Lecture 10: Mechanism Design

Lecturer: Yishay Mansour *Scribe: Vera Vsevolozhsky, Nadav Wexler*

10.1 Mechanisms with money

10.1.1 Introduction

As we have seen in the previous lesson, Mechanism Design is challenging. We chose two reasonable conditions:

1. Unanimity - If everyone agrees on a choice, it is accepted.
2. Strategy Proof - no player can alter the outcome to his favor by reporting a vote different than his actual preference.

As we've seen, The results of Arrow's Impossibility Theorem for social welfare functions and Gibbard-Satterthwaite theorem for social choice functions, showed us that the resulting function will always be dictatorial. In this lesson we'll try to solve this problem by adding the concept of "Money". The utility of a player will be divided into two parts: the value of the result and the payments.

10.1.2 A Quasi-Linear Model

Definition: A model for n players is given by:

- $\mathbb{N}$ - a group of n Players
- A - a group of the Alternative options (products)
- Each player $i \in \mathbb{N}$ has a value function $v_i \in V_i$, for each $a \in A$
- Each player $i \in \mathbb{N}$ is given m_i worth of money
- Utility function for player i is $u_i(a, m_i) = v_i(a) + m_i$

The utility function is quasi-linear, and the money is added to the utility of the products. The money given can be a negative or positive, depending on the game played.

Example: Selling a single item

- $\mathbb{N}$ - a group of n Players
- $A = \{"i - wins"|i \in \mathbb{N}\}$
- for each player i, $v_i(a) = \begin{cases} w_i, & \text{if a="i-wins"} \\ 0, & \text{else} \end{cases}$
- for each player i, if $a = "i - wins", m_i = p$
- Therefore, each player $i \in \mathbb{N}$ has a utility function $u_i = \begin{cases} w_i - p, & \text{if a="i-wins"} \\ 0, & \text{else} \end{cases}$

Our goal: give the product to the player with the highest value for it. If we just ask them to report their valuation, and give the product to the player with the highest valuation, the players will just say high values and try to win.
More practical options:

- first-price auction
- second-price auction

Theorem 10.1 *A second-price auction is Strategy Proof.*

Proof: We separate into cases, depending on the player. We fix the player i as the winner.
Player i has no incentive to change his bid:

- Changing above second bid - Won't change the outcome as he will still win and pay the second bid.
- Lowering below second bid - player i will lose the auction, making his profit 0.

For player $j \neq i$:

- Raising above v_i - raising his bid above his value to v_i will result in a loss: $u_j = v_j - v_i < 0$, since $v_j < v_i$. Player j will pay at least as the actual winner bid and that is more than the value he gives for the product.
- Changing under v_i - he can't win, and his profit will stay 0, as before.

Thus, player j has no incentive to change his bid. $\square$

10.1.3 General Models

Definition: A Mechanism is called Direct if it defines the following:

- $f : V_1 \times V_2 \times ... \times V_n \longrightarrow A$ - social choice function
- $p_i : V_1 \times V_2 \times ... \times V_n \longrightarrow \mathbb{R}$ - price for each player i

Definition: A mechanism $(f, p_1, p_2, ..., p_n)$ is called Strategy Proof (or Incentive Compatible, or Truthfull) if:

$$\forall i \in \mathbb{N}, \forall v_i \in V_i : a = f(v_i, v_{-i}), a' = f(v'_i, v_{-i}), \text{ then:}$$
$$v_i(a) - p_i(v_i, v_{-i}) \geq v_i(a') - p_i(v'_i, v_{-i})$$

In words: A SP Mechanism ensures that for any value function a player in the game has, he cannot gain from reporting a different function. Thus, it promotes the players to report their real value.

10.1.4 VCG Mechanism

Definition: A mechanism $(f, p_1, p_2, ..., p_n)$ is called Vickery-Clarke-Grove (VCG) if:

- $f(v_1, .., v_n) = \arg\max_{a \in A} \sum_{i \in \mathbb{N}} v_i(a)$
- $p_i(v_1.., v_n) = h_i(v_{-i}) - \sum_{j \neq i} v_j(f(v_1, ..., v_n))$, for any function h_i that does not depend on v_i.

