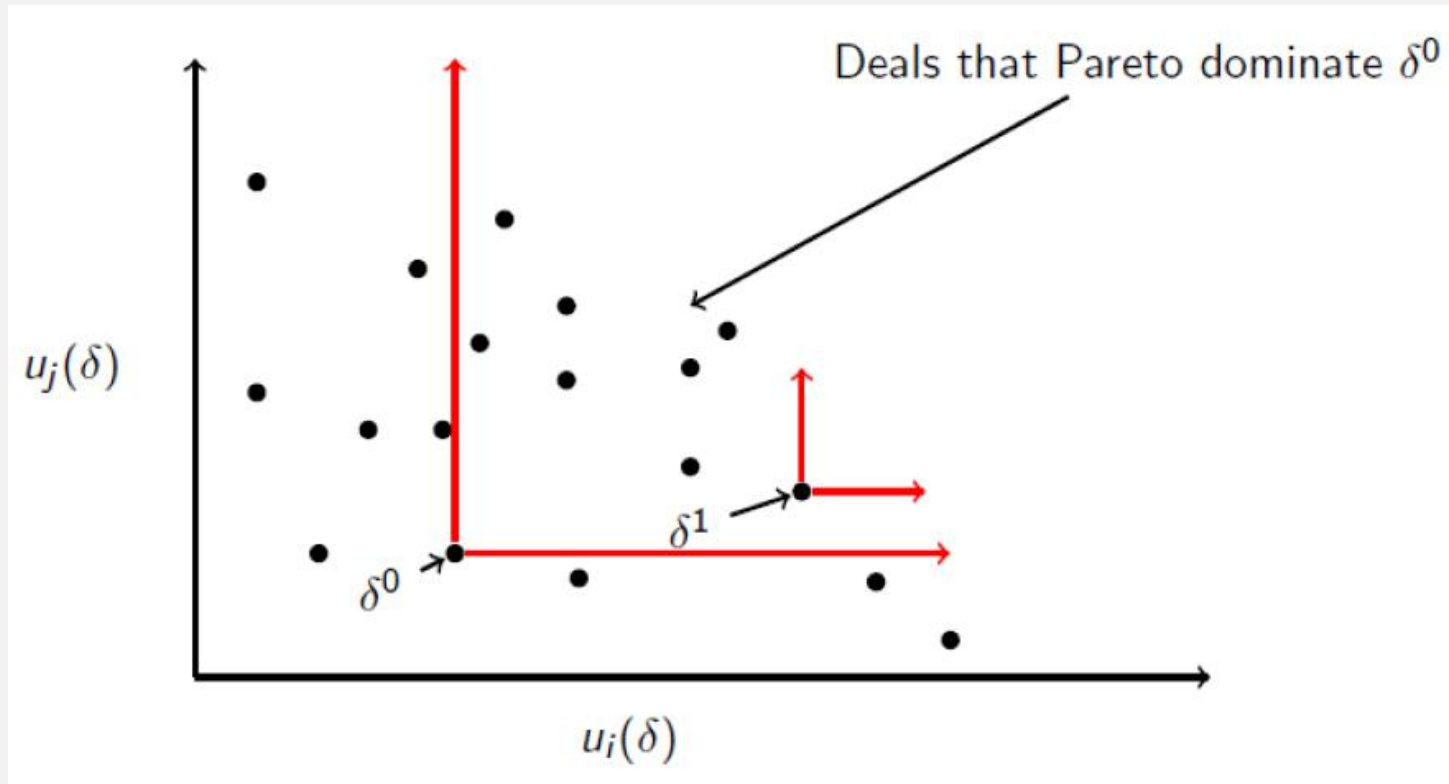


Együttműködés versengés közepette – koordinálás + feladatmegosztás tárgyalással/2

Tárgyalás, mint egy elosztott keresés - Hegymászó tárgyalás



(Bamba) Törekvés a domináló (Pareto) megoldás felé, pl. δ_0 – ből kiindulva δ_1 -be: Pareto, de nem szociális, Nash, stb. optimum.

Feladat-orientált domén – Task Oriented Domain (TOD \subset SOD)

Ágensek aktivitásai – a feladatai

Feladatvégrehajtás – mások segítsége, megzavarása nélkül

Minden erőforrás rendelkezésre áll

Hasznosságok = a feladatok értéke – feladatok végrehajtásának költségei

Feladatok közös jó érdekében **újrafeloszthatók, hasznosság átvállalható**

Állapot-orientált domén – State Oriented Domain (SOD \subset WOD)

Elfogadható környezeti végállapotok az ágensek céljai

Lehetnek pozitív/negatív kölcsönhatások (segítségnyújtás, megzavarás)

Javaslatok a közös tervek és ütemezések (szerepek és azok költségei)

Várható hasznosság maximalizálása, hasznosságátvállalás nincs

Hasznosságok = a célok értéke – a célok elérésének költségei

Érték-orientált domén – Worth Oriented Domain (WOD)

Egyes környezeti állapotoknak értéke lehet $W(\text{állapot}, \text{ágens})$

A cél a legértékesebb környezeti állapot elérése

Közös tervek/költségek halmaza $c(\text{terv}, \text{ágens})$

Javaslatok a közös tervek, ütemezések, célrelaxációk és az azokhoz szükséges erőforrások és eszközök

Feladat-orientált domén – Task Oriented Domain (TOD)

Lehetséges feladatok (véges) halmaza $T = \{t_1, t_2, \dots, t_n\}$ $\langle T, Ag, c \rangle$

Ágenshalmaz $Ag = \{1, 2, \dots, n\}$

Egy feladathalmaz végrehajtásának költsége $c: 2^T \rightarrow \mathbb{R}^+$

Költség-függvény monoton $T_1 \subseteq T_2 \rightarrow c(T_1) \leq c(T_2)$

A „semmitevés” költsége $c(\emptyset) = 0$

Összeütközés (*encounter*) – a taszkok hozzárendelése $\langle T_1, T_2, \dots, T_n \rangle$

$\langle T_1, T_2 \rangle$ összeütközésben egy **üzlet**/(le)osztás (*deal*) $\delta = \langle D_1, D_2 \rangle$

$T_1 \cup T_2 = D_1 \cup D_2$ a feladatok új kiosztása

Egy ágens részére az üzlet haszna: $u_i(\delta) = c(T_i) - c(D_i)$

Ha nincs üzlet és mindegyik ágens az ő eredeti feladatát végzi, $\delta_- = \langle T_1, T_2 \rangle$
az a **konfliktus-üzlet** („nincs-üzlet” üzlet)

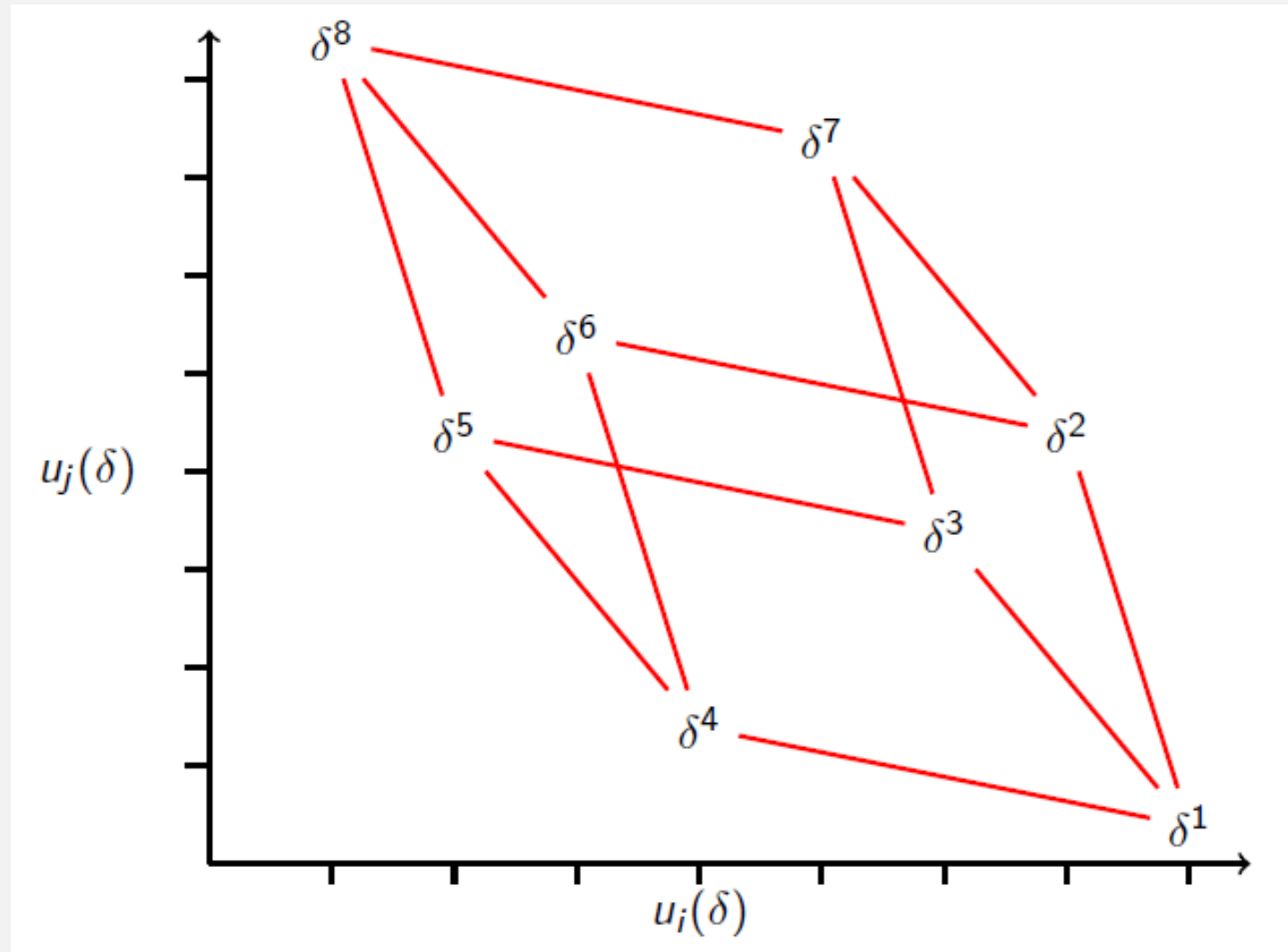
Feladathozzárendelési probléma - Task Allocation Problem

