Title: 시장설계이론1, 이산 자원의 분배 (3)

- ✓ **Instructor: 최연구**
- ✓ **Institution: 연세대학교**
- ✓ **Dictated: 강은경, 강성호, 김신희, 김종백, 신원대, 현소형**

[00:00]

Last class, there were one more in the Friday.
So my plan is to finish up random assignment, mechanism today maybe even turn to a school choice
application even today's class.

But if not, at least we will deal with that in friday's class.

So, this step, problem step number 2 was assigned last Friday and due this Friday.

And then there were another problem set that will be assigned this friday, it will be due next Wednesday.

Because there is no more class I want you to, I want to keep you up occupied at least with the little bit more of the matching theory

And then I am planning on a having an exam, maybe week from next Wednesday.

Maybe using our class, maybe two and half hours, I will give you more about what's involved.

But you know, what I am thinking at least is that it will be based on problem sets as well as course material.

I mean, I am just giving you incentive to go over the material and sort of remember the argument.

So my plan basically, is that the there not be questions asking you to prove something that is really complicated and very difficult or not so important.

But among the material, frist of all that I am covering, I am discussing in the class or I have discussed I the class.

Result that are sufficiently important and prove and, who's proof are not very complicated,

I am hoping that remember the arguments enough.

So it's one thing to understand by going through the proof.

I think quite another to understand it well enough to remember.

So I want you to reach the second stage, you know, being able to even replicate the argument by yourself.

And this certainly requires a little bit of training and memorization to a degree.

In the beginning, that's what it takes.

Later on, you may forget about the argument, but next time within a few years, you look at it again, it now comes to you much quicker.

You know, at that point , it looks to you to be sort of very intuitive argument.

So that's kind of the part of trainning.

That's involved something that I am hoping to, I am intending to faciliate by giving you the exam.

So that's what's involved.

I am debating in my mind, whether to assign you additional sort of paper summary, I think I should probably do that also.

So my philosophy of teaching is that certain degree of torture is needed.

I mean, it is all very paternalistic and very well sort out the philosophy I think, in my mind, something that, you know,

That's jointly beneficial, also beneficial to you.

But you know, It's hard to commit yourself, subject yourself to the stable,,really push yourself,

You are willing to push your self, challenge yourself but, you know, certainly additional incentive cannot help.

So I engage that , if I torture you enough, at some point, you know, develop the Stockholm Syndrome

which is somehow get in to like a torturer and material.

So anyway,

Its all very gentle by the way, so in terms of what we talked about last time, we talked about,

let me just make two remarks, also sort of resense if you will.

We talked about serial dictatorship not random dictatorship just general, serial dictatorship with arbitrary serial order.

Now is there any connection between particular serial dictatorship and something that we had studied

before in the two-sided matching?

And the answer is yes,

Serial dictatorship coincides with a particular form of deffered acceptance algorithm.

[05:00]

So if you remember what happens in the serial dictatorship, there is serial order.

So the order of agent, okay, and then agent of course has, each agent has his own strict ranking, preference ranking of the objects.

So on one side, we have the right ingredients, we needed for deffered acceptance algorithm.

Now what, on the object side, you can trick objects and as agents on the other side.

There are sort of fungible objects and you can sort of think of them as agents on the other side.

They are matched to the agent, the real agent.

Now what about their preference ordering?

Okay, so to have one deffered acceptance algorithm, we need their preference ranking.

The serial order in fact, of the agents other preference ranking on the side of objects.

So in some sense, serial dictatorship is a special case of deffered acceptance algorithm, where all objects are seen as agent.

So on one side, have the same preference ranking that coincide with serial order.

In other words that, serial order once like i1 i2,,,in, there are n objects, let's say

And this is serial order, that means that, all objects like this guy is the best, this guy is second best, and so on.

And think about running, then now, run the deffered acceptance algorithm,

So let's see, which one to one,

Okay, so, suppose that, we run, let's try just one see if that works.

Let's say agents, agents all propose that,,

Let's do the object first, if the objects all propose to

That's easiler one, you can do the other way as well.

They will both produce same.

Because this actually, corresponds to the same the problem you did, the last problem set of the first, last problem with first problem set,

where if you have a common ranking on one side, then you have unique stable matching.

So it's either sort of version of the deffered acceptance algorithm, result in the same matching.

But one is easier than the other right.

So as you might have seen maybe not yet, but I mean you see in the answer,

one version is easier the other version is slightly more involved into use inductive argument.

