First Welfare Theorem *

Julian Parsert Cezary Kaliszyk

December 7, 2022

Abstract

Contents

*This work is supported by the Austrian Science Fund (FWF) project P26201 and the European Research Council (ERC) grant no 714034 *SMART*.

1 Introducing Syntax

Syntax, abbreviations and type-synonyms

theory *Syntax*
imports *Main*
begin

type-synonym $'a\ relation = ('a \times 'a)\ set$

abbreviation *gen-weak-stx* :: $'a \Rightarrow 'a\ relation \Rightarrow 'a \Rightarrow bool$
(- ⪰[-] - [*51*,*100*,*51*] *60*)
where
$x \succeq[P]\ y \equiv (x,\ y) \in P$

abbreviation *gen-indif-stx* :: $'a \Rightarrow 'a\ relation \Rightarrow 'a \Rightarrow bool$
(- ≈[-] - [*51*,*100*,*51*] *60*)
where
$x \approx[P]\ y \equiv x \succeq[P]\ y \wedge y \succeq[P]\ x$

abbreviation *gen-strc-stx* :: $'a \Rightarrow 'a$ *relation* $\Rightarrow 'a \Rightarrow$ *bool*
(- ≻[-] - [*51*,*100*,*51*] *60*)
where
$x \succ[P]\ y \equiv\ x \succeq[P]\ y \wedge \neg y \succeq[P]\ x$

end

2 Arg Min and Arg Max sets

theory *Argmax*
imports
Complex-Main
begin

2.1 Definitions and Lemmas by Julian Parsert

definition of argmax and argmin returing a set.

definition *arg-min-set* :: $('a \Rightarrow 'b::ord) \Rightarrow 'a\ set \Rightarrow 'a\ set$
where
arg-min-set f S = $\{x.$ *is-arg-min f* $(\lambda x.\ x \in S)\ x\}$

definition *arg-max-set* :: $('a \Rightarrow 'b::ord) \Rightarrow 'a\ set \Rightarrow 'a\ set$
where
arg-max-set f S = $\{x.$ *is-arg-max f* $(\lambda x.\ x \in S)\ x\}$

Useful lemmas for *arg-max-set* and *arg-min-set*.

lemma *no-better-in-s*:
assumes $x \in$ *arg-max-set f S*
shows $\nexists y.\ y \in S \wedge (f\ y) > (f\ x)$
⟨*proof*⟩

lemma *argmax-sol-in-s*:
assumes $x \in$ *arg-max-set f S*
shows $x \in S$
⟨*proof*⟩

lemma *leq-all-in-sol*:
fixes $f :: 'a \Rightarrow ('b :: preorder)$
assumes $x \in$ *arg-max-set f S*
shows $\forall y \in S.\ f\ y \geq f\ x \longrightarrow y \in$ *arg-max-set f S*
⟨*proof*⟩

lemma *all-leq*:
fixes $f :: 'a \Rightarrow ('b :: linorder)$
assumes $x \in$ *arg-max-set f S*
shows $\forall y \in S.\ f\ x \geq f\ y$
⟨*proof*⟩

lemma *all-in-argmax-equal*:
  fixes $f :: 'a \Rightarrow ('b :: linorder)$
  assumes $x \in$ *arg-max-set f S*
  shows $\forall\, y \in$ *arg-max-set f S*. $f\ x = f\ y$
    $\langle proof \rangle$

end

3 Preference Relations

Preferences modeled as a set of pairs

theory *Preferences*
  imports
    HOL−Analysis.Multivariate-Analysis
    Syntax
begin

3.1 Basic Preference Relation

Basic preference relation locale with carrier and relation modeled as a set of pairs.

locale *preference* =
  fixes *carrier* :: *'a set*
  fixes *relation* :: *'a relation*
  assumes *not-outside*: $(x,y) \in relation \Longrightarrow x \in carrier$
    and $(x,y) \in relation \Longrightarrow y \in carrier$
  assumes *trans-refl*: *preorder-on carrier relation*

context *preference*
begin

abbreviation *geq* (- $\succeq$ - [51,51] 60)
  where
    $x \succeq y \equiv x \succeq[relation]\ y$

abbreviation *str-gr* (- $\succ$ - [51,51] 60)
  where
    $x \succ y \equiv x \succeq y \wedge \neg y \succeq x$

abbreviation *indiff* (- $\approx$ - [51,51] 60)
  where
    $x \approx y \equiv x \succeq y \wedge y \succeq x$

lemma *reflexivity*: *refl-on carrier relation*
  $\langle proof \rangle$

lemma *transitivity*: *trans relation*
⟨*proof*⟩

lemma *indiff-trans* [*simp*]: $x \approx y \Longrightarrow y \approx z \Longrightarrow x \approx z$
⟨*proof*⟩

end

3.1.1 Contour sets

definition *at-least-as-good* :: $'a \Rightarrow 'a$ *set* $\Rightarrow 'a$ *relation* $\Rightarrow 'a$ *set*
where
at-least-as-good $x\ B\ P = \{y \in B.\ y \succeq[P]\ x\ \}$

definition *no-better-than* :: $'a \Rightarrow 'a$ *set* $\Rightarrow 'a$ *relation* $\Rightarrow 'a$ *set*
where
no-better-than $x\ B\ P = \{y \in B.\ x \succeq[P]\ y\}$

definition *as-good-as* :: $'a \Rightarrow 'a$ *set* $\Rightarrow 'a$ *relation* $\Rightarrow 'a$ *set*
where
as-good-as $x\ B\ P = \{y \in B.\ x \approx[P]\ y\}$

lemma *at-lst-asgd-ge*:
assumes $x \in$ *at-least-as-good* $y\ B\ Pr$
shows $x \succeq[Pr]\ y$
⟨*proof*⟩

lemma *strict-contour-is-diff*:
$\{a \in B.\ a \succ[Pr]\ y\} =$ *at-least-as-good* $y\ B\ Pr\ -$ *as-good-as* $y\ B\ Pr$
⟨*proof*⟩

lemma *strict-countour-def* [*simp*]:
(*at-least-as-good* $y\ B\ Pr$) $-$ *as-good-as* $y\ B\ Pr = \{x \in B.\ x \succ[Pr]\ y\}$
⟨*proof*⟩

lemma *at-least-as-goodD* [*dest*]:
assumes $z \in$ *at-least-as-good* $y\ B\ Pr$
shows $z \succeq[Pr]\ y$
⟨*proof*⟩

3.2 Rational Preference Relation

Rational preferences add totality to the basic preferences.

locale *rational-preference* = *preference* +
assumes *total*: *total-on carrier relation*
begin

lemma *compl*: $\forall x \in$ *carrier* . $\forall y \in$ *carrier* . $x \succeq y \lor y \succeq x$
⟨*proof*⟩

lemma *strict-not-refl-weak* [*iff*]: $x \in carrier \wedge y \in carrier \Longrightarrow \neg (y \succeq x) \longleftrightarrow x \succ y$
⟨*proof*⟩

lemma *strict-trans* [*simp*]: $x \succ y \Longrightarrow y \succ z \Longrightarrow x \succ z$
⟨*proof*⟩

lemma *completeD* [*dest*]: $x \in carrier \Longrightarrow y \in carrier \Longrightarrow x \neq y \Longrightarrow x \succeq y \vee y \succeq x$
⟨*proof*⟩

lemma *pref-in-at-least-as*:
assumes $x \succeq y$
shows $x \in$ *at-least-as-good y carrier relation*
⟨*proof*⟩

lemma *worse-in-no-better*:
assumes $x \succeq y$
shows $y \in$ *no-better-than y carrier relation*
⟨*proof*⟩

lemma *strict-is-neg-transitive* :
assumes $x \in carrier \wedge y \in carrier \wedge z \in carrier$
shows $x \succ y \Longrightarrow x \succ z \vee z \succ y$
⟨*proof*⟩

lemma *weak-is-transitive*:
assumes $x \in carrier \wedge y \in carrier \wedge z \in carrier$
shows $x \succeq y \Longrightarrow y \succeq z \Longrightarrow x \succeq z$
⟨*proof*⟩