Theorem 10.2 *for any h_i, a VCG mechanism is Strategy Proof.*

Proof: We shall fix a player i and a value vector v_{-i} for the other players. we shall also define v_i as the real value for the player and v'_i as the false reported value. we also define $a = f(v_1, ..., v_n), a' = f(v'_i, v_{-i})$. the player's utility when he reports the true value is:

$$u_i = v_i(a) + \sum_{j \neq i} v_j(a) - h_i(v_{-i}) = \sum_{k \in \mathbb{N}} v_k(a) - h_i(v_{-i})$$

while in a false report the utility is:

$$u'_i = v_i(a') + \sum_{j \neq i} v_j(a') - h_i(v_{-i}) = \sum_{k \in \mathbb{N}} v_k(a') - h_i(v_{-i})$$

Since a maximizes the social welfare, $\sum_{k \in \mathbb{N}} v_k(a') \leq \sum_{k \in \mathbb{N}} v_k(a)$. we conclude that $u_i \geq u'_i$ and thus a VCG mechanism is SP. $\square$

Grove's Mechanism

We would like to choose a function h_i. As we've seen, this can be done by any function, but we'd like to choose function that complies with certain reasonable conditions.

Definition:

1. Individual Rationality - if a player participates, his utility cannot be negative:
   $u_i = v_i(f(v_1, ..., v_n)) - p_i(v_1, ..., v_n) \geq 0$.
2. No Positive Transfer - we sell an item, therefore the players cannot gain money. The prices that players pay cannot be negative:
   $p_i(v_1, ..., v_n) \geq 0$

Definition: Grove's function
We define the function $h_i(v_{-i}) = \max_{b \in A} \sum_{j \neq i} v_j(b)$.
The payments associated with this functions are: $p_i(v_1, ..., v_n) = \max_{b \in A} \sum_{j \neq i} v_j(b) - \sum_{j \neq i} v_j(a)$, where $a = f(v_1, ..., v_n)$.
The payment function is defined so that the payments are the difference between the value when player i is participating in the game and the value without player i.

Lemma 10.3 *A VCG mechanism with h_i as defined above is Individual Rational and has No Positive Transfers.*

Proof: As before, $a = f(v_1, ..., v_n)$. We shall assume that $v_i(a) \geq 0$, which is reasonable, since a player will not participate if the value is negative.

- Individual Rationality:
  $u_i(a) = v_i(a) - p_i = v_i(a) + \sum_{j \neq i} v_j(a) - \sum_{j \neq i} v_j(b) \geq \sum_{j \in \mathbb{N}} v_j(a) - \sum_{j \in \mathbb{N}} v_j(b) \geq 0$
  For the first inequality, we only added a negative term, thus reducing the value. For the second inequality from the definition of a: it is the choice that maximizes the social welfare - the sum of values.
- No Positive Transfer:
  $p_i = \sum_{j \neq i} v_j(b) - \sum_{j \neq i} v_j(a) \geq 0$.
  This inequality is obvious by looking at the definition of b: it is the choice that maximizes the function h_i.

$\square$

10.1.5 VCG Examples

Single Item Auction

We define a single item auction by:

- $A = \{"i - wins" | i \in \mathbb{N}\}$
- $v_i(a) = \begin{cases} w_i, & \text{a="i-wins"} \\ 0, & \text{else} \end{cases}$

In this case, f is reduced into $\sum_{j \in \mathbb{N}} v_j("k - wins") = w_k$, f will choose player i that has the maximum value of w_i.
If i has the maximum of w_i, then $\sum_{j \neq i} v_j("i - wins") = 0$ and $\sum_{j \neq i} v_i(b) = w_s$, where w_s is the second highest value. Therefore $p_i = w_s$ and this model is identical to a second-price Auction.

Trade

- $A = \{"trade", "no - trade"\}$
- Seller - has value v_s for the product
- Buyer - has value v_b for the product
- (p_b, p_s) - the price of trade

We would like to have a trade iff $v_b > v_s$. We also define: $u_b(\text{"trade"}) = v_b - p_b, u_s(\text{"trade"}) = p_s - v_s, u_b(\text{"no-trade"}) = u_s(\text{"no-trade"}) = 0$. If $v_b > v_s$ bringing f to maximum can be achieved only by "trade". The prices associated are: $p_b = v_s, p_s = -v_b$. In this results that the mechanism is subsidizing the Trade in the amount of $v_b - v_s \geq 0$.

