A \to B$

theorem rule overrideAssociates $[X, Y]$
$\forall Q, R, S : X \leftrightarrow Y \bullet (Q \oplus R) \oplus S = Q \oplus (R \oplus S)$

theorem rule overrideWithNull $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet R \oplus \{\} = R$

The following theorem has as a special case $\{\} \oplus R = R$.

theorem disabled rule overrideEverything $[X, Y]$
$\forall Q, R : X \leftrightarrow Y \mid \mathrm{dom}\, Q \subseteq \mathrm{dom}\, R \bullet Q \oplus R = R$

theorem rule overrideNull $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet \{\} \oplus R = R$

theorem rule domOverride $[X, Y]$
$\forall Q, R : X \leftrightarrow Y \bullet \mathrm{dom}(Q \oplus R) = (\mathrm{dom}\, Q) \cup (\mathrm{dom}\, R)$

theorem rule overrideAppliesTo $[X, Y]$
$\forall f, g : X \leftrightarrow Y \bullet (f \oplus g)\ \mathit{applies\$to}\ x \Leftrightarrow g\ \mathit{applies\$to}\ x \lor (\neg\, x \in \mathrm{dom}\, g \land f\ \mathit{applies\$to}\ x)$

theorem disabled rule applyOverride $[X, Y]$
$\forall f, g : X \leftrightarrow Y;\ x : X \mid (f \oplus g)\ \mathit{applies\$to}\ x \bullet (f \oplus g)(x) = \mathbf{if}\ g\ \mathit{applies\$to}\ x\ \mathbf{then}\ g(x)\ \mathbf{else}\ f(x)$

theorem rule applyOverride1 $[X, Y]$
$\forall f, g : X \leftrightarrow Y;\ x : X \mid g\ \mathit{applies\$to}\ x \bullet (f \oplus g)(x) = g(x)$

theorem rule applyOverride2 $[X, Y]$
$\forall f, g : X \leftrightarrow Y;\ x : X \mid \neg\, x \in \mathrm{dom}\, g \land f\ \mathit{applies\$to}\ x \bullet (f \oplus g)(x) = f(x)$

10.11 Transitive closure

Definitions

syntax $^{+}$ *postfun* `\plus`
syntax * *postfun* `\star`

$[X]$

$\_^{+}, \_^{*} : (X \leftrightarrow X) \rightarrow (X \leftrightarrow X)$

$\langle\!\langle$ disabled rule plusDef $\rangle\!\rangle$
$\forall R : X \leftrightarrow X \bullet R^{+} = \bigcap \{ Q : X \leftrightarrow X \mid R \subseteq Q \land Q \mathbin{;} Q \subseteq Q \}$

$\langle\!\langle$ disabled rule starDef $\rangle\!\rangle$
$\forall R : X \leftrightarrow X \bullet R^{*} = \bigcap \{ Q : X \leftrightarrow X \mid \mathrm{id}\, X \subseteq Q \land R \subseteq Q \land Q \mathbin{;} Q \subseteq Q \}$

Theorems

The minimality of R^{+} can be expressed in three different ways. These are expressed as disabled rules, since the choice of which to apply, and when, must be made by the user.

theorem disabled rule plusSubset1 $[X]$
$\forall Q, R : X \leftrightarrow X \mid R \subseteq Q \land Q \mathbin{;} Q \subseteq Q \bullet R^{+} \in \mathbb{P}\, Q$

theorem disabled rule plusSubset2 $[X]$
$\forall Q, R : X \leftrightarrow X \mid R \subseteq Q \land R \mathbin{;} Q \subseteq Q \bullet R^{+} \in \mathbb{P}\, Q$

theorem disabled rule plusSubset3 $[X]$
$\forall Q, R : X \leftrightarrow X \mid R \subseteq Q \land Q \mathbin{;} R \subseteq Q \bullet R^{+} \in \mathbb{P}\, Q$

theorem plusContainsSelf $[X]$
$\forall R : X \leftrightarrow X \bullet R \subseteq R^{+}$

theorem plusIsTransitive $[X]$
$\forall R : X \leftrightarrow X \bullet R^{+} \mathbin{;} R^{+} \subseteq R^{+}$

theorem disabled rule plusOfTransitive $[X]$
$\forall R : X \leftrightarrow X \mid R \mathbin{;} R \subseteq R \bullet R^{+} = R$

The minimality of R^{*} can be expressed in two different ways, depending on which side we want to compose R and Q. As for the rules about R^{+}, these rules are disabled, and can be applied by the user.

theorem disabled rule starSubset1 $[X]$
$\forall Q, R : X \leftrightarrow X \mid \mathrm{id}\, X \subseteq Q \land R \mathbin{;} Q \subseteq Q \bullet R^{*} \in \mathbb{P}\, Q$

theorem disabled rule starSubset2 $[X]$
$\forall Q, R : X \leftrightarrow X \mid \mathrm{id}\, X \subseteq Q \land Q \mathbin{;} R \subseteq Q \bullet R^{*} \in \mathbb{P}\, Q$

An alernative defintion of R^{*} may be simper to use in some proofs:

theorem disabled rule starDef2 $[X]$
$\forall R : X \leftrightarrow X \bullet R^{*} = \mathrm{id}\, X \cup R^{+}$

theorem starContainsSelf $[X]$
$\forall R : X \leftrightarrow X \bullet R \subseteq R^*$

theorem rule starIsTransitive $[X]$
$\forall R : X \leftrightarrow X \bullet R^* \mathbin{;} R^* = R^*$

theorem rule domPlus $[X]$
$\forall R : X \leftrightarrow X \bullet \mathrm{dom}(R^+) = \mathrm{dom}\, R$

theorem rule ranPlus $[X]$
$\forall R : X \leftrightarrow X \bullet \mathrm{ran}(R^+) = \mathrm{ran}\, R$

theorem rule plusInRel $[X]$
$\forall A, B : \mathbb{P}\, X;\ R : X \leftrightarrow X \bullet R^+ \in A \leftrightarrow B \Leftrightarrow R \in A \leftrightarrow B$

theorem rule domStar $[X]$
$\forall R : X \leftrightarrow X \bullet \mathrm{dom}(R^*) = X$

theorem rule ranStar $[X]$
$\forall R : X \leftrightarrow X \bullet \mathrm{ran}(R^*) = X$

theorem rule starInRel $[X]$
$\forall R : X \leftrightarrow X \bullet R^* \in A \leftrightarrow B \Leftrightarrow X \in \mathbb{P}\, A \land X \in \mathbb{P}\, B$

theorem rule nullPlus $[X]$
$\{\}^+[X] = \{\}$

theorem rule nullStar $[X]$
$\{\}^* = \mathrm{id}\, X$

theorem rule crossPlus $[X]$
$\forall A, B : \mathbb{P}\, X \bullet (A \times B)^+ = A \times B$

theorem rule idPlus $[X]$
$\forall S : \mathbb{P}\, X \bullet (\mathrm{id}\, S)^+ = \mathrm{id}\, S$

theorem rule idStar $[X]$
$\forall S : \mathbb{P}\, X \bullet (\mathrm{id}\, S)^* = \mathrm{id}\, X$

theorem plusMonotonic $[X]$
$\forall Q, R : X \leftrightarrow X \mid Q \subseteq R \bullet Q^+ \subseteq R^+$

theorem starMonotonic $[X]$
$\forall Q, R : X \leftrightarrow X \mid Q \subseteq R \bullet Q^* \subseteq R^*$

Many more theorems should be added!

11 Functions

11.1 Function spaces

Definitions

Partial and total function spaces are predefined; the definitions are

$$X \pfun Y == \{ f : X \rel Y \mid \forall x : X;\ y, y' : Y \mid (x, y) \in f \land (x, y') \in f \bullet y = y' \}$$

and

$$X \fun Y == \{ f : X \pfun Y \mid \forall x : X \bullet \exists y : Y \bullet (x, y) \in f \}.$$

Theorems

theorem disabled rule pfunDef $[X, Y]$
$\forall R : X \leftrightarrow Y \bullet R \in X \pfun Y \Leftrightarrow R^{\sim} \comp R \subseteq \mathrm{id}\, Y$

theorem rule nullInPfun
$\{\} \in A \pfun B$

theorem rule nullInFun
$\{\} \in A \rightarrow B \Leftrightarrow A = \{\}$

theorem rule unitInPfun
$\{p\} \in A \pfun B \Leftrightarrow p \in A \times B$

theorem rule cupInPfun $[X, Y]$
$\forall f, g : \mathbb{P}(X \times Y);\ A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet$
$(f \cup g) \in A \pfun B$
$\Leftrightarrow$
$f \in A \pfun B$
$\land\ g \in A \pfun B$
$\land\ (\forall x : A \mid x \in \mathrm{dom}\, f \land x \in \mathrm{dom}\, g \bullet f(x) = g(x))$

theorem disabled rule cupInFun $[X, Y]$
$\forall f, g : \mathbb{P}(X \times Y);\ A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet$
$(f \cup g) \in A \rightarrow B$
$\Leftrightarrow$
$f \in A \pfun B$
$\land\ g \in A \pfun B$
$\land\ (\forall x : A \mid x \in \mathrm{dom}\, f \land x \in \mathrm{dom}\, g \bullet f(x) = g(x))$
$\land\ (\mathrm{dom}\, f) \cup (\mathrm{dom}\, g) = A$

theorem subsetOfPfun
$f \in A \pfun B \land g \in \mathbb{P}\, f \Rightarrow g \in A \pfun B$

theorem pfunExtensionality $[X, Y]$
$\forall f, g : X \pfun Y \bullet f = g \Leftrightarrow \mathrm{dom}\, f = \mathrm{dom}\, g \land (\forall x : \mathrm{dom}\, f \bullet f(x) = g(x))$

theorem funExtensionality $[X, Y]$
$\forall f, g : X \rightarrow Y \bullet f = g \Leftrightarrow (\forall x : X \bullet f(x) = g(x))$

theorem pfunIsFun $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet \forall f : A \pfun B \bullet f \in A \rightarrow B \Leftrightarrow \mathrm{dom}\, f = A$

Automation

theorem grule pfun_type $[X, Y]$
$X \pfun Y \in \mathbb{P}(X \leftrightarrow Y)$

theorem rule pfun_sub $[X, Y]$
$\forall A : \mathbb{P}\, X;\; B : \mathbb{P}\, Y \bullet A \pfun B \in \mathbb{P}(X \pfun Y)$

theorem rule pfun_ideal $[X, Y]$
$\mathbb{P}\, Z \in \mathbb{P}(X \pfun Y) \Leftrightarrow Z \in X \pfun Y$

theorem grule fun_type $[X, Y]$
$X \fun Y \in \mathbb{P}(X \pfun Y)$

theorem rule fun_sub $[X, Y]$
$\forall B : \mathbb{P}\, Y \bullet X \fun B \in \mathbb{P}(X \fun Y)$

theorem rule domFunction
$\mathit{KnownMember}[A \fun B] \land A \in \mathbb{P}\, X \land B \in \mathbb{P}\, Y \Rightarrow \mathrm{dom}[X, Y]\mathit{element} = A$

The following theorem might lead to non-maximal generic actuals in $dom[A, B]$.

theorem rule applicationInDeclaredRangePfun $[A, B]$
$\mathit{KnownMember}[A \pfun B] \land x \in \mathrm{dom}\, \mathit{element} \land B \in \mathbb{P}\, X \Rightarrow \mathit{element}(x) \in X$

theorem rule applicationInDeclaredRangeFun $[A, B]$
$\mathit{KnownMember}[A \fun B] \land x \in A \land B \in \mathbb{P}\, X \Rightarrow \mathit{element}(x) \in X$

11.2 Application

Function application is part of the Z syntax, so no definitions are needed.