lemma *no-better-than-nonepty*:
assumes $carrier \neq \{\}$
shows $\bigwedge x.\ x \in carrier \Longrightarrow$ (*no-better-than x carrier relation*) $\neq \{\}$
⟨*proof*⟩

lemma *no-better-subset-pref* :
assumes $x \succeq y$
shows *no-better-than y carrier relation* $\subseteq$ *no-better-than x carrier relation*
⟨*proof*⟩

lemma *no-better-thansubset-rel* :
assumes $x \in carrier$ **and** $y \in carrier$
assumes *no-better-than y carrier relation* $\subseteq$ *no-better-than x carrier relation*
shows $x \succeq y$
⟨*proof*⟩

lemma *nbt-nest* :

shows (*no-better-than* y *carrier relation* $\subseteq$ *no-better-than* x *carrier relation*) $\vee$
(*no-better-than* x *carrier relation* $\subseteq$ *no-better-than* y *carrier relation*)
⟨*proof*⟩

lemma *at-lst-asgd-not-ge*:
assumes *carrier* $\neq \{\}$
assumes $x \in$ *carrier* **and** $y \in$ *carrier*
assumes $x \notin$ *at-least-as-good* y *carrier relation*
shows $\neg\, x \succeq y$
⟨*proof*⟩

lemma *as-good-as-sameIff*[*iff*]:
$z \in$ *as-good-as* y *carrier relation* $\longleftrightarrow z \succeq y \wedge y \succeq z$
⟨*proof*⟩

lemma *same-at-least-as-equal*:
assumes $z \approx y$
shows *at-least-as-good* z *carrier relation* $=$
at-least-as-good y *carrier relation* (**is** *?az* $=$ *?ay*)
⟨*proof*⟩

lemma *as-good-asIff* [*iff*]:
$x \in$ *as-good-as* y *carrier relation* $\longleftrightarrow x \approx$[*relation*] y
⟨*proof*⟩

lemma *nbt-subset*:
assumes *finite carrier*
assumes $x \in$ *carrier* **and** $y \in$ *carrier*
shows *no-better-than* x *carrier relation* $\subseteq$ *no-better-than* x *carrier relation* $\vee$
no-better-than x *carrier relation* $\subseteq$ *no-better-than* x *carrier relation*
⟨*proof*⟩

lemma *fnt-carrier-fnt-rel*: *finite carrier* $\Longrightarrow$ *finite relation*
⟨*proof*⟩

lemma *nbt-subset-carrier*:
assumes $x \in$ *carrier*
shows *no-better-than* x *carrier relation* $\subseteq$ *carrier*
⟨*proof*⟩

lemma *xy-in-eachothers-nbt*:
assumes $x \in$ *carrier* $y \in$ *carrier*
shows $x \in$ *no-better-than* y *carrier relation* $\vee$
$y \in$ *no-better-than* x *carrier relation*
⟨*proof*⟩

lemma *same-nbt-same-pref*:
assumes $x \in$ *carrier* $y \in$ *carrier*
shows $x \in$ *no-better-than* y *carrier relation* $\wedge$

$y \in$ *no-better-than x carrier relation* $\longleftrightarrow x \approx y$
⟨*proof*⟩

lemma *indifferent-imp-weak-pref*:
assumes $x \approx y$
shows $x \succeq y$ $y \succeq x$
⟨*proof*⟩

3.3 Finite carrier

context
assumes *finite carrier*
begin

lemma *fnt-carrier-fnt-nbt*:
shows $\forall x\in$*carrier*. *finite* (*no-better-than x carrier relation*)
⟨*proof*⟩

lemma *nbt-subset-imp-card-leq*:
assumes $x \in$ *carrier* **and** $y \in$ *carrier*
shows *no-better-than x carrier relation* $\subseteq$ *no-better-than y carrier relation* $\longleftrightarrow$
card (*no-better-than x carrier relation*) $\leq$ *card* (*no-better-than y carrier relation*)
(**is** *?nbt* $\longleftrightarrow$ *?card*)
⟨*proof*⟩

lemma *card-leq-pref*:
assumes $x \in$ *carrier* **and** $y \in$ *carrier*
shows *card* (*no-better-than x carrier relation*) $\leq$ *card* (*no-better-than y carrier relation*)
$\longleftrightarrow y \succeq x$
⟨*proof*⟩

lemma *finite-ne-remove-induct*:
assumes *finite B* $B \neq \{\}$
$\bigwedge A.$ *finite A* $\Longrightarrow A \subseteq B \Longrightarrow A \neq \{\} \Longrightarrow$
$(\bigwedge x.\ x \in A \Longrightarrow A - \{x\} \neq \{\} \Longrightarrow P\ (A - \{x\})) \Longrightarrow P\ A$
shows $P\ B$
⟨*proof*⟩

lemma *finite-nempty-preorder-has-max*:
assumes *finite B* $B \neq \{\}$ *refl-on B R trans R total-on B R*
shows $\exists x \in B.\ \forall y \in B.\ (x, y) \in R$
⟨*proof*⟩

lemma *finite-nempty-preorder-has-min*:
assumes *finite B* $B \neq \{\}$ *refl-on B R trans R total-on B R*
shows $\exists x \in B.\ \forall y \in B.\ (y, x) \in R$
⟨*proof*⟩

lemma *finite-nonempty-carrier-has-maximum*:
  assumes $carrier \neq \{\}$
  shows $\exists\, e \in carrier.\ \forall\, m \in carrier.\ e \succeq[relation]\ m$
  ⟨*proof*⟩

lemma *finite-nonempty-carrier-has-minimum*:
  assumes $carrier \neq \{\}$
  shows $\exists\, e \in carrier.\ \forall\, m \in carrier.\ m \succeq[relation]\ e$
  ⟨*proof*⟩

end

lemma *all-carrier-ex-sub-rel*:
  $\forall\, c \subseteq carrier.\ \exists\, r \subseteq relation.\ rational\text{-}preference\ c\ r$
⟨*proof*⟩

end

3.4 Local Non-Satiation

Defining local non-satiation.

definition *local-nonsatiation*
  where
  $local\text{-}nonsatiation\ B\ P \longleftrightarrow$
   $(\forall\, x{\in}B.\ \forall\, e{>}0.\ \exists\, y{\in}B.\ norm\ (y - x) \leq e \wedge y \succ[P]\ x)$

Alternate definitions and intro/dest rules with them

lemma *lns-alt-def1* [*iff*] :
  shows $local\text{-}nonsatiation\ B\ P \longleftrightarrow (\forall\, x \in B.\ \forall\, e{>}0.\ (\exists\, y \in B.\ dist\ y\ x \leq e \wedge y \succ[P]\ x))$
  ⟨*proof*⟩

lemma *lns-normI* [*intro*]:
  assumes $\bigwedge x\ e.\ x \in B \Longrightarrow e > 0 \Longrightarrow (\exists\, y{\in}B.\ norm\ (y - x) \leq e \wedge y \succ[P]\ x)$
  shows $local\text{-}nonsatiation\ B\ P$
  ⟨*proof*⟩

lemma *lns-distI* [*intro*]:
  assumes $\bigwedge x\ e.\ x \in B \Longrightarrow e > 0 \Longrightarrow (\exists\, y{\in}B.\ (dist\ y\ x) \leq e \wedge y \succ[P]\ x)$
  shows $local\text{-}nonsatiation\ B\ P$
  ⟨*proof*⟩

lemma *lns-alt-def2* [*iff*]:
  $local\text{-}nonsatiation\ B\ P \longleftrightarrow (\forall\, x{\in}B.\ \forall\, e{>}0.\ (\exists\, y.\ \ y \in (ball\ x\ e) \wedge y \in B \wedge y \succ[P]\ x))$
⟨*proof*⟩

lemma *lns-normD* [*dest*]:
assumes *local-nonsatiation B P*
shows $\forall x \in B.\ \forall e{>}0.\ \exists y \in B.\ (norm\ (y - x) \leq e \wedge y \succ[P]\ x)$
⟨*proof*⟩