δ	$s_i(\delta)$	$s_j(\delta)$	$c_i(\delta)$	$c_j(\delta)$	$u_i(\delta)$	$u_j(\delta)$
δ^1	\emptyset	$\{t_1, t_2, t_3\}$	0	8	8	0
δ^2	$\{t_1\}$	$\{t_2, t_3\}$	1	4	7	4
δ^3	$\{t_2\}$	$\{t_1, t_3\}$	2	5	6	3
δ^4	$\{t_3\}$	$\{t_1, t_2\}$	4	7	4	1
δ^5	$\{t_2, t_3\}$	$\{t_1\}$	6	4	2	4
δ^6	$\{t_1, t_3\}$	$\{t_2\}$	5	3	3	5
δ^7	$\{t_1, t_2\}$	$\{t_3\}$	3	1	5	7
δ^8	$\{t_1, t_2, t_3\}$	\emptyset	7	0	1	8

Hasznosság számítása: $u_i(\delta) = 8 - c_i(\delta)$, $u_j(\delta) = 8 - c_j(\delta)$

Üzletek számítása: egyszerre egyetlenegy feladat üzletelése
(üzletek váltásánál egyszerre csak egyetlenegy feladat változhat)

Feladathozzárendelési probléma - Task Allocation Problem



Feladathozzárendelési probléma - Task Allocation Problem

Mi van, ha nemigen van egy jó üzlet,
amit lehetne javasolni és elfogadásra találni?

Feladathozzárendelési probléma - Task Allocation Problem

Mi van, ha nemigen van egy jó üzlet,
amit lehetne javasolni és elfogadásra találni?

Fizetések:

(pénzbeli) fizetések bevezetése – növeli a lehetséges üzletek választékát
(a Vállalkozási Hálók protokoll bevezetésének hajdani igazi oka)

Feladathozzárendelési probléma - Task Allocation Problem

Fizetések: (Vállalkozási Hálók)

- a menedzser egy feladat megoldását „megveszi” és a vállalkozók benevezik (költségük alapján) a fizetendő árat,
- menedzser választ vállalkozókat (egybevetve a hasznot és a költséget), kiadja a feladatot és fizet.

Tegyük fel:

Az i . ágens számára a pillanatnyi δ feladathozzárendelés költséges, mert az egyik t feladata (számára) igen drága.

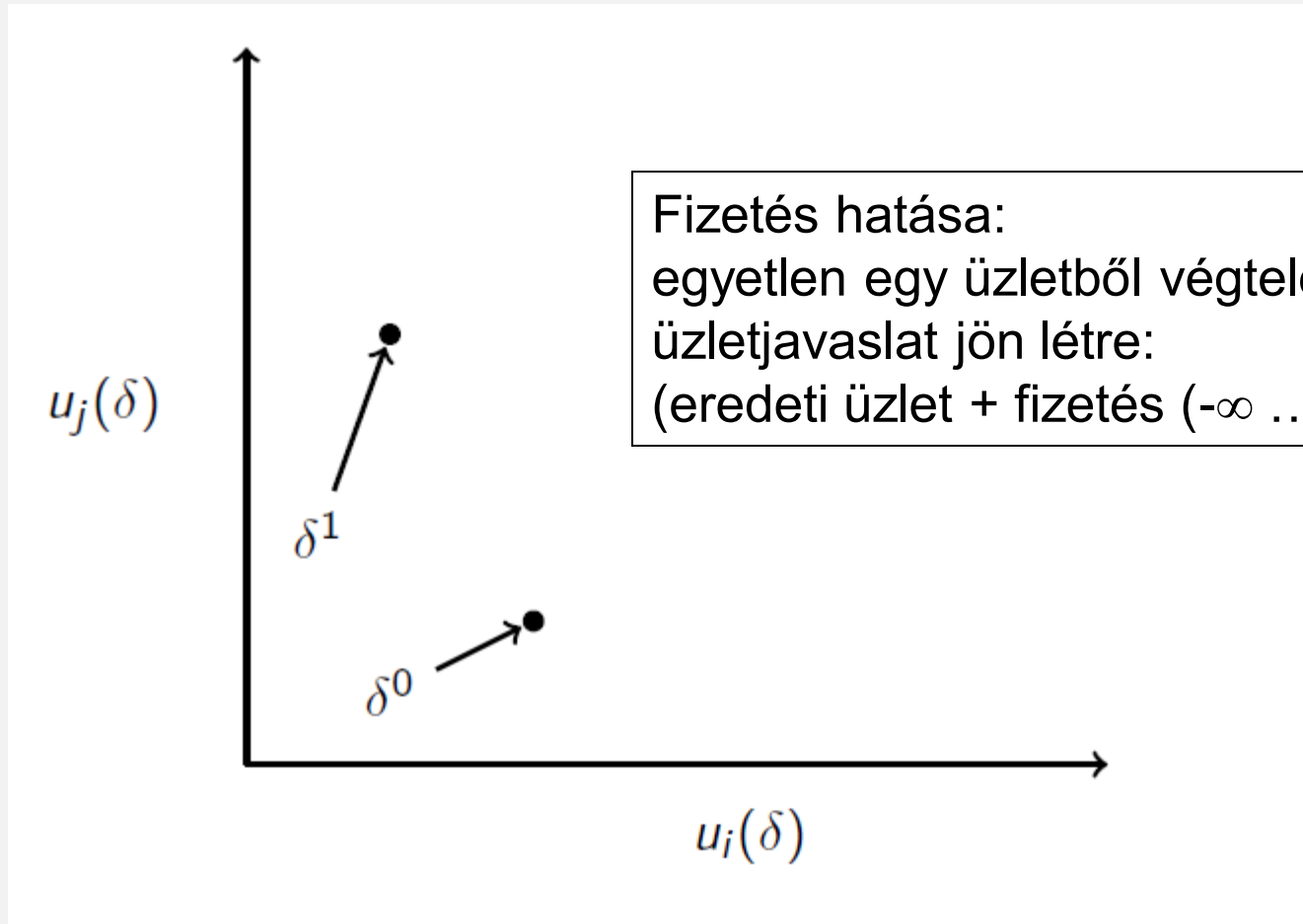
Az i hajlandó akár Δc -t is fizetni másnak, hogy a t feladatot átvegye (a feladat átadásával járó hasznosságnyereség erejéig, többet fizetni tiszta veszteség lenne).

