Intro to the Fair Allocation

CMSC 828M

Rangfu Hu

The model of cutting a divisible cake

- Heterogeneous: different ingredients and different toppings
- Divisible: cut without destroying their values
- Agents: several partners with different preference over different ingredients
- Subjectively Fair: each agent receive a piece, that he or she believes to be a fair share
- A problem of dividing a divisible, heterogeneous and desirable resource is called fair cake-cutting, can be used to other resources: land estates, advertisement space, broadcast time

The model of cutting a divisible cake: Math

- The cake is the interval $[0,1]$
- Set of agents $N=\{1,...,n\}$
- Each of agent has a valuation function V_i over pieces of cake
- Additive: if $X \cap Y = \varnothing$ then $V_i(X) + V_i(Y) = V_i(X \cup Y)$
- $\forall i \in N$, $V_i([0,1]) = 1$
- Find an allocation $A = A_1, ..., A_n$

Fairness Definitions

- Envy Free(EF):
  - Each agent receives a piece that values at least as much as every other piece
  - $\forall i,j \in N$, $V_i(A_i) \geq V_i(A_j)$
- Proportionality(PR):
  - Each agent receives a piece that values at least 1/n of the value of the entire cake
  - $\forall i \in N$, $V_i(A_i) \geq 1/n$

Example cake:

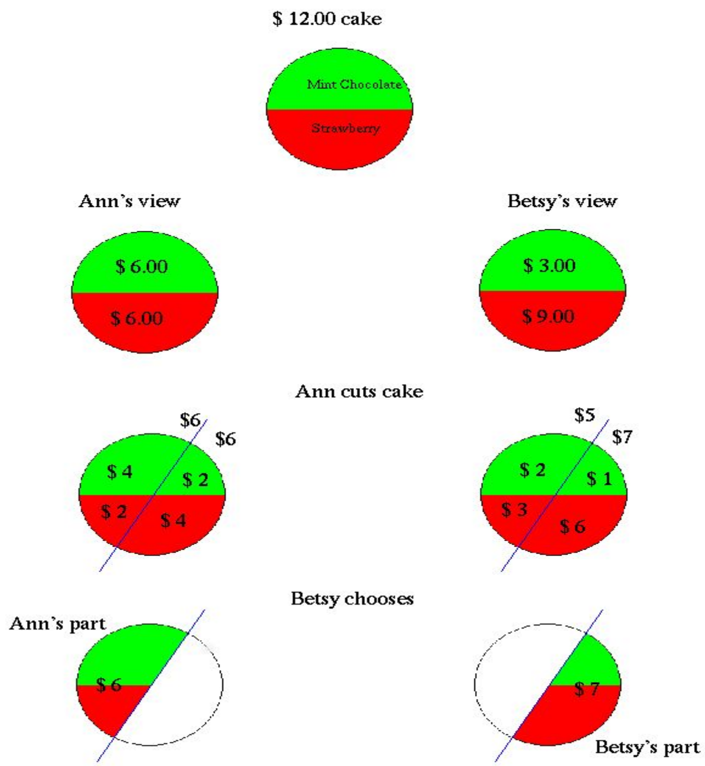
- The cake has two parts: fruit and cookie
- Two agents: Alice and George
- Alice values the cookie as 0.9 and fruit as 0.1
- George values the cookie as 0.3 and fruit as 0.7
- Give all cookies to Alice and all fruit to George
- EF and PR

Fairness Definitions: with additive

Agents	Valuations	EF implies PR?	PR implies EF?
2	additive	Yes	Yes
2	general	No	Yes
3+	additive	Yes	No
3+	general	No	No

- For n=2 with additive, EF and PR are always equivalent
- For n> 2 with additive, EF can implies to PR but not the other way

Cut and Choose Algorithm: 2 players



- A cake with strawberry(½)and chocolate(½)
- Ann's valuation: strawberry(½) and chocolate(½)
- Betsy's valuation: strawberry(¼) and chocolate(¾)
- Step 1: Let Ann cut the cake into two pieces, where those two pieces has the same value from Ann's valuation
  - For Ann: Each of the two pieces worth ½
  - For Betsy: one piece worth 5/12 and the other 7/12
- Step 2: Let Betsy choose one piece and Ann will get the rest one
  - For Ann: get ½
  - For Besty: get 7/12 EF and PR

Cut and Choose Algorithm: 3 players

Stage 1: Cut

1. Player 1 divides cake into 3 equal pieces
2. Player 2 trims the largest pieces such that the remaining part of this piece equals to the second largest pieces
3. Now we call the trimmed part cake 2 and the rest forms cake 1

Cut and Choose Algorithm: 3 players

Stage 2: Choose Cake 1

Players are going to choose in order of player 3, player 2, and player 1

1. Player 3 choose the largest piece
   - Choose the trimmed remaining piece
   - not choose the trimmed remaining piece
2. Player 2 will choose the trimmed remaining pieces if player 3 didn't

