Cardinality of Set Partitions

Lukas Bulwahn

September 18, 2020

Abstract

The theory's main theorem states that the cardinality of set partitions of size k on a carrier set of size n is expressed by Stirling numbers of the second kind. In Isabelle, Stirling numbers of the second kind are defined in the AFP entry 'Discrete Summation' [1] through their well-known recurrence relation. The main theorem relates them to the alternative definition as cardinality of set partitions. The proof follows the simple and short explanation in Richard P. Stanley's 'Enumerative Combinatorics: Volume 1' [2] and Wikipedia [3], and unravels the full details and implicit reasoning steps of these explanations.

Contents

1 Set Partitions

theory *Set-Partition*
imports
HOL−Library.Disjoint-Sets

$HOL{-}Library.FuncSet$
begin

1.1 Useful Additions to Main Theories

lemma *set-eqI'*:
assumes $\bigwedge x.\ x \in A \Longrightarrow x \in B$
assumes $\bigwedge x.\ x \in B \Longrightarrow x \in A$
shows $A = B$
$\langle proof \rangle$

lemma *comp-image*:
$((`)\ f \circ (`)\ g) = (`)\ (f\ o\ g)$
$\langle proof \rangle$

1.2 Introduction and Elimination Rules

The definition of *partition-on* is in $HOL{-}Library.Disjoint\text{-}Sets$.

lemma *partition-onI*:
assumes $\bigwedge p.\ p \in P \Longrightarrow p \neq \{\}$
assumes $\bigcup P = A$
assumes $\bigwedge p\ p'.\ p \in P \Longrightarrow p' \in P \Longrightarrow p \neq p' \Longrightarrow p \cap p' = \{\}$
shows *partition-on* $A\ P$
$\langle proof \rangle$

lemma *partition-onE*:
assumes *partition-on* $A\ P$
obtains $\bigwedge p.\ p \in P \Longrightarrow p \neq \{\}$
$\bigcup P = A$
$\bigwedge p\ p'.\ p \in P \Longrightarrow p' \in P \Longrightarrow p \neq p' \Longrightarrow p \cap p' = \{\}$
$\langle proof \rangle$

1.3 Basic Facts on Set Partitions

lemma *partition-onD4*: *partition-on* $A\ P \Longrightarrow p \in P \Longrightarrow q \in P \Longrightarrow x \in p \Longrightarrow x \in q \Longrightarrow p = q$
$\langle proof \rangle$

lemma *partition-subset-imp-notin*:
assumes *partition-on* $A\ P\ X \in P$
assumes $X' \subset X$
shows $X' \notin P$
$\langle proof \rangle$

lemma *partition-on-Diff*:
assumes P: *partition-on* $A\ P$ **shows** $Q \subseteq P \Longrightarrow$ *partition-on* $(A - \bigcup Q)\ (P - Q)$
$\langle proof \rangle$

lemma *partition-on-UN*:
  assumes A: *partition-on* A B **and** B: $\bigwedge b.\ b \in B \Longrightarrow$ *partition-on* b $(P\ b)$
  shows *partition-on* A $(\bigcup b{\in}B.\ P\ b)$
⟨*proof*⟩

lemma *partition-on-notemptyI*:
  assumes *partition-on* A P
  assumes $A \neq \{\}$
  shows $P \neq \{\}$
⟨*proof*⟩

lemma *partition-on-disjoint*:
  assumes *partition-on* A P
  assumes *partition-on* B Q
  assumes $A \cap B = \{\}$
  shows $P \cap Q = \{\}$
⟨*proof*⟩

lemma *partition-on-eq-implies-eq-carrier*:
  assumes *partition-on* A Q
  assumes *partition-on* B Q
  shows $A = B$
⟨*proof*⟩

lemma *partition-on-insert*:
  assumes *partition-on* A P
  assumes *disjnt* A X $X \neq \{\}$
  assumes $A \cup X = A'$
  shows *partition-on* A' (*insert* X P)
⟨*proof*⟩

An alternative formulation of ⟦*partition-on* $?A$ $?P$; *disjnt* $?A$ $?X$; $?X \neq \{\}$; $?A \cup ?X = ?A'$⟧ $\Longrightarrow$ *partition-on* $?A'$ (*insert* $?X$ $?P$)

lemma *partition-on-insert'*:
  assumes *partition-on* $(A - X)$ P
  assumes $X \subseteq A$ $X \neq \{\}$
  shows *partition-on* A (*insert* X P)
⟨*proof*⟩

lemma *partition-on-insert-singleton*:
  assumes *partition-on* A P $a \notin A$ *insert* a $A = A'$
  shows *partition-on* A' (*insert* $\{a\}$ P)
⟨*proof*⟩

lemma *partition-on-remove-singleton*:
  assumes *partition-on* A P $X \in P$ $A - X = A'$
  shows *partition-on* A' $(P - \{X\})$
⟨*proof*⟩