Theorems

theorem rule pfunAppliesTo $[X, Y]$
$\forall f : X \pfun Y \bullet f\ \mathit{applies\$to}\ x \Leftrightarrow x \in \operatorname{dom} f$

theorem applyInRanPfun $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet \forall f : A \pfun B \bullet \forall a : \operatorname{dom} f \bullet f(a) \in \operatorname{ran} f \land f(a) \in B$

theorem applyInRanFun $[X, Y]$
$\forall f : X \rightarrow Y;\ a : X \bullet f(a) \in Y$

theorem pairInFunction $[X, Y]$
$\forall f : X \pfun Y \bullet (x, y) \in f \Rightarrow y = f(x)$

theorem rule applyUnit
$z = x \Rightarrow \{(x, y)\}(z) = y$

theorem rule applyCupLeft $[X, Y]$
$\forall f, g : X \leftrightarrow Y \bullet (f \cup g) \in X \pfun Y \land x \in \operatorname{dom} f \Rightarrow (f \cup g)(x) = f(x)$

theorem rule applyCupRight $[X, Y]$
$\forall f, g : X \leftrightarrow Y \bullet (f \cup g) \in X \pfun Y \land x \in \operatorname{dom} g \Rightarrow (f \cup g)(x) = g(x)$

theorem applySubset $[X, Y]$
$\forall f, g : X \pfun Y;\ x : X \mid f \subseteq g \land x \in \operatorname{dom} f \bullet f(x) = g(x)$

11.3 Injections

Definitions

syntax $\pinj$ *ingen* `\pinj`
syntax $\inj$ *ingen* `\inj`

$$X \pinj Y == \{ f : X \pfun Y \mid f^{\sim} \in Y \pfun X \}$$

$$X \inj Y == (X \pinj Y) \cap (X \fun Y)$$

Theorems

theorem rule nullInPinj
$\{\} \in A \pinj B$

theorem rule unitInPinj
$\{p\} \in A \pinj B \Leftrightarrow p \in A \times B$

theorem rule nullInInj
$\{\} \in A \inj B \Leftrightarrow A = \{\}$

theorem disabled rule cupInPinj $[X, Y]$
$\forall f, g : \mathbb{P}(X \times Y);\ A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet$
$(f \cup g) \in A \pinj B$
$\Leftrightarrow$
$f \in A \pfun B$
$\land\ g \in A \pfun B$
$\land\ (\forall x : A \mid x \in \operatorname{dom} f \land x \in \operatorname{dom} g \bullet f(x) = g(x))$
$\land\ (\forall y : B \mid y \in \operatorname{ran} f \land y \in \operatorname{ran} g \bullet f^{\sim}(y) = g^{\sim}(y))$

theorem pinjApplicationsEqual $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet \forall f : A \pinj B \bullet \forall x, y : \operatorname{dom} f \bullet f(x) = f(y) \Rightarrow x = y$

theorem subsetOfPinjIsPinj $[X, Y]$
$\forall f : X \pinj Y \bullet \forall g : \mathbb{P}\, f \bullet g \in X \pinj Y$

theorem applyInverse $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \mid f \in A \pinj B \land x \in \operatorname{dom} f \bullet f^{\sim}(f(x)) = x$

Automation

theorem grule pinj_type
$X \pinj Y \in \mathbb{P}(X \pfun Y)$

theorem rule inj_type
$(X \pinj Y \in \mathbb{P}\, Z \lor X \fun Y \in \mathbb{P}\, Z) \Rightarrow X \inj Y \in \mathbb{P}\, Z$

theorem rule pinj_sub $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet A \pinj B \in \mathbb{P}(X \pinj Y)$

theorem rule inj_sub $[X, Y]$
$\forall B : \mathbb{P}\, Y \bullet X \rightarrowtail B \in \mathbb{P}(X \rightarrowtail Y)$

theorem rule pinj_ideal
$\mathbb{P}\, R \in \mathbb{P}(A \pinj B) \Leftrightarrow R \in A \pinj B$

theorem rule applicationInDeclaredRangePinj $[A, B]$
$\mathit{KnownMember}[A \pinj B] \land x \in \mathrm{dom}\, \mathit{element} \land B \in \mathbb{P}\, X \Rightarrow \mathit{element}(x) \in X$

theorem rule domInjection
$\mathit{KnownMember}[A \rightarrowtail B] \land A \in \mathbb{P}\, X \land B \in \mathbb{P}\, Y \Rightarrow \mathrm{dom}[X, Y]\mathit{element} = A$

theorem rule applicationInDeclaredRangeInj $[A, B]$
$\mathit{KnownMember}[A \rightarrowtail B] \land x \in A \land B \in \mathbb{P}\, X \Rightarrow \mathit{element}(x) \in X$

We do not have enough rules to "compute" membership of a set construction in $A \rightarrowtail B$.

11.4 Surjections

Definitions

syntax $\pfun\!\!\!\twoheadrightarrow$ *ingen* `\psurj`
syntax $\twoheadrightarrow$ *ingen* `\surj`

$$X \twoheadrightarrow\!\!\!\!\!+\; Y == \{ f : X \nrightarrow Y \mid \operatorname{ran} f = Y \}$$

$$X \twoheadrightarrow Y == (X \twoheadrightarrow\!\!\!\!\!+\; Y) \cap (X \rightarrow Y)$$

Theorems

theorem nullInPsurj
$\{\} \in A \twoheadrightarrow\!\!\!\!\!+\; B \Leftrightarrow B = \{\}$

theorem unitInPsurj
$\{p\} \in A \twoheadrightarrow\!\!\!\!\!+\; B \Leftrightarrow (p \in A \times B \land B = \{p.2\})$

theorem rule cupInPsurj $[X, Y]$
$\forall f, g : \mathbb{P}(X \times Y);\ A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet$
$(f \cup g) \in A \twoheadrightarrow\!\!\!\!\!+\; B$
$\Leftrightarrow$
$f \in A \nrightarrow B$
$\land\ g \in A \nrightarrow B$
$\land\ (\forall x : A \mid x \in \operatorname{dom} f \land x \in \operatorname{dom} g \bullet f(x) = g(x))$
$\land\ (\operatorname{ran} f) \cup (\operatorname{ran} g) = B$

Automation

theorem grule psurj_type
$X \twoheadrightarrow\!\!\!\!\!+\; Y \in \mathbb{P}(X \nrightarrow Y)$

theorem rule psurj_sub $[X, Y]$
$\forall A : \mathbb{P}\, X \bullet A \twoheadrightarrow\!\!\!\!\!+\; Y \in \mathbb{P}(X \twoheadrightarrow\!\!\!\!\!+\; Y)$

theorem rule surj_type
$(X \twoheadrightarrow\!\!\!\!\!+\; Y \in \mathbb{P}\, Z \lor X \rightarrow Y \in \mathbb{P}\, Z) \Rightarrow X \twoheadrightarrow Y \in \mathbb{P}\, Z$

theorem rule ranPsurj $[X, Y]$
$\mathit{KnownMember}[A \twoheadrightarrow\!\!\!\!\!+\; B] \land A \in \mathbb{P}\, X \land B \in \mathbb{P}\, Y \Rightarrow \operatorname{ran}[X, Y]\mathit{element} = B$

theorem rule applicationInDeclaredRangePsurj $[A, B]$
$\mathit{KnownMember}[A \twoheadrightarrow\!\!\!\!\!+\; B] \land x \in \operatorname{dom} \mathit{element} \land B \in \mathbb{P}\, X \Rightarrow \mathit{element}(x) \in X$

theorem rule domSurjection $[X, Y]$
$\mathit{KnownMember}[A \twoheadrightarrow B] \land A \in \mathbb{P}\, X \land B \in \mathbb{P}\, Y \Rightarrow \operatorname{dom}[X, Y]\mathit{element} = A$

theorem rule ranSurjection $[X, Y]$
$\mathit{KnownMember}[A \twoheadrightarrow B] \land A \in \mathbb{P}\, X \land B \in \mathbb{P}\, Y \Rightarrow \operatorname{ran}[X, Y]\mathit{element} = B$

theorem rule applicationInDeclaredRangeSurj $[A, B]$
$\mathit{KnownMember}[A \twoheadrightarrow B] \land x \in A \land B \in \mathbb{P}\, X \Rightarrow \mathit{element}(x) \in X$

We do not have enough rules to "compute" membership of a set construction in $A \twoheadrightarrow B$.

11.5 Bijections

Definitions

syntax $\rightarrowtail\!\!\!\!\twoheadrightarrow$ *ingen* `\bij`

$$X \rightarrowtail\!\!\!\!\twoheadrightarrow Y == (X \twoheadrightarrow Y) \cap (X \rightarrowtail Y)$$

Theorems

theorem rule nullInBij
$\{\} \in A \rightarrowtail\!\!\!\!\twoheadrightarrow B \Leftrightarrow (A = \{\} \land B = \{\})$

theorem rule unitInBij
$\{p\} \in A \rightarrowtail\!\!\!\!\twoheadrightarrow B \Leftrightarrow (A = \{p.1\} \land B = \{p.2\})$

Automation

theorem rule bij_type
$(X \rightarrowtail Y \in \mathbb{P}\,Z \lor X \twoheadrightarrow Y \in \mathbb{P}\,Z) \Rightarrow X \rightarrowtail\!\!\!\!\twoheadrightarrow Y \in \mathbb{P}\,Z$

theorem grule id_type
$\mathrm{id}\,X \in X \rightarrowtail\!\!\!\!\twoheadrightarrow X$

theorem rule domBijection $[X, Y]$
$KnownMember[A \rightarrowtail\!\!\!\!\twoheadrightarrow B] \land A \in \mathbb{P}\,X \land B \in \mathbb{P}\,Y \Rightarrow \mathrm{dom}[X, Y]element = A$

theorem rule ranBijection $[X, Y]$
$KnownMember[A \rightarrowtail\!\!\!\!\twoheadrightarrow B] \land A \in \mathbb{P}\,X \land B \in \mathbb{P}\,Y \Rightarrow \mathrm{ran}[X, Y]element = B$

theorem rule applicationInDeclaredRangeBij $[A, B]$
$KnownMember[A \rightarrowtail\!\!\!\!\twoheadrightarrow B] \land x \in A \land B \in \mathbb{P}\,X \Rightarrow element(x) \in X$

We do not have enough rules to "compute" membership of a set construction in $A \rightarrowtail\!\!\!\!\twoheadrightarrow B$.

11.6 Inversion and function spaces

Theorems

theorem rule inverseInPfun $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet \forall f : B ⤔ A \bullet f^{\sim} \in A ⇸ B$

theorem rule inverseInFun $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet \forall f : B ⤔ A \bullet f^{\sim} \in A \rightarrow B \Leftrightarrow \mathrm{ran}\, f = A$

theorem rule inverseInPinj $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y;\ f : Y \leftrightarrow X \bullet f^{\sim} \in A ⤔ B \Leftrightarrow f \in B ⤔ A$

theorem rule inverseInInj $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y;\ f : Y \leftrightarrow X \bullet f^{\sim} \in A \rightarrowtail B \Leftrightarrow f \in B ⤔ A \land \mathrm{ran}\, f = A$

theorem rule inverseInPsurj $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \mid f \in B \rightarrowtail A \bullet f^{\sim} \in A ⤀ B$

theorem rule inverseInSurj $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \mid f \in B ⤖ A \bullet f^{\sim} \in A \twoheadrightarrow B$

theorem rule inverseBij $[X, Y]$
$\forall A : \mathbb{P}\, Y;\ B : \mathbb{P}\, X;\ f : X \leftrightarrow Y \bullet f^{\sim} \in A ⤖ B \Leftrightarrow f \in B ⤖ A$

11.7 Constant functions

There are two simple ways to define a constant function with value v for all arguments in domain D: as $D \times \{v\}$, or using a lambda-term $\lambda x : D \bullet y$. The following theorems give the basic properties of these functions.

The rules about lambda-terms are applicable to all such terms denoting constant partial functions, because of the way Z/EVES represents lambda terms.

theorem rule constFunctionInPfun $[X, Y]$
$\forall D : \mathbb{P} X;\ y : Y \bullet D \times \{y\} \in X \pfun Y$

theorem rule constFunctionInFun $[X, Y]$
$\forall y : Y \bullet X \times \{y\} \in X \rightarrow Y$

theorem rule applyConstFunction $[D]$
$\forall x : D \bullet (D \times \{y\})(x) = y$

theorem rule lambdaConstFnIsRel $[X, Y]$
$\forall D : \mathbb{P} X;\ y : Y \bullet (\lambda x : D \bullet y) \in X \leftrightarrow Y$

theorem rule lambdaConstFnIsPfun $[X, Y]$
$\forall D : \mathbb{P} X;\ y : Y \bullet (\lambda x : D \bullet y) \in X \pfun Y$

theorem rule lambdaConstFnIsFun $[X, Y]$
$\forall y : Y \bullet (\lambda x : X \bullet y) \in X \rightarrow Y$

theorem rule applyLambdaConstFn $[D]$
$\forall a : D \bullet (\lambda x : D \bullet y)(a) = y$

theorem rule domLambdaConstFn $[X, Y]$
$\forall D : \mathbb{P} X;\ y : Y \bullet \mathrm{dom}(\lambda x : D \bullet y) = D$

12 Numbers

Not all theorems about numbers appear explicitly in the Toolkit; instead, the prover has a decision procedure for linear arithmetic. This applies to equalities and inequalities using addition, subtraction, or multiplication by constants, where all the terms are known to be integers. There are also a few other built-in facts, such as the rule of signs for multiplication.

theorem integersExist
$\neg (\mathbb{Z} = \{\})$

Function *the$integer* can be generated by a proof step; it is applied to some expression whose value could not be determined to be integer.

theorem rule theIntegerElimination
$\forall i : \mathbb{Z} \bullet \mathit{the\$integer}(i) = i$

12.1 Arithmetic functions

The arithmetic functions $\_ + \_$, $\_ - \_$, $(-)$, $\_ * \_$, $\_ \mathsf{div} \_$ and $\_ \mathsf{mod} \_$ are predefined.

