Ramsey's Theorem

Tom Ridge

December 17, 2016

Abstract

The infinite form of Ramsey's Theorem is proved following Boolos and Jeffrey, Chapter 26.

Contents

1 Infinite Sets and Related Concepts **1**
1.1 Infinitely Many and Almost All 2
1.2 Enumeration of an Infinite Set 5

2 Ramsey's Theorem **6**
2.1 Library lemmas 6
2.2 Dependent Choice Variant 6
2.3 Partitions 7
2.4 Ramsey's theorem 7

1 Infinite Sets and Related Concepts

theory *Infinite-Set*
imports *Main*
begin

The set of natural numbers is infinite.

lemma *infinite-nat-iff-unbounded-le*: $infinite\ (S{::}nat\ set) \longleftrightarrow (\forall m.\ \exists n{\geq}m.\ n \in S)$
⟨*proof*⟩

lemma *infinite-nat-iff-unbounded*: $infinite\ (S{::}nat\ set) \longleftrightarrow (\forall m.\ \exists n{>}m.\ n \in S)$
⟨*proof*⟩

lemma *finite-nat-iff-bounded*: $finite\ (S{::}nat\ set) \longleftrightarrow (\exists k.\ S \subseteq \{..{<}k\})$
⟨*proof*⟩

lemma *finite-nat-iff-bounded-le*: $finite\ (S{::}nat\ set) \longleftrightarrow (\exists k.\ S \subseteq \{..\ k\})$
⟨*proof*⟩

lemma *finite-nat-bounded*: *finite* $(S::nat\ set) \Longrightarrow \exists\, k.\ S \subseteq \{..<k\}$
⟨*proof*⟩

For a set of natural numbers to be infinite, it is enough to know that for any number larger than some k, there is some larger number that is an element of the set.

lemma *unbounded-k-infinite*: $\forall\, m>k.\ \exists\, n>m.\ n \in S \Longrightarrow$ *infinite* $(S::nat\ set)$
⟨*proof*⟩

lemma *nat-not-finite*: *finite* $(UNIV::nat\ set) \Longrightarrow R$
⟨*proof*⟩

lemma *range-inj-infinite*:
inj $(f::nat \Rightarrow 'a) \Longrightarrow$ *infinite* $(range\ f)$
⟨*proof*⟩

The set of integers is also infinite.

lemma *infinite-int-iff-infinite-nat-abs*: *infinite* $(S::int\ set) \longleftrightarrow$ *infinite* $((nat\ o\ abs)\ `\ S)$
⟨*proof*⟩

proposition *infinite-int-iff-unbounded-le*: *infinite* $(S::int\ set) \longleftrightarrow (\forall\, m.\ \exists\, n.\ |n| \geq m \wedge n \in S)$
⟨*proof*⟩

proposition *infinite-int-iff-unbounded*: *infinite* $(S::int\ set) \longleftrightarrow (\forall\, m.\ \exists\, n.\ |n| > m \wedge n \in S)$
⟨*proof*⟩

proposition *finite-int-iff-bounded*: *finite* $(S::int\ set) \longleftrightarrow (\exists\, k.\ abs\ `\ S \subseteq \{..<k\})$
⟨*proof*⟩

proposition *finite-int-iff-bounded-le*: *finite* $(S::int\ set) \longleftrightarrow (\exists\, k.\ abs\ `\ S \subseteq \{..\ k\})$
⟨*proof*⟩

1.1 Infinitely Many and Almost All

We often need to reason about the existence of infinitely many (resp., all but finitely many) objects satisfying some predicate, so we introduce corresponding binders and their proof rules.

lemma *not-INFM* [*simp*]: $\neg\ (INFM\ x.\ P\ x) \longleftrightarrow (MOST\ x.\ \neg\ P\ x)$ ⟨*proof*⟩
lemma *not-MOST* [*simp*]: $\neg\ (MOST\ x.\ P\ x) \longleftrightarrow (INFM\ x.\ \neg\ P\ x)$ ⟨*proof*⟩

lemma *INFM-const* [*simp*]: $(INFM\ x::'a.\ P) \longleftrightarrow P \wedge$ *infinite* $(UNIV::'a\ set)$
⟨*proof*⟩

lemma *MOST-const* [*simp*]: $(MOST\ x::'a.\ P) \longleftrightarrow P \vee$ *finite* $(UNIV::'a\ set)$