1.4 The Unique Part Containing an Element in a Set Partition

lemma *partition-on-partition-on-unique*:
  assumes *partition-on* A P
  assumes $x \in A$
  shows $\exists! X.\ x \in X \wedge X \in P$
$\langle proof \rangle$

lemma *partition-on-the-part-mem*:
  assumes *partition-on* A P
  assumes $x \in A$
  shows $(THE\ X.\ x \in X \wedge X \in P) \in P$
$\langle proof \rangle$

lemma *partition-on-in-the-unique-part*:
  assumes *partition-on* A P
  assumes $x \in A$
  shows $x \in (THE\ X.\ x \in X \wedge X \in P)$
$\langle proof \rangle$

lemma *partition-on-the-part-eq*:
  assumes *partition-on* A P
  assumes $x \in X$ $X \in P$
  shows $(THE\ X.\ x \in X \wedge X \in P) = X$
$\langle proof \rangle$

lemma *the-unique-part-alternative-def*:
  assumes *partition-on* A P
  assumes $x \in A$
  shows $(THE\ X.\ x \in X \wedge X \in P) = \{y.\ \exists X \in P.\ x \in X \wedge y \in X\}$
$\langle proof \rangle$

lemma *partition-on-all-in-part-eq-part*:
  assumes *partition-on* A P
  assumes $X' \in P$
  shows $\{x \in A.\ (THE\ X.\ x \in X \wedge X \in P) = X'\} = X'$
$\langle proof \rangle$

lemma *partition-on-part-characteristic*:
  assumes *partition-on* A P
  assumes $X \in P$ $x \in X$
  shows $X = \{y.\ \exists X \in P.\ x \in X \wedge y \in X\}$
$\langle proof \rangle$

lemma *partition-on-no-partition-outside-carrier*:
  assumes *partition-on* A P
  assumes $x \notin A$
  shows $\{y.\ \exists X \in P.\ x \in X \wedge y \in X\} = \{\}$

⟨*proof*⟩

1.5 Cardinality of Parts in a Set Partition

lemma *partition-on-le-set-elements*:
  assumes *finite* A
  assumes *partition-on* A P
  shows *card* $P \leq$ *card* A
⟨*proof*⟩

1.6 Operations on Set Partitions

lemma *partition-on-union*:
  assumes $A \cap B = \{\}$
  assumes *partition-on* A P
  assumes *partition-on* B Q
  shows *partition-on* $(A \cup B)$ $(P \cup Q)$
⟨*proof*⟩

lemma *partition-on-split1*:
  assumes *partition-on* A $(P \cup Q)$
  shows *partition-on* $(\bigcup P)$ P
⟨*proof*⟩

lemma *partition-on-split2*:
  assumes *partition-on* A $(P \cup Q)$
  shows *partition-on* $(\bigcup Q)$ Q
⟨*proof*⟩

lemma *partition-on-intersect-on-elements*:
  assumes *partition-on* $(A \cup C)$ P
  assumes $\forall X \in P.\ \exists x.\ x \in X \cap C$
  shows *partition-on* C $((\lambda X.\ X \cap C)$ ` $P)$
⟨*proof*⟩

lemma *partition-on-insert-elements*:
  assumes $A \cap B = \{\}$
  assumes *partition-on* B P
  assumes $f \in A \rightarrow_E P$
  shows *partition-on* $(A \cup B)$ $((\lambda X.\ X \cup \{x \in A.\ f\,x = X\})$ ` $P)$ (**is** *partition-on* - *?P*)
⟨*proof*⟩

lemma *partition-on-map*:
  assumes *inj-on* f A
  assumes *partition-on* A P
  shows *partition-on* $(f$ ` $A)$ $(($`$)$ f ` $P)$
⟨*proof*⟩

lemma *set-of-partition-on-map*:

assumes *inj-on f A*
shows $(`)\ ((`)\ f)\ `\ \{P.\ \textit{partition-on}\ A\ P\} = \{P.\ \textit{partition-on}\ (f\ `\ A)\ P\}$
⟨*proof*⟩

end

2 Combinatorial Basics

theory *Injectivity-Solver*
imports
HOL−Library.Disjoint-Sets
HOL−Library.Monad-Syntax
HOL−Eisbach.Eisbach
begin

2.1 Preliminaries

These lemmas shall be added to the Disjoint Set theory.