Auction with Several Identical Items

We have an auction with k identical items.

- $A = \{"S - wins" | S \subseteq \mathbb{N}, |S| = k\}$
- $v_i("S - wins") = \begin{cases} w_i, & i \in S \\ 0, & \text{else} \end{cases}$

The maximum value of f means that S includes the k players with the highest values. We fix a player i and calculate his price: $p_i = \max_{S'} \sum_{j \neq i} v_j(S') - \sum_{j \neq i} v_j(S) = w_{k+1}$, where S' includes all players from S except i and also the "$k + 1$" player which is the player with the $k + 1$ value. So each winning player pays a price of w_{k+1}.

Public Project

In a public project like a bridge, the players benefit from building. The problem is how to divide the cost of building.

- $A = \{"build", "no-build"\}$
- A total build cost of C.
- Each player has value w_i from the project.

The social utility is $\sum w_i - C$. If $\sum w_i > C \Rightarrow$ the project will be built.
What will the payments be?
Each player will pay only if $w_i > C - \sum_{j \neq i} w_j$. Therefore, $p_i = C - \sum_{j \neq i} w_j$. It is clear that $\sum_{i \in \mathbb{N}} p_i < C$.

Assume we want to enforce that the project would be budget balanced. The following example will show that in this case there will be payments even if the project is not built.
Example:
Let's assume $C = 3$ and $w_1 = w_2 \in \{0, 2\}$.

1. If $w_1 = w_2 = 2$: The cost must be covered by the payments so $p_1(2) + p_2(2) \geq 3$.
2. If $w_1 = w_2 = 0$: Since the mechanism has NPT, $p_1(0) + p_2(0) \geq 0$.

From the two inequalities we conclude that either $p_1(2) + p_2(0) \geq \frac{3}{2}$ or $p_1(0) + p_2(2) \geq \frac{3}{2}$. In both cases, the project is not built. Therefore, we conclude that although the project is not built, at least one player is paying.

Buying Edges in Network

We define a graph $G = (V, E)$. Our goal is to build a path from s to t.

- $e \in E$ are the players
- $v_e(a) = \begin{cases} -c_e, & e \text{ is used in the network} \\ 0, & \text{else} \end{cases}$

the solution for f is the shortest solution in respect to values. the payments for player e will be: $p_e =$ "cost of shortest path without e" $-$ "path's cost without e's cost"

10.1.6 The revelation principle

We are now ready to formalize the notion of a general - nondirect revelation mechanism.
Definition: A mechanism for n players is given by:

- Players' type spaces $T_1, T_2..., T_n$. Each player $i \in \mathbb{N}$ has some private information $t_i \in T_i$ that captures his preference over a set of alternatives A.
- Players' action spaces $X_1, X_2..., X_n$
- Each player $i \in \mathbb{N}$ has some strategy $s_i(t_i)$, where $s_i : T_i \longrightarrow X_i$
- Each player $i \in \mathbb{N}$ has utility function $u_i(t_i, s_1(t_2), s_1(t_2), ..., s_n(t_n))$

At first sight it seems that the more general definition of mechanisms will allow us to do more than is possible using strategy proof direct revelation mechanisms. But as we will see shortly this turns out to be false.

Theorem 10.4 ***Revelation principle***
If there exists a strategy proof mechanism that implements f then there exists a direct strategy proof mechanism that implements f. The payments of the players in the new direct strategy proof mechanism and original one are identical.

Proof: The new direct mechanism will simply simulate the original mechanism. That is, let $s_i(t_i)$ be a dominant strategy of the original mechanism for some player i with type t_i. Now player i reveals his private information t_i. And the new direct mechanism will play strategy $s_i(t_i)$ for player i. But, because $s_i(t_i)$ is a dominant strategy in the original mechanism, then t_i is a dominant action in the new direct mechanism. And the payments of the players are identical (according to the simulation). $\square$

10.1.7 Bayesian - Nash Equilibrium

Now we will talk about equilibrium models in mechanisms.Each of the participants is familiar with his private information, but not the others'. However, he can derive the expected value of the others' private information knowing their distribution.