Az j . ágens átveheti a feladatot, ha a fizetés több, mint a feladat végzésével járó hasznosságcsökkenés.

$$c_i(s_i(\delta)) - c_i(s_i(\delta) - t) = \text{nyereség} > \Delta c = \text{fizetés}$$

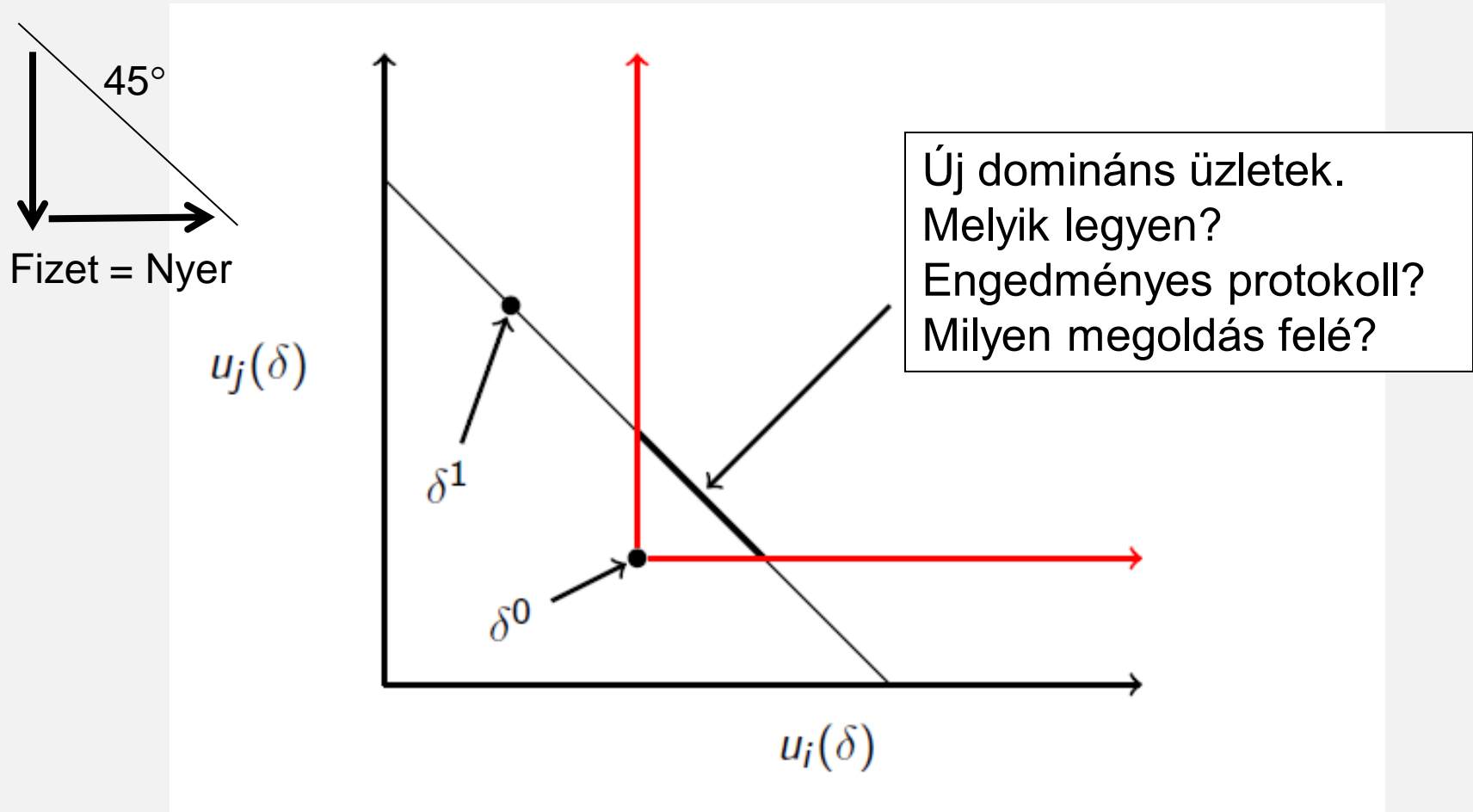
$$c_j(s_j(\delta)) - c_j(s_j(\delta) + t) + \Delta c = \text{veszteség} + \text{fizetés} > 0$$

Feladathozzárendelési probléma - Task Allocation Problem



Fizetés hatása:
egyetlen egy üzletből végtelen sok
üzletjavaslat jön létre:
(eredeti üzlet + fizetés $(-\infty \dots + \infty)$)

Feladathozzárendelési probléma - Task Allocation Problem



Feladathozzárendelési probléma - Task Allocation Problem

δ_1 nem dominálja a δ_0 -t, a tárgyalás könnyűszerrel a δ_0 -ban terminálódik, ami pl. nem egy utilitárius megoldás.

A Vállalkozási Hálók protokoll beszúrása, ami a feladat átvállallását tisztázza, fizetés ellenében, lehetőséget ad egy utilitárius megoldás megegyezésére. (nem garantált – a költségfüggvény alakjától függ)

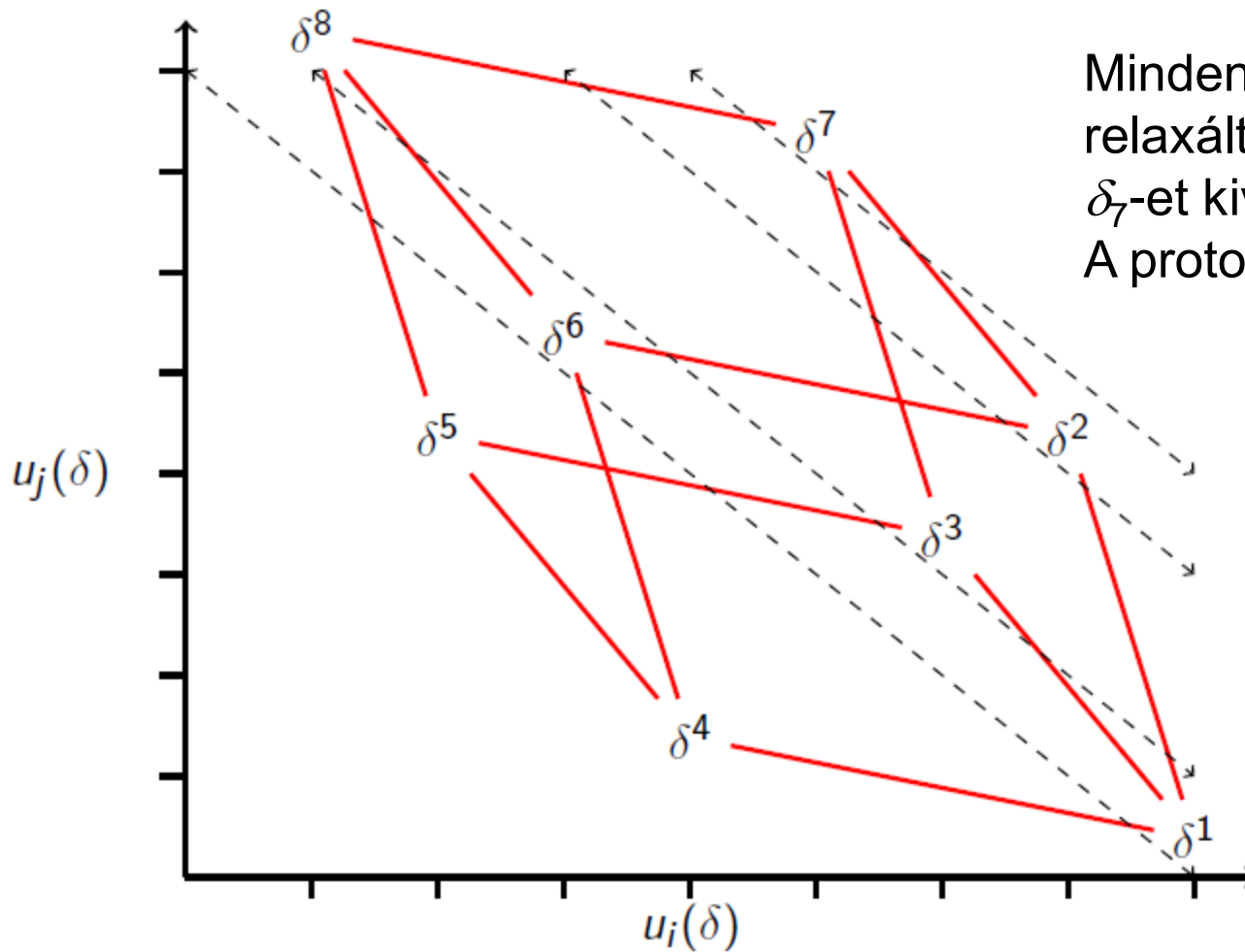
A $c(s)$ függvény **additív költségfüggvény**, ha minden feladat-hozzárendelés esetén, a hozzárendelt feladatok összköltsége az egyes feladatok költségösszege.

$$\forall S. S \subseteq T, \quad c(S) = \sum_{t \in S} c(t)$$

Tétel:

Ha egy **additív** költségfüggvénnyel rendelkező feladathozzárendelési problémában **egyszerre egyetlenegy** feladat cseréje történik, minden olyan tárgyalási protokoll, ami lehetővé teszi a **fizetést** és mindig a **domináns** üzlet **felé** lép, végeredményben az **utilitárius** megegyezéshez fog konvergálni (2006).

Feladathozzárendelési probléma - Task Allocation Problem



Minden üzletet a fizetéssel relaxált változat dominálja, δ_7 -et kivéve. A protokoll ott köt ki.