2.1.1 Injectivity and Disjoint Families

lemma *inj-on-impl-disjoint-family-on-singleton*:
assumes *inj-on f A*
shows *disjoint-family-on* $(\lambda x.\ \{f\ x\})\ A$
⟨*proof*⟩

2.1.2 Cardinality Theorems for Set.bind

lemma *card-bind*:
assumes *finite S*
assumes $\forall X \in S.\ \textit{finite}\ (f\ X)$
assumes *disjoint-family-on f S*
shows $\textit{card}\ (S \ggg f) = (\sum x{\in}S.\ \textit{card}\ (f\ x))$
⟨*proof*⟩

lemma *card-bind-constant*:
assumes *finite S*
assumes $\forall X \in S.\ \textit{finite}\ (f\ X)$
assumes *disjoint-family-on f S*
assumes $\bigwedge x.\ x \in S \Longrightarrow \textit{card}\ (f\ x) = k$
shows $\textit{card}\ (S \ggg f) = \textit{card}\ S * k$
⟨*proof*⟩

lemma *card-bind-singleton*:
assumes *finite S*
assumes *inj-on f S*
shows $\textit{card}\ (S \ggg (\lambda x.\ \{f\ x\})) = \textit{card}\ S$
⟨*proof*⟩

2.2 Third Version of Injectivity Solver

Here, we provide a third version of the injectivity solver. The original first version was provided in the AFP entry 'Spivey's Generalized Recurrence for Bell Numbers'. From that method, I derived a second version in the AFP entry 'Cardinality of Equivalence Relations'. At roughly the same time, Makarius improved the injectivity solver in the development version of the first AFP entry. This third version now includes the improvements of the second version and Makarius improvements to the first, and it further extends the method to handle the new cases in the cardinality proof of this AFP entry.

As the implementation of the injectivity solver only evolves in the development branch of the AFP, the submissions of the three AFP entries that employ the injectivity solver, have to create clones of the injectivity solver for the identified and needed method adjustments. Ultimately, these three clones should only remain in the stable branches of the AFP from Isabelle2016 to Isabelle2017 to work with their corresponding release versions. In the development version, I have now consolidated the three versions here. In the next step, I will move this version of the injectivity solver in the *HOL−Library.Disjoint-Sets* and it will hopefully only evolve further there.

lemma *disjoint-family-onI*:
  assumes $\bigwedge i\ j.\ i \in I \wedge j \in I \Longrightarrow i \neq j \Longrightarrow (A\ i) \cap (A\ j) = \{\}$
  shows *disjoint-family-on* $A\ I$
⟨*proof*⟩

lemma *disjoint-bind*: $\bigwedge S\ T\ f\ g.\ (\bigwedge s\ t.\ S\ s \wedge T\ t \Longrightarrow f\ s \cap g\ t = \{\}) \Longrightarrow (\{s.\ S\ s\} \gg= f) \cap (\{t.\ T\ t\} \gg= g) = \{\}$
⟨*proof*⟩

lemma *disjoint-bind'*: $\bigwedge S\ T\ f\ g.\ (\bigwedge s\ t.\ s \in S \wedge t \in T \Longrightarrow f\ s \cap g\ t = \{\}) \Longrightarrow (S \gg= f) \cap (T \gg= g) = \{\}$
⟨*proof*⟩

lemma *injectivity-solver-CollectE*:
  assumes $a \in \{x.\ P\ x\} \wedge a' \in \{x.\ P'\ x\}$
  assumes $(P\ a \wedge P'\ a') \Longrightarrow W$
  shows W
⟨*proof*⟩

lemma *injectivity-solver-prep-assms-Collect*:
  assumes $x \in \{x.\ P\ x\}$
  shows $P\ x \wedge P\ x$
⟨*proof*⟩

lemma *injectivity-solver-prep-assms*: $x \in A \Longrightarrow x \in A \wedge x \in A$
  ⟨*proof*⟩

lemma *disjoint-terminal-singleton*: $\bigwedge s\ t\ X\ Y.\ s \neq t \Longrightarrow (X = Y \Longrightarrow s = t) \Longrightarrow$
$\{X\} \cap \{Y\} = \{\}$
⟨*proof*⟩

lemma *disjoint-terminal-Collect*:
  assumes $s \neq t$
  assumes $\bigwedge x\ x'.\ S\ x \wedge T\ x' \Longrightarrow x = x' \Longrightarrow s = t$
  shows $\{x.\ S\ x\} \cap \{x.\ T\ x\} = \{\}$
⟨*proof*⟩

lemma *disjoint-terminal*:
  $s \neq t \Longrightarrow (\bigwedge x\ x'.\ x \in S \wedge x' \in T \Longrightarrow x = x' \Longrightarrow s = t) \Longrightarrow S \cap T = \{\}$
⟨*proof*⟩