Definition: A game with independent private values and incomplete information on a set of n players is given by the following ingredients:

1. For every player i, a set of actions X_i, where $X = X_1 \times X_2 \times ... \times X_n$.

2. For every player i, a set of types T_i, and a distribution $D_i \sim \triangle(T_i)$. A value $t_i \in T_i$ is the private information that i has, and $t_i \sim D_i$ where $D_i(t_i)$ is the a probability that i gets type t_i.

3. For every player i, a utility function $u_i : T_i \times X \longrightarrow \mathbb{R}$, where $u_i(t_i, x_1, x_2..., x_n)$ is the utility achieved by player i, if his type (private information) is t_i, and the profile of actions taken by all players is $x_1, x_2, ..., x_n$.

The main idea that we wish to capture with this definition is that each player i must choose his action x_i when knowing t_i but not the other $t_j\ 's$ $(j \neq i)$, but rather only knowing the prior distribution D_j on each other t_j ($partial\ information$).
The behavior of player i in such a setting is captured by a function $s_i : T_i \longrightarrow X_i$ that specifies which action x_i is taken for every possible type t_i - this is termed a *strategy*. It is these strategies that we would want to be in equilibrium.

Definition: Bayesian - Nash Equilibrium

$s_1, s_2..., s_n$ is a $Bayesian - Nash\ equilibrium$ if for every player i and every $t_i \in T_i$ we have as follows:

$$\forall x_i \in X_i : E_{D_{-i}}[u_i(t_i), s_i(t_i), s_{-i}(t_{-i})] \geq E_{D_{-i}}[u_i(t_i), s_i'(t_i), s_{-i}(t_{-i})]$$

($s_i(t_i)$ is the best response that i has to $s_{-i}()$ when his type is t_i, in expectation over the types of the other players. And $E_{D_{-i}}$ [ ] denotes the expectation over the other types t_{-i} being chosen according to distribution D_{-i}).

This now allows us to define implementation in the Bayesian sense.

Definition: A Bayesian mechanism for n players is given by:

- Players' action spaces $X_1, X_2..., X_n$
- Players' type spaces $T_1, T_2..., T_n$ and distributions on them $D_1, D_2..., D_n$
- An alternative set A
- Players' valuations functions $v_i : T_i \times A \longrightarrow \mathbb{R}$
- An outcome function $f : X \longrightarrow A$
- Payment functions $p_1, p_2, ..., p_n$, where $p_i : X \longrightarrow \mathbb{R}$

Definition: A direct Bayesian mechanism

A mechanism is direct in the Bayesian sense if two conditions are held:

- The type spaces are equal to the action spaces – $T_i = X_i$
- The truthful strategies $s_i(t_i) = t_i$ are a Bayesian-Nash equilibrium.

Analysis of Bayesian–Nash Equilibrium in First–Price Auction

As an example of Bayesian analysis we study the standard first price auction in a simple setting: a single item is auctioned between two players, Alice and Bob. Each has a private value for the item: $t_{Alice} = a$ is Alice's value and $t_{Bob} = b$ is Bob's value; the distribution D_a over a and D_b over b. Every player will announce a value smaller than his real value. Let x denote the announced value of Alice(where x is a function of a). Let y denote the announced value of Bob(where y is a function of b). If $x > y$ Alice wins, otherwise Bob wins.

While we already saw that a second price auction will allocate the item to the one with higher value, here we ask what would happen if the auction rules are the usual first–price ones: the highest bidder pays his own bid. Certainly Alice will not bid her value a since if she does, even if she wins her utility will be 0. She will thus need to bid some $x < a$, but how much lower? If she knew that Bob would bid y, she would certainly bid $x = y + \epsilon$ (as long as $x \leq a$). But she does not know y or even b which y would depend on - she only knows the distribution D_b over b.

Let us observe what happen in the Bayesian – Nash equilibrium. In general, finding Bayesian-Nash equilibria is not an easy thing. However, for the symmetric case where $D_a = D_b$, the situation is simpler and a closed form expression for the equilibrium strategies may be found. We will prove it for the special case where the private information, t_{Alice} and t_{Bob}, is uniformly distributed on the interval [0, 1] ($D_a = D_b = U([0, 1])$).