Feladathozzárendelési probléma - Task Allocation Problem

Üzlettípusok (javaslatnyelvek, üzlet-gráfok):

O-üzlet (O (original) contract) – egy feladat átadása 2 ágens között

C-üzlet (C (cluster) contract) – legalább 2 feladat átadása 2 ágens között

S-üzlet (S (swap) contract) – egy-egy feladat kölcsönös átadása

M-üzlet (M (multiagent) contract) – pontosan 3 feladat átadása 3 ágens között, egy ágens egyetlen egy feladatot adja át, de kaphat kettőt is

OCSM-üzlet (combined)

Optimális megegyezés elérése tetszőleges (nem additív) költségfüggvény esetében nem garantált (merre megy a hegymászó? Optimum számítása – komplexitás!)

Ha az üzletgráf teljesen összekötött (pl. OCSM-javaslatnyelv mellett), akkor minden protokoll, ami a domináns üzletek sorozatából áll, utilitáriánus megoldást eredményez. (de a domináns üzletek számítási komplexitása!)

Pragmatikus cél: a javasolható üzletek számának mérséklése (az üzletek hozzáférhetőségének mérséklése, a gráf élei), de az optimum megegyezés garantálása (még nincsenek eredmények)

Feladathozzárendelési probléma - Task Allocation Problem

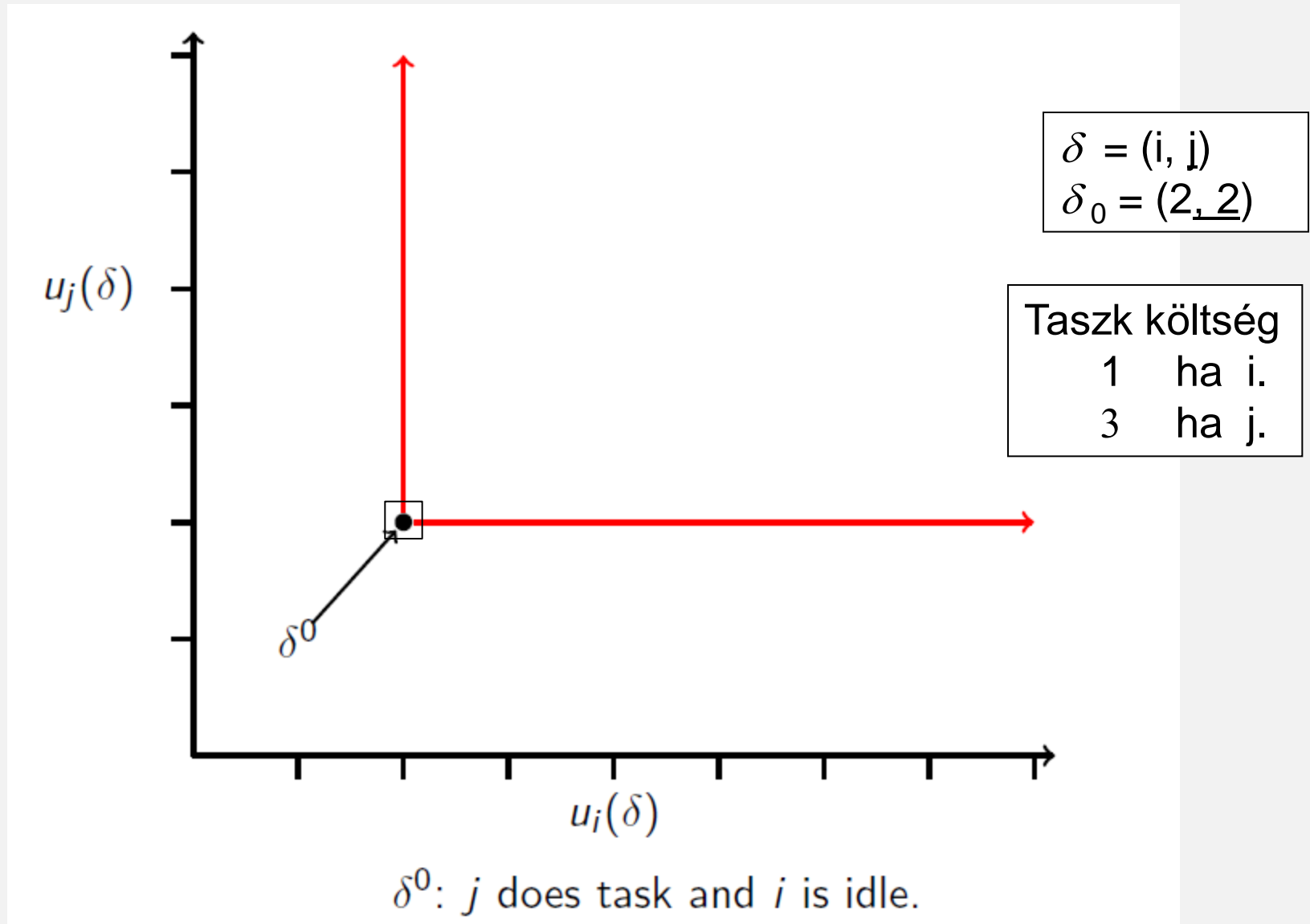
Megegyezés és **visszalépés** az üzlettől (de-committment), mert:

- időközben jobb ajánlatok,
- a környezet dinamikájának hatása,
- ágens „vis major”,
- ...

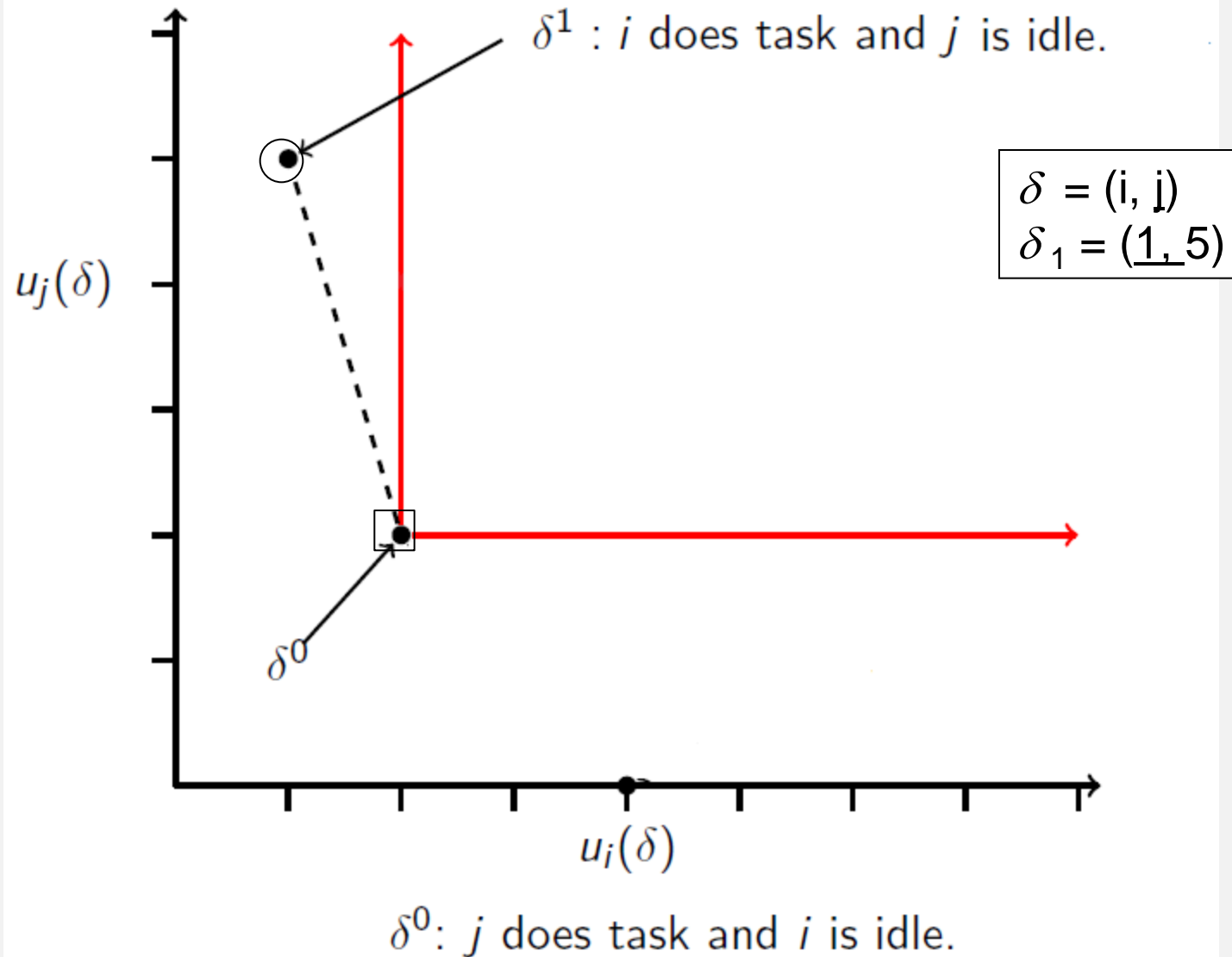
Az üzletjavaslatba ágyazott büntetés csökkenti a rizikót.