So suppose you run this one,

I think this is easy one , then so every object now, proposed to i1.

Because they have same ranking.

i1, every object propose to i1,

Then i1 choose whatever his best, and then reject everything else.

And so, therefore, now we have one pure object availiable,

And all those guy now have to repropose to the agent which have not yet, who has not yet rejected.

And that means that, every agent except for that guy, and then who there repropose to, this guy.

So you can see that, what's happening essencially, with the object proposing deffered acceptance

algorithm is excatly same thing as serial dictatorship.

[student speaking]

Say that this, so, of course, when object propose,

Are you talking about serial dictatorship or deffered acceptance?

[student speaking]

So now, way that, I mean, there is no number of, there is no [?10:3] no good.

[10:00]

Null good, you can always take.

So let me just one through the scenario, see what happen.

Let's say that, there are some objects,

Let's say that, at some stage, at some round, the best object that, an object that's best remaining object from agent okay, is unacceptable.

So every other object is unacceptable to an agent.

You reach a stage, at which, agent finds no object acceptable.

It can simply take the null good at that point.

And then he is done.

So there is no shortage of null good.

This works also for [?11:6] many to one matching .

Null good can be assigned to many agents if that you asking, that's fine.

So there will be the special in this case.

Yeah, it can be assigned to many agents.

But you can see the coincidence of the two mechanisms.

Because, it is excatly same thing happen right.

So at edge stage, we can learn in this inductively.

So suppose that there are case steps of so, the object're assigned, so the case step, case for step for the serial dictatorship are gone,

And then at that point, you run deffered acceptance, along with same agent.

Up to case, the objects that are signed, up to case step, assigned to the same agent.

And also remove them, and you start deffered acceptance algorithm or you start serial dictatorship,

you get exactly same assignment for the next step.

So you can sort of iternatively argue that you get the same.

And the other thing that I kind of miss, there is one thing, so this is something that I should clarify I forgot about it.

So I am not have done it, now second thing that I kind of didn't speak correctly is this,

particular mechanism called 'you look press at my house and I get your turn' thing.

So everything I said is right, except that when you assign serial order, you assign serial order

to everybody even those who own something.

Because you know, if you assign serial order only to those who don't own any thing,

it could be that those who own something can be stuck to their own object.

So it may not allow them to trade their endowment for better good.

But if you assign, if you, serial order includes everybody, includes those even though those

who own some houses in the beginning, then I mean idea is exactly the same as before.

Okay, you run through the serial order, it could be that the first guy in the turn, serial order may be

the one who own his house.

He may point to some house, if it's not owned, he gets it.

If it's owned, then that owner moves up in the ranking.

Now the guy then claims another one, if that's owned, okay, that guys need moved up in before.

Keep going on to form your cycle, once you have a cycle then you execute the cycle,

and then you go on from their after.

Untill you know, guy who claims house, it's not owned.

There are two things.

So one thing that we, in the random assignment mechanism, big motivation is the fairness,

again I just remind you what we are discussing.

Serial deterministic assignment mechanism, there are two different ways to achieve efficiency

and strategic proofness.

Okay, one is TTC, and the other is serial dictatorship.

But, so, you use serial dictatorship in an environment where there are no ownership structure.

And you use TTC in an environment where there are ownership structure.

So in the first setting, where houses are socially owned, in other words, there are no private ownership,

[15:00]

Then the mechanism to use serial dictatorship which we know pareto efficiency and starategy proof,

unfortunately the outcome is unfair.

So therefore, what you use is random version of the serial dictatorship which is called

random serial dictatorship or random priority rule.

which in addition to satisfying the two properties, pareto efficiency and starategy proofness, is also symmetric.

Treating people with same preference ordering in the same lane.

In what sense?, in the sense that if we take two different individual who have the same preference orders, they end up getting the same lottery of goods.

In other words that they will end up getting that same probability distribution over different alternative objects.

The other thing that I noticed at that point is you could have also achieve the same outcome,

in terms of the collection of lottery is that each individual gets by randomly assigning the objects initially.

One for each agent, and then run TTC.

And turns out that it has shown to produce same collection of lotteries for the agent as random serial dictatorship.

And random priority rule or random serial dictatorship is the most commonly used preference based

random assignment mechanism.

So whenever you have some assignment of describe resources which care to reflect

the preference of agents, you know, I gain more than, more likely than not.

The mechanism would be random priority.

Okay, that's the most widely used mechanism.