Lemma 10.5 *In a first price auction among two players with distributions $D_a = D_b$ of the private values a, b uniform over the interval [0, 1], the strategies $x(a) = a/2$ and $y(b) = b/2$ are in Bayesian-Nash equilibrium.*

Proof: Let us consider which bid x is Alice's optimal response to Bob's strategy $y = b/2$, when Alice has value a. The utility for Alice is 0 if she loses and $a - x$ if she wins and pays x. Then, $u_{Alice}(x) = Pr[Alice\ wins\ with\ bid\ x] * (a - x)$, where $Pr[Alice\ wins\ with\ bid\ x] = Pr_{b \sim D_b}[x > y(b)]$. Note, that if Alice chooses to bid $x \leq 0$ then she never wins. If she chooses to bid $x \geq 1/2$ then she always wins. Therefore we need to consider only $x \in [0, 1/2]$. Let

us calculate $Pr_{b \sim D_b}[x > y(b)]$ for $y(b) = b/2$, where b is uniform over the interval [0, 1] and $x \in [0, 1/2]$:
$Pr_{b \sim D_b}[x > b/2] = Pr_{b \sim D_b}[2x > b] = 2x$. Hence, $u_{Alice}(x) = 2x(a - x)$. Thus, to optimize the value of x, we need to find the maximum of the function $u_{Alice}(x)$ over the range $x \in [0, 1/2]$. For the maximization we compute:

$$\frac{\partial u_{Alice}(x)}{\partial x} = 2a - 4x = 0,$$

and derive $x = a/2$ as required. Note, that Alice and Bob are symmetric players. Then Bob's optimal response to Alice's strategy, $x = a/2$, will be $y = b/2$. □

10.1.8 Revenue Equivalence

Let us now attempt comparing the first price auction and the second price auction. The social choice function implemented is exactly the same: giving the item to the player with highest private value. How about the payments? Where does the auctioneer get a higher revenue?
–The revenue of the first–price auction is $max(a/2, b/2)$.
–The revenue of the second–price auction is $min(a, b)$.
Then, for a first–price auction we get that $E_{a,b}[max(a/2, b/2)] = \frac{1}{2}E_{a,b}[max(a,b)] = \frac{1}{2}\int_a \int_b max(a,b)$ $= \frac{1}{2}\int_a [ab|_0^a + \frac{b^2}{2}|_a^1] = \frac{1}{4}\int_a (1 + a^2) = \frac{1}{4}(a + \frac{a^3}{3})|_0^1 = \frac{1}{3}$, where a and b are chosen uniformly in [0, 1]. Similarly calculations for a second–price auction will reveal that $E_{a,b}[\text{min(a, b)}] = \frac{1}{3}$. Thus, both auctions generate equivalent revenue in expectation! This is no coincidence.

Theorem 10.6 ***The Revenue Equivalence Principle***
Under certain weak assumptions(to be detailed in the proof body), for every two Bayesian-Nash mechanisms that implement the same social choice function f, we have:

1. *If for some type t_i^0 of player i, the expected payment of player i is the same in the two mechanisms, then it is the same for every value of t_i.*

2. *If for each player i there exists a type t_i^0 where the two mechanisms have the same expected payment for player i, then the two mechanisms have the same expected payments from each player and their expected revenues are the same.*

Intuition - we can choose t_i^0 for player i to be zero value for the product. Thus, the expected payment of player i will be zero.

Corollary 10.7 *Every mechanism that allocates the item to the player with highest value will have identical expected revenue(under mild assumption).*

Proof: Using the revelation principle, we can first limit ourselves to mechanisms that are direct in the Bayesian-Nash sense. Let us denote by V_i the space of valuation functions $v_i(t_i, \cdot)$ over all t_i.
Assumption 1: Each V_i is a convex set.
Take any type $t_i^1 \in T_i$. We will derive a formula for the expected payment for this type that depends only on the expected payment for type $t_i^0 \in T_i$ and on the social choice function f. Thus any two mechanisms that implement the same social choice function and have identical expected payments at $t_i^0 \in T_i$ will also have identical expected payments at $t_i^1 \in T_i$. For this, let us now introduce some notations:

- v_0 is the valuation $v_i(t_i^0, \cdot)$(i.e., v_0=$v_i(t_i^0, \cdot)$). Similar, $v_1 = v_i(t_i^1, \cdot)$. We will look at these as vectors in $V_i \subseteq \mathbb{R}^A$. Their convex combinations is $v^\lambda = \lambda v_0 + (1-\lambda)v_1$. The convexity of V_i implies that $v^\lambda \in V_i$ and thus there exists some type t_i^λ such that $v^\lambda = v_i(t_i^\lambda, \cdot)$.
- p_i^λ is the expected payment of player i at type t_i^λ: $p_i^\lambda = E_{t_{-i}}[p_i(t_i^\lambda, t_{-i})]$.
- w^λ is the probability distribution of $f(t_i^\lambda, \cdot)$, i.e., for every $a \in A$, $w^\lambda(a) = Pr_{t_{-i}}[f(t_i^\lambda, t_{-i}) = a]$.