Lássúk ...

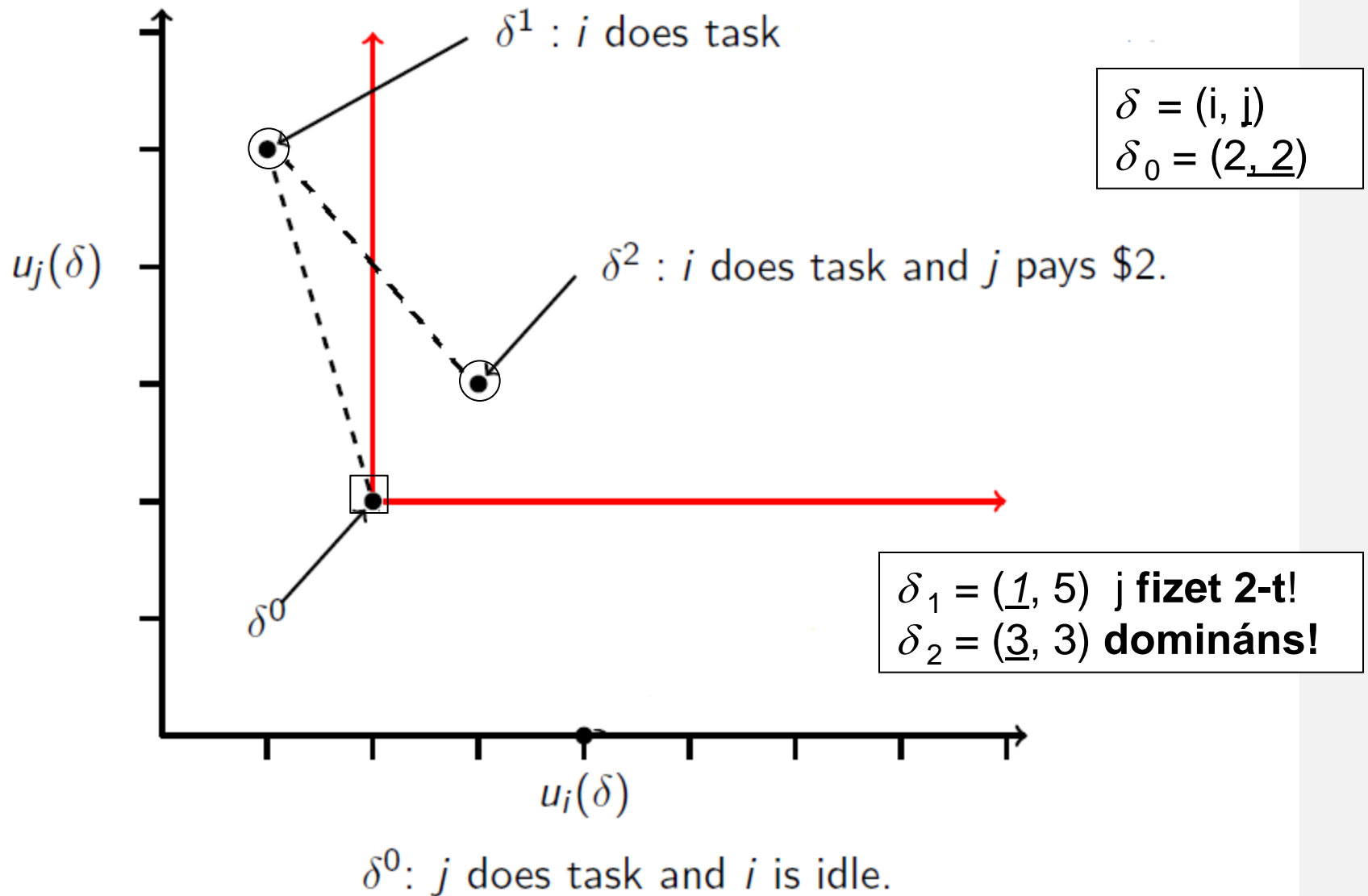
Feladathozzárendelési probléma - Task Allocation Problem



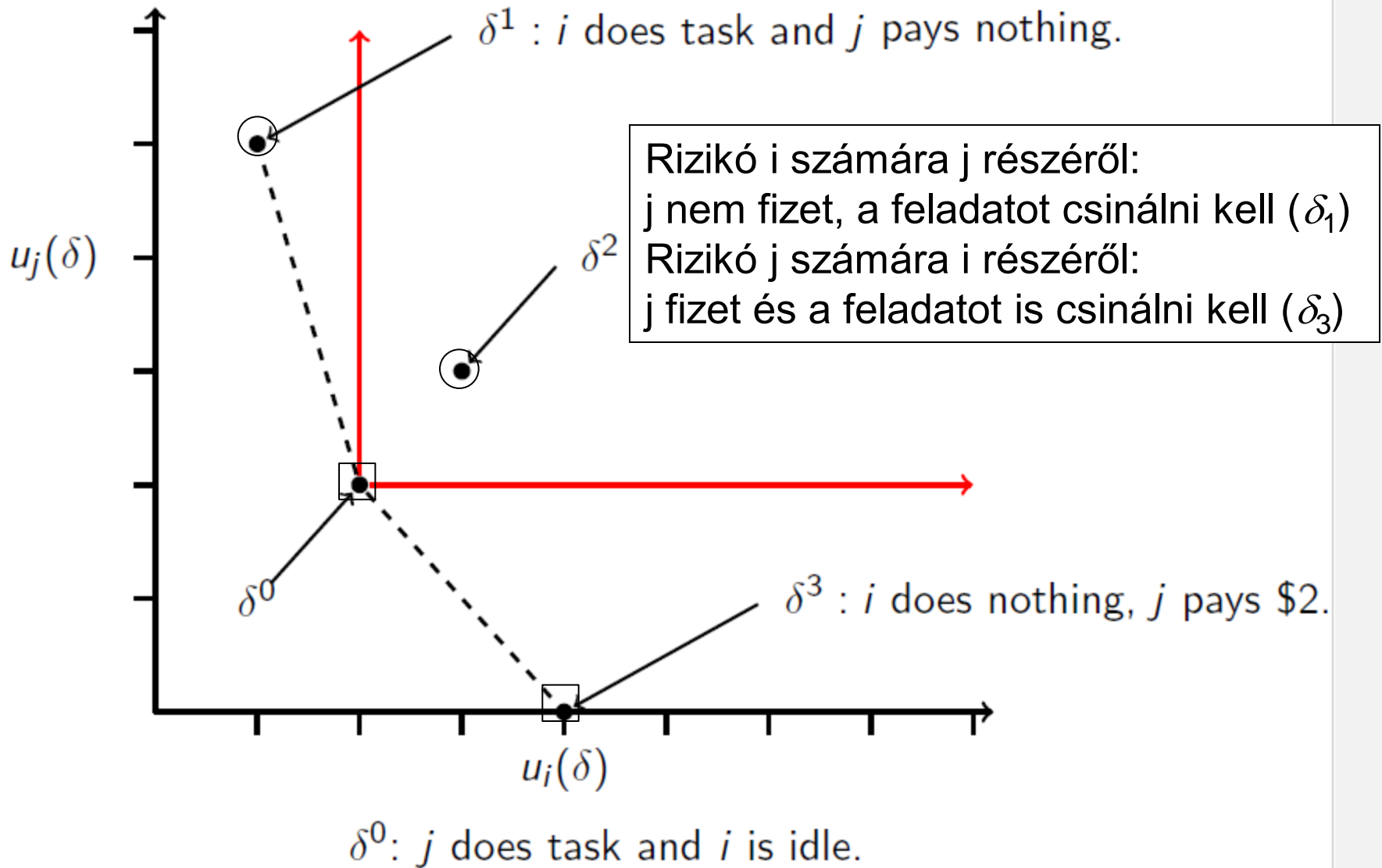
Feladathozzárendelési probléma - Task Allocation Problem