$\langle proof \rangle$

lemma *INFM-imp-distrib*: $(INFM\ x.\ P\ x \longrightarrow Q\ x) \longleftrightarrow ((MOST\ x.\ P\ x) \longrightarrow (INFM\ x.\ Q\ x))$
$\langle proof \rangle$

lemma *MOST-imp-iff*: $MOST\ x.\ P\ x \Longrightarrow (MOST\ x.\ P\ x \longrightarrow Q\ x) \longleftrightarrow (MOST\ x.\ Q\ x)$
$\langle proof \rangle$

lemma *INFM-conjI*: $INFM\ x.\ P\ x \Longrightarrow MOST\ x.\ Q\ x \Longrightarrow INFM\ x.\ P\ x \wedge Q\ x$
$\langle proof \rangle$

Properties of quantifiers with injective functions.

lemma *INFM-inj*: $INFM\ x.\ P\ (f\ x) \Longrightarrow inj\ f \Longrightarrow INFM\ x.\ P\ x$
$\langle proof \rangle$

lemma *MOST-inj*: $MOST\ x.\ P\ x \Longrightarrow inj\ f \Longrightarrow MOST\ x.\ P\ (f\ x)$
$\langle proof \rangle$

Properties of quantifiers with singletons.

lemma *not-INFM-eq* [*simp*]:
$\neg\ (INFM\ x.\ x = a)$
$\neg\ (INFM\ x.\ a = x)$
$\langle proof \rangle$

lemma *MOST-neq* [*simp*]:
$MOST\ x.\ x \neq a$
$MOST\ x.\ a \neq x$
$\langle proof \rangle$

lemma *INFM-neq* [*simp*]:
$(INFM\ x{::}'a.\ x \neq a) \longleftrightarrow infinite\ (UNIV{::}'a\ set)$
$(INFM\ x{::}'a.\ a \neq x) \longleftrightarrow infinite\ (UNIV{::}'a\ set)$
$\langle proof \rangle$

lemma *MOST-eq* [*simp*]:
$(MOST\ x{::}'a.\ x = a) \longleftrightarrow finite\ (UNIV{::}'a\ set)$
$(MOST\ x{::}'a.\ a = x) \longleftrightarrow finite\ (UNIV{::}'a\ set)$
$\langle proof \rangle$

lemma *MOST-eq-imp*:
$MOST\ x.\ x = a \longrightarrow P\ x$
$MOST\ x.\ a = x \longrightarrow P\ x$
$\langle proof \rangle$

Properties of quantifiers over the naturals.

lemma *MOST-nat*: $(\forall_\infty n.\ P\ (n{::}nat)) \longleftrightarrow (\exists\, m.\ \forall\, n{>}m.\ P\ n)$
$\langle proof \rangle$

lemma *MOST-nat-le*: $(\forall_\infty n.\ P\ (n{::}nat)) \longleftrightarrow (\exists\, m.\ \forall\, n{\ge}m.\ P\ n)$
  ⟨*proof*⟩

lemma *INFM-nat*: $(\exists_\infty n.\ P\ (n{::}nat)) \longleftrightarrow (\forall\, m.\ \exists\, n{>}m.\ P\ n)$
  ⟨*proof*⟩

lemma *INFM-nat-le*: $(\exists_\infty n.\ P\ (n{::}nat)) \longleftrightarrow (\forall\, m.\ \exists\, n{\ge}m.\ P\ n)$
  ⟨*proof*⟩

lemma *MOST-INFM*: $infinite\ (UNIV{::}'a\ set) \Longrightarrow MOST\ x{::}'a.\ P\ x \Longrightarrow INFM\ x{::}'a.\ P\ x$
  ⟨*proof*⟩

lemma *MOST-Suc-iff*: $(MOST\ n.\ P\ (Suc\ n)) \longleftrightarrow (MOST\ n.\ P\ n)$
  ⟨*proof*⟩

lemma
  shows *MOST-SucI*: $MOST\ n.\ P\ n \Longrightarrow MOST\ n.\ P\ (Suc\ n)$
    and *MOST-SucD*: $MOST\ n.\ P\ (Suc\ n) \Longrightarrow MOST\ n.\ P\ n$
  ⟨*proof*⟩