Theorems

theorem rule domDiv
$\mathrm{dom}(\_ \mathsf{div} \_) = \mathbb{Z} \times (\mathbb{Z} \setminus \{0\})$

theorem rule domMod
$\mathrm{dom}(\_ \mathsf{mod} \_) = \mathbb{Z} \times (\mathbb{Z} \setminus \{0\})$

theorem grule divModRelation
$\forall x, y : \mathbb{Z} \mid \neg y = 0 \bullet x = (x \mathsf{div} y) * y + (x \mathsf{mod} y)$

theorem modRange1
$\forall x, y : \mathbb{Z} \mid y > 0 \bullet 0 \leq x \mathsf{mod} y < y$

theorem modRange2
$\forall x, y : \mathbb{Z} \mid y < 0 \bullet y < x \mathsf{mod} y \leq 0$

Automation

theorem grule modLowerBound1
$\forall x, y : \mathbb{Z} \mid y > 0 \bullet 0 \leq x \mathsf{mod} y$

theorem grule modUpperBound1
$\forall x, y : \mathbb{Z} \mid y > 0 \bullet x \mathsf{mod} y < y$

theorem grule modLowerBound2
$\forall x, y : \mathbb{Z} \mid y < 0 \bullet y < x \mathsf{mod} y$

theorem grule modUpperBound2
$\forall x, y : \mathbb{Z} \mid y < 0 \bullet x \mathsf{mod} y \leq 0$

12.2 Arithmetic relations

The relations $\_ < \_$, $\_ \leq \_$, $\_ \geq \_$, and $\_ > \_$ are predefined.

Theorems

theorem disabled rule timesMonotonic1
$\forall a : \mathbb{Z};\ b, c : \mathbb{Z} \mid a \geq 1 \bullet a * b \leq a * c \Leftrightarrow b \leq c$

Note that putting $b == c + 1, c == b$ in *timesMonotonic*1 and simplifying gives the conclusion $b \leq c \Leftrightarrow a * b < a * c + a$

theorem disabled rule timesMonotonic2
$\forall a, b, c, d : \mathbb{Z} \mid 0 \leq a \leq c \land 0 \leq b \leq d \bullet a * b \leq c * d$

theorem rule divLowerBound
$\forall x, y, d : \mathbb{Z} \mid d > 0 \bullet y \leq x \mathsf{\ div\ } d \Leftrightarrow d * y \leq x$

theorem rule divUpperBound
$\forall x, y, d : \mathbb{Z} \mid d > 0 \bullet x \mathsf{\ div\ } d \leq y \Leftrightarrow x < d + d * y$

theorem rule lessthanInv
$(\_ < \_)^{\sim} = (\_ > \_)$

theorem rule leqInv
$(\_ \leq \_)^{\sim} = (\_ \geq \_)$

theorem rule greaterthanInv
$(\_ > \_)^{\sim} = (\_ < \_)$

theorem rule geqInv
$(\_ \geq \_)^{\sim} = (\_ \leq \_)$

theorem rule compLeqLeq
$(\_ \leq \_) \fatsemi (\_ \leq \_) = (\_ \leq \_)$

theorem rule compGeqGeq
$(\_ \geq \_) \fatsemi (\_ \geq \_) = (\_ \geq \_)$

We could have a number of additional theorems about composition of the arithmetic relations.

12.3 Naturals

Definitions

$\mathbb{N} == \{ n : \mathbb{Z} \mid n \geq 0 \}$

$\mathbb{N}_1 == \{ n : \mathbb{N} \mid n \geq 1 \}$

$$\begin{array}{|l} succ : \mathbb{N} \rightarrowtail\!\!\!\!\twoheadrightarrow \mathbb{N}_1 \\ \hline \langle\!\langle \text{rule succDef} \rangle\!\rangle \\ \forall n : \mathbb{N} \bullet succ(n) = n + 1 \end{array}$$

Theorems

theorem rule inNat
$x \in \mathbb{N} \Leftrightarrow x \in \mathbb{Z} \land x \geq 0$

theorem rule inNat1
$x \in \mathbb{N}_1 \Leftrightarrow x \in \mathbb{Z} \land x \geq 1$

theorem natsExist
$\neg\, \mathbb{N} = \{\}$

theorem nat1sExist
$\neg\, \mathbb{N}_1 = \{\}$

The following theorem shows that functions on the naturals can be defined inductively.

theorem primitiveRecursion $[X]$
$\forall\, base : X;\ step : X \times \mathbb{N} \rightarrow X \bullet$
$\quad \exists\, f : \mathbb{N} \rightarrow X \bullet$
$\quad\quad f(0) = base \land (\forall\, n : \mathbb{N} \bullet f(n + 1) = step(f(n), n))$

Here is another version of the primitive recursion theorem, where we allow the defined function to have additional parameters.

theorem generalPrimitiveRecursion $[Result, Parameter]$
$\forall\, base : Parameter \rightarrow Result;\ step : Result \times \mathbb{N} \times Parameter \rightarrow Result \bullet$
$\quad \exists\, f : \mathbb{N} \times Parameter \rightarrow Result \bullet$
$\quad\quad \forall\, p : Parameter \bullet$
$\quad\quad\quad f(0, p) = base(p) \land (\forall\, n : \mathbb{N} \bullet f(n + 1, p) = step(f(n, p), n, p))$

Automation

theorem grule natType
$\mathbb{N} \in \mathbb{P}\, \mathbb{Z}$

theorem grule nat1_type
$\mathbb{N}_1 \in \mathbb{P}\, \mathbb{N}$

12.4 Relational iteration

Definitions

$[X]$

$iter : \mathbb{Z} \rightarrow (X \leftrightarrow X) \rightarrow (X \leftrightarrow X)$

$\langle\!\langle$ rule iter0 $\rangle\!\rangle$
$\forall R : X \leftrightarrow X \bullet iter\,0\,R = \mathrm{id}\,X$

$\langle\!\langle$ iterNegative $\rangle\!\rangle$
$\forall R : X \leftrightarrow X;\; n : \mathbb{Z} \mid n < 0 \bullet iter\,n\,R = iter\,(-n)\,(R^{\sim})$

$\langle\!\langle$ iterPositive $\rangle\!\rangle$
$\forall R : X \leftrightarrow X;\; n : \mathbb{N} \bullet iter\,(n+1)\,R = R \mathbin{\S} (iter\,n\,R)$

Theorems

theorem rule iterateId $[X]$
$\forall n : \mathbb{Z};\; S : \mathbb{P}\,X \bullet (\mathrm{id}\,S)^n = \mathbf{if}\ n = 0\ \mathbf{then}\ \mathrm{id}\,X\ \mathbf{else}\ \mathrm{id}\,S$

theorem rule iterateEmpty $[X]$
$\forall n : \mathbb{Z} \mid \neg\, n = 0 \bullet iter[X]\,n\,\{\} = \{\}$

theorem rule oneIteration $[X]$
$\forall R : X \leftrightarrow X \bullet R^1 = R$

theorem rule minusOneIteration $[X]$
$\forall R : X \leftrightarrow X \bullet R^{-1} = R^{\sim}$

theorem disabled rule composePositiveIterates $[X]$
$\forall n, k : \mathbb{N};\; R : X \leftrightarrow Y \bullet R^{n+k} = R^n \mathbin{\S} R^k$

theorem inverseOfIteration $[X]$
$\forall R : X \leftrightarrow X;\; n : \mathbb{Z} \bullet (R^n)^{\sim} = R^{-n}$

theorem iterInPlus $[X]$
$\forall n : \mathbb{N}_1;\; R : X \leftrightarrow X \bullet R^n \subseteq R^+$

theorem iterInStar $[X]$
$\forall n : \mathbb{N};\; R : X \leftrightarrow X \bullet R^n \subseteq R^*$

Many more theorems should be added.

12.5 Ranges

Definitions

syntax $..$ *infun*2 `\upto`

$$\_ \mathrel{..} \_ : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{P}\,\mathbb{Z}$$

$$\langle\!\langle \text{ disabled rule rangeDef } \rangle\!\rangle$$
$$\forall a, b : \mathbb{Z} \bullet a \mathrel{..} b = \{ k : \mathbb{Z} \mid a \leq k \leq b \}$$

Theorems

theorem rule inRange
$$\forall a, b : \mathbb{Z} \bullet x \in a \mathrel{..} b \Leftrightarrow a \leq x \leq b$$

theorem rule rangeNull
$$a > b \Rightarrow a \mathrel{..} b = \{\}$$

theorem rule rangeUnit
$$\forall a : \mathbb{Z} \bullet a \mathrel{..} a = \{a\}$$

theorem rule rangeSubsetNat
$$\forall a, b : \mathbb{Z} \bullet a \mathrel{..} b \in \mathbb{P}\,\mathbb{N} \Leftrightarrow a \in \mathbb{N} \lor b < a$$

theorem rule rangeSubsetNat1
$$\forall a, b : \mathbb{Z} \bullet a \mathrel{..} b \in \mathbb{P}\,\mathbb{N}_1 \Leftrightarrow a \in \mathbb{N}_1 \lor b < a$$

theorem rule rangeSubsetRange
$$\forall a, b, c, d : \mathbb{Z} \bullet a \mathrel{..} b \in \mathbb{P}(c \mathrel{..} d) \Leftrightarrow b < a \lor (c \leq a \land b \leq d)$$

theorem rule rangeEqualRange
$$\forall a, b, c, d : \mathbb{Z} \bullet a \mathrel{..} b = c \mathrel{..} d \Leftrightarrow (a = c \land b = d) \lor (b < a \land d < c)$$

theorem rangeSplits
$$\forall a, b, c : \mathbb{Z} \mid a \leq b \leq c \bullet a \mathrel{..} c = (a \mathrel{..} b) \cup (b + 1 \mathrel{..} c)$$

There should be theorems for unions, intersections, and differences of ranges.