Unfortunately, we've seen that the mechanism is not ex ante efficient.

So it's ex-post efficient clearly, once you realize the serial order and run dictatorship,

Then at that point, there is no way to relocate the objects you know, in a way to makes everybody better off.

But at the level of lotteries, from the ex ante perspective, you can do better.

Okay, so this is an example, I mean, this doesn't aright all the time.

But you could find an example where random priority mechanism or random serial dictatorship produces lottery for the agent, that entail inefficiency.

So we mention a little bit about the nature of ineifficiency.

It's something called ordinal inefficient.

So, let me just define ordinal efficiency and inefficiency means, to talk about ordinal efficiency, we have to talk about ordinal domination.

So random assignment, what is random assignment?

Random assignment is collection of lottaries one for each agent.

So the easiest way to describe it in the house allocation problem, in particular when you say house allocation is n object, assign to be assigned to n agent one for each.

Okay, so every object is acceptable, you know, the standard some sort of very special case, we are dealing with a little more general case.

In other words that we are dealing with the case where the number of agent do not match up does not equal to the number of object,

So that, some agent may be unassigned, some object can be unassigned, that's fine.

But when you talk about house allocation, we are dealing with the special case of that where the numbers are each the equal.

Okay, so P, we describe the random assignment by P, okay, and you can think of this as matrix.

So matrix, the each row represents agent I, each column represents object a okay.

And so Pia represents the probability that agent I gets object a.

If you read through the given row, what you get, you get a lottery of object, probably distribution over different objects that agent I gets.

So, in the house allocation problem particular, this must sum up to one clearly.

Even in a more general case, where we allow agent to be unassigned, we can add the possible null there.

[20:00]

And then, require the each row, sums to one.

In the house allocation problem, each column sums to one as well.

Because each object must be assigned to some agent.

But in general, that may not be the case.

In general, in other words, what you mean by general, in a situation with number of object is not equal to the number of agent.

So in that case, this is, must sum level, number that's less than equal to one.

So the agent, object a may or may not assign

We can insist upon each agent lottery sums to one by adding a null good.

So the matrix of this form, when its row sums to one, column sums to a number less than equal to one.

It's called substochastic matrix.

But in the house allocation problem, where each row sums to one and each column sums to one,

we call it bistochastic matrix.

The other one is called, more general case is substochastic matrix.

So, that's what we mean by random assignment.

Anyway, so we say, random assignment P, ordinally dominates another random assignment, so think of first of all, bistochastic matrix.

But the same should be the case for substochastic matrix as well.

If for every agent [?21:56] assignment on the P, so we are talking about row by row to matrixes focus on i through for i-th agent.

Then we are talking about two different lotteries.

On the assignment, on the P, her lottery on the P, the i through P, first stochastically dominates those under P', there same i through under P', matrix P'.

According to her preference, if this is the case, what is this,

We already fixed I, for any good a, the probability that agent I gets a or better good under P should be no less than the probability that the agent I gets a or better good, under P'

We require this to be the case for all a.

So the probability of getting a good, in the any upper contour set.

upper contour set define relative to any arbitrary good, okey?

should be higher under P than under P^2, okey?

in that case we say and this is true for all"I", okey?

in that case we say P ordinally dominates P^2 or P first pof the stochastically dominates P^2.

so people use different, two different terminologies.

and then we require at less one strict inequality for at less one pair of I and A, okey?

because if you this is all same for all "A" then we're talking about the same matrixes, right?

a random assignmnet is ordinally efficient.

if it is not ordinally dominated by any other random assignment, okey?

so, so I think to know here it is that if P ordinally dominates P^2, okey?

then if we, we haven't talked about cardinal preferences yet.

we our position in terms of describing preferences so far is that only ordinal preferences are well defined.

but we can imagine yourself world where everybody is in doubt with Von Neumann–Morgenstern utility values, okey?

so we are sort of entering into a world of cardinal preferences.

now as long as that Von Neumann–Morgenstern utility the numbers that we assign, we can do so,

we are free to assign any Von Neumann–Morgenstern utility number that we want.

subjects to the constrain that they are consistent with ordinal preferences, okey?

so in other words that if one agent likes A more than B, his Von Neumann–Morgenstern number for A must be more higher than that of B, okey?

as long as you assign any Von Neumann–Morgenstern utility numbers subject to this constraint of preserving ordinal preferences.