Either player 2 or player 3 is going to choose the trimmed remaining piece; call that player T (trimmed)and the other T'

3. Player 1 chooses the remaining(untrimmed)piece

Cut and Choose Algorithm: 3 players

Stage 3: Allocate Cake 2

Cut:

Player T’ cut cake 2 into 3 equal pieces

Choose:

Players are going to choose in order of player T, player 1, and playerT’

Cut and Choose Algorithm: 3 players

EF for cake 1:

Cut: player 1 cut and player 2 trimmed; Choose order: Player 3, player 2, Player 1

1. Player 3 chooses first shouldn't envy player 1 or player 2
2. Player 2 likes the trimmed remaining piece and the original second largest piece the same and also better than the third piece, so player 2 will not envy player 1 or player 3
3. Player 1 like those two untrimmed piece the same of $\frac{1}{3}$ and also better than the trimmed remaining piece, so player 1 will not envy player 2 and player 3

The allocation of cake 1 is EF

Cut and Choose Algorithm: 3 players

EF for cake 2:

Cut: player T’; Choose order: Player T, player 1, Player T’

1. Player T choose first so shouldn’t envy player T’ or player 1
2. Player T’ is indifferent weighing the three pieces of cake 2, so player 2 will not envy player T or Player 1
3. Player 1 choose before player T’ so will not envy player T’; even if player T gets the whole cake 2, it will be totally $\frac{1}{3}$ of the whole cake combine the allocation of cake1 and cake2 to player T, so player 1 will not envy player T

The cut and choose algorithm is EF for 3 players

Successive Pair Algorithm: n players

1. Recursively divide the cake for n-1 players to get their pieces
   - Assume everyone is happy to divide among n-1 player with $\forall i \in n-1$, $V_i \geq 1/(n-1)$
2. Let n-1 players cut their own pieces into n equal pieces and let the last player n join, each piece for every player: $\forall i \in n-1$, $V_i \geq (1/(n-1))/n$
3. The last playerN will choose 1 largest piece separately from n-1 player's part
4. The n-1 player will get the remaining n-1 equal pieces from their own part

For n-1 player: $V' \geq (n-1)*(1/(n-1)/n)$, so $V' \geq 1/n$ for them.

For Player n: $V\text{-}n \geq 1*(V1/n) + 1*(V2/n)\ldots + 1*(V(n-1)/n)$

$=(V1+\ldots+Vn-1)/n$

$=1/n$, where we can guarantee PR for it but not necessarily EF

Continuously Moving Knife Algorithm

- Moves the knife continuously across the cake until some player say stop
- This player will get this piece
  - Tell lies means risk of gaining less in this game
  - $V=1/n$ or this player will not yell out
- The rest player continues
  - $(n-1)$ will get $1/n$ or they will not yell
  - But for the last player:
    - He never yells: those $V(1, N-1)<(n-1)*1/n$
    - The remaining value $Vn>1-(n-1)/n$
    - The last player will get $Vn>1/n$, PR
- However, EF can not Guaranteed here

Maximin Share

$60

$40

$14

$5

$10

Step 1: Let player1 to put items into 3 bundles

Step 2: Player 1 get the least valued bundle (bundle with minimum value)

Maximin Share

Total: $60

$60

Total: $40

$40

Total: $29

$14

$5

$10

Maximin: player1 needs to find a strategy which will maximizes the value of minimum valued bundle: here is $29

Maximin Share Guarantee

MMS guarantees that of player *i*:

- $\forall n \geq 3$ there exist additive valuation functions that do not admit an MMS allocation
- It is always possible to give each player at least $\frac{2}{3}$ of his MMS

--Procaccia & Wang[EC-2014]

p.s possible of giving each player at least 3/4 of MMS has been proved

--Ghodsi et al. [EC-2018]

Maximin Share Guarantee: Algorithm for ½ MMS

1. Allocate those items with value greater than ½ MMS to players and ignore this part: Left items each values less than ½ MMS
2. Start to put those items into bundles, when one bundle first reaches the value of ½ MMS for one player, this player is going to yell stop and get this bundle.
3. Repeat this process until all items in bundles has been allocated to the rest n players

Look at one of those bundles:

1. Call this bundle A of several items: $V(A) \geq \frac{1}{2}$ MMS
2. Call the last piece into this bundle item b: $V(A-b) \leq \frac{1}{2}$ MMS, $V(b) \leq \frac{1}{2}$ MMS
3. The allocation for n players: $V \geq n*MMS$
4. $V(A)=V(A-b)+V(b) \leq$ MMS, $\frac{1}{2}$ MMS $\leq V(A) \leq$ MMS
5. ½ MMS is guaranteed for n players in the allocation of smaller items in bundle