12.6 Finiteness

Definitions

syntax $\mathbb{F}$ *pregen* `\finset`

$$\mathbb{F}\,X == \{\, S : \mathbb{P}\,X \mid \exists\, n : \mathbb{N} \bullet \exists\, f : 1 \mathinner{..} n \rightarrow S \bullet \operatorname{ran} f = S \,\}$$

$$\mathbb{F}_1\,X == (\mathbb{F}\,X) \setminus \{\{\}\}$$

Theorems

theorem rule inFinset1 $[X]$
$x \in \mathbb{F}_1\,X \Leftrightarrow x \in \mathbb{F}\,X \land \lnot\, x = \{\}$

theorem rule nullFinite $[X]$
$\{\} \in \mathbb{F}\,X$

theorem rule unitFinite $[X]$
$\{x\} \in \mathbb{F}\,X \Leftrightarrow x \in X$

theorem rule cupFinite $[X]$
$\forall\, A, B : \mathbb{P}\,X \bullet A \cup B \in \mathbb{F}\,Y \Leftrightarrow (A \in \mathbb{F}\,Y \land B \in \mathbb{F}\,Y)$

theorem rule numIsInfinite
$\lnot\,(\mathbb{Z} \in \mathbb{F}\,\mathbb{Z})$

theorem rule natIsInfinite
$\lnot\,(\mathbb{N} \in \mathbb{F}\,\mathbb{Z})$

theorem rule nat1IsInfinite
$\lnot\,(\mathbb{N}_1 \in \mathbb{F}\,\mathbb{Z})$

theorem rule powersetInFinset $[X]$
$\mathbb{P}\,X \in \mathbb{F}(\mathbb{P}\,Y) \Leftrightarrow X \in \mathbb{F}\,Y$

theorem rule rangeInFinset
$\forall\, a, b : \mathbb{Z} \bullet a \mathinner{..} b \in \mathbb{F}\,X \Leftrightarrow a \mathinner{..} b \in \mathbb{P}\,X$

theorem rule crossIsFinite2 $[X, Y]$
$\lnot\,(A \times B = \{\}) \Rightarrow (A \times B \in \mathbb{F}(X \times Y) \Leftrightarrow A \in \mathbb{F}\,X \land B \in \mathbb{F}\,Y)$

theorem disabled rule finiteInduction $[X]$
$\forall\, S : \mathbb{P}(\mathbb{P}\,X) \mid \{\} \in S \land (\forall\, x : X;\; Y : S \bullet \{x\} \cup Y \in S) \bullet \mathbb{F}\,X \subseteq S$

theorem disabled rule finite1Induction $[X]$
$\forall\, S : \mathbb{P}(\mathbb{P}\,X) \mid (\forall\, x : X \bullet \{x\} \in S) \land (\forall\, A, B : S \bullet A \cup B \in S) \bullet \mathbb{F}_1\,X \subseteq S$

Automation

theorem grule finset_type $[X]$
$\mathbb{F}\, X \in \mathbb{P}(\mathbb{P}\, X)$

theorem rule finset_sub
$\mathbb{F}\, X \in \mathbb{P}(\mathbb{F}\, Y) \Leftrightarrow X \in \mathbb{P}\, Y$

theorem rule finset_ideal
$\mathbb{P}\, X \in \mathbb{P}(\mathbb{F}\, Y) \Leftrightarrow X \in \mathbb{F}\, Y$

theorem grule finset1_type $[X]$
$\mathbb{F}_1\, X \in \mathbb{P}(\mathbb{F}\, X)$

theorem rule finset1_sub
$\mathbb{F}_1\, X \in \mathbb{P}(\mathbb{F}_1\, Y) \Leftrightarrow X \in \mathbb{P}\, Y$

12.7 Cardinality

Definitions

syntax $\#$ *word* **\#**

$$
\begin{array}{|l}
\hline
[X] \\
\# : \mathbb{F}\, X \rightarrow \mathbb{N} \\
\hline
\langle\!\langle \mathrm{sizeDef} \rangle\!\rangle \\
\forall S : \mathbb{F}\, X \bullet \exists f : 1 \mathrel{..} (\# S) \rightarrowtail\!\!\!\!\rightarrow S \bullet \mathit{true} \\
\hline
\end{array}
$$

Theorems

theorem disabled rule ranCard $[X]$
$\mathrm{ran}(\#[X]) = \mathbf{if}\ X \in \mathbb{F}\, X\ \mathbf{then}\ 0 \mathrel{..} \# X\ \mathbf{else}\ \mathbb{N}$

theorem rule sizeNull $[X]$
$\#[X]\{\} = 0$

theorem rule sizeUnit $[X]$
$\forall x : X \bullet \#\{x\} = 1$

theorem rule sizeRange
$\forall a, b : \mathbb{Z} \bullet \#(a \mathrel{..} b) = \mathbf{if}\ a \leq b\ \mathbf{then}\ 1 + b - a\ \mathbf{else}\ 0$

theorem sizeOfSubset $[X]$
$\forall T : \mathbb{F}\, X \mid S \in \mathbb{P}\, T \bullet 0 \leq \# S \leq \# T$

theorem rule cardAddElement $[X]$
$\forall x : X;\ S : \mathbb{F}\, X \mid \neg\, x \in S \bullet \#(\{x\} \cup S) = 1 + \# S$

theorem cardCup $[X]$
$\forall S, T : \mathbb{F}\, X \bullet \# S + \# T = \#(S \cup T) + \#(S \cap T)$

theorem cardDiff $[X]$
$\forall S : \mathbb{F}\, X;\ T : \mathbb{P}\, X \bullet \#(S \setminus T) = \# S - \#(S \cap T)$

theorem rule card0 $[X]$
$\forall S : \mathbb{F}\, X \bullet \# S = 0 \Leftrightarrow S = \{\}$

theorem cardIsNonNegative $[X]$
$\forall S : \mathbb{F}\, X \bullet \# S \geq 0$

12.8 Finite function spaces

Definitions

syntax $\ffun$ *ingen* `\ffun`
syntax $\finj$ *ingen* `\finj`

$$X \ffun Y == (X \pfun Y) \cap \mathbb{F}(X \times Y)$$

$$X \finj Y == (X \ffun Y) \cap (X \pinj Y)$$

Theorems

theorem rule nullInFFun
$\{\} \in A \ffun B$

theorem rule unitInFFun
$\{p\} \in A \ffun B \Leftrightarrow p \in A \times B$

theorem rule cupInFfun $[X, Y]$
$\forall f, g : \mathbb{P}(X \times Y);\ A : \mathbb{P}\,X;\ B : \mathbb{P}\,Y \bullet$
$(f \cup g) \in A \ffun B$
$\Leftrightarrow$
$f \in A \ffun B$
$\land\ g \in A \ffun B$
$\land\ (\forall x : A \mid x \in \operatorname{dom} f \land x \in \operatorname{dom} g \bullet f(x) = g(x))$

theorem rule nullInFinj
$\{\} \in A \finj B$

theorem rule unitInFinj
$\{p\} \in A \finj B \Leftrightarrow p \in A \times B$

theorem disabled rule cupInFinj $[X, Y]$
$\forall f, g : \mathbb{P}(X \times Y);\ A : \mathbb{P}\,X;\ B : \mathbb{P}\,Y \bullet$
$(f \cup g) \in A \finj B$
$\Leftrightarrow$
$f \in A \finj B$
$\land\ g \in A \finj B$
$\land\ (\forall x : A \mid x \in \operatorname{dom} f \land x \in \operatorname{dom} g \bullet f(x) = g(x))$
$\land\ (\forall y : \operatorname{dom} f;\ z : \operatorname{dom} g \mid f(y) = g(z) \bullet y = z)$

theorem functionFinite $[X, Y]$
$\forall A : \mathbb{P}\,X;\ B : \mathbb{P}\,Y \bullet f \in A \ffun B \Leftrightarrow (f \in A \pfun B \land \operatorname{dom} f \in \mathbb{F}\,X)$

theorem finiteFunction $[X, Y]$
$\forall f : X \ffun Y \bullet \operatorname{dom} f \in \mathbb{F}\,X \land \operatorname{ran} f \in \mathbb{F}\,Y \land \#(\operatorname{ran} f) \leq \#(\operatorname{dom} f) = \#f$

Automation

theorem rule ffun_type
$(X \pfun Y \in \mathbb{P}\, Z \lor \mathbb{F}(X \times Y) \in \mathbb{P}\, Z) \implies X \ffun Y \in \mathbb{P}\, Z$

theorem rule finj_type
$(X \ffun Y \in \mathbb{P}\, Z \lor X \pinj Y \in \mathbb{P}\, Z) \implies X \finj Y \in \mathbb{P}\, Z$

theorem rule ffun_ideal
$\mathbb{P}\, R \in \mathbb{P}(X \ffun Y) \iff R \in X \ffun Y$

theorem rule finj_ideal
$\mathbb{P}\, R \in \mathbb{P}(X \finj Y) \iff R \in X \finj Y$

theorem rule ffun_sub $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet A \ffun B \in \mathbb{P}(X \ffun Y)$

theorem rule finj_sub $[X, Y]$
$\forall A : \mathbb{P}\, X;\ B : \mathbb{P}\, Y \bullet A \finj B \in \mathbb{P}(X \finj Y)$

theorem rule applicationInDeclaredRangeFfun $[A, B]$
$\mathit{KnownMember}[A \ffun B] \land x \in \operatorname{dom} \mathit{element} \land B \in \mathbb{P}\, X \implies \mathit{element}(x) \in X$

theorem rule applicationInDeclaredRangeFinj $[A, B]$
$\mathit{KnownMember}[A \finj B] \land x \in \operatorname{dom} \mathit{element} \land B \in \mathbb{P}\, X \implies \mathit{element}(x) \in X$

12.9 Min and max

Definitions

$$
\begin{array}{l}
min, max : \mathbb{P}_1 \mathbb{Z} \pfun \mathbb{Z} \\
\hline
\langle\!\langle \text{minDef} \rangle\!\rangle \\
min = \{ S : \mathbb{P}\mathbb{Z};\ m : \mathbb{Z} \mid m \in S \land (\forall n : S \bullet m \leq n) \} \\
\langle\!\langle \text{maxDef} \rangle\!\rangle \\
max = \{ S : \mathbb{P}\mathbb{Z};\ m : \mathbb{Z} \mid m \in S \land (\forall n : S \bullet n \leq m) \}
\end{array}
$$

Theorems

theorem maxProperty
$S \in \mathrm{dom}\, max \Rightarrow max\, S \in S \land (\forall n : S \bullet n \leq max\, S)$

theorem minProperty
$S \in \mathrm{dom}\, min \Rightarrow min\, S \in S \land (\forall n : S \bullet min\, S \leq n)$

theorem minBound
$\forall S : \mathbb{P}\mathbb{Z};\ x : \mathbb{Z} \mid (\forall n : S \bullet x \leq n) \bullet S \in \mathrm{dom}\, min \land x \leq min\, S$

theorem maxBound
$\forall S : \mathbb{P}\mathbb{Z};\ x : \mathbb{Z} \mid (\forall n : S \bullet n \leq x) \bullet S \in \mathrm{dom}\, max \land max\, S \leq x$

theorem explicitMin
$\forall S : \mathbb{P}\mathbb{Z} \mid x \in S \land (\forall n : S \bullet x \leq n) \bullet S \in \mathrm{dom}\, min \land min\, S = x$

theorem explicitMax
$\forall S : \mathbb{P}\mathbb{Z} \mid x \in S \land (\forall n : S \bullet n \leq x) \bullet S \in \mathrm{dom}\, max \land max\, S = x$

theorem rule natIsWellFounded
$\forall S : \mathbb{P}\mathbb{N} \bullet S \in \mathrm{dom}\, min \Leftrightarrow \neg\, S = \{\}$

theorem rule finiteSetHasMin
$S \in \mathbb{F}_1 \mathbb{Z} \Rightarrow S \in \mathrm{dom}\, min$

theorem rule finiteSetHasMax
$S \in \mathbb{F}_1 \mathbb{Z} \Rightarrow S \in \mathrm{dom}\, max$

theorem rule minUnit
$\forall x : \mathbb{Z} \bullet min\{x\} = x$

theorem rule minRange
$\forall a, b : \mathbb{Z} \mid a \leq b \bullet min(a \mathinner{..} b) = a$

theorem rule maxUnit
$\forall x : \mathbb{Z} \bullet max\{x\} = x$

theorem rule maxRange
$\forall a, b : \mathbb{Z} \mid a \leq b \bullet max(a \mathrel{..} b) = b$

theorem rule cupInDomMin
$S \neq \{\} \land T \neq \{\} \Rightarrow (S \cup T \in \mathrm{dom}\, min \Leftrightarrow (S \in \mathrm{dom}\, min \land T \in \mathrm{dom}\, min))$

theorem rule minCup
$S \in \mathrm{dom}\, min \land T \in \mathrm{dom}\, min \Rightarrow min(S \cup T) = \mathbf{if}\ min\, S < min\ T\ \mathbf{then}\ min\, S\ \mathbf{else}\ min\ T$

theorem rule cupInDomMax
$S \neq \{\} \land T \neq \{\} \Rightarrow (S \cup T \in \mathrm{dom}\, max \Leftrightarrow (S \in \mathrm{dom}\, max \land T \in \mathrm{dom}\, max))$

theorem rule maxCup
$S \in \mathrm{dom}\, max \land T \in \mathrm{dom}\, max \Rightarrow max(S \cup T) = \mathbf{if}\ max\, S < max\ T\ \mathbf{then}\ max\ T\ \mathbf{else}\ max\, S$

12.10 Induction

We express the induction schemes using set variables. In order to use induction to show $\forall n : \mathbb{N} \bullet P(n)$ for some property P, one first forms the set $P\_values == \{ n : \mathbb{N} \mid P(n) \}$, then uses one of the induction theorems to show $\mathbb{N} \subseteq P\_values$. Rewriting this (using *subDef* and *inPower*) gives the original goal.