[25:00]

then if P ordinally dominates P^2 then P pareto dominates P^2 as well, okey?

ex ante pareto dominates, okey?

ex ante meaning that in terms of expect utility.

so we calculate expect utility for everybody, then everybody's expect utility is under P should be higher

that everybody is expect utility under P^2, and strictly for at least one agent.

and strictly for at less one agent.

so in some sense

so if P ordinally dominates P^2 than regardless of how we determine regard, no matter what the cardinal preferences that may have, okey?

P pareto dominates P^2 in terms of ex ante expect utilities, okey?

now in a sense of stronger notion.

now suppose that P does not ordinally dominates P^2, okey?

but P^2 does not, I mean, so they are ordinally incompararble, okey?

P and P^2 are ordinally incomparable meaning that neither ordinally dominates the other which is possible.

because final relation given by ordinal domination only forms partial order, not a complete order, okey?

so in that case even in that case it's possible that cardinally P might pareto dimates P^2, okey?

so in some sense, ordinal domination is a more conservative notion of welfare, welfare comparison.

if, even though two random assignments may be ordinally incomparable, it may be that under some cardinal preferences.

one cardinally dominates the other.

that's possible, okey?

but you know, but if one ordinally dominates definitely then the former cardinally dominates the later, okey?

no matter how we assign cardinal preferences.

and two as an example, we can see that this random assignmnet ordinally dominates that random assignment.

because fix each agent let's say agent one or two, okey?

fix a good and look at the associated upper contour set.

what's the probability of getting A are better.

it's one half here.

it's something less than one half here, okey?

that's fine.

now we checked one.

what about the upper contour set corresponding to object B.

well, the probability getting B are better is one half here.

and one half here.

so it's still consistent with this lottery first of the stochastically dominating that lottery.

two here, I mean if in this case, fix A, A are better for this second dime, it's one half, it's one half, right?

because this guy like that one.

we'll be up to evaluate they according to the preference of each guy, right?

upper contour set is define according to preference clearly.

it’s one half, one half the same.

look at here, one half, something less than one half, okey?

so you can see that for each agent the lottery that one gets from here.

first of the stochastically dominates the lottery that one gets here, okey, according evaluate

according to each agence preference, ordinal preference, okey?

therefore and there is a strict inequality clearly, right?

so therefore this random assignment ordinally dominates that random assignment, okey?

and further you can show actually that you can kind of intuitive C that this assignment is ordinally efficient.

not only that we'd ordinally dominates I mean, one could, one can dominates ordinally something else and can still the ordinally inefficient.

this case, in fact it is ordinally efficient as well.

there is no assignment.

no other assignment ordinally dominates that guy, okey.

so here is the characterization of ordinally efficiency.

because ordinally efficiency itself is so nice intuitive notion.

[30:00]

it may be difficult to check however, okey?

so it may be useful to have a different way of checking to see if a given random assignment is ordinally efficient or not.

and this lemma due to Bogomolnaia and Moulin offers characterization meaning ordinally effiiciency meaning that here are alternative ways of cheking for ordinal efficiency.

and hopefully these are easier to check than ordinal efficiency itself.

so one is non-wastfulness, okey?

what he says is this, okey?

if one agence say "I" if there's an agence "I" who strictly prefers A over B.

and yet consumes B with positive probability, okey?

consumes B means that there lottery that he gets.

so we are talking about ordinal efficiency of particular random assignment.

so we up to fix a random assignment P first, okey.

this is what's we going on in P, okey?

so if in P under P if agent "I" has a lottery.

the lottery that he gets under P gives him good B with positive probability.

that's what we mean by he consumes B, okey?

so we sort of use the term consuming means that gets that good with positive probability under P.

if that were the case and if in addition, he likes A strictly more than B,okey?

then A must be fully assigned, okey?

A must be fully assigned, okey?

meaning that that the probability of A being assigned to somebody must equal to one.

if some, in other words that the column corresponding to A must some to one.

for P to be ordinally efficient, okey?

that has be the case.

now if you think about a situation where of the house allocation problem, okey?

where every good there are same number of goods as does as the number of agents, okey?

and every good is acceptable to every agent, okey?

in that case, suppose that we are focusing on a random assignment which forms a bistochastic matrix, okey?

then this will be automatically satisfied.

because every object is consumed fully.

there is no object that is not fully consumed, okey?

so this has a byte.