3.5 Convex preferences

definition *weak-convex-pref* :: ('*a*::*real-vector*) *relation* ⇒ *bool*
where
$weak\text{-}convex\text{-}pref\ Pr \longleftrightarrow (\forall x\ y.\ x \succeq[Pr]\ y \longrightarrow$
$(\forall \alpha\ \beta.\ \alpha + \beta = 1 \wedge \alpha > 0 \wedge \beta > 0 \longrightarrow \alpha *_R x + \beta *_R y \succeq[Pr]\ y))$

definition *convex-pref* :: ('*a*::*real-vector*) *relation* ⇒ *bool*
where
$convex\text{-}pref\ Pr \longleftrightarrow (\forall x\ y.\ x \succ[Pr]\ y \longrightarrow$
$(\forall \alpha.\ 1 > \alpha \wedge \alpha > 0 \longrightarrow \alpha *_R x + (1{-}\alpha) *_R y \succ[Pr]\ y))$

definition *strict-convex-pref* :: ('*a*::*real-vector*) *relation* ⇒ *bool*
where
$strict\text{-}convex\text{-}pref\ Pr \longleftrightarrow (\forall x\ y.\ x \succeq[Pr]\ y \wedge x \neq y \longrightarrow$
$(\forall \alpha.\ 1 > \alpha \wedge \alpha > 0 \longrightarrow \alpha *_R x + (1{-}\alpha) *_R y \succ[Pr]\ y))$

lemma *convex-ge-imp-conved*:
assumes $\forall x\ y.\ x \succeq[Pr]\ y \longrightarrow (\forall \alpha\ \beta.\ \alpha + \beta = 1 \wedge \alpha \geq 0 \wedge \beta \geq 0 \longrightarrow \alpha *_R x + \beta *_R y \succeq[Pr]\ y)$
shows *weak-convex-pref Pr*
⟨*proof*⟩

lemma *weak-convexI* [*intro*]:
assumes $\bigwedge x\ y\ \alpha\ \beta.\ x \succeq[Pr]\ y \Longrightarrow \alpha + \beta = 1 \Longrightarrow 0 < \alpha \Longrightarrow 0 < \beta \Longrightarrow \alpha *_R x + \beta *_R y \succeq[Pr]\ y$
shows *weak-convex-pref Pr*
⟨*proof*⟩

lemma *weak-convexD* [*dest*]:
assumes *weak-convex-pref Pr* **and** $x \succeq[Pr]\ y$ **and** $0 < u$ **and** $0 < v$ **and** $u + v = 1$
shows $u *_R x + v *_R y \succeq[Pr]\ y$
⟨*proof*⟩

3.6 Real Vector Preferences

Preference relations on real vector type class.

locale *real-vector-rpr* = *rational-preference carrier relation*
for *carrier* :: '*a*::*real-vector set*
and *relation* :: '*a relation*

sublocale *real-vector-rpr* ⊆ *rational-preference carrier relation*
⟨*proof*⟩

context *real-vector-rpr*
begin

lemma *have-rpr*: *rational-preference carrier relation*
⟨*proof*⟩

Multiple convexity alternate definitions intro/dest rules.

lemma *weak-convex1D* [*dest*]:
assumes *weak-convex-pref relation* **and** $x \succeq[relation]\ y$ **and** $0 \leq u$ **and** $0 \leq v$ **and** $u + v = 1$
shows $u *_R x + v *_R y \succeq[relation]\ y$
⟨*proof*⟩

lemma *weak-convex1I* [*intro*] :
assumes $\forall x.$ *convex* (*at-least-as-good x carrier relation*)
shows *weak-convex-pref relation*
⟨*proof*⟩

Definition of convexity in "Handbook of Social Choice and Welfare"[1].

lemma *convex-def-alt*:
assumes *rational-preference carrier relation*
assumes *weak-convex-pref relation*
shows ($\forall x \in$ *carrier*. *convex* (*at-least-as-good x carrier relation*))
⟨*proof*⟩

lemma *convex-imp-convex-str-upper-cnt*:
assumes $\forall x \in$ *carrier*. *convex* (*at-least-as-good x carrier relation*)
shows *convex* (*at-least-as-good x carrier relation* − *as-good-as x carrier relation*)
(**is** *convex* (*?a* − *?b*))
⟨*proof*⟩

end

3.6.1 Monotone preferences

definition *weak-monotone-prefs* :: *'a set* ⇒ (*'a::ord*) *relation* ⇒ *bool*
where
weak-monotone-prefs B P ⟷ ($\forall x \in B.\ \forall y \in B.\ x \geq y \longrightarrow x \succeq[P] y$)

definition *monotone-preference* :: *'a set* ⇒ (*'a::ord*) *relation* ⇒ *bool*
where
monotone-preference B P ⟷ ($\forall x \in B.\ \forall y \in B.\ x > y \longrightarrow x \succ[P]\ y$)

Given a carrier set that is unbounded above (not the "standard" mathematical definition), monotonicity implies local non-satiation.

lemma *unbounded-above-mono-imp-lns*:
assumes $\forall M \in$ *carrier*. ($\forall x > M.\ x \in$ *carrier*)

assumes *mono*: *monotone-preference* (*carrier*:: $'a$::*ordered-euclidean-space set*) *relation*
shows *local-nonsatiation carrier relation*
⟨*proof*⟩

end

4 Utility Functions

Utility functions and results involving them.

theory *Utility-Functions*
imports
Preferences
begin

4.1 Ordinal utility functions

Ordinal utility function locale

locale *ordinal-utility* =
fixes *carrier* :: $'a$ *set*
fixes *relation* :: $'a$ *relation*
fixes *u* :: $'a \Rightarrow$ *real*
assumes *util-def*[*iff*]: $x \in carrier \Longrightarrow y \in carrier \Longrightarrow x \succeq[relation]\ y \longleftrightarrow u\ x \geq u\ y$
assumes *not-outside*: $x \succeq[relation]\ y \Longrightarrow x \in carrier$
and $x \succeq[relation]\ y \Longrightarrow y \in carrier$
begin

lemma *util-def-conf*: $x \in carrier \Longrightarrow y \in carrier \Longrightarrow u\ x \geq u\ y \longleftrightarrow x \succeq[relation]\ y$
⟨*proof*⟩

lemma *relation-subset-crossp*:
$relation \subseteq carrier \times carrier$
⟨*proof*⟩

Utility function implies totality of relation

lemma *util-imp-total*: *total-on carrier relation*
⟨*proof*⟩

lemma *x-y-in-carrier*: $x \succeq[relation]\ y \Longrightarrow x \in carrier \land y \in carrier$
⟨*proof*⟩

Utility function implies transitivity of relation.

lemma *util-imp-trans*: *trans relation*
⟨*proof*⟩

lemma *util-imp-refl*: *refl-on carrier relation*
⟨*proof*⟩

lemma *affine-trans-is-u*:
shows $\forall\, \alpha > 0.\ (\forall\, \beta.\ \textit{ordinal-utility}\ \ \textit{carrier relation}\ (\lambda x.\ u(x) * \alpha + \beta))$
⟨*proof*⟩

This utility function definition is ordinal. Hence they are only unique up to a monotone transformation.

lemma *ordinality-of-utility-function* :
fixes $f :: \textit{real} \Rightarrow \textit{real}$
assumes *monot*: $\textit{monotone}\ (>)\ (>)\ f$
shows $(f \circ u)\ x > (f \circ u)\ y \longleftrightarrow u\ x > u\ y$
⟨*proof*⟩

corollary *utility-prefs-corresp* :
fixes $f :: \textit{real} \Rightarrow \textit{real}$
assumes *monotonicity* : $\textit{monotone}\ (>)\ (>)\ f$
shows $\forall\, x \in \textit{carrier}.\ \forall\, y \in \textit{carrier}.\ (x,y) \in \textit{relation} \longleftrightarrow (f \circ u)\ x \geq (f \circ u)\ y$
⟨*proof*⟩

corollary *monotone-comp-is-utility*:
fixes $f :: \textit{real} \Rightarrow \textit{real}$
assumes *monot*: $\textit{monotone}\ (>)\ (>)\ f$
shows $\textit{ordinal-utility carrier relation}\ (f \circ u)$
⟨*proof*⟩

lemma *ordinal-utility-left*:
assumes $x \succeq[\textit{relation}]\ y$
shows $u\ x \geq u\ y$
⟨*proof*⟩