Theorems

theorem disabled rule natInduction
$\forall S : \mathbb{P}\mathbb{Z} \mid 0 \in S \land (\forall x : S \bullet x + 1 \in S) \bullet \mathbb{N} \subseteq S$

theorem disabled rule nat1Induction
$\forall S : \mathbb{P}\mathbb{Z} \mid 1 \in S \land (\forall x : S \bullet x + 1 \in S) \bullet \mathbb{N}_1 \subseteq S$

theorem disabled rule natStrongInduction
$\forall S : \mathbb{P}\mathbb{Z} \mid (\forall x : \mathbb{N} \mid (\forall y : \mathbb{N} \mid y < x \bullet y \in S) \bullet x \in S) \bullet \mathbb{N} \subseteq S$

theorem disabled rule nat1StrongInduction
$\forall S : \mathbb{P}\mathbb{Z} \mid (\forall x : \mathbb{N}_1 \mid (\forall y : \mathbb{N}_1 \mid y < x \bullet y \in S) \bullet x \in S) \bullet \mathbb{N}_1 \subseteq S$

13 Sequences

Definitions

syntax seq *pregen* `\seq`
syntax iseq *pregen* `\iseq`

$$\mathrm{seq}\, X == \{ f : \mathbb{N} \pfun X \mid \exists\, n : \mathbb{N} \bullet \mathrm{dom}\, f = 1 \mathinner{..} n \}$$

$$\mathrm{seq}_1\, X == \{ f : \mathrm{seq}\, X \mid \# f > 0 \}$$

$$\mathrm{iseq}\, X == (\mathrm{seq}\, X) \cap (\mathbb{N} \pinj X)$$

Theorems

It is sometimes useful to be able to convert sequence extensions to set extensions.

theorem disabled rule nullSeqDef
$\langle\rangle = \{\}$

theorem disabled rule unitSeqDef
$\langle x \rangle = \{(1, x)\}$

theorem rule unitInSeq $[X]$
$\langle x \rangle \in \mathrm{seq}\, X \Leftrightarrow x \in X$

theorem rule unitInIseq $[X]$
$\langle x \rangle \in \mathrm{iseq}\, X \Leftrightarrow x \in X$

theorem rule inSeq1 $[X]$
$s \in \mathrm{seq}_1\, X \Leftrightarrow s \in \mathrm{seq}\, X \land \lnot\, s = \langle\rangle$

theorem rule applyUnitSeq
$\langle x \rangle(1) = x$

theorem rule sizeNullSeq $[X]$
$\#[\mathbb{Z} \times X]\langle\rangle = 0$

theorem rule sizeUnitSeq $[X]$
$\forall\, x : X \bullet \# \langle x \rangle = 1$

theorem domSeq $[X]$
$\forall\, s : \mathrm{seq}\, X \bullet \mathrm{dom}\, s = 1 \mathinner{..} \# s$

theorem rule unitIsNullSeq1
$\lnot\, \langle x \rangle = \langle\rangle$

theorem rule unitIsNullSeq2
$\lnot\, \langle\rangle = \langle x \rangle$

theorem rule unitsEqual
$\langle x \rangle = \langle y \rangle \Leftrightarrow x = y$

theorem rule ranNullSeq $[X]$
$\mathrm{ran}[\mathbb{Z}, X]\langle\rangle = \{\}$

theorem rule ranUnitSeq $[X]$
$\forall x : X \bullet \mathrm{ran}\langle x \rangle = \{x\}$

theorem ranSeqInPower $[X]$
$\forall s : \mathrm{seq}\, X \bullet \mathrm{ran}\, s \in \mathbb{P}\, Y \Leftrightarrow s \in \mathrm{seq}\, Y$

theorem rule dresSeqInSeq $[X]$
$\forall S : \mathbb{P}\, X \bullet \forall a, b : \mathbb{Z};\ s : \mathrm{seq}\, S \bullet (a \mathinner{..} b) \lhd s \in \mathrm{seq}\, S \Leftrightarrow (a \leq 1 \lor b < a \lor a > \# s)$

theorem rule seqSize0 $[X]$
$\forall s : \mathrm{seq}\, X \bullet \# s = 0 \Leftrightarrow s = \langle\rangle$

Automation

theorem grule seq_type $[X]$
$\mathrm{seq}\, X \in \mathbb{P}(\mathbb{N}_1 \mathrel{+\!\!\!\twoheadrightarrow} X)$

theorem grule seq1_type $[X]$
$\mathrm{seq}_1\, X \in \mathbb{P}(\mathrm{seq}\, X)$

theorem rule iseq_type $[X]$
$(\mathrm{seq}\, X \in \mathbb{P}\, Z \lor \mathbb{N}_1 \mathrel{\rightarrowtail\!\!\!\!\!+\!\!\!\twoheadrightarrow} X \in \mathbb{P}\, Z) \Rightarrow \mathrm{iseq}\, X \in \mathbb{P}\, Z$

theorem grule nullSeqType
$\langle\rangle \in \mathrm{iseq}\{\}$

theorem grule unitSeqType
$\langle x \rangle \in \mathrm{iseq}\{x\}$

theorem rule seq_sub $[Y]$
$\mathrm{seq}\, X \in \mathbb{P}(\mathrm{seq}\, Y) \Leftrightarrow X \in \mathbb{P}\, Y$

theorem rule seq1_sub $[Y]$
$\mathrm{seq}_1\, X \in \mathbb{P}(\mathrm{seq}_1\, Y) \Leftrightarrow X \in \mathbb{P}\, Y$

theorem rule iseq_sub $[Y]$
$\mathrm{iseq}\, X \in \mathbb{P}(\mathrm{iseq}\, Y) \Leftrightarrow X \in \mathbb{P}\, Y$

theorem rule domSeqRule $[X]$
$KnownMember[\mathrm{seq}\, A] \land A \in \mathbb{P}\, X \Rightarrow \mathrm{dom}\, element = 1 \mathrel{..} \# element$

theorem rule applicationInDeclaredRangeSeq $[A, B]$
$KnownMember[\mathrm{seq}\, A] \land 1 \leq x \leq \# element \land A \in \mathbb{P}\, X \Rightarrow element(x) \in X$

theorem rule domIseqRule $[X]$
$KnownMember[\mathrm{iseq}\, A] \land A \in \mathbb{P}\, X \Rightarrow \mathrm{dom}\, element = 1 \mathrel{..} \# element$

theorem rule applicationInDeclaredRangeIseq $[A, B]$
$KnownMember[\mathrm{iseq}\, A] \land 1 \leq x \leq \# element \land A \in \mathbb{P}\, X \Rightarrow element(x) \in X$

13.1 Concatenation

Definitions

syntax $\frown$ *infun*3 `\cat`

$[X]$

$\_ \frown \_ : \mathrm{seq}\, X \times \mathrm{seq}\, X \rightarrow \mathrm{seq}\, X$

Theorems

theorem rule domCat
$\mathrm{seq}\, X \times \mathrm{seq}\, X \in \mathbb{P}\, A \land \mathrm{seq}\, X \in \mathbb{P}\, B \Rightarrow \mathrm{dom}[A, B](\_ \frown \_)[X] = \mathrm{seq}\, X \times \mathrm{seq}\, X$

theorem rule catInSeq $[X]$
$\forall s, t : \mathrm{seq}\, X \bullet (s \frown t) \in \mathrm{seq}\, Y \Leftrightarrow (s \in \mathrm{seq}\, Y \land t \in \mathrm{seq}\, Y)$

theorem rule sizeCat $[X]$
$\forall s, t : \mathrm{seq}\, X \bullet \#(s \frown t) = (\#s) + (\#t)$

theorem rule applyCat $[X]$
$\forall s, t : \mathrm{seq}\, X \bullet \forall n : 1 \mathinner{..} \#s + \#t \bullet (s \frown t)(n) = \textbf{if } n \leq \#s \textbf{ then } s(n) \textbf{ else } t(n - \#s)$

theorem rule ranCat $[X]$
$\forall s, t : \mathrm{seq}\, X \bullet \mathrm{ran}(s \frown t) = \mathrm{ran}\, s \cup \mathrm{ran}\, t$

theorem rule nullCat $[X]$
$\forall s : \mathrm{seq}\, X \bullet \langle\rangle \frown s = s$

theorem rule catNull $[X]$
$\forall s : \mathrm{seq}\, X \bullet s \frown \langle\rangle = s$

theorem disabled rule catRightCancellation $[X]$
$\forall s, t, u : \mathrm{seq}\, X \bullet (s \frown u = t \frown u) \Leftrightarrow s = t$

theorem disabled rule catLeftCancellation $[X]$
$\forall s, t, u : \mathrm{seq}\, X \bullet (s \frown t = s \frown u) \Leftrightarrow t = u$

theorem rule catsEqual $[X]$
$\forall x, y : X;\ s, t : \mathrm{seq}\, X \bullet \langle x\rangle \frown s = \langle y\rangle \frown t \Leftrightarrow x = y \land s = t$

theorem rule catAssociates $[X]$
$\forall s, t, u : \mathrm{seq}\, X \bullet (s \frown t) \frown u = s \frown (t \frown u)$

theorem rule catYieldsNullseq $[X]$
$\forall s, t : \mathrm{seq}\, X \bullet ((s \frown t) = \langle\rangle) \Leftrightarrow s = \langle\rangle \land t = \langle\rangle$

13.2 Sequence decomposition

Definitions

$[X]$

$head, last : \mathrm{seq}_1\, X \rightarrow X$
$tail, front : \mathrm{seq}_1\, X \rightarrow \mathrm{seq}\, X$

$\langle\!\langle$ disabled rule headDef $\rangle\!\rangle$
$\forall s : \mathrm{seq}_1\, X \bullet head\ s = s(1)$

$\langle\!\langle$ disabled rule lastDef $\rangle\!\rangle$
$\forall s : \mathrm{seq}_1\, X \bullet last\ s = s(\#s)$

$\langle\!\langle$ disabled rule tailDef $\rangle\!\rangle$
$\forall s : \mathrm{seq}_1\, X \bullet tail\ s = (\lambda\, n : 1 \mathrel{..} \#s - 1 \bullet s(n+1))$

$\langle\!\langle$ disabled rule frontDef $\rangle\!\rangle$
$\forall s : \mathrm{seq}_1\, X \bullet front\ s = (\lambda\, n : 1 \mathrel{..} \#s - 1 \bullet s(n))$

Theorems

theorem rule headInSet $[X]$
$\forall Y : \mathbb{P}\, X \bullet \forall s : \mathrm{seq}_1\, Y \bullet head\ s \in Y$

theorem rule lastInSet $[X]$
$\forall Y : \mathbb{P}\, X \bullet \forall s : \mathrm{seq}_1\, Y \bullet last\ s \in Y$

theorem rule tailInSeq $[X]$
$\forall Y : \mathbb{P}\, X \bullet \forall s : \mathrm{seq}_1\, Y \bullet tail\ s \in \mathrm{seq}\, Y$

theorem rule frontInSeq $[X]$
$\forall Y : \mathbb{P}\, X \bullet \forall s : \mathrm{seq}_1\, Y \bullet front\ s \in \mathrm{seq}\, Y$

theorem headTailComposition $[X]$
$\forall s : \mathrm{seq}_1\, X \bullet s = \langle head\ s \rangle \frown (tail\ s)$

theorem frontLastComposition $[X]$
$\forall s : \mathrm{seq}_1\, X \bullet s = (front\ s) \frown \langle last\ s \rangle$

theorem rule cardTail $[X]$
$\forall s : \mathrm{seq}_1\, X \bullet \#(tail\ s) = (\#s) - 1$

theorem rule applyTail $[X]$
$\forall s : \mathrm{seq}_1\, X \mid 1 \leq n < \#s \bullet (tail\ s)(n) = s(n+1)$

theorem rule cardFront $[X]$
$\forall s : \mathrm{seq}_1\, X \bullet \#(front\ s) = (\#s) - 1$

theorem rule applyFront $[X]$
$\forall s : \mathrm{seq}_1 X \mid 1 \leq n < \#s \bullet (\mathit{front}\ s)(n) = s(n)$

theorem rule headUnit $[X]$
$\forall x : X \bullet \mathit{head}\ \langle x \rangle = x$

theorem rule headCat $[X]$
$\forall s, t : \mathrm{seq}\, X \mid \neg\ s = \langle\rangle \bullet \mathit{head}\ (s \frown t) = \mathit{head}\ s$

theorem rule tailUnit $[X]$
$\forall x : X \bullet \mathit{tail}\ \langle x \rangle = \langle\rangle$

theorem rule tailCat $[X]$
$\forall s, t : \mathrm{seq}\, X \mid \neg\ s = \langle\rangle \bullet \mathit{tail}\ (s \frown t) = (\mathit{tail}\ s) \frown t$

theorem rule lastUnit $[X]$
$\forall x : X \bullet \mathit{last}\ \langle x \rangle = x$

theorem rule lastCat $[X]$
$\forall s, t : \mathrm{seq}\, X \mid \neg\ t = \langle\rangle \bullet \mathit{last}\ (s \frown t) = \mathit{last}\ t$

theorem rule frontUnit $[X]$
$\forall x : X \bullet \mathit{front}\ \langle x \rangle = \langle\rangle$

theorem rule frontCat $[X]$
$\forall s, t : \mathrm{seq}\, X \mid \neg\ t = \langle\rangle \bullet \mathit{front}\ (s \frown t) = s \frown (\mathit{front}\ t)$