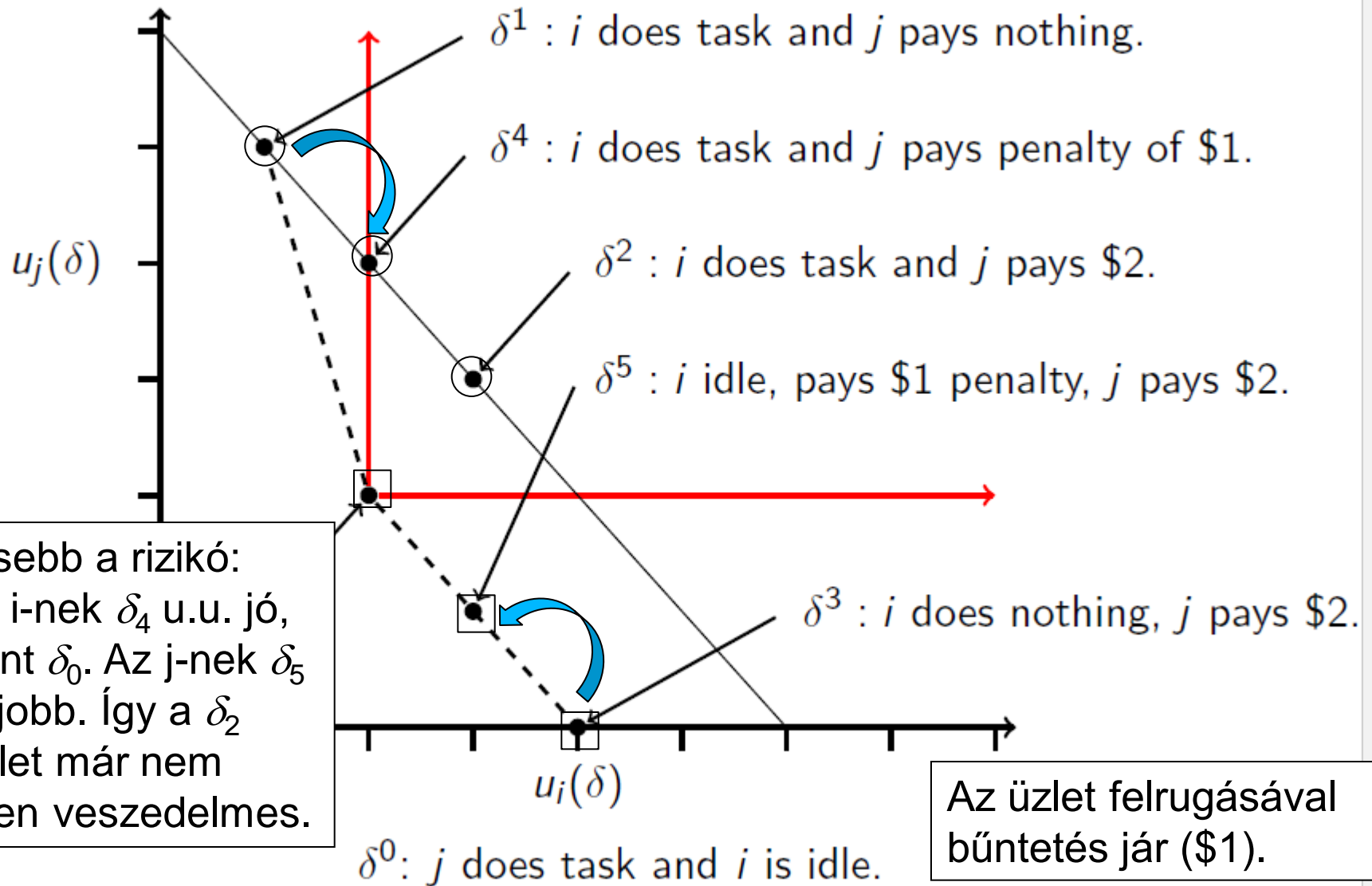
Feladathozzárendelési probléma - Task Allocation Problem



Feladathozzárendelési probléma - Task Allocation Problem



Feladathozzárendelési probléma - Task Allocation Problem



Feladathozzárendelési probléma - Task Allocation Problem

Hazudni néha jó – tárgyalni fantom feladatokról

- feladatot elhallgattatni,
- álfeladatot kitalálni.

Nash-tárgyalási megoldás
a δ_1 , az i -nek nem tetszik.

δ	$s_i(\delta)$	$s_j(\delta)$	$u_i(\delta)$	$u_j(\delta)$
δ^1	\emptyset	$\{t_1\}$	1	3
δ^2	$\{t_1\}$	\emptyset	2	1

Az i kitalál egy fiktív t_2 feladatot és tárgyalásba viszi be.

Most a Nash-tárgyalási megoldás a δ_4 .

Ezzel az i minden feladatot felvállal, de belőle csak t_1 a valódi.

Így magához ragadta a tényleges feladatot, nagyobb hasznossággal.

(általános költségfüggvények)

δ	$s_i(\delta)$	$s_j(\delta)$	$u_i(\delta)$	$u_j(\delta)$
δ^1	\emptyset	$\{t_1, t_2\}$	1	5
δ^2	$\{t_1\}$	$\{t_2\}$	2	3
δ^3	$\{t_2\}$	$\{t_1\}$	2	3
δ^4	$\{t_1, t_2\}$	\emptyset	8	1

Feladathozzárendelési probléma - Task Allocation Problem

Komplex üzletek

Üzlet komponensei: x_1, \dots, x_n változók

- elosztott korlátozás kielégítés (optimalizálás)
- ágensek nem uralják a változók értékét

PI.

$$u_i(\delta) = c_1 u_i^1(x_1) + c_2 u_i^2(x_2) + \dots + c_n u_i^n(x_n)$$

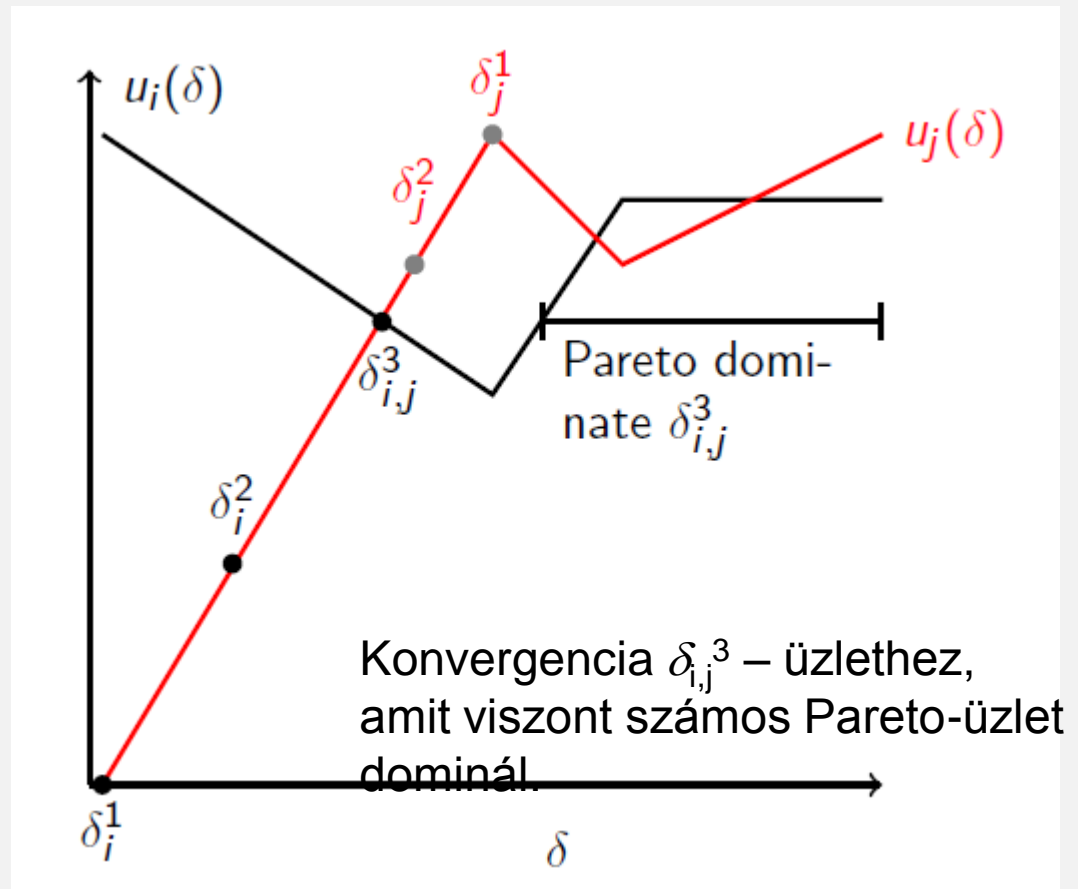
Feladathozzárendelési probléma - Task Allocation Problem

Komplex üzletek

Üzletek generálása
lokális (mert sok van).

Engedményes,
hegymászó, stb. protokoll
leállhat Pareto megoldások
előtt.

Indulunk a legjobbaktól.
Nagy a keresési tér.
Azok környezetében
módosítunk engedmények
érdekében =
legfeljebb lokális min.



Agenda = lehet pl. tárgyalt dimenziók sorozata
(pl. típus, évjárat, extrák, szín, garanciaidő, ...)

Feladathozzárendelési probléma - Task Allocation Problem

Komplex üzletek – tárgyalás **közvetítő ágens** segítségével

akik a tárgyalásban résztvesznek

„Hegymászó tárgyaló” (**Hill Climber**)

- elfogadott rezervációs ára $u_i(\delta_-)$, majd mindenkor eddigi legjobb elfogadott üzlet felett. Azaz monoton módon növeli a rezervációs árát.