lemma *MOST-ge-nat*: $MOST\ n{::}nat.\ m \le n$
  ⟨*proof*⟩

lemma *Inf-many-def*: $Inf\text{-}many\ P \longleftrightarrow infinite\ \{x.\ P\ x\}$ ⟨*proof*⟩
lemma *Alm-all-def*: $Alm\text{-}all\ P \longleftrightarrow \neg\ (INFM\ x.\ \neg\ P\ x)$ ⟨*proof*⟩
lemma *INFM-iff-infinite*: $(INFM\ x.\ P\ x) \longleftrightarrow infinite\ \{x.\ P\ x\}$ ⟨*proof*⟩
lemma *MOST-iff-cofinite*: $(MOST\ x.\ P\ x) \longleftrightarrow finite\ \{x.\ \neg\ P\ x\}$ ⟨*proof*⟩
lemma *INFM-EX*: $(\exists_\infty x.\ P\ x) \Longrightarrow (\exists\, x.\ P\ x)$ ⟨*proof*⟩
lemma *ALL-MOST*: $\forall\, x.\ P\ x \Longrightarrow \forall_\infty x.\ P\ x$ ⟨*proof*⟩
lemma *INFM-mono*: $\exists_\infty x.\ P\ x \Longrightarrow (\bigwedge x.\ P\ x \Longrightarrow Q\ x) \Longrightarrow \exists_\infty x.\ Q\ x$ ⟨*proof*⟩
lemma *MOST-mono*: $\forall_\infty x.\ P\ x \Longrightarrow (\bigwedge x.\ P\ x \Longrightarrow Q\ x) \Longrightarrow \forall_\infty x.\ Q\ x$ ⟨*proof*⟩
lemma *INFM-disj-distrib*: $(\exists_\infty x.\ P\ x \lor Q\ x) \longleftrightarrow (\exists_\infty x.\ P\ x) \lor (\exists_\infty x.\ Q\ x)$ ⟨*proof*⟩
lemma *MOST-rev-mp*: $\forall_\infty x.\ P\ x \Longrightarrow \forall_\infty x.\ P\ x \longrightarrow Q\ x \Longrightarrow \forall_\infty x.\ Q\ x$ ⟨*proof*⟩
lemma *MOST-conj-distrib*: $(\forall_\infty x.\ P\ x \land Q\ x) \longleftrightarrow (\forall_\infty x.\ P\ x) \land (\forall_\infty x.\ Q\ x)$ ⟨*proof*⟩
lemma *MOST-conjI*: $MOST\ x.\ P\ x \Longrightarrow MOST\ x.\ Q\ x \Longrightarrow MOST\ x.\ P\ x \land Q\ x$ ⟨*proof*⟩
lemma *INFM-finite-Bex-distrib*: $finite\ A \Longrightarrow (INFM\ y.\ \exists\, x{\in}A.\ P\ x\ y) \longleftrightarrow (\exists\, x{\in}A.\ INFM\ y.\ P\ x\ y)$ ⟨*proof*⟩
lemma *MOST-finite-Ball-distrib*: $finite\ A \Longrightarrow (MOST\ y.\ \forall\, x{\in}A.\ P\ x\ y) \longleftrightarrow (\forall\, x{\in}A.\ MOST\ y.\ P\ x\ y)$ ⟨*proof*⟩
lemma *INFM-E*: $INFM\ x.\ P\ x \Longrightarrow (\bigwedge x.\ P\ x \Longrightarrow thesis) \Longrightarrow thesis$ ⟨*proof*⟩
lemma *MOST-I*: $(\bigwedge x.\ P\ x) \Longrightarrow MOST\ x.\ P\ x$ ⟨*proof*⟩
lemmas *MOST-iff-finiteNeg* = *MOST-iff-cofinite*

1.2 Enumeration of an Infinite Set

The set's element type must be wellordered (e.g. the natural numbers).

Could be generalized to $enumerate'\ S\ n = (SOME\ t.\ t \in s \wedge finite\ \{s \in S.\ s < t\} \wedge card\ \{s \in S.\ s < t\} = n)$.

primrec (**in** *wellorder*) $enumerate :: 'a\ set \Rightarrow nat \Rightarrow 'a$
where
  enumerate-0: $enumerate\ S\ 0 = (LEAST\ n.\ n \in S)$
| *enumerate-Suc*: $enumerate\ S\ (Suc\ n) = enumerate\ (S - \{LEAST\ n.\ n \in S\})\ n$

lemma *enumerate-Suc'*: $enumerate\ S\ (Suc\ n) = enumerate\ (S - \{enumerate\ S\ 0\})\ n$
  ⟨*proof*⟩

lemma *enumerate-in-set*: $infinite\ S \Longrightarrow enumerate\ S\ n \in S$
  ⟨*proof*⟩

declare *enumerate-0* [*simp del*] *enumerate-Suc* [*simp del*]

lemma *enumerate-step*: $infinite\ S \Longrightarrow enumerate\ S\ n < enumerate\ S\ (Suc\ n)$
  ⟨*proof*⟩

lemma *enumerate-mono*: $m < n \Longrightarrow infinite\ S \Longrightarrow enumerate\ S\ m < enumerate\ S\ n$
  ⟨*proof*⟩