Automation

theorem rule tail_result $[X]$
$\forall s : \mathrm{seq}_1 X \mid \mathrm{seq}(\mathrm{ran}\, s) \in \mathbb{P}\, Z \bullet \mathit{tail}\ s \in Z$

theorem rule front_result $[X]$
$\forall s : \mathrm{seq}_1 X \mid \mathbb{P}\, s \in \mathbb{P}\, Z \bullet \mathit{front}\ s \in Z$

theorem rule head_result $[X]$
$\forall s : \mathrm{seq}_1 X \mid \mathrm{ran}\, s \in \mathbb{P}\, Z \bullet \mathit{head}\ s \in Z$

theorem rule last_result $[X]$
$\forall s : \mathrm{seq}_1 X \mid \mathrm{ran}\, s \in \mathbb{P}\, Z \bullet \mathit{last}\ s \in Z$

13.3 Reversal

Definitions

$$[X]$$

$$rev : \operatorname{seq} X \rightarrow \operatorname{seq} X$$

$\langle\!\langle$ rule revNull $\rangle\!\rangle$
$rev\ \langle\rangle = \langle\rangle$

$\langle\!\langle$ rule revUnit $\rangle\!\rangle$
$\forall x : X \bullet rev\langle x\rangle = \langle x\rangle$

$\langle\!\langle$ rule revCat $\rangle\!\rangle$
$\forall s, t : \operatorname{seq} X \bullet rev(s \frown t) = (rev\ t) \frown (rev\ s)$

Theorems

theorem rule revInSeq $[X]$
$\forall s : \operatorname{seq} X \bullet rev\ s \in \operatorname{seq} Y \Leftrightarrow s \in \operatorname{seq} Y$

theorem rule revInIseq $[X]$
$\forall s : \operatorname{seq} X \bullet rev\ s \in \operatorname{iseq} Y \Leftrightarrow s \in \operatorname{iseq} Y$

theorem rule revRev $[X]$
$\forall s : \operatorname{seq} X \bullet rev(rev\ s) = s$

theorem rule domRev $[X]$
$\forall s : \operatorname{seq} X \bullet \operatorname{dom}(rev\ s) = \operatorname{dom} s$

theorem rule ranRev $[X]$
$\forall s : \operatorname{seq} X \bullet \operatorname{ran}(rev\ s) = \operatorname{ran} s$

theorem rule cardRev $[X]$
$\forall s : \operatorname{seq} X \bullet \#(rev\ s) = \#s$

theorem rule tailRev $[X]$
$\forall s : \operatorname{seq}_1 X \bullet tail(rev\ s) = rev\ (front\ s)$

theorem rule frontRev $[X]$
$\forall s : \operatorname{seq}_1 X \bullet front(rev\ s) = rev\ (tail\ s)$

theorem rule headRev $[X]$
$\forall s : \operatorname{seq}_1 X \bullet head(rev\ s) = last\ s$

theorem rule lastRev $[X]$
$\forall s : \operatorname{seq}_1 X \bullet last(rev\ s) = head\ s$

theorem rule applyRev $[X]$
$\forall s : \operatorname{seq} X \mid 1 \leq n \leq \#s \bullet (rev\ s)(n) = s(1 + (\#s) - n)$

13.4 Filtering

Definitions

syntax $\upharpoonright$ *infun4* `\filter`
syntax $\restriction$ *infun4* `\extract`

$$[X]$$
$$\_ \restriction \_ : \mathbb{P}\,\mathbb{Z} \times \operatorname{seq} X \rightarrow \operatorname{seq} X$$
$$\_ \upharpoonright \_ : \operatorname{seq} X \times \mathbb{P}\, X \rightarrow \operatorname{seq} X$$
$$squash : (\mathbb{N}_1 \mathrel{+\!\!\!\!\twoheadrightarrow} X) \rightarrow \operatorname{seq} X$$

$\langle\!\langle$ extractDef $\rangle\!\rangle$
$\forall E : \mathbb{P}\,\mathbb{Z};\ s : \operatorname{seq} X \bullet E \restriction s = squash(E \lhd s)$

$\langle\!\langle$ filterDef $\rangle\!\rangle$
$\forall s : \operatorname{seq} X;\ F : \mathbb{P}\, X \bullet s \upharpoonright F = squash(s \rhd F)$

$\langle\!\langle$ squashDef $\rangle\!\rangle$
$\forall f : \mathbb{N}_1 \mathrel{+\!\!\!\!\twoheadrightarrow} X \bullet$
$\quad \exists g : 1 \mathinner{..} \# f \rightarrowtail\!\!\!\!\!\twoheadrightarrow (\operatorname{dom} f)$
$\quad\quad | (\forall i, j : \operatorname{dom} g \mid i < j \bullet g(i) < g(j))$
$\quad\quad \bullet squash(f) = g \circ f$

Spivey specifies $\_ \restriction \_ : \mathbb{P}\,\mathbb{N}_1 \times \ldots$; there seemed to be no obvious reason why that domain could not be enlarged.

Theorems

theorem rule extractNull $[X]$
$\forall E : \mathbb{P}\,\mathbb{Z} \bullet E \restriction [X]\langle\rangle = \langle\rangle$

theorem disabled rule extractUnit $[X]$
$\forall E : \mathbb{P}\,\mathbb{Z};\ x : X \bullet E \restriction \langle x \rangle = \textbf{if } 1 \in E \textbf{ then } \langle x \rangle \textbf{ else } \langle\rangle$

theorem rule extractUnit1 $[X]$
$\forall E : \mathbb{P}\,\mathbb{Z};\ x : X \mid 1 \in E \bullet E \restriction \langle x \rangle = \langle x \rangle$

theorem rule extractUnit2 $[X]$
$\forall E : \mathbb{P}\,\mathbb{Z};\ x : X \mid 1 \notin E \bullet E \restriction \langle x \rangle = \langle\rangle$

theorem disabled rule extractAll $[X]$
$\forall E : \mathbb{P}\,\mathbb{Z};\ s : \operatorname{seq} X \mid \operatorname{dom} s \in \mathbb{P}\, E \bullet E \restriction s = s$

theorem disabled rule extractNone $[X]$
$\forall E : \mathbb{P}\,\mathbb{Z};\ s : \operatorname{seq} X \mid (\operatorname{dom} s) \cap E = \{\} \bullet E \restriction s = \langle\rangle$

theorem rule nullExtract $[X]$
$\forall s : \operatorname{seq} X \bullet \{\} \restriction s = \langle\rangle$

theorem rule extractIsSeq $[X]$
$\forall E : \mathbb{P}\,\mathbb{Z};\ Y : \mathbb{P}\,X \bullet \forall s : \mathrm{seq}\,Y \bullet E \upharpoonleft s \in \mathrm{seq}\,Y$

theorem rule extractIsIseq $[X]$
$\forall E : \mathbb{P}\,\mathbb{Z};\ Y : \mathbb{P}\,X \bullet \forall s : \mathrm{iseq}\,Y \bullet E \upharpoonleft s \in \mathrm{iseq}\,Y$

theorem disabled rule sizeExtract $[X]$
$\forall E : \mathbb{P}\,\mathbb{Z};\ s : \mathrm{seq}\,X \bullet \#(E \upharpoonleft s) = \#(E \cap (1 \mathrel{..} \#s))$

theorem rule nullFilter $[X]$
$\forall F : \mathbb{P}\,X \bullet \langle\rangle \restriction F = \langle\rangle$

theorem disabled rule filterUnit $[X]$
$\forall F : \mathbb{P}\,X;\ x : X \bullet \langle x\rangle \restriction F = \mathbf{if}\ x \in F\ \mathbf{then}\ \langle x\rangle\ \mathbf{else}\ \langle\rangle$

theorem rule filterUnit1 $[X]$
$\forall F : \mathbb{P}\,X \bullet \forall x : F \bullet \langle x\rangle \restriction F = \langle x\rangle$

theorem rule filterUnit2 $[X]$
$\forall F : \mathbb{P}\,X;\ x : X \mid \neg\, x \in F \bullet \langle x\rangle \restriction F = \langle\rangle$

theorem rule filterCat $[X]$
$\forall F : \mathbb{P}\,X;\ s, t : \mathrm{seq}\,X \bullet (s \frown t) \restriction F = (s \restriction F) \frown (t \restriction F)$

theorem disabled rule filterAll $[X]$
$\forall F : \mathbb{P}\,X;\ s : \mathrm{seq}\,X \mid F \cap \mathrm{ran}\,s = \{\} \bullet s \restriction F = \langle\rangle$

theorem disabled rule filterNone $[X]$
$\forall F : \mathbb{P}\,X;\ s : \mathrm{seq}\,X \mid \mathrm{ran}\,s \subseteq F \bullet s \restriction F = s$

theorem rule filterNull $[X]$
$\forall s : \mathrm{seq}\,X \bullet s \restriction \{\} = \langle\rangle$

theorem rule filterInSeq1 $[X]$
$\forall s : \mathrm{seq}\,X;\ F : \mathbb{P}\,Z \bullet (s \restriction F) \in \mathrm{seq}\,Z$

theorem rule filterInSeq2 $[X]$
$\forall s : \mathrm{seq}\,Z;\ F : \mathbb{P}\,X \bullet (s \restriction F) \in \mathrm{seq}\,Z$

theorem rule filterInIseq1 $[X]$
$\forall s : \mathrm{iseq}\,X;\ F : \mathbb{P}\,Z \bullet (s \restriction F) \in \mathrm{iseq}\,Z$

theorem rule filterInIseq2 $[X]$
$\forall s : \mathrm{iseq}\,Z;\ F : \mathbb{P}\,X \bullet (s \restriction F) \in \mathrm{iseq}\,Z$

theorem rule ranFilter $[X]$
$\forall s : \mathrm{seq}\, X;\ F : \mathbb{P}\, X \bullet \mathrm{ran}(s \upharpoonright F) = (\mathrm{ran}\, s) \cap F$

theorem rule revFilter $[X]$
$\forall s : \mathrm{seq}\, X;\ F : \mathbb{P}\, X \bullet rev(s \upharpoonright F) = (rev\ s) \upharpoonright F$

theorem rule sizeFilter $[X]$
$\forall s : \mathrm{seq}\, X;\ F : \mathbb{P}\, X \bullet \#(s \upharpoonright F) \leq \# s$

theorem rule squashInSeq $[X]$
$\forall Y : \mathbb{P}\, X \bullet \forall f : \mathbb{N}_1 \mathrel{+\!\!\!\!\!\!\twoheadrightarrow} Y \bullet squash(f) \in \mathrm{seq}\ Y$

theorem rule squashInIseq $[X]$
$\forall Y : \mathbb{P}\, X \bullet \forall f : \mathbb{N}_1 \mathrel{\rightarrowtail\!\!\!\!\!\!+} Y \bullet squash(f) \in \mathrm{iseq}\ Y$

theorem rule squashNull $[X]$
$squash[X](\{\}) = \langle\rangle$

theorem rule squashUnit $[X]$
$\forall p : \mathbb{N}_1 \times X \bullet squash\{p\} = \langle p.2 \rangle$

There should be other rules about squash.

Automation

We should perhaps offer *filterResult*, *extractResult*, and *squashResult*.