lemma *add-right*:
assumes $\bigwedge x\ y.\ x \succeq[\textit{relation}]\ y \Longrightarrow f\ x \geq f\ y$
shows $\textit{ordinal-utility carrier relation}\ (\lambda x.\ u\ x + f\ x)$
⟨*proof*⟩

lemma *add-left*:
assumes $\bigwedge x\ y.\ x \succeq[\textit{relation}]\ y \Longrightarrow f\ x \geq f\ y$
shows $\textit{ordinal-utility carrier relation}\ (\lambda x.\ f\ x + u\ x)$
⟨*proof*⟩

lemma *ordinal-utility-scale-transl*:
assumes $(c::\textit{real}) > 0$
shows $\textit{ordinal-utility carrier relation}\ (\lambda x.\ c * (u\ x) + d)$
⟨*proof*⟩

lemma *strict-prefernce-iff-strict-utility*:

assumes $x \in carrier$
assumes $y \in carrier$
shows $x \succ[relation]\ y \longleftrightarrow u\ x > u\ y$
⟨*proof*⟩

end

A utility function implies a rational preference relation. Hence a utility function contains exactly the same amount of information as a RPR

sublocale *ordinal-utility* ⊆ *rational-preference carrier relation*
⟨*proof*⟩

Given a finite carrier set. We can guarantee that given a rational preference relation, there must also exist a utility function representing this relation. Construction of witness roughly follows from.

theorem *fnt-carrier-exists-util-fun*:
assumes *finite carrier*
assumes *rational-preference carrier relation*
shows $\exists\, u.$ *ordinal-utility carrier relation u*
⟨*proof*⟩

corollary *obt-u-fnt-carrier*:
assumes *finite carrier*
assumes *rational-preference carrier relation*
obtains *u* **where** *ordinal-utility carrier relation u*
⟨*proof*⟩

theorem *ordinal-util-imp-rat-prefs*:
assumes *ordinal-utility carrier relation u*
shows *rational-preference carrier relation*
⟨*proof*⟩

4.2 Utility function on Euclidean Space

locale *eucl-ordinal-utility* = *ordinal-utility carrier relation u*
for *carrier* :: ($'a$::*euclidean-space*) *set*
and *relation* :: $'a$ *relation*
and *u* :: $'a \Rightarrow real$

sublocale *eucl-ordinal-utility* ⊆ *rational-preference carrier relation*
⟨*proof*⟩

lemma *ord-eucl-utility-imp-rpr*: *eucl-ordinal-utility s rel u* ⟶ *real-vector-rpr s rel*
⟨*proof*⟩

context *eucl-ordinal-utility*
begin

Local non-satiation on utility functions

lemma *lns-pref-lns-util* [*iff*]:
local-nonsatiation carrier relation $\longleftrightarrow$
($\forall$ *x*$\in$*carrier*. $\forall$ *e* > *0*. $\exists$ *y*$\in$*carrier*.
norm (*y* − *x*) $\leq$ *e* $\wedge$ *u y* > *u x*) (**is** - $\longleftrightarrow$ *?alt*)
⟨*proof*⟩

end

lemma *finite-carrier-rpr-iff-u*:
assumes *finite carrier*
and (*relation*::'*a relation*) $\subseteq$ *carrier* $\times$ *carrier*
shows *rational-preference carrier relation* $\longleftrightarrow$ ($\exists$ *u*. *ordinal-utility carrier relation u*)
⟨*proof*⟩

end

5 Consumers

Consumption sets

theory *Consumers*
imports
HOL−Analysis.Multivariate-Analysis
../Syntax
begin

5.1 Pre Arrow-Debreu consumption set

It turns out that the First Welfare Theorem does not require any particular limitations on the consumption set

locale *pre-arrow-debreu-consumption-set* =
fixes *consumption-set* :: ('*a*::*euclidean-space*) *set*
assumes *x* $\in$ (*UNIV*:: '*a set*) $\Longrightarrow$ *x* $\in$ *consumption-set*
begin
end

5.2 Arrow-Debreu model consumption set

The Arrow-Debreu model consumption set includes more and stricter assumptions which are necessary for further results.

locale *gen-pre-arrow-debreu-consum-set* =
fixes *consumption-set* :: ('*a*::*ordered-euclidean-space*) *set*
begin

end

locale *arrow-debreu-consum-set* =
  fixes *consumption-set* :: ($'a$::*ordered-euclidean-space*) *set*
  assumes *r-plus*: *consumption-set* $\subseteq \{(x::'a).\ x \geq 0\}$
  assumes *closed*: *closed consumption-set*
  assumes *convex*: *convex consumption-set*
  assumes *non-empty*: *consumption-set* $\neq \{\}$
  assumes $\forall M \in$ *consumption-set*. ($\forall x > M.\ x \in$ *consumption-set*)
begin

lemma *x-larger-0*: $x \in$ *consumption-set* $\Longrightarrow x \geq 0$
  ⟨*proof*⟩

lemma *larger-in-consump-set*:
  $x \in$ *consumption-set* $\wedge\ y \geq x \Longrightarrow y \in$ *consumption-set*
  ⟨*proof*⟩

end

end

theory *Common*
  imports
    ../*Preferences*
    ../*Utility-Functions*
    ../*Argmax*
begin

6 Pareto Ordering

Allows us to define a Pareto Ordering.

locale *pareto-ordering* =
  fixes *agents* :: $'i$ *set*
  fixes U :: $'i \Rightarrow 'a \Rightarrow$ *real*
begin
notation U (U[-])

definition *pareto-dominating* (**infix** $\succ$*Pareto* *60*)
  where
    $X \succ$*Pareto* $Y \longleftrightarrow$
      ($\forall i \in$ *agents*. $U[i]\ (X\ i) \geq U[i]\ (Y\ i)$) $\wedge$
      ($\exists i \in$ *agents*. $U[i]\ (X\ i) > U[i]\ (Y\ i)$)

lemma *trans-strict-pareto*: $X \succ$*Pareto* $Y \Longrightarrow Y \succ$*Pareto* $Z \Longrightarrow X \succ$*Pareto* Z

⟨*proof*⟩

lemma *anti-sym-strict-pareto*: $X \succ Pareto\ Y \Longrightarrow \neg Y \succ Pareto\ X$
⟨*proof*⟩

end

6.1 Budget constraint

Definition returns all afforedable bundles given wealth W

f is a function that computes the value given a bundle

definition *budget-constraint*
where
$budget\text{-}constraint\ f\ S\ W = \{x \in S.\ f\ x \leq W\}$

6.2 Feasiblity

definition *feasible-private-ownership*
where
$feasible\text{-}private\text{-}ownership\ A\ F\ \mathcal{E}\ Cs\ Ps\ X\ Y \longleftrightarrow$
$(\sum i{\in}A.\ X\ i) \leq (\sum i{\in}A.\ \mathcal{E}\ i) + (\sum j{\in}F.\ Y\ j) \wedge$
$(\forall i{\in}A.\ X\ i \in Cs) \wedge (\forall j{\in}F.\ Y\ j \in Ps\ j)$

lemma *feasible-private-ownershipD*:
assumes $feasible\text{-}private\text{-}ownership\ A\ F\ \mathcal{E}\ Cs\ Ps\ X\ Y$
shows $(\sum i{\in}A.\ X\ i) \leq (\sum i{\in}A.\ \mathcal{E}\ i) + (\sum j{\in}F.\ Y\ j)$
and $(\forall i{\in}A.\ X\ i \in Cs)$ **and** $(\forall j{\in}F.\ Y\ j \in Ps\ j)$
⟨*proof*⟩

end

theory *Exchange-Economy*
imports
../Preferences
../Utility-Functions
../Argmax
Consumers
Common
begin