„Szimulált lehűtő tárgyaló” (**Annealer**)

- T hőmérséklet szerint elfogad kis valószínűséggel az eddigi elfogadott legjobb üzletnél rosszabb javaslatot is.

$$P = \exp\left(-\frac{\Delta U}{T}\right)$$

Feladathozzárendelési probléma - Task Allocation Problem

Elmozdulás lokális minimumtól a globális minimum felé

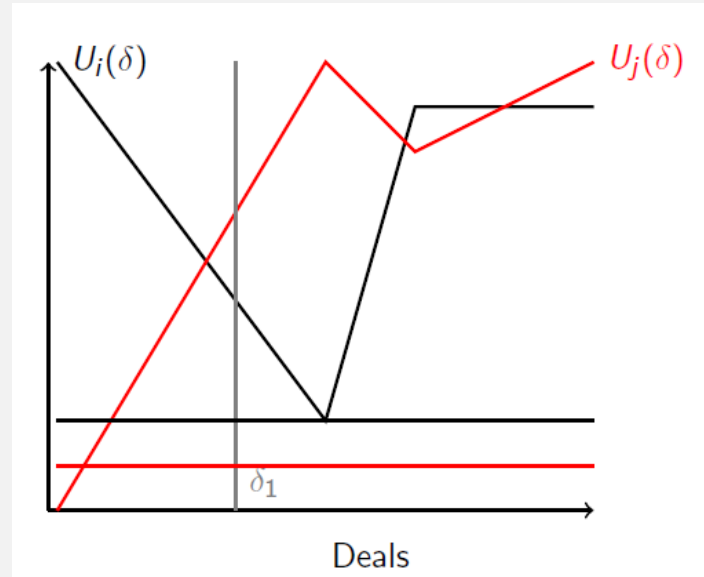
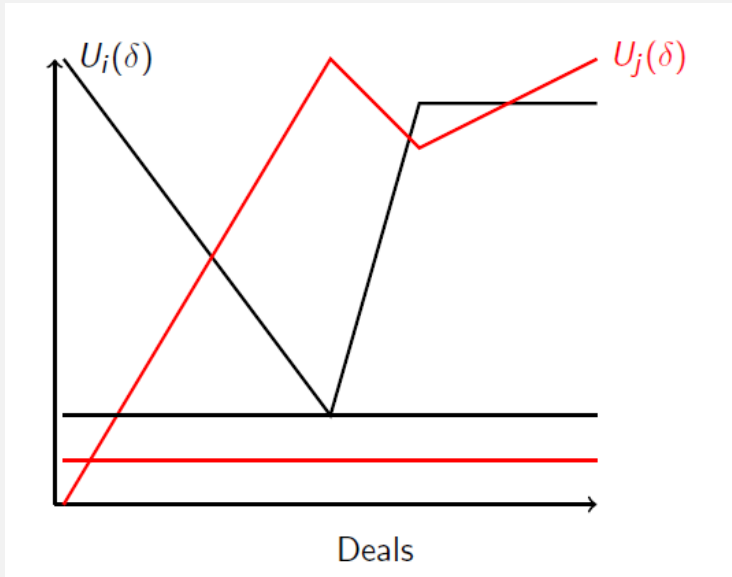
-- **szimulált lehűtés** algoritmus

Kilépés: stabil hasznosságok
egyik ágens számára sincs
már javulás.

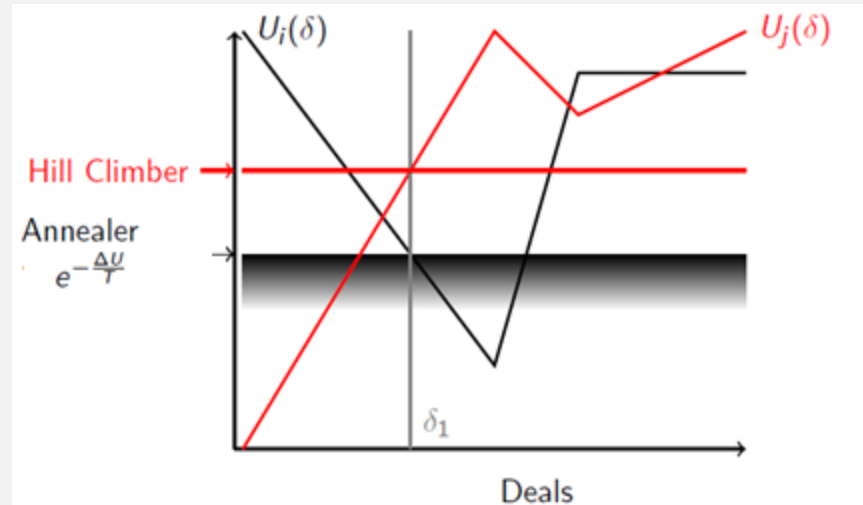
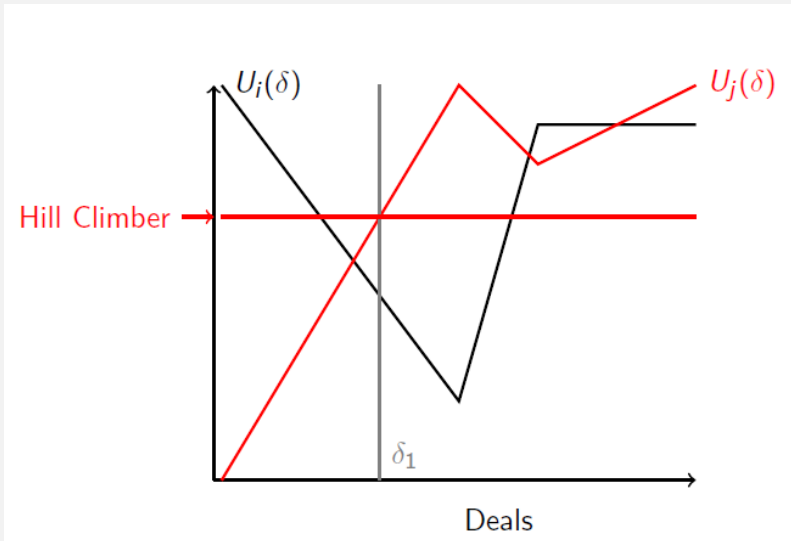
ANNEALING-MEDIATOR

```
1  Generate random deal  $\delta$ .
2   $\delta_{\text{accepted}} \leftarrow \delta$ 
3  Present  $\delta$  to agents.
4  if both accept
5      then  $\delta_{\text{accepted}} \leftarrow \delta$ 
6           $\delta \leftarrow \text{mutate}(\delta)$ 
7          goto 3
8  if one or more reject
9      then  $\delta \leftarrow \text{mutate}(\delta_{\text{accepted}})$ 
10 goto 3
```