13.5 Mapping over a sequence

Mapping a function f over a sequence $s = \langle x_1, x_2, \ldots \rangle$ results in the sequence $\langle f(x_1), f(x_2), \ldots \rangle$. There is no special function for this in Z; since sequences are functions, composition can be used instead. The above result can be expressed as $f \circ s$ or $s \mathbin{;} f$.

Theorems

The following three theorems allow for the computation of mapping of a function over a literal sequence.

theorem rule mapSeqNull $[X, Y]$
$\forall f : X \leftrightarrow Y \bullet \langle\rangle \mathbin{;} f = \langle\rangle$

theorem rule mapSeqUnit $[X, Y]$
$\forall f : X \pfun Y;\ x : X \mid x \in \operatorname{dom} f \bullet \langle x \rangle \mathbin{;} f = \langle f(x) \rangle$

theorem rule mapSeqUnitOffDomain $[X, Y]$
$\forall f : X \pfun Y;\ x : X \mid x \notin \operatorname{dom} f \bullet \langle x \rangle \mathbin{;} f = \langle\rangle$

theorem disabled rule mapSeqUnit2 $[X, Y]$
$\forall f : X \pfun Y;\ x : X \bullet \langle x \rangle \mathbin{;} f = \mathbf{if}\ x \in \operatorname{dom} f\ \mathbf{then}\ \langle f(x) \rangle\ \mathbf{else}\ \langle\rangle$

theorem rule mapSeqCat $[X, Y]$
$\forall f : X \pfun Y;\ s, t : \operatorname{seq} X \bullet (s \frown t) \mathbin{;} f = (s \mathbin{;} f) \frown (t \mathbin{;} f)$

13.6 Relations between sequences

Definitions

syntax prefix *inrel* `\prefix`
syntax suffix *inrel* `\suffix`
syntax in *inrel* `\inseq`

$[X]$

$\_ \mathsf{prefix} \_, \_ \mathsf{suffix} \_, \_ \mathsf{in} \_ : \mathrm{seq}\, X \leftrightarrow \mathrm{seq}\, X$

$\langle\!\langle$ disabled rule prefixDef $\rangle\!\rangle$
$\forall s, t : \mathrm{seq}\, X \bullet s \mathrel{\mathsf{prefix}} t \Leftrightarrow (\exists u : \mathrm{seq}\, X \bullet s \frown u = t)$

$\langle\!\langle$ disabled rule suffixDef $\rangle\!\rangle$
$\forall s, t : \mathrm{seq}\, X \bullet s \mathrel{\mathsf{suffix}} t \Leftrightarrow (\exists u : \mathrm{seq}\, X \bullet u \frown s = t)$

$\langle\!\langle$ disabled rule inseqDef $\rangle\!\rangle$
$\forall s, t : \mathrm{seq}\, X \bullet s \mathrel{\mathsf{in}} t \Leftrightarrow (\exists u, v : \mathrm{seq}\, X \bullet u \frown s \frown v = t)$

Theorems

theorem rule nullPrefix $[X]$
$\forall t : \mathrm{seq}\, X \bullet \langle\rangle \mathrel{\mathsf{prefix}} t$

theorem rule prefixNull $[X]$
$\forall s : \mathrm{seq}\, X \bullet s \mathrel{\mathsf{prefix}} \langle\rangle \Leftrightarrow s = \langle\rangle$

theorem rule nullSuffix $[X]$
$\forall t : \mathrm{seq}\, X \bullet \langle\rangle \mathrel{\mathsf{suffix}} t$

theorem rule suffixNull $[X]$
$\forall s : \mathrm{seq}\, X \bullet s \mathrel{\mathsf{suffix}} \langle\rangle \Leftrightarrow s = \langle\rangle$

theorem rule nullInseq $[X]$
$\forall t : \mathrm{seq}\, X \bullet \langle\rangle \mathrel{\mathsf{in}} t$

theorem rule inseqNull $[X]$
$\forall s : \mathrm{seq}\, X \bullet s \mathrel{\mathsf{in}} \langle\rangle \Leftrightarrow s = \langle\rangle$

theorem prefixRev $[X]$
$\forall s, t : \mathrm{seq}\, X \bullet s \mathrel{\mathsf{prefix}} t \Leftrightarrow rev(s) \mathrel{\mathsf{suffix}} rev(t)$

theorem inSeqRev $[X]$
$\forall s, t : \mathrm{seq}\, X \bullet rev(s) \mathrel{\mathsf{in}} rev(t) \Rightarrow s \mathrel{\mathsf{in}} t$

The partial order laws should be added.

13.7 Distributed concatenation

Definitions

$[X]$

$\frown/ : \mathrm{seq}(\mathrm{seq}\, X) \rightarrow \mathrm{seq}\, X$

$\langle\!\langle \text{rule dcatNull} \rangle\!\rangle$
$\frown/\langle\rangle = \langle\rangle$

$\langle\!\langle \text{rule dcatUnit} \rangle\!\rangle$
$\forall s : \mathrm{seq}\, X \bullet \frown/\langle s\rangle = s$

$\langle\!\langle \text{rule dcatCat} \rangle\!\rangle$
$\forall s, t : \mathrm{seq}(\mathrm{seq}\, X) \bullet \frown/(s \frown t) = (\frown/\, s) \frown (\frown/\, t)$

Theorems

theorem rule dcatInSeq $[X]$
$\forall s : \mathrm{seq}(\mathrm{seq}\, X);\ Y : \mathbb{P}\, X \bullet \frown/\, s \in \mathrm{seq}\, Y \Leftrightarrow s \in \mathrm{seq}(\mathrm{seq}\, Y)$

13.8 Disjointness and partitioning

Definitions

syntax disjoint *prerel* `\disjoint`
syntax partition *inrel* `\partition`

$[I, X]$

$\mathsf{disjoint}\ \_ : \mathbb{P}(I \pfun \mathbb{P}\, X)$

$\_\ \mathsf{partition}\ \_ : (I \pfun \mathbb{P}\, X) \leftrightarrow \mathbb{P}\, X$

$\langle\!\langle$ disabled rule disjointDef $\rangle\!\rangle$
$\forall S : I \pfun \mathbb{P}\, X \bullet \mathsf{disjoint}\ S \Leftrightarrow (\forall i, j : \mathrm{dom}\, S \mid \neg\, i = j \bullet S(i) \cap S(j) = \{\})$

$\langle\!\langle$ rule partitionDef $\rangle\!\rangle$
$\forall S : I \pfun \mathbb{P}\, X;\ T : \mathbb{P}\, X \bullet S\ \mathsf{partition}\ T \Leftrightarrow \mathsf{disjoint}\ S \land \bigcup(\mathrm{ran}\, S) = T$

Theorems

theorem rule disjointEmpty $[I, X]$
$\mathsf{disjoint}\ [I, X]\, \{\}$

theorem rule disjointNull $[X]$
$\mathsf{disjoint}\ [\mathbb{Z}, X]\, \langle\rangle$

theorem rule disjointUnit $[I, X]$
$\forall x : I \times \mathbb{P}\, X \bullet \mathsf{disjoint}\ \{x\}$

theorem rule disjointUnitSeq $[X]$
$\forall x : \mathbb{P}\, X \bullet \mathsf{disjoint}\ \langle x\rangle$

theorem disabled rule disjointCat $[X]$
$\forall s, t : \mathrm{seq}(\mathbb{P}\, X) \bullet \mathsf{disjoint}\ (s \frown t) \Leftrightarrow \mathsf{disjoint}\ s \land \mathsf{disjoint}\ t \land (\bigcup(\mathrm{ran}\, s)) \cap (\bigcup(\mathrm{ran}\, t)) = \{\}$

13.9 Induction

The comments on integer induction apply equally well here.

theorem disabled rule seqInduction $[X]$
$\forall A : \mathbb{P} X \bullet \forall S : \mathbb{P}(\mathrm{seq}\, A) \mid \langle\rangle \in S \land (\forall x : A \bullet \langle x \rangle \in S) \land (\forall s, t : S \bullet s \frown t \in S) \bullet \mathrm{seq}\, A \subseteq S$

theorem disabled rule seqLeftInduction $[X]$
$\forall A : \mathbb{P} X;\ S : \mathbb{P}(\mathrm{seq}\, X) \mid \langle\rangle \in S \land (\forall x : A;\ s : S \bullet \langle x \rangle \frown s \in S) \bullet \mathrm{seq}\, A \subseteq S$

theorem disabled rule seqRightInduction $[X]$
$\forall A : \mathbb{P} X;\ S : \mathbb{P}(\mathrm{seq}\, X) \mid \langle\rangle \in S \land (\forall x : A;\ s : S \bullet s \frown \langle x \rangle \in S) \bullet \mathrm{seq}\, A \subseteq S$

theorem disabled rule seq1Induction $[X]$
$\forall A : \mathbb{P} X \bullet \forall S : \mathbb{P}(\mathrm{seq}\, A) \mid (\forall x : A \bullet \langle x \rangle \in S) \land (\forall s, t : S \bullet s \frown t \in S) \bullet \mathrm{seq}_1\, A \subseteq S$

13.10 Constant sequences

Constant sequences are a special case of constant functions. As described in Section 11.7, two different idioms can be used.

theorem rule constFnIsSeq $[Y]$
$\forall D : \mathbb{P}\,\mathbb{Z};\ y : Y \bullet D \times \{y\} \in \mathrm{seq}\ Y \Leftrightarrow (\exists n : \mathbb{N} \bullet D = 1 \mathrel{..} n)$

theorem rule lambdaConstFnIsSeq $[Y]$
$\forall D : \mathbb{P}\,\mathbb{Z};\ y : Y \bullet (\lambda x : D \bullet y) \in \mathrm{seq}\ Y \Leftrightarrow (\exists n : \mathbb{N} \bullet D = 1 \mathrel{..} n)$

14 Bags

Definitions

syntax bag *pregen* `\bag`

We define $\mathrm{bag}\,X$ as $X \nrightarrow \mathbb{N}_1$ rather than $X \rightarrow \mathbb{N}$ so that $A \subseteq B$ implies $\mathrm{bag}\,A \subseteq \mathrm{bag}\,B$.

$$\mathrm{bag}\,X == X \nrightarrow \mathbb{N}_1$$

Theorems

It is sometimes useful to turn bag extensions into set extensions.

theorem disabled rule nullBagDef
$[\![]\!] = \{\}$

theorem disabled rule unitBagDef
$[\![x]\!] = \{(x, 1)\}$

theorem grule unitBagType
$[\![x]\!] \in \mathrm{bag}\{x\}$

theorem rule unitInBag
$[\![x]\!] \in \mathrm{bag}\,X \Leftrightarrow x \in X$

Some specifiers use set constructions as bags; the following three rules account for that:

theorem rule nullsetInBag $[X]$
$\{\} \in \mathrm{bag}\,X$

theorem rule unitsetInBag $[X]$
$\{x\} \in \mathrm{bag}\,X \Leftrightarrow x \in X \times \mathbb{N}_1$

theorem rule cupInBag $[X]$
$\forall S, T : \mathbb{P}(X \times \mathbb{Z});\ Y : \mathbb{P}\,X \mid (\mathrm{dom}\,S) \cap \mathrm{dom}\,T = \{\}$
$\quad \bullet S \cup T \in \mathrm{bag}\,Y \Leftrightarrow S \in \mathrm{bag}\,Y \wedge T \in \mathrm{bag}\,Y$

theorem rule sizeNullBag $[X]$
$\#[X \times \mathbb{Z}][\![]\!] = 0$

theorem rule sizeUnitBag $[X]$
$\forall x : X \bullet \#[\![x]\!] = 1$

theorem rule unitIsNullBag1
$\neg\ [\![x]\!] = [\![]\!]$

theorem rule unitIsNullBag2
$\neg\ [\![]\!] = [\![x]\!]$

theorem rule unitBagsEqual
$[\![x]\!] = [\![y]\!] \Leftrightarrow x = y$

If B is a bag, $\mathrm{dom}\, B$ gives the set of elements of the bag.

theorem rule domNullBag $[X]$
$\mathrm{dom}[X, \mathbb{Z}][\![\,]\!] = \{\}$

theorem rule domUnitBag $[X]$
$\forall x : X \bullet \mathrm{dom}\, [\![x]\!] = \{x\}$