I mean we use a term byte means that matters, okey?

relavent only with a mor general environment where random assignment will be descrived the substochastic matrix.

it's possible the some column may not some ot one.

all at saying is that object, okey.

whose column associated with a column doesn't some to one, better be better not be the object.

that strictly better than something that is consumed by somebody, okey?

so second requirement a little bit more interesting requirement in a sense.

it is called acyclicity like the terms suggest

that is a requirement that they should not be a particular form of cycle, okey?

to determine to define a cycle we have to first of all define a binal relation.

so we are ging to define a new binal relation, okey, with a different rotation, okey?

so again this think of as a another form inequaility okey?

binal relation on "O" given under, so we habe to again fix the random assignment.

so binal relation depends both on the preferences as well as on of the random assignment in question.

so we have to have random assignment to be able to talk about binal relation.

we need to have random assignment as well as preferences fixed to talk about a binal relation.

so binal relation related to set up preferences and and the underlying random assignment, okey?

we say A dominates B or A is bigger than B according to this binla relation.

if there ecists an agent such who likes A more than B.

and yet consumes B with positive probability, okey?

that's a definition of A dominating B.

sort of mouthful but you know it's feel intuitive.

[35:00]

so A is bigger than B it there is an agent who likes A more than B and yet consumes B with positive probability under this assignment, okey?

this is this is the part of well, we use a particular random assignment.

without being a random assignment we cannot say talk about that.

so[35:28] means that should not be a cycle given, defined by this binal relation, okey?

so what's the cycle, cycle this is a binal relation define over object, right?

and cycle will be therefore a list of objects, okey?

such that A dominates B, B dominates C, C dominates D and so on and so forth.

and then finally the last guy dominates A.

the cycle define according to binal relation.

now what's the significance of there being such a cycle.

well, what does he say?

there is a guy who likes A more than B and yet he gets B with positive probability.

and there is a guy who likes B more than C and yet gets C with positive probabiltiy and so on and so forth, okey?

now once we have cycle of this form, we can kind of C that they can trade according to along the cycle.

so imagine that A, B and let's say C and A, okey?

then so, okey, so this similarly exchange, okey?

there is an gent I, J and I, okey?

so he likes A more than B and yet he gets positive share of B.

J gets B more than C and yet he likes, he has, gets share of C, okey?

and so on

so what can we do?

let’s say the probability of "I" getting B is something, probability of "J" getting C something probability of K getting A something.

choose the minimal one, get smallest one, okey?

so let's say that's the probability of A, okey?

PIB, PJC, PKA okey?

take the minimum of the three.

and let's call P head, okey?

now here is what we can do, we can have B, A.

we can, we can have "I",okey?

give up this much probability share of A to K, okey?

and then we can have K give up I mean, and then K give up.

so K I'm sorry.

that's not what I meant

so we are, we can have K give up.

this much share of A and give it to J, right?

no give it to A, give it to "I", okey?

and then we can have J give up, this much share of C and give it to K, okey?

we can have "I" give up this much share of B and give it to J, right?

right, okey?

now we once do that everybody give up this much something and gains back much of different thing, okey?

once we go through this trading, okey?

what happens is that each agence gives up this much of less prefered good share of this much share of less prefered good.

in again the same share of more prefered good, okey?

so everybody is better off, okey?

the fact that we can find such a cycle means that there is a room for improving everybody's welfare sort of trading a probability shares.

so the example that we dealt with before is a special case, right?

sepcial case here, okey?

so he will likes A more than B and yet has a share well, this guy one and two, okey?

[40:00]

so in this case we can say A dominates B.

because there is an agent who likes A more than B and yet under assignment RP assignment gets positive share of B, okey?

and then for this guy then also this is true.

because there is an agent who likes B more than A and yet has a positive share of less prefered good, okey?

and then they could trade to arrive there.

so the fact that you cannot find such a cycle is a good thing, in fact.

says something about something nice about the underlying random assignment, okey?

so that's the requirement of acyclicity.

we cannot find a cycle defined by this binal relation.

so in fact what they show is that ordinal efficiency is equivalent to this pair of a condition.

so in other words if random assignment is ordinally efficient then that will that random assignment must be non-wasteful and acyclitic

acylitic,   acyclic, okey.

and and then if a random assignment the non-wasteful and sarisfied acyclicity.

then it must also be ordinally efficient.

so what we do is to apply this core character relation from then from now on to see if a random assignment is ordinally efficient or not.

now here is second random assignment machanism.

so we learned one which is RP, which we already show is some nice properties ex post efficiency and symmetric and strategy proofness and yet may be inefficient ex ante in the ordinal sense, ordinal inefficient, okey?