7 Exchange Economy

Define the exchange economy model

locale *exchange-economy* =

fixes *consumption-set* :: ('*a*::*ordered-euclidean-space*) *set*
fixes *agents* :: '*i set*
fixes $\mathcal{E}$ :: '*i* ⇒ '*a*
fixes *Pref* :: '*i* ⇒ '*a relation*
fixes *U* :: '*i* ⇒ '*a* ⇒ *real*
assumes *cons-set-props*: *pre-arrow-debreu-consumption-set consumption-set*
assumes *agent-props*: *i* ∈ *agents* ⟹ *eucl-ordinal-utility consumption-set* (*Pref* *i*) (*U i*)
assumes *finite-agents*: *finite agents* **and** *agents* ≠ {}

sublocale *exchange-economy* ⊆ *pareto-ordering agents U*
⟨*proof*⟩

context *exchange-economy*
begin

context
begin

notation *U* (*U*[-])
notation *Pref* (*Pr*[-])
notation $\mathcal{E}$ ($\mathcal{E}$[-])

lemma *base-pref-is-ord-eucl-rpr*: *i* ∈ *agents* ⟹ *rational-preference consumption-set* *Pr*[*i*]
⟨*proof*⟩ **abbreviation** *calculate-value*
where
calculate-value P x ≡ *P* · *x*

7.1 Feasibility

definition *feasible-allocation*
where
feasible-allocation A E ⟷
($\sum$ *i*∈*agents*. *A i*) ≤ ($\sum$ *i*∈*agents*. *E i*)

7.2 Pareto optimality

definition *pareto-optimal-endow*
where
pareto-optimal-endow X E ⟷
(*feasible-allocation X E* ∧
(∄ *X'*. *feasible-allocation X' E* ∧ *X'* ≻*Pareto X*))

7.3 Competitive Equilibrium in Exchange Economy

Competitive Equilibirum or Walrasian Equilibrium definition.

definition *comp-equilib-endow*
where

comp-equilib-endow P X E $\equiv$
feasible-allocation X E $\wedge$
$(\forall\, i \in$ *agents*. $X\ i \in$ *arg-max-set* $U[i]$
(*budget-constraint* (*calculate-value* P) *consumption-set* $(P \cdot E\ i)))$

7.4 Lemmas for final result

lemma *utility-function-def* [*iff*]:
assumes $i \in$ *agents*
shows $U[i]\ x \geq U[i]\ y \longleftrightarrow x \succeq[Pr[i]]\ y$
⟨*proof*⟩

lemma *budget-constraint-is-feasible*:
assumes $i \in$ *agents*
assumes $X \in$ (*budget-constraint* (*calculate-value* P) *consumption-set* $(P \cdot \mathcal{E}[i])$)
shows $P \cdot X \leq P \cdot \mathcal{E}[i]$
⟨*proof*⟩

lemma *arg-max-set-therefore-no-better* :
assumes $i \in$ *agents*
assumes $x \in$ *arg-max-set* $U[i]$ (*budget-constraint* (*calculate-value* P) *consumption-set* $(P \cdot \mathcal{E}[i])$)
shows $U[i]\ y > U[i]\ x \longrightarrow y \notin$ *budget-constraint* (*calculate-value* P) *consumption-set* $(P \cdot \mathcal{E}[i])$
⟨*proof*⟩

Since we need no restriction on the consumption set for the First Welfare Theorem

lemma *consumption-set-member*: $\forall\, x.\ x \in$ *consumption-set*
⟨*proof*⟩

Under the assumption of Local non-satiation, agents will utilise their entire budget.

lemma *argmax-entire-budget* :
assumes $i \in$ *agents*
assumes *local-nonsatiation consumption-set* $Pr[i]$
assumes $X \in$ *arg-max-set* $U[i]$ (*budget-constraint* (*calculate-value* P) *consumption-set* $(P \cdot \mathcal{E}[i])$)
shows $P \cdot X = P \cdot \mathcal{E}[i]$
⟨*proof*⟩

All bundles that would be strictly preferred to any argmax result, are more expensive.

lemma *pref-more-expensive*:
assumes $i \in$ *agents*
assumes $x \in$ *arg-max-set* $U[i]$ (*budget-constraint* (*calculate-value* P) *consumption-set* $(P \cdot \mathcal{E}[i])$)
assumes $U[i]\ y > U[i]\ x$

shows $y \cdot P > P \cdot \mathcal{E}[i]$
⟨*proof*⟩

Greater or equal utility implies greater or equal price.

lemma *same-util-is-equal-or-more-expensive*:
assumes $i \in agents$
assumes *local-nonsatiation consumption-set* $Pr[i]$
assumes $x \in$ *arg-max-set* $U[i]$ (*budget-constraint* (*calculate-value* P) *consumption-set* ($P \cdot \mathcal{E}[i]$))
assumes $U[i]\ y \geq U[i]\ x$
shows $y \cdot P \geq P \cdot \mathcal{E}[i]$
⟨*proof*⟩

lemma *all-in-argmax-same-price*:
assumes $i \in agents$
assumes *local-nonsatiation consumption-set* $Pr[i]$
assumes $x \in$ *arg-max-set* $U[i]$ (*budget-constraint* (*calculate-value* P) *consumption-set* ($P \cdot \mathcal{E}[i]$))
and $y \in$ *arg-max-set* $U[i]$ (*budget-constraint* (*calculate-value* P) *consumption-set* ($P \cdot \mathcal{E}[i]$))
shows $P \cdot x = P \cdot y$
⟨*proof*⟩

All rationally acting agents (which is every agent by assumption) will not decrease his utility

lemma *individual-rationalism* :
assumes *comp-equilib-endow* $P\ X\ \mathcal{E}$
shows $\forall\, i \in agents.\ X\ i \succeq[Pref\ i]\ \mathcal{E}[i]$
⟨*proof*⟩

lemma *walras-law-per-agent* :
assumes $\bigwedge i.\ i \in agents \Longrightarrow$ *local-nonsatiation consumption-set* $Pr[i]$
assumes *comp-equilib-endow* $P\ X\ \mathcal{E}$
shows $\forall\, i \in agents.\ P \cdot X\ i = P \cdot \mathcal{E}[i]$
⟨*proof*⟩

Walras Law holds in our Exchange Economy model. It states that in an equilibrium, demand equals supply

lemma *walras-law*:
assumes $\bigwedge i.\ i \in agents \Longrightarrow$ *local-nonsatiation consumption-set* $Pr[i]$
assumes *comp-equilib-endow* $P\ X\ \mathcal{E}$
shows $(\sum i{\in}agents.\ P \cdot (X\ i)) - (\sum i{\in}agents.\ P \cdot \mathcal{E}[i]) = 0$
⟨*proof*⟩

lemma *inner-with-ge-0*: $(P::(real,\ 'n::finite)\ vec) > 0 \Longrightarrow A \geq B \Longrightarrow\ P \cdot A \geq P \cdot B$
⟨*proof*⟩

7.5 First Welfare Theorem in Exchange Economy

We prove the first welfare theorem in our Exchange Economy model.

theorem *first-welfare-theorem-exchange*:
assumes *lns* : $\bigwedge i.\ i \in agents \Longrightarrow local\text{-}nonsatiation\ consumption\text{-}set\ Pr[i]$
and *price-cond*: $Price > 0$
assumes *equilibrium* : $comp\text{-}equilib\text{-}endow\ Price\ \mathcal{X}\ \mathcal{E}$
shows $pareto\text{-}optimal\text{-}endow\ \mathcal{X}\ \mathcal{E}$
⟨*proof*⟩

Monotone preferences can be used instead of local non-satiation. Many textbooks etc. do not introduce the concept of local non-satiation and use monotonicity instead.

corollary *first-welfare-exch-thm-monot*:
assumes $\forall M \in carrier.\ (\forall x > M.\ x \in carrier)$
assumes $\bigwedge i.\ i \in agents \Longrightarrow monotone\text{-}preference\ consumption\text{-}set\ Pr[i]$
and *price-cond*: $Price > 0$
assumes $comp\text{-}equilib\text{-}endow\ Price\ \mathcal{X}\ \mathcal{E}$
shows $pareto\text{-}optimal\text{-}endow\ \mathcal{X}\ \mathcal{E}$
⟨*proof*⟩

end

end

end

8 Pre Arrow-Debreu model

Model similar to Arrow-Debreu model but with fewer assumptions, since we only need assumptions strong enough to proof the First Welfare Theorem.