lemma *elim-singleton*:
  assumes $x \in \{s\} \wedge x' \in \{t\}$
  obtains $x = s \wedge x' = t$
⟨*proof*⟩

```
method injectivity-solver uses rule =
  insert method-facts,
  use nothing in ‹
   ((drule injectivity-solver-prep-assms-Collect | drule injectivity-solver-prep-assms)+)?;
    rule disjoint-family-onI;
    ((rule disjoint-bind | rule disjoint-bind')+)?;
    (erule elim-singleton)?;
   (erule disjoint-terminal-singleton | erule disjoint-terminal-Collect | erule disjoint-terminal);
    (elim injectivity-solver-CollectE)?;
    rule rule;
    assumption+
  ›
```

end

3 Cardinality of Set Partitions

theory *Card-Partitions*
imports
  HOL−Library.Stirling
  Set-Partition
  Injectivity-Solver
begin

lemma *set-partition-on-insert-with-fixed-card-eq*:
  assumes *finite A*
  assumes $a \notin A$
  shows $\{P.\ partition\text{-}on\ (insert\ a\ A)\ P \wedge card\ P = Suc\ k\} = (do\ \{$
    $P <- \{P.\ partition\text{-}on\ A\ P \wedge card\ P = Suc\ k\};$
    $p <- P;$

$\{insert\ (insert\ a\ p)\ (P - \{p\})\}$
$\})$
$\cup$ (*do* $\{$
P <− $\{P.\ partition\text{-}on\ A\ P \wedge card\ P = k\}$;
$\{insert\ \{a\}\ P\}$
$\})$ (**is** $?S = ?T$)
$\langle proof \rangle$

lemma *injectivity-subexpr1*:
assumes $a \notin A$
assumes $X \in P \wedge X' \in P'$
assumes $insert\ (insert\ a\ X)\ (P - \{X\}) = insert\ (insert\ a\ X')\ (P' - \{X'\})$
assumes $(partition\text{-}on\ A\ P \wedge card\ P = Suc\ k') \wedge (partition\text{-}on\ A\ P' \wedge card\ P' = Suc\ k')$
shows $P = P'$ **and** $X = X'$
$\langle proof \rangle$

lemma *injectivity-subexpr2*:
assumes $a \notin A$
assumes $insert\ \{a\}\ P = insert\ \{a\}\ P'$
assumes $(partition\text{-}on\ A\ P \wedge card\ P = k') \wedge partition\text{-}on\ A\ P' \wedge card\ P' = k'$
shows $P = P'$
$\langle proof \rangle$

theorem *card-partition-on*:
assumes *finite* A
shows $card\ \{P.\ partition\text{-}on\ A\ P \wedge card\ P = k\} = Stirling\ (card\ A)\ k$
$\langle proof \rangle$

theorem *card-partition-on-at-most-size*:
assumes *finite* A
shows $card\ \{P.\ partition\text{-}on\ A\ P \wedge card\ P \leq k\} = (\sum j \leq k.\ Stirling\ (card\ A)\ j)$
$\langle proof \rangle$

theorem *partition-on-size1*:
assumes *finite* A
shows $\{P.\ partition\text{-}on\ A\ P \wedge (\forall X \in P.\ card\ X = 1)\} = \{(\lambda a.\ \{a\})\ `\ A\}$
$\langle proof \rangle$

theorem *card-partition-on-size1*:
assumes *finite* A
shows $card\ \{P.\ partition\text{-}on\ A\ P \wedge (\forall X \in P.\ card\ X = 1)\} = 1$
$\langle proof \rangle$

lemma *card-partition-on-size1-eq-1*:
assumes *finite* A
assumes $card\ A \leq k$
shows $card\ \{P.\ partition\text{-}on\ A\ P \wedge card\ P \leq k \wedge (\forall X \in P.\ card\ X = 1)\} = 1$
$\langle proof \rangle$

lemma *card-partition-on-size1-eq-0*:
  assumes *finite A*
  assumes $k < card\ A$
  shows $card\ \{P.\ partition\text{-}on\ A\ P \wedge card\ P \leq k \wedge (\forall X{\in}P.\ card\ X = 1)\} = 0$
⟨*proof*⟩

end

References

[1] F. Haftmann. Discrete summation. *Archive of Formal Proofs*, Apr. 2014. http://isa-afp.org/entries/Discrete_Summation.shtml, Formal proof development.

[2] R. P. Stanley. *Enumerative Combinatorics: Volume 1*. Cambridge University Press, second edition, 2012.

[3] Wikipedia. Stirling numbers of the second kind — wikipedia, the free encyclopedia, 2015. https://en.wikipedia.org/w/index.php?title=Stirling_numbers_of_the_second_kind&oldid=693800357, [Online; accessed 12-December-2015].