here is a machamism that is designed to do better In terms of ordinal inefficiency, ordinal efficiency.

but at the expense of something.

why because, why do we know that we cannot be perfect really now that we cannot be perfect because of this impossibility result, okey?

so there is no machanism achieves ordinal efficiency symmetric and stretegy proofness.

now ordnial efficiency implies actually ex post efficiency, okey?

it's not clear cut.

but that's in fact the case.

but it's important in one sense.

so if a random assignment arbiturary matrix like this is ordinally efficient, okey?

then actually I've been talked about the composition.

but then you can there is an way of realizing.

let me not talk about, I'm going to talk about it little bit later when I talked about once I've talked about the composition issue.

but there is a way to..

so if you are given a random assignment, okey?

suppose you are given assignment.

now we are asked to implement it, okey?

it’s not trivial to implement it okey?

so let me show you why it's not trivial to implement it.

but what I'm saying is that if you given a random assignment.

there is ordinally efficient.

there is a way ti implement it in a ex post efficient way, okey?

in that sense ordinal efficiency implies ex post efficiency.

and we know that you know.

so RP satisfies this that plus ex post efficiency but not this, okey?

the one that I'm about to present to you satisfies ordinal efficiency and symmetry in fact just satisfy stronger form of symmetry or fairness.

but it will not satisfy strategy proofness, okey?

here is how we define score simultaneous of eating algorithm or probalistic serial machanism, okey?

that's the name.

so RP is one, this is PS, okey?

it's based on simultaneous eating algorithm.

now the idea basically is that in sort of thinking about which good and as indicisible eventhough it is invisible.

think of it as divisible, okey, in one unit probability units.

we can talk about point 3 share of object A okey?

[45:00]

so we can imagine agent I obtaining the good A with probabilistic point three.

we can think of it as an event where agent "I "consume, okey?

point three fraction of good A.

so think of its good as being divisible in the sense of probability.

and imagine that the time runs continuously from zero.

so it's sort of dynamic machanism.

time runs from zero to one, okey?

and imagine that its agent eats at unit's bit, okey, from time zero one to time one, okey?

if you eat at unit's bit from during this, duration of a time interval.

at the end of that once you reach time equals with one, you at eaten a unit share of the good, okey?

time equal zero one. You let eaten unit share of the good. Okay?

So, at the end of day, you will get a rotary. That's sum of 2, 1. Okay?

Now, how do you eat? You eat at each point whatever is remaining over given good.

Assuming as step good is your best good. Okay?

So, in other words, at time zero, you start eating the most prefered good. Okay?

Now, it may not be the most prefered good for just you. It may be the most prefered good to many others. Okay?

Let's say that K, number of agents who find particular good. Let's say A to be the most prefered good. Okay?

In that case, each start eating the good. Right? At unit speed. Okay?

So, what that means is that if there are K people eating good A, how long will the object A last? Okay?

One of K times. So, time interval here. Now, only K people are eating at unit speed.

They run out of the good. Because each good is available up to one unit. Okay?

So, we decided to regard each good as divisible. Right? There is unit capacity of each good.

That's all exhausted at time one of K, because each agent eating at the smae speed. Okay?

Without 2 agents they run out of that good at time equal to point 5. Okay?

But then, it's more complicated. Because there may be other good.

So, bunch of people are eating A, bunch of people are eating B.

But, if more people are eating A, then A will be exhausted first. And then, what do they do?

Once you run out of A, they switch it to the most prefered available good. And start eating.

So, that means that if more people are eating A than B, let's say two people are eating A.

And, three people are eating, I'm sorry. Three people are eating A. And two people are eating B. Okay?

Then, at time 1/3, A is done. Okay? So, this is more than that. But then this is A.

And then, A is done. But then what do they do? They then turn to B. Okay?

So, B has been eating by two people up to that point. Okay? So, there is some.. How much of B left? Okay?

So, 1/3 times. So, 2/3 has been eaten. 2/3 of B has been eaten by that time.

Because, there is 1/3.. Two people are eating. Right? So, one guy is eating 1/3 of B.

The other guy is eating another 1/3 of B. Okay? So, 2/3 of B are enough that uh.. 1/3 of B remaining. Okay?

But, from then on, these two are not the only guy is eating. Because the first three guys

who have been eating A, once we are done with A, they will switch to eating B.

Okay? There are 5 of them now eatiing B. B will be consumed very quickly. Right?