theory *Private-Ownership-Economy*
imports
../Preferences
../Preferences
../Utility-Functions
../Argmax
Consumers
Common
begin

locale *pre-arrow-debreu-model* =
fixes *production-sets* :: $'f \Rightarrow ('a{::}ordered\text{-}euclidean\text{-}space)\ set$
fixes *consumption-set* :: $'a\ set$
fixes *agents* :: $'i\ set$
fixes *firms* :: $'f\ set$

fixes $\mathcal{E}$ :: $'i \Rightarrow 'a$ ($\mathcal{E}$[-])
fixes *Pref* :: $'i \Rightarrow 'a$ *relation* (*Pr*[-])
fixes *U* :: $'i \Rightarrow 'a \Rightarrow$ *real* (*U*[-])
fixes Θ :: $'i \Rightarrow 'f \Rightarrow$ *real* (Θ[-,-])
assumes *cons-set-props*: *pre-arrow-debreu-consumption-set consumption-set*
assumes *agent-props*: $i \in$ *agents* $\Longrightarrow$ *eucl-ordinal-utility consumption-set* (*Pr*[*i*]) (*U*[*i*])
assumes *firms-comp-owned*: $j \in$ *firms* $\Longrightarrow$ ($\sum i{\in}agents.\ \Theta[i,j]$) = *1*
assumes *finite-nonepty-agents*: *finite agents* **and** *agents* $\neq$ {}

sublocale *pre-arrow-debreu-model* $\subseteq$ *pareto-ordering agents U*
⟨*proof*⟩

context *pre-arrow-debreu-model*
begin

No restrictions on consumption set needed

lemma *all-larger-zero-in-csset*: $\forall x.\ x \in$ *consumption-set*
⟨*proof*⟩

context
begin

Calculate wealth of individual i in context of Private Ownership economy.

private abbreviation *poe-wealth*
where
poe-wealth P i Y $\equiv P \cdot \mathcal{E}[i] + (\sum j{\in}firms.\ \Theta[i,j] *_R (P \cdot Y\ j))$

8.1 Feasiblity

private abbreviation *feasible*
where
feasible X Y $\equiv$ *feasible-private-ownership agents firms* $\mathcal{E}$ *consumption-set production-sets X Y*

private abbreviation *calculate-value*
where
calculate-value P x $\equiv P \cdot x$

8.2 Profit maximisation

In a production economy we need to specify profit maximisation.

definition *profit-maximisation*
where
profit-maximisation P S = *arg-max-set* ($\lambda x.\ P \cdot x$) *S*

8.3 Competitive Equilibrium

Competitive equilibrium in context of production economy with private ownership. This includes the profit maximisation condition.

definition *competitive-equilibrium*
where
$competitive\text{-}equilibrium\ P\ X\ Y \longleftrightarrow feasible\ X\ Y \wedge$
$(\forall j \in firms.\ (Y\ j) \in profit\text{-}maximisation\ P\ (production\text{-}sets\ j)) \wedge$
$(\forall i \in agents.\ (X\ i) \in arg\text{-}max\text{-}set\ U[i]\ (budget\text{-}constraint\ (calculate\text{-}value\ P)\ consumption\text{-}set\ (poe\text{-}wealth\ P\ i\ Y)))$

lemma *competitive-equilibriumD* [*dest*]:
assumes $competitive\text{-}equilibrium\ P\ X\ Y$
shows $feasible\ X\ Y \wedge$
$(\forall j \in firms.\ (Y\ j) \in profit\text{-}maximisation\ P\ (production\text{-}sets\ j)) \wedge$
$(\forall i \in agents.\ (X\ i) \in arg\text{-}max\text{-}set\ U[i]\ (budget\text{-}constraint\ (calculate\text{-}value\ P)\ consumption\text{-}set\ (poe\text{-}wealth\ P\ i\ Y)))$
$\langle proof \rangle$

lemma *compet-max-profit*:
assumes $j \in firms$
assumes $competitive\text{-}equilibrium\ P\ X\ Y$
shows $Y\ j \in profit\text{-}maximisation\ P\ (production\text{-}sets\ j)$
$\langle proof \rangle$

8.4 Pareto Optimality

definition *pareto-optimal*
where
$pareto\text{-}optimal\ X\ Y \longleftrightarrow$
$(feasible\ X\ Y \wedge$
$(\nexists X'\ Y'.\ feasible\ X'\ Y' \wedge X' \succ Pareto\ X))$

lemma *pareto-optimalI* [*intro*]:
assumes $feasible\ X\ Y$
and $\nexists X'\ Y'.\ feasible\ X'\ Y' \wedge X' \succ Pareto\ X$
shows $pareto\text{-}optimal\ X\ Y$
$\langle proof \rangle$

lemma *pareto-optimalD*[*dest*]:
assumes $pareto\text{-}optimal\ X\ Y$
shows $feasible\ X\ Y$ **and** $\nexists X'\ Y'.\ feasible\ X'\ Y' \wedge X' \succ Pareto\ X$
$\langle proof \rangle$

lemma *util-fun-def-holds*: $i \in agents \Longrightarrow x \succeq[Pr[i]]\ y \longleftrightarrow U[i]\ x \geq U[i]\ y$
$\langle proof \rangle$

lemma *base-pref-is-ord-eucl-rpr*: $i \in agents \Longrightarrow rational\text{-}preference\ consumption\text{-}set\ Pr[i]$

⟨*proof*⟩

lemma *prof-max-ge-all-in-pset*:
assumes $j \in$ *firms*
assumes $Y\ j \in$ *profit-maximisation* P (*production-sets* j)
shows $\forall\, y \in$ *production-sets* $j.\ P \cdot Y\ j \geq P \cdot y$
⟨*proof*⟩

8.5 Lemmas for final result

Strictly preferred bundles are strictly more expensive.

lemma *all-prefered-are-more-expensive*:
assumes *i-agt*: $i \in$ *agents*
assumes *equil*: *competitive-equilibrium* $P\ \mathcal{X}\ \mathcal{Y}$
assumes $z \in$ *consumption-set*
assumes $(U\ i)\ z > (U\ i)\ (\mathcal{X}\ i)$
shows $z \cdot P > P \cdot (\mathcal{X}\ i)$
⟨*proof*⟩

Given local non-satiation, argmax will use the entire budget.

lemma *am-utilises-entire-bgt*:
assumes *i-agts*: $i \in$ *agents*
assumes *lns* : *local-nonsatiation consumption-set* $Pr[i]$
assumes *argmax-sol* : $X \in$ *arg-max-set* $U[i]$ (*budget-constraint* (*calculate-value* P) *consumption-set* (*poe-wealth* $P\ i\ Y$))
shows $P \cdot X = P \cdot \mathcal{E}[i] + (\sum j{\in}\mathit{firms}.\ \Theta[i,j] *_R (P \cdot Y\ j))$
⟨*proof*⟩

corollary *x-equil-x-ext-budget*:
assumes *i-agt*: $i \in$ *agents*
assumes *lns* : *local-nonsatiation consumption-set* $Pr[i]$
assumes *equilibrium* : *competitive-equilibrium* $P\ X\ Y$
shows $P \cdot X\ i = P \cdot \mathcal{E}[i] + (\sum j{\in}\mathit{firms}.\ \Theta[i,j] *_R (P \cdot Y\ j))$
⟨*proof*⟩

lemma *same-price-in-argmax* :
assumes *i-agt*: $i \in$ *agents*
assumes *lns* : *local-nonsatiation consumption-set* $Pr[i]$
assumes $x \in$ *arg-max-set* $(U[i])$ (*budget-constraint* (*calculate-value* P) *consumption-set* (*poe-wealth* $P\ i\ Y$))
assumes $y \in$ *arg-max-set* $(U[i])$ (*budget-constraint* (*calculate-value* P) *consumption-set* (*poe-wealth* $P\ i\ Y$))
shows $(P \cdot x) = (P \cdot y)$
⟨*proof*⟩

Greater or equal utility implies greater or equal value.