Feladathozzárendelési probléma - Task Allocation Problem

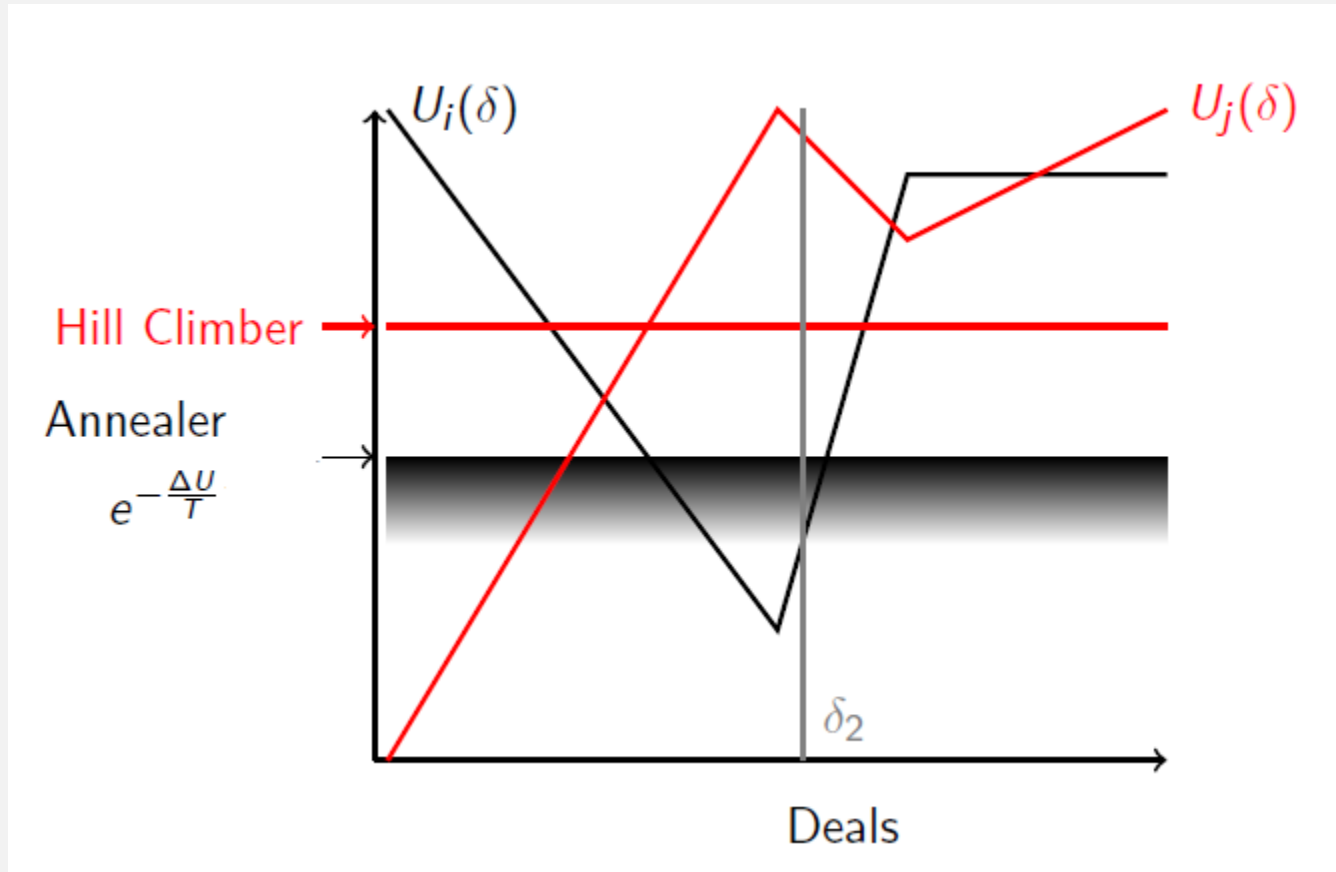


Rezervációs árszintek módosítása üzletek függvényében (**HC**, **ANN**)



Jose M Vidal

Feladathozzárendelési probléma - Task Allocation Problem



Feladathozzárendelési probléma - Task Allocation Problem

Szimulációk:

Mindkettő **HC** – alacsonyabb hasznosság (mindkettő kielégítése)

ANN jelenléte – mindig növeli a jólétet.

Mindkettő **ANN** – kb. 40% jobb hasznosság. Jólét érdekében hajlandók időnként magán profit maximálásáról lemondani.

	Hill Climber	Annealer
Hill Climber	400, 400	700, 180
Annealer	180, 700	550, 550

800	880
880	1100

Problémák:

- random üzletgenerálás – kölcsönös ismeret egymás hasznosságáról elvész,
- rosszabb üzlet elfogadása – irracionális lépés.

Feladathozzárendelési probléma - Task Allocation Problem

Tárgyalási hálók

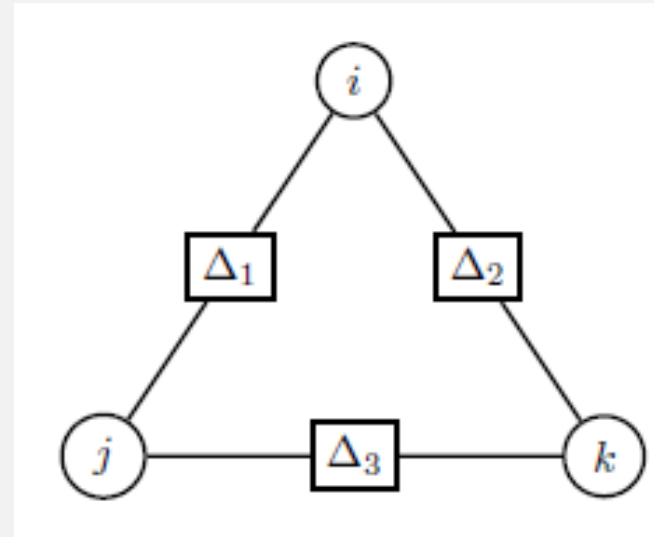
Több ágens szimultán módon tárgyal egymással, több üzletet tartalmazó Δ_i üzleti csomagokról.

Ágensek üzleti csomagjai nem feltétlenül konzisztensek (benne van mindegyikben a konfliktus üzlet is).

A bennük lévő üzletek lehetnek kölcsönhatásban (pl. több eladóval tárgyalni ugyanannak a dolognak az eladásáról, visszalépések!).

Egy-egy üzleti csomagot tárgyaló ágensek a teljes MAS valamilyen részhalmaza.

A tárgyalási háló megoldása: az üzleti csomagokból (mindegyikéből) egy-egy üzletet meríteni, úgy hogy az azokból álló üzlethalmaz mindegyik ágensre nézve konzisztens.



Feladathozzárendelési probléma - Task Allocation Problem

Érvelés alapú tárgyalás

- Szabványos tárgyalási nyelvek még nincsenek
(érvelési protokollok, preferencia alapú kommunikációs nyelvek, ...)

- Benyújtott üzletjavaslat

Kritika információ másik hasznossági függvényéről
jövőbeli esélytelen javaslatok kiszűrése

Ellenjavaslat erősen korrelált a kritizált javaslattal
társult érvelés – meggyőzés érdekében

Rábeszélés (további információ megadása)

Fenyegetés. Jutalmazás, hatásgyakorlás.

...

Ezzel lehetséges pl.

megbecsülni az ellenfél hasznosságfüggvényét

üzletjavaslatokat eliminálni (keresés komplexitása!)

befolyásolni ellenfél hasznosságfüggvényét

módosítani saját hasznosságfüggvényét

Érvelés

Érvelés – mások meggyőzése a dolgok igaz/hamis állásáról

Érvelés módja (Gilbert)

Logikus érvelés

mint egy logikai bizonyítás

Emocionális érvelés

emociók, attitűdök irányába

(„Hogyan érezné magát, ha ...”)

Fizikai (visceral – „zsigeri”) érvelés

fizikai, szociális aspektusok („idősebb vagyok, erősebb vagyok ...”)

Kisceral érvelés

intuitív, misztikus, vallásos érvek, ...

Logikus érvelés

Bizonyítás szerkesztése az adott javaslatra (üzletre) vonatkozólag
mellette
ellene

Formálisan: egy mondat kikövetkeztetése tudásbázisból: $D \mid - f$

Ha a tudásbázis egy részhalmaza elegendő a
mondat kikövetkeztetéséhez:

G – **alap igazságok** (grounds)

$$G \subseteq D : G \mid - f$$

Egy **érv** = $\langle G, f \rangle$ a **D felett**.

Egy érv **nem triviális**, ha a G konzisztens.

Két f és g mondat esetén f **támadja** a g -t, ha $f \equiv \neg g$

Logikus érvelés

Két f és g mondat esetén f **támadja** a g -t, ha $f \equiv \neg g$

$\langle G1, f1 \rangle$ érv kétféleképpen megtámadható:

$\langle G2, f2 \rangle$ **megcáfolja** a $\langle G1, f1 \rangle$ -t, ha $f2$ támadja a $f1$ -t

$\langle G2, f2 \rangle$ **aláássa** a $\langle G1, f1 \rangle$ -t, ha $f2$ valamilyen $h \in G1$ -beli állítást támad meg

Logikus érvelés

Elfogadható érvek hierarchiája

A1: D feletti összes érv osztálya

A2: D feletti nem triviális érvek osztálya

A3: D feletti olyan érvek osztálya, amikhez nem léteznek az azokat megcáfoló érvek

A4: D feletti olyan érvek osztálya, amikhez nem léteznek az azokat aláásó érvek

A5: D feletti érvényes érvek osztálya

Elfogadhatósági hierarchia:

$$A1(D) < A2(D) < A3(D) < A4(D) < A5(D)$$

Érvelő párbeszéd

Absztrakt érvelés

Logikus érvelés

Absztrakt érvelés $\langle X, \rightarrow \rangle$

X – érvek halmaza

\rightarrow reláció $X \times X$ –ben, az érvek közötti reláció - a **támadás** kifejezése

Cél:

Definiálni, mely érvhalmaz „biztonságos”, „elfogadható”, kölcsönösen védhető, nem megtámadható, ...

Egy érv $x \in X$ **megtámadott** Y érvhalmaz **által** (Y része az X -nek), ha létezik legalább egy $y \in Y$, hogy $y \rightarrow x$.

Egy érv $x \in X$ **elfogadható az Y -ra nézve**, ha minden Y -beli támadója az x -nek szintén megtámadott.

Egy Y érvhalmaz **konfliktusmentes**, ha egy Y -beli érv sem támad más Y -beli érvet.

Egy Y érvhalmaz **megengedett**, ha Y minden érve az Y -ra nézve elfogadható.

Logikus érvelés

$Y = \{a, b, c, d, e, f, g, h, i, j, k, l, m, n, p, q\}$