So, this is how it works. Okay? Right. Okay. So, once you run thorugh this mechanism,

and, reach time period one, you can look at really what each person has been eating. Okay?

And, how long. Okay? Now, you'll see that at the end of day, the total share of good that each person has eaten is equal to one.

[50:00]

So, it's nice in the sense that the you know shares of different good sum to one. Okay?

You get a nice rotary for each agents. And then he might have eaten, spent first /3 of time eating A.

Second, sometime I don't know may be you know up to one half he has been eating B.

Up to some probability, let's say four/fifth he has been eating C. And then remaining time. There is nothing left.

He has eaten all good. Okay? So, what does that mean? That is that his probability share is that

the rotary they will get is that he gets good A with the probability 1/3.

He gets good B with probability 1/6. Okay? 2 half minus 1/3.

So, you count up the duration of time spent for eating a given good. And that's the probability of getting that good. Okay?

Here in this case, I don't know. It's uh.. 8 minus.. So 4/5 one half is 8*5/10.

So, here is the probability of getting C is 3/10. Okay? That's how it works.

And, we can show that this mechanism once everybody report truthfully of course that turned out not to be the case,

assuming that everybody report truthfully about their ordinal rankings.

I mean so relative to the preferences that they state. Okay?

There also random assignment resulting from PS is ordinaly efficient. Okay? That's what we're going to show.

Okay. So, here is my claim. Once you.. We have so far looked at what happen on the PS

from the perspective agents. We can now look at what happens on the PS from the perspective of each object.

Okay? Each object is eaten at some point, at some time. And then at some point it's all eaten away. Okay?

Now, at the time point at which an object disappears. You can think of the and define as expiration date. Okay?

An object expires at a given point between 0 and 1. Okay? Once you pin down those, okay?

Once you pin down those, once you fix the expiration date for each object, we will have determined the random assignment for every agent. Okay? Of course to do so, we need to have ordinal preferences.

Give me ordinal preferences, give me expiration date, we will pin down everything. Okay?

So, here is the way. So, capital Ta, we use this notation to denote the expiration date for object A. Okay?

Okay. So, let's say object. Here is the.. So, let's just you know so change things.

Let's say this is the expiration date for B. This is the expiration date for A. Okay?

That's turns out to be the.. Look at what happen to the object and see when each object goes away. Okay?

And this turns out to be the case. Let's say. Okay? Now, give me any ordinal preferneces,

I can determine the rotary of good that you know, give me any agent and his ordinal preference,

then I can determine the rotary that he gets. Okay?

Say that his preference is.. Let's say. Let me give you 2 examples. Let's say agent I.

Agent I likes A more than B more than C. Okay? Let's say that you know.

There are 3 goods and in fact, Tc is equal to 1. Okay? It could mean that you know..

This may not be uh.. The object C may not be fully consumed by anybody.

Anyway this is 1. In other words that whoever wants to get C you know has no problem getting it. Okay?

C is available until the last end. I mean until the end. Okay?

[55:00]

So, what's the probability of this guy getting A? Okay. Let me just do the easy one.

And then, do slightly more interesting one. Let's start with B, A, C. Okay?

What's the probability for this guy getting B? That's TB. Right?

That's the probability of this guy getting B. That's TB. What's the probability of this guy getting..

Why? Because if he likes B, so he will start eating B. Okay? Until when?

Until he runs out B. And this is when B runs out. Okay? And the duration of time is spent eating B is TB.

What about A? That's TA - TB. Right? What about C? That's TC which is equal to 1 - TB in this case. So, very easy. Okay?

Now one of guy J whose preference is actually ABC. He likes A more than B more than C in this order. Okay?

Now, he starts eating A. So, what's his probability of getting A? That's TA.

Now, what's his probability of getting B? 0. Because he starts eating A. And he is done here.

And, he is about to switch to eating B. Only to realize at that point that these are all gone. Even gone before A.

Okay? He cannot consume B. Okay? And C is exactly same TC. However minus TA. Okay?

So, give me.. Once you pin down the expiriation date for each object, and give preference,

I can determine the rotary. Okay? What this means is that given the set of profile of preferences,

All you need to know to see what happens in terms of random assignment is the expiration date. Okay?

Now, I am going to prove to you that the random assignment that you get from PS will be nonwasteful.

And then, A cyclic. Okay? A cyclic. So, first random assignment, first nonwastefulness..

What does nonwastefulness say? If A is better than B for some agent I under P, Okay?

That must be that A must be fully consumed, meaning that the column associated with A must sum to what?