lemma *le-enumerate*:
  assumes S: $infinite\ S$
  shows $n \le enumerate\ S\ n$
  ⟨*proof*⟩

lemma *enumerate-Suc''*:
  fixes $S :: 'a::wellorder\ set$
  assumes $infinite\ S$
  shows $enumerate\ S\ (Suc\ n) = (LEAST\ s.\ s \in S \wedge enumerate\ S\ n < s)$
  ⟨*proof*⟩

lemma *enumerate-Ex*:
  assumes S: $infinite\ (S::nat\ set)$
  shows $s \in S \Longrightarrow \exists n.\ enumerate\ S\ n = s$
⟨*proof*⟩

lemma *bij-enumerate*:
  fixes $S :: nat\ set$
  assumes S: $infinite\ S$
  shows $bij\text{-}betw\ (enumerate\ S)\ UNIV\ S$
⟨*proof*⟩

A pair of weird and wonderful lemmas from HOL Light

lemma *finite-transitivity-chain*:
  assumes *finite* A
    and R: $\bigwedge x.\ \sim\ R\ x\ x\ \bigwedge x\ y\ z.\ [\![R\ x\ y;\ R\ y\ z]\!] \Longrightarrow R\ x\ z$
    and A: $\bigwedge x.\ x \in A \Longrightarrow \exists\, y.\ y \in A \wedge R\ x\ y$
  shows $A = \{\}$
⟨*proof*⟩

corollary *Union-maximal-sets*:
  assumes *finite* $\mathcal{F}$
  shows $\bigcup\{T \in \mathcal{F}.\ \forall\, U{\in}\mathcal{F}.\ \neg\ T \subset U\} = \bigcup\mathcal{F}$
  (**is** *?lhs* = *?rhs*)
⟨*proof*⟩

end

2 Ramsey's Theorem

theory *Ramsey*
imports *Main* ~~/*src*/*HOL*/*Library*/*Infinite-Set*
begin

declare [[*simp-depth-limit* = *5*]]

2.1 Library lemmas

lemma *infinite-inj-infinite-image*: *infinite* Z ==> *inj-on* f Z ==> *infinite* (f ‘ Z)
  ⟨*proof*⟩

lemma *infinite-dom-finite-rng*: [| *infinite* A; *finite* (f ‘ A) |] ==> ? b : f ‘ A. *infinite* $\{a : A.\ f\ a = b\}$
  ⟨*proof*⟩

lemma *infinite-mem*: *infinite* X ==> ? x. x : X
  ⟨*proof*⟩

lemma *not-empty-least*: (Y::*nat set*) ~= {} ==> ? m. m : Y & (! m'. m' : Y --> m <= m')
  ⟨*proof*⟩

2.2 Dependent Choice Variant

  —

primrec *choice* :: ('*a* => *bool*) => ('*a* => '*a* => *bool*) => *nat* => '*a* **where**
  choice P R *0* = (*SOME* x. P x)
| *choice* P R (*Suc* n) = (*let* x = *choice* P R n *in SOME* y. P y & R x y)
  —

lemma *dc*:
(! *x y z*. *R x y* & *R y z* --> *R x z*)
& (? *x0*. *P x0*)
& (! *x*. *P x* --> (? *y*. *P y* & *R x y*))
--> (? *f*::*nat*=>'*b*. (! *n*. *P* (*f n*)) & (! *n m*. *R* (*f n*) (*f* (*n*+*m*+*1*))))

⟨*proof*⟩

2.3 Partitions

definition
part :: *nat* => *nat* => '*a set* => ('*a set* => *nat*) => *bool* **where**
part r s Y f = (! *X*. *X* <= *Y* & *finite X* & *card X* = *r* --> *f X* < *s*)

lemma *part*: [| *infinite YY*; *part* (*Suc n*) *s YY f*; *yy* : *YY* |] ==> *part n s* (*YY* − {*yy*}) (%*u*. *f* (*insert yy u*))
⟨*proof*⟩

lemma *part-subset*: *part* (*Suc n*) *s YY f* ==> *Y* <= *YY* ==> *part* (*Suc n*) *s Y* *f*
⟨*proof*⟩

2.4 Ramsey's theorem

lemma *ramsey*:
! (*s*::*nat*) (*r*::*nat*) (*YY*::'*a set*) (*f*::'*a set* => *nat*).
infinite YY
& (! *X*. *X* <= *YY* & *finite X* & *card X* = *r* --> *f X* < *s*)
--> (? *Y' t'*.
Y' <= *YY*
& *infinite Y'*
& *t'* < *s*
& (! *X*. *X* <= *Y'* & *finite X* & *card X* = *r* --> *f X* = *t'*))
⟨*proof*⟩

end