Assumption 2: w^λ is continuous and differentiable in λ. (This assumption is not really needed, but allows us to simply take derivatives and integrals as convenient.)
Then, we have:

1. The expected utility of player i with type t_i^λ that declares $t_i^{\lambda'}$ is given by $v^\lambda \cdot w^{\lambda'} - p^{\lambda'}$.
2. Since a player with type t_i^λ prefers reporting the truth rather than $t_i^{\lambda+\epsilon}$
$\Rightarrow v^\lambda \cdot w^\lambda - p^\lambda \geq v^\lambda \cdot w^{\lambda+\epsilon} - p^{\lambda+\epsilon}$
3. Similarly, a player with type $t_i^{\lambda+\epsilon}$ prefers reporting the truth rather than t_i^λ.
$\Rightarrow v^{\lambda+\epsilon} \cdot w^{\lambda+\epsilon} - p^{\lambda+\epsilon} \geq v^{\lambda+\epsilon} \cdot w^\lambda - p^\lambda$.

Note, that 2 and 3 follow from strategy proof property.
Re–arranging and combining, we get: $v^\lambda(w^{\lambda+\epsilon} - w^\lambda) \leq p^{\lambda+\epsilon} - p^\lambda \leq v^{\lambda+\epsilon}(w^{\lambda+\epsilon} - w^\lambda)$. Now divide throughout by ϵ and let ϵ approach 0.
$\Rightarrow \frac{v^\lambda(w^{\lambda+\epsilon} - w^\lambda)}{\epsilon} \leq \frac{p^{\lambda+\epsilon} - p^\lambda}{\epsilon} \leq \frac{v^{\lambda+\epsilon}(w^{\lambda+\epsilon} - w^\lambda)}{\epsilon}$
$\Rightarrow v^{\lambda+\epsilon} \to v^\lambda$; $\frac{w^{\lambda+\epsilon} - w^\lambda}{\epsilon} \to \frac{\partial w^\lambda(\lambda)}{\partial \lambda} = w'(\lambda)$
and thus the derivative of p^λ is defined and is continuous and equal to $w'(\lambda)v^\lambda$. Integrating, we get $p_1 = p_0 + \int_0^1 v^\lambda w'(\lambda) d\lambda$.

□

The revenue equivalence theorem tells us that we cannot increase revenue without changing appropriately the allocation rule (social choice function) itself. However, if we are willing to modify the social choice function, then we can certainly increase revenue.

Example: Assume two players(bidders) with valuations distributed uniformly in [0, 1]. Put a reservation price of 1/2, and then sell to the player with maximum bid for a price that is the maximum of the low bid and the reservation price, 1/2. If both players bid below the reservation price, then none of them wins. Otherwise, the player with maximum bid wins and pays the maximum between 1/2 and a second price. Then a quick calculation will reveal that the expected revenue of this auction is 5/12 which is more than the 1/3 obtained by the regular second price or first price auctions. Let us calculate the expected revenue of this auction. Let us analyze three cases:

1. Two players bid below 1/2. This happens with probability 1/4 and the revenue is 0.
2. Both bids greater than or equal 1/2. This happens with probability 1/4 and the expected revenue is the expected value of the lowest bid assuming that both are greater than 1/2 - $E(min(x, y) \mid x, y \in [0.5, 1])$. Then the expected revenue is 1/2+1/3*1/2=2/3(second price).
3. One player bids above 1/2 and second bids below 1/2. This happens with probability 1/2 and the revenue is 1/2.
   Putting all this together, we get that expected revenue of this auction is:
   1/4*0+1/4*2/3+1/2*1/2=5/12.

References:

[1] Vijay V. Vazirani; Nisan, Noam; Tim Roughgarden; va Tardos (2007). Algorithmic Game Theory. Cambridge, UK: Cambridge University