Okay? I am going to prove the contrapositive. In other words that if A is not fully consumed, Okay?

The column associated with A doesn't sum to 1, sum to number less than 1. Okay?

Then it must be that nobody is consuming any object. Okay? That is strictly worse than A. Okay?

So, I am going to prove the contrapositive which is equivalently proving the whole thing, only thing contrapositive meaning.

Not A in plus not let. Not the latter in plus not the former. Okay?

So, imagine that A is not fully consumed. What does that mean in terms of what happens in this algorithm.

At time equal to 1, A is still remaining. So, the expiration date for A is equal to 1. okay?

So, meaning that I mean A is not expiring at the end by the end of the time. Okay?

A remains, so the expiration date for A is this. Okay? So, what that means is the given the fact that the agents are eating in their order of their preferences. Okay? Anything that is worse than A for any agent

will never be consumed. Right? Is that right? So, we are done. You never get there. Right?

[60:00]

If some other object B let's say, is worse for agent I than A. Okay? Then he left before he is done eating A.

Before he reachs B. But he never be done by the end of the time eating A. okay?

So, you never get to B meaning that we count up the duration of time spent by that guy on eating different object

in particular B, it will be 0. Okay? That is meaning that B is never consumed by any people. It maybe.

There are people who like B more than A. They may consume B which is fine. B's expiration date may be

run earlier which is fine. Okay? So, the PS random assignment is nonwasteful in this sense. Okay?

Now, I am going to show you next that PS random assignment is also acyclic. Okay? Acyclic.

To show that, I am going to first show you that if A dominates B, according to this binal relation,

it has to mean A must be expired before B on the PS. Remember to define for the given binal relation

presupposes a random assignment. So, we have a random assignment in mind already. Right?

So, that random assignment is PS random assignment. So, the binal relation will define relative to that random assignment.

According to that binal relation, if A dominates B, then A must be expired befor B. That's what I am saying. Okay?

To show that suppose otherwise, suppose not meaning that TB is TA. Okay? It means that this is the case. Okay?

Suppose that this is true and yet this is the case. Okay? So, meaning that again I mean eventhough I am surpressing

the dependence of this binal relation on the underline random assignment, it is there. Right?

So, remember that. Even if this is true, and yet this is also the true case. Okay?

We are going to generate some contradiction here. So, here is TB and here is TA. I mean they could be the same.

It doesn't matter. Okay? I will show you. I will deal with the case where they are the same a little bit later. Okay?

Now, there is an A guy who consumes B eventhough he likes A more than B. Okay? So, pick any guy. Okay?

For.. Okay. If.. Okay? What is true? If he likes A more than B, what happens under PS random assignment?

You start eating A or something even better than A. Okay? We don't know wether he will be eating A. Okay?

But the point here is that.. Because this A, B may be you know they may be other goods. It may be eaten before or later

even between doesn't matter. Okay? But the point here is that anybody who likes A more than B will never get to

consume B at all. Okay? Because he has something better to eat and something better to eat than B

remains available even after B is gone. So, whenever, whatever he likes the best at the point that time is available.

[65:00]

He will never get to reach a point where he will actually consume B. Because he has something else

that is better off for him that is still available. Okay? So, that means that the probability of I consumes B is 0. okay?

So, that means that we cannot possibly have this.

Because what does that mean? If A is bigger than B, it means there must be who likes A more than B.

And yet, consume B with positive probability. Because whenever a guy likes A more than B,

if this were the case, he will never consume B. Okay? So, we cannot possibly have this binal relation holding.

Okay? And this is true. The argument holds even when the other is same. Okay?

So, let me say this is true. Okay? So, what happens if there were a cycle? Okay?

So, this is O not 0 by the way. What happens if a cycle wer.. This must be true. Right?

Right? This can possibly B set is 5. Okay? Why? Because the order in terms of expiration date

it just order defined on the real number. Okay? Expiration date is a number between 0 and 1.

So, they are linearly ordered. Okay? So, we cannot possibly have a cycle. Okay?

The fact that there is a cycle here means that you know you cannot have a cycle here.

Because we need to have a cycle here, we should have a cycle here. Okay?

That means that random assignment that we will get from PS must be nonwasteful. Okay? And also acyclic. Okay?

And those are sufficient for the underline random assignment B ordinaly efficient.

So, this is proof for the ordinal efficiency of the random assignment that you get from PS.

Let me take 3 minutes break. And then come back.