Automation

theorem grule bag_type
$\mathrm{bag}\, X \in \mathbb{P}(X \mathrel{+\!\!\!\rightarrow} \mathbb{N}_1)$

theorem rule bag_sub
$\mathrm{bag}\, X \in \mathbb{P}(\mathrm{bag}\, Y) \Leftrightarrow X \in \mathbb{P}\, Y$

theorem rule bag_ideal
$\mathbb{P}\, R \in \mathbb{P}(\mathrm{bag}\, X) \Leftrightarrow R \in \mathrm{bag}\, X$

theorem grule nullBagType
$[\![\,]\!] \in \mathrm{bag}\{\}$

14.1 Bag count

Definitions

syntax in *inrel* `\inbag`
syntax $\sharp$ *infun*5 `\bcount`

$$[X]$$

$$\_\ \mathrm{in}\ \_ : X \leftrightarrow \mathrm{bag}\, X$$
$$count : \mathrm{bag}\, X \rightarrow (X \rightarrow \mathbb{N})$$
$$\_ \sharp \_ : \mathrm{bag}\, X \times X \rightarrow \mathbb{N}$$

$\langle\!\langle$ disabled rule inbagDef $\rangle\!\rangle$
$$\forall x : X;\ B : \mathrm{bag}\, X \bullet x\ \mathrm{in}\ B \Leftrightarrow x \in \mathrm{dom}\, B$$

$\langle\!\langle$ rule countDef $\rangle\!\rangle$
$$\forall x : X;\ B : \mathrm{bag}\, X \bullet (count\ B)x = B \sharp x$$

$\langle\!\langle$ disabled rule bcountDef $\rangle\!\rangle$
$$\forall x : X;\ B : \mathrm{bag}\, X \bullet B \sharp x = \mathbf{if}\ x\ \mathrm{in}\ B\ \mathbf{then}\ B(x)\ \mathbf{else}\ 0$$

Theorems

theorem rule domCount $[X]$
$$\forall B : \mathrm{bag}\, X \bullet \mathrm{dom}(count\ B) = X$$

theorem rule inNullBag $[X]$
$$\neg\, x\ \mathrm{in}\ [X][\![\,]\!]$$

theorem rule inUnitBag $[X]$
$$x\ \mathrm{in}\ [\![y]\!] \Leftrightarrow x \in X \wedge x = y$$

theorem rule bcountNullBag $[X]$
$$\forall x : X \bullet [\![\,]\!] \sharp x = 0$$

theorem rule bcountUnitBag $[X]$
$$\forall x, y : X \bullet [\![x]\!] \sharp y = \mathbf{if}\ x = y\ \mathbf{then}\ 1\ \mathbf{else}\ 0$$

theorem bagExtensionality $[X]$
$$\forall A, B : \mathrm{bag}\, X \bullet A = B \Leftrightarrow (\forall x : X \bullet A \sharp x = B \sharp x)$$

14.2 Subbags

Definitions

syntax $\sqsubseteq$ *inrel* `\subbageq`

$$[X]$$

$$\_ \sqsubseteq \_ : \mathrm{bag}\, X \leftrightarrow \mathrm{bag}\, X$$

$$\langle\!\langle \text{disabled rule subbagDef} \rangle\!\rangle$$
$$\forall A, B : \mathrm{bag}\, X \bullet A \sqsubseteq B \Leftrightarrow (\forall x : X \bullet A \sharp x \leq B \sharp x)$$

Theorems

theorem rule nullBagSubbag $[X]$
$\forall B : \mathrm{bag}\, X \bullet [\![]\!] \sqsubseteq B$

theorem rule unitBagSubbag $[X]$
$\forall x : X;\ B : \mathrm{bag}\, X \bullet [\![x]\!] \sqsubseteq B \Leftrightarrow x \mathrel{\mathrm{in}} B$

theorem rule subbagSelf $[X]$
$\forall B : \mathrm{bag}\, X \bullet B \sqsubseteq B$

We need more rules about subbags, e.g., transitivity.

14.3 Bag scaling

Definitions

syntax $\otimes$ *infun*5 `\otimes`

$[X]$

$\_ \otimes \_ : \mathbb{N} \times \mathrm{bag}\, X \rightarrow \mathrm{bag}\, X$

$\langle\!\langle$ rule bcountBagScale $\rangle\!\rangle$
$\forall n : \mathbb{N};\ B : \mathrm{bag}\, X;\ x : X \bullet (n \otimes B) \sharp x = n * (B \sharp x)$

Theorems

theorem rule bagscaleBy0 $[X]$
$\forall B : \mathrm{bag}\, X \bullet 0 \otimes B = [\![\,]\!]$

theorem rule bagscaleBy1 $[X]$
$\forall B : \mathrm{bag}\, X \bullet 1 \otimes B = B$

theorem rule bagscaleNull $[X]$
$\forall n : \mathbb{N} \bullet n \otimes [X][\![\,]\!] = [\![\,]\!]$

theorem rule bagscalebagscale $[X]$
$\forall n, k : \mathbb{N};\ B : \mathrm{bag}\, X \bullet n \otimes (k \otimes B) = (n * k) \otimes B$

theorem rule domBagscale $[X]$
$\forall n : \mathbb{N}_1;\ B : \mathrm{bag}\, X \bullet \mathrm{dom}(n \otimes B) = \mathrm{dom}\, B$

theorem rule ranBagscale $[X]$
$\forall n : \mathbb{N}_1;\ B : \mathrm{bag}\, X \bullet \mathrm{ran}(n \otimes B) = \mathrm{ran}\, B$

theorem rule inbagscale $[X]$
$\forall x : X;\ n : \mathbb{N};\ B : \mathrm{bag}\, X \bullet x \mathrel{\mathrm{in}} n \otimes B \Leftrightarrow x \mathrel{\mathrm{in}} B \land \neg\, n = 0$

theorem rule bagscaleInBag $[X]$
$\forall n : \mathbb{N};\ B : \mathrm{bag}\, X;\ Y : \mathbb{P}\, X \bullet (n \otimes B) \in \mathrm{bag}\, Y \Leftrightarrow n = 0 \lor B \in \mathrm{bag}\, Y$

Automation

The following rules allow the computation of scaling of bags expressed by set comprehensions.

theorem rule bagScaleNullset $[X]$
$\forall n : \mathbb{N} \bullet n \otimes [X]\{\} = \{\}$

theorem rule bagScaleUnitSet $[X]$
$\forall n, k : \mathbb{N};\ x : X \bullet n \otimes \{(x, k)\} = \{(x, n * k)\}$

theorem rule bagScaleUnion $[X]$
$\forall S, T : \mathbb{P}(X \times \mathbb{Z}) \mid (S \cup T) \in \mathrm{bag}\, X \bullet n \otimes (S \cup T) = (n \otimes S) \cup (n \otimes T)$

14.4 Bag union

syntax $\uplus$ *infun*3 `\uplus`
Function $\_ \uplus \_$ is predefined.

Theorems

theorem rule domBagUnionFunction $[X]$
$\mathrm{bag}\, X \times \mathrm{bag}\, X \in \mathbb{P}\, A \land \mathrm{bag}\, X \in \mathbb{P}\, B \Rightarrow \mathrm{dom}[A, B](\_ \uplus \_)[X] = \mathrm{bag}\, X \times \mathrm{bag}\, X$

theorem rule ranBagUnionFunction $[X]$
$\mathrm{bag}\, X \times \mathrm{bag}\, X \in \mathbb{P}\, A \land \mathrm{bag}\, X \in \mathbb{P}\, B \Rightarrow \mathrm{ran}[A, B](\_ \uplus \_)[X] = \mathrm{bag}\, X$

theorem rule domBagUnion $[X]$
$\forall A, B : \mathrm{bag}\, X \bullet \mathrm{dom}(A \uplus B) = (\mathrm{dom}\, A) \cup (\mathrm{dom}\, B)$

theorem rule inBagUnion $[X]$
$\forall A, B : \mathrm{bag}\, X \bullet x \mathrel{\mathrm{in}} (A \uplus B) \Leftrightarrow (x \mathrel{\mathrm{in}} A) \lor (x \mathrel{\mathrm{in}} B)$

theorem rule countBagUnion $[X]$
$\forall A, B : \mathrm{bag}\, X;\ x : X \bullet (A \uplus B) \sharp x = A \sharp x + B \sharp x$

theorem rule bagUnionInBag $[X]$
$\forall Y : \mathbb{P}\, X;\ A, B : \mathrm{bag}\, X \bullet A \uplus B \in \mathrm{bag}\, Y \Leftrightarrow A \in \mathrm{bag}\, Y \land B \in \mathrm{bag}\, Y$

theorem rule bagUnionNullLeft $[X]$
$\forall B : \mathrm{bag}\, X \bullet [\![\,]\!] \uplus B = B$

theorem rule bagUnionNullRight $[X]$
$\forall B : \mathrm{bag}\, X \bullet B \uplus [\![\,]\!] = B$

theorem rule bagUnionCommutes $[X]$
$\forall A, B : \mathrm{bag}\, X \bullet A \uplus B = B \uplus A$

theorem rule bagUnionAssociates $[X]$
$\forall A, B, C : \mathrm{bag}\, X \bullet (A \uplus B) \uplus C = A \uplus (B \uplus C)$

theorem rule bagUnionPermutes $[X]$
$\forall A, B, C : \mathrm{bag}\, X \bullet A \uplus (B \uplus C) = B \uplus (A \uplus C)$

14.5 Bag difference

Definitions

syntax $\uminus$ *infun*3 `\uminus`

$$
\begin{array}{l}
[X] \\
\hline
\_ \uminus \_ : \mathrm{bag}\, X \times \mathrm{bag}\, X \rightarrow \mathrm{bag}\, X \\
\hline
\langle\!\langle \text{rule bcountUminus} \rangle\!\rangle \\
\forall A, B : \mathrm{bag}\, X;\; x : X \bullet (A \uminus B) \sharp x = max\{0, (A \sharp x) - (B \sharp x)\}
\end{array}
$$

Theorems

theorem rule inBagDifference $[X]$
$\forall A, B : \mathrm{bag}\, X;\; x : X \bullet x \mathrel{\mathrm{in}} (A \uminus B) \Leftrightarrow A \sharp x > B \sharp x$

theorem rule bagDifferenceNullLeft $[X]$
$\forall B : \mathrm{bag}\, X \bullet [\![]\!] \uminus B = [\![]\!]$

theorem rule bagDifferenceNullRight $[X]$
$\forall B : \mathrm{bag}\, X \bullet B \uminus [\![]\!] = B$

theorem rule bagDifferenceSubbag $[X]$
$\forall A, B, C : \mathrm{bag}\, X \bullet (A \uminus B) \sqsubseteq C \Leftrightarrow A \sqsubseteq B \uplus C$

14.6 Items

Definition

$$[X]$$
$$items : \mathrm{seq}\, X \rightarrow \mathrm{bag}\, X$$

Theorems

theorem rule itemsNullSeq $[X]$
$items[X]\langle\rangle = [\![\,]\!]$

theorem rule itemsUnitSeq $[X]$
$\forall x : X \bullet items[X]\langle x\rangle = [\![x]\!]$

theorem rule itemsCat $[X]$
$\forall s, t : \mathrm{seq}\, X \bullet items(s \frown t) = (items\ s) \uplus (items\ t)$

theorem rule inItems $[X]$
$\forall s : \mathrm{seq}\, X \bullet x \ \mathrm{in}\ (items\ s) \Leftrightarrow x \in \mathrm{ran}\, s$

theorem rule itemsInBag $[X]$
$\forall s : \mathrm{seq}\, X;\ Y : \mathbb{P}\, X \bullet items\ s \in \mathrm{bag}\, Y \Leftrightarrow s \in \mathrm{seq}\, Y$

theorem rule domItems $[X]$
$\forall s : \mathrm{seq}\, X \bullet \mathrm{dom}(items\ s) = \mathrm{ran}\, s$

theorem disabled rule countItems $[X]$
$\forall s : \mathrm{seq}\, X;\ x : X \bullet (items\ s) \sharp\, x = \#(s \rhd \{x\})$

References

[1] ISO SC22 Working Group 19. Z notation. Technical report, ISO/IEC JTC1/SC22 N1970, 1995. ISO CD 13568; Committee Draft of the proposed Z Standard.

[2] Irwin Meisels and Mark Saaltink. The Z/EVES 2.0 Reference Manual. Technical Report TR-99-5493-03e, ORA Canada, October 1999.

[3] J. M. Spivey. *The Z Notation: A Reference Manual*. Prentice Hall International Series in Computer Science, 2nd edition, 1992.