lemma *utility-ge-price-ge* :
assumes *agts*: $i \in$ *agents*

assumes *lns* : *local-nonsatiation consumption-set* $Pr[i]$
assumes *equil*: *competitive-equilibrium* $P\ X\ Y$
assumes *geq*: $U[i]\ z \geq U[i]\ (X\ i)$
and $z \in$ *consumption-set*
shows $P \cdot z \geq P \cdot (X\ i)$
⟨*proof*⟩

lemma *commutativity-sums-over-funs*:
fixes X :: *'x set*
fixes Y :: *'y set*
shows $(\sum i{\in}X.\ \sum j{\in}Y.\ (f\ i\ j *_R C \cdot g\ j)) = (\sum j{\in}Y. \sum i{\in}X.\ (f\ i\ j *_R C \cdot g\ j))$
⟨*proof*⟩

lemma *assoc-fun-over-sum*:
fixes X :: *'x set*
fixes Y :: *'y set*
shows $(\sum j{\in}Y.\ \sum i{\in}X.\ f\ i\ j *_R C \cdot g\ j) = (\sum j{\in}Y.\ (\sum i{\in}X.\ f\ i\ j) *_R C \cdot g\ j)$
⟨*proof*⟩

Walras' law in context of production economy with private ownership. That is, in an equilibrium demand equals supply.

lemma *walras-law*:
assumes $\bigwedge i.\ i{\in}agents \Longrightarrow$ *local-nonsatiation consumption-set* $Pr[i]$
assumes $(\forall\, i \in agents.\ (X\ i) \in$ *arg-max-set* $U[i]$ (*budget-constraint* (*calculate-value* P) *consumption-set* (*poe-wealth* $P\ i\ Y$)))
shows $P \cdot (\sum i{\in}agents.\ (X\ i)) = P \cdot ((\sum i{\in}agents.\ \mathcal{E}[i]) + (\sum j{\in}firms.\ Y\ j))$
⟨*proof*⟩

lemma *walras-law-in-compeq*:
assumes $\bigwedge i.\ i{\in}agents \Longrightarrow$ *local-nonsatiation consumption-set* $Pr[i]$
assumes *competitive-equilibrium* $P\ X\ Y$
shows $P \cdot ((\sum i{\in}agents.\ (X\ i)) - (\sum i{\in}agents.\ \mathcal{E}[i]) - (\sum j{\in}firms.\ Y\ j)) = 0$
⟨*proof*⟩

8.6 First Welfare Theorem

Proof of First Welfare Theorem in context of production economy with private ownership.

theorem *first-welfare-theorem-priv-own*:
assumes $\bigwedge i.\ i \in agents \Longrightarrow$ *local-nonsatiation consumption-set* $Pr[i]$
and $Price > 0$
assumes *competitive-equilibrium* $Price\ \mathcal{X}\ \mathcal{Y}$
shows *pareto-optimal* $\mathcal{X}\ \mathcal{Y}$
⟨*proof*⟩

Equilibrium cannot be Pareto dominated.

lemma *equilibria-dom-eachother*:
assumes $\bigwedge i.\ i \in agents \Longrightarrow$ *local-nonsatiation consumption-set* $Pr[i]$

and *Price* > 0
assumes *equil*: *competitive-equilibrium Price* $\mathcal{X}$ $\mathcal{Y}$
shows $\nexists X'\ Y'.$ *competitive-equilibrium P* $X'\ Y' \wedge X' \succ$*Pareto* $\mathcal{X}$
⟨*proof*⟩

Using monotonicity instead of local non-satiation proves the First Welfare Theorem.

corollary *first-welfare-thm-monotone*:
assumes $\forall M \in$ *carrier*. ($\forall x > M$. $x \in$ *carrier*)
assumes $\bigwedge i.\ i \in$ *agents* $\Longrightarrow$ *monotone-preference consumption-set* $Pr[i]$
and *Price* > 0
assumes *competitive-equilibrium Price* $\mathcal{X}$ $\mathcal{Y}$
shows *pareto-optimal* $\mathcal{X}$ $\mathcal{Y}$
⟨*proof*⟩

end

end

end

9 Arrow-Debreu model

theory *Arrow-Debreu-Model*
imports
../Preferences
../Preferences
../Utility-Functions
../Argmax
Consumers
Common
begin

locale *pre-arrow-debreu-model* =
fixes *production-sets* :: $'f \Rightarrow ('a$::*ordered-euclidean-space*) *set*
fixes *consumption-set* :: $'a$ *set*
fixes *agents* :: $'i$ *set*
fixes *firms* :: $'f$ *set*
fixes $\mathcal{E}$ :: $'i \Rightarrow 'a$ ($\mathcal{E}$[-])
fixes *Pref* :: $'i \Rightarrow 'a$ *relation* (*Pr*[-])
fixes U :: $'i \Rightarrow 'a \Rightarrow$ *real* (U[-])
fixes Θ :: $'i \Rightarrow 'f \Rightarrow$ *real* (Θ[-,-])
assumes *cons-set-props*: *arrow-debreu-consum-set consumption-set*
assumes *agent-props*: $i \in$ *agents* $\Longrightarrow$ *eucl-ordinal-utility consumption-set* ($Pr[i]$) ($U[i]$)
assumes *firms-comp-owned*: $j \in$ *firms* $\Longrightarrow$ ($\sum i \in$*agents*. $\Theta[i,j]$) = *1*
assumes *finite-nonepty-agents*: *finite agents* **and** *agents* $\neq$ {}

sublocale *pre-arrow-debreu-model* $\subseteq$ *pareto-ordering agents U*

$\langle proof \rangle$

context *pre-arrow-debreu-model*
begin

Calculate wealth of individual i in context of Private Ownership economy.

context
begin

private abbreviation *poe-wealth*
where
$poe\text{-}wealth\ P\ i\ Y \equiv P \cdot \mathcal{E}[i] + (\sum j{\in}firms.\ \Theta[i,j] *_R (P \cdot Y\ j))$

9.1 Feasiblity

private abbreviation *feasible*
where
$feasible\ X\ Y \equiv feasible\text{-}private\text{-}ownership\ agents\ firms\ \mathcal{E}\ consumption\text{-}set\ production\text{-}sets\ X\ Y$

private abbreviation *calculate-value*
where
$calculate\text{-}value\ P\ x \equiv P \cdot x$

9.2 Profit maximisation

In a production economy (which this is) we need to specify profit maximisation.

definition *profit-maximisation*
where
$profit\text{-}maximisation\ P\ S = arg\text{-}max\text{-}set\ (\lambda x.\ P \cdot x)\ S$

9.3 Competitive Equilibirium

Competitive equilibrium in context of production economy with private ownership. This includes the profit maximisation condition.

definition *competitive-equilibrium*
where
$competitive\text{-}equilibrium\ P\ X\ Y \longleftrightarrow feasible\ X\ Y \wedge$
$(\forall j \in firms.\ (Y\ j) \in profit\text{-}maximisation\ P\ (production\text{-}sets\ j)) \wedge$
$(\forall i \in agents.\ (X\ i) \in arg\text{-}max\text{-}set\ U[i]\ (budget\text{-}constraint\ (calculate\text{-}value\ P)\ consumption\text{-}set\ (poe\text{-}wealth\ P\ i\ Y)))$

lemma *competitive-equilibriumD* [*dest*]:
assumes $competitive\text{-}equilibrium\ P\ X\ Y$
shows $feasible\ X\ Y \wedge$

$(\forall j \in \mathit{firms}.\ (Y\ j) \in \mathit{profit\text{-}maximisation}\ P\ (\mathit{production\text{-}sets}\ j)) \wedge$
$(\forall i \in \mathit{agents}.\ (X\ i) \in \mathit{arg\text{-}max\text{-}set}\ U[i]\ (\mathit{budget\text{-}constraint}\ (\mathit{calculate\text{-}value}\ P)\ \mathit{consumption\text{-}set}\ (\mathit{poe\text{-}wealth}\ P\ i\ Y)))$
⟨*proof*⟩

lemma *compet-max-profit*:
assumes $j \in \mathit{firms}$
assumes *competitive-equilibrium* $P\ X\ Y$
shows $Y\ j \in \mathit{profit\text{-}maximisation}\ P\ (\mathit{production\text{-}sets}\ j)$
⟨*proof*⟩

9.4 Pareto Optimality

definition *pareto-optimal*
where
$\mathit{pareto\text{-}optimal}\ X\ Y \longleftrightarrow$
$(\mathit{feasible}\ X\ Y \wedge$
$(\nexists X'\ Y'.\ \mathit{feasible}\ X'\ Y' \wedge X' \succ\!\mathit{Pareto}\ X))$

lemma *pareto-optimalI*[*intro*]:
assumes *feasible* $X\ Y$
and $\nexists X'\ Y'.\ \mathit{feasible}\ X'\ Y' \wedge X' \succ\!\mathit{Pareto}\ X$
shows *pareto-optimal* $X\ Y$
⟨*proof*⟩

lemma *pareto-optimalD*[*dest*]:
assumes *pareto-optimal* $X\ Y$
shows *feasible* $X\ Y$ **and** $\nexists X'\ Y'.\ \mathit{feasible}\ X'\ Y' \wedge X' \succ\!\mathit{Pareto}\ X$
⟨*proof*⟩

lemma *util-fun-def-holds*:
assumes $i \in \mathit{agents}$
and $x \in \mathit{consumption\text{-}set}$
and $y \in \mathit{consumption\text{-}set}$
shows $x \succeq[\mathit{Pr}[i]]\ y \longleftrightarrow U[i]\ x \geq U[i]\ y$
⟨*proof*⟩

lemma *base-pref-is-ord-eucl-rpr*: $i \in \mathit{agents} \Longrightarrow \mathit{rational\text{-}preference}\ \mathit{consumption\text{-}set}\ \mathit{Pr}[i]$
⟨*proof*⟩

lemma *prof-max-ge-all-in-pset*:
assumes $j \in \mathit{firms}$
assumes $Y\ j \in \mathit{profit\text{-}maximisation}\ P\ (\mathit{production\text{-}sets}\ j)$
shows $\forall y \in \mathit{production\text{-}sets}\ j.\ P \cdot Y\ j \geq P \cdot y$
⟨*proof*⟩

9.5 Lemmas for final result

Strictly preferred bundles are strictly more expensive.

lemma *all-prefered-are-more-expensive*:
  assumes *i-agt*: $i \in agents$
  assumes *equil*: *competitive-equilibrium* $P\ \mathcal{X}\ \mathcal{Y}$
  assumes $z \in consumption\text{-}set$
  assumes $(U\ i)\ z > (U\ i)\ (\mathcal{X}\ i)$
  shows $z \cdot P > P \cdot (\mathcal{X}\ i)$
$\langle proof \rangle$

Given local non-satiation, argmax will use the entire budget.

lemma *am-utilises-entire-bgt*:
  assumes *i-agts*: $i \in agents$
  assumes *lns* : *local-nonsatiation consumption-set* $Pr[i]$
  assumes *argmax-sol* : $X \in$ *arg-max-set* $U[i]$ (*budget-constraint* (*calculate-value* P) *consumption-set* (*poe-wealth* $P\ i\ Y$))
  shows $P \cdot X = P \cdot \mathcal{E}[i] + (\sum j \in firms.\ \Theta[i,j] *_R (P \cdot Y\ j))$
$\langle proof \rangle$

corollary *x-equil-x-ext-budget*:
  assumes *i-agt*: $i \in agents$
  assumes *lns* : *local-nonsatiation consumption-set* $Pr[i]$
  assumes *equilibrium* : *competitive-equilibrium* $P\ X\ Y$
  shows $P \cdot X\ i = P \cdot \mathcal{E}[i] + (\sum j \in firms.\ \Theta[i,j] *_R (P \cdot Y\ j))$
$\langle proof \rangle$

lemma *same-price-in-argmax* :
  assumes *i-agt*: $i \in agents$
  assumes *lns* : *local-nonsatiation consumption-set* $Pr[i]$
  assumes $x \in$ *arg-max-set* $(U[i])$ (*budget-constraint* (*calculate-value* P) *consumption-set* (*poe-wealth* $P\ i\ Y$))
  assumes $y \in$ *arg-max-set* $(U[i])$ (*budget-constraint* (*calculate-value* P) *consumption-set* (*poe-wealth* $P\ i\ Y$))
  shows $(P \cdot x) = (P \cdot y)$
  $\langle proof \rangle$

Greater or equal utility implies greater or equal value.

lemma *utility-ge-price-ge* :
  assumes *agts*: $i \in agents$
  assumes *lns* : *local-nonsatiation consumption-set* $Pr[i]$
  assumes *equil*: *competitive-equilibrium* $P\ X\ Y$
  assumes *geq*: $U[i]\ z \geq U[i]\ (X\ i)$
    and $z \in consumption\text{-}set$
  shows $P \cdot z \geq P \cdot (X\ i)$
$\langle proof \rangle$

lemma *commutativity-sums-over-funs*:
  fixes $X :: {}'x\ set$

fixes Y :: $'y$ *set*
shows $(\sum i{\in}X.\ \sum j{\in}Y.\ (f\ i\ j *_R\ C \cdot g\ j)) = (\sum j{\in}Y.\sum i{\in}X.\ (f\ i\ j *_R\ C \cdot g\ j))$
⟨*proof*⟩

lemma *assoc-fun-over-sum*:
fixes X :: $'x$ *set*
fixes Y :: $'y$ *set*
shows $(\sum j{\in}Y.\ \sum i{\in}X.\ f\ i\ j *_R\ C \cdot g\ j) = (\sum j{\in}Y.\ (\sum i{\in}X.\ f\ i\ j) *_R\ C \cdot g\ j)$
⟨*proof*⟩

Walras' law in context of production economy with private ownership. That is, in an equilibrium demand equals supply.

lemma *walras-law*:
assumes $\bigwedge i.\ i{\in}agents \Longrightarrow$ *local-nonsatiation consumption-set* $Pr[i]$
assumes $(\forall\, i \in agents.\ (X\ i) \in$ *arg-max-set* $U[i]$ (*budget-constraint* (*calculate-value* P) *consumption-set* (*poe-wealth* $P\ i\ Y$)))
shows $P \cdot (\sum i{\in}agents.\ (X\ i)) = P \cdot ((\sum i{\in}agents.\ \mathcal{E}[i]) + (\sum j{\in}firms.\ Y\ j))$
⟨*proof*⟩

lemma *walras-law-in-compeq*:
assumes $\bigwedge i.\ i{\in}agents \Longrightarrow$ *local-nonsatiation consumption-set* $Pr[i]$
assumes *competitive-equilibrium* $P\ X\ Y$
shows $P \cdot ((\sum i{\in}agents.\ (X\ i)) - (\sum i{\in}agents.\ \mathcal{E}[i]) - (\sum j{\in}firms.\ Y\ j)) = 0$
⟨*proof*⟩

9.6 First Welfare Theorem

Proof of First Welfare Theorem in context of production economy with private ownership.

theorem *first-welfare-theorem-priv-own*:
assumes $\bigwedge i.\ i \in agents \Longrightarrow$ *local-nonsatiation consumption-set* $Pr[i]$
and $Price > 0$
assumes *competitive-equilibrium* $Price\ \mathcal{X}\ \mathcal{Y}$
shows *pareto-optimal* $\mathcal{X}\ \mathcal{Y}$
⟨*proof*⟩

Equilibrium cannot be Pareto dominated.

lemma *equilibria-dom-eachother*:
assumes $\bigwedge i.\ i \in agents \Longrightarrow$ *local-nonsatiation consumption-set* $Pr[i]$
and $Price > 0$
assumes *equil*: *competitive-equilibrium* $Price\ \mathcal{X}\ \mathcal{Y}$
shows $\nexists X'\ Y'.$ *competitive-equilibrium* $P\ X'\ Y' \wedge X' \succ$*Pareto* $\mathcal{X}$
⟨*proof*⟩

Using monotonicity instead of local non-satiation proves the First Welfare Theorem.

corollary *first-welfare-thm-monotone*:
assumes $\forall\, M \in carrier.\ (\forall\, x > M.\ x \in carrier)$

assumes $\bigwedge i.\ i \in agents \Longrightarrow monotone\text{-}preference\ consumption\text{-}set\ Pr[i]$
and $Price > 0$
assumes $competitive\text{-}equilibrium\ Price\ \mathcal{X}\ \mathcal{Y}$
shows $pareto\text{-}optimal\ \mathcal{X}\ \mathcal{Y}$
$\langle proof \rangle$

end

end

end

10 Related work

[2]

References

[1] K. J. Arrow, A. Sen, and K. Suzumura. *Handbook of Social Choice and Welfare*, volume 2. Elsevier, 2010.

[2] S. Tadelis. *Game Theory: An Introduction*. Princeton University Press, 2013.