

14.12 Oyun Teorisi

Muhamet Yıldız

Güz 2005

Ödev 4

1. Alttaki statik oyunun 5 kez tekrarlandığı bir oyun düşünelim:

	L	R
A	2,1	0,0
B	0,0	1,2

Varsayalım ki, kazançlar, her tekrardan gelen kazançlar toplamı olsun. Statik oyundaki her strateji vektörü için, tekrarlı oyunun bir saf stratejili alt-oyun tam dengesini bulunuz, öyle ki, oyuncular strateji vektörünü ilk gün oynasınlar.

2. Goliath Software, X adında bir internet tarayıcısı satan büyük bir yazılım firmasıdır. Her yıl yeni bir firma gelir. Yeni firma ya X ile uyumlu yeni bir tarayıcı ya da bir arama motoru üretir. Yeni firmanın ne ürettiğini gören Goliath Software X'i ya günceller ya da güncellemez. Yılın sonunda, sonuçtan bağımsız olarak, yeni firma kaybolur, yerini bir sonraki yeni firmaya bırakır. Her durum için yıllık karlar alttaki tabloda verilmiştir, öyle ki, ilk sayı Goliath Software'in karıdır.

	Tarayıcı	Arama Motoru
Güncelle	1,0	3,1
Güncelleme	2,2	4,1

(Yukarıdaki bir tablodur, oyun değildir.) Eğer Goliath Yazılım 0 yılında π_0 , 1 yılında π_1, \dots , t yılında π_t, \dots , yıllık kar elde ediyorsa toplam kazancı

$$\pi_0 + .9\pi_1 + (.9)^2\pi_2 + \dots + (.9)^t\pi_t + \dots \quad (1)$$

idir. Her yeni firmanın kazancı onun yıllık karıdır. Tüm geçmiş gözlemlenebilmektedir ve oyun hiçbir zaman sona ermez. Alttaki tüm strateji vektörleri için, bir alt-oyun tam denge olup olmadıklarını kontrol edin ve strateji vektöründen çıkacak sonucu bulun.

- (a)
- Yeni firma tarayıcı üretirse ve Goliath Yazılım X'i, geçmişte yeni firma tarayıcı ürettiği her durumda güncellediyse,¹ Goliath Yazılım ürününü günceller;
 - aksi takdirde, güncellemez.
 - Yeni firma, Goliath Yazılım geçmişte her yeni firma tarayıcı ürettiğinde X'i güncellediyse, arama motoru üretir;
 - aksi takdirde, yeni firma tarayıcı üretir.
- (b) Her t yılında, Goliath Yazılım tarayıcısını ancak ve ancak yeni firma (t yılında) tarayıcı üretirse günceller. Yeni firma her zaman arama motoru üretir.
- (c) 100'ün katları olan yıllar kutsal yıllardır. Kutsal yıllarda, Goliath Yazılım (yeni firmanın ne yaptığından bağımsız olarak) tarayıcısını güncellemez ve yeni firma da bir tarayıcı üretir. Kutsal olmayan yıllarda ise alttaki stratejiler sözkonusu:
- Yeni firma tarayıcı üretirse ve Goliath Yazılım X'i, kutsal yıllar dışındaki geçmişte yeni firma tarayıcı ürettiği her durumda güncellediyse, Goliath Yazılım ürününü günceller;
 - aksi takdirde, güncellemez.
 - Yeni firma, Goliath Yazılım kutsal yıllar dışındaki geçmişte her yeni firma tarayıcı ürettiğinde X'i güncellediyse, arama motoru üretir;
 - aksi takdirde, yeni firma tarayıcı üretir.
- (d) (c)'deki kutsal yılları kutsal olmayan yıllarla ter değiştirin, yani, 100,200,... yılları artık kutsal değil ama geri kalan yıllar kutsal yıllardır.

¹ $t = 0$ yılında, geçmişte Goliath Yazılım her yeni firma tarayıcı ürettiğinde X'i güncellediği doğrudur

3. Bir Firma ve bir İşçi düşünelim. Firma işçiye $\omega > 0$ ödeyip ödememeye (iş almaya) karar verir ve sonra işçi çalışıp çalışmamaya karar verir, bu işçiye $c > 0$ 'a mal olur ve firma için π üretir, öyle ki, $\pi > \omega > c$. Kazançlar alttaki tabloda verilmiştir:

	Firma	İşçi
öde, çalış	$\pi - \omega$	$\omega - c$
öde, çalışma	$-\omega$	ω
ödeme, çalış	π	$-c$
ödeme, çalışma	0	0

- (a) Tüm Nash dengelerini bulun.
- (b) Şimdi bu statik oyunun sonsuz defa tekrarlandığını ve oyuncuların δ iskonto faktörüne sahip olduklarını düşünelim. Altta, bu tekrarlı oyun için strateji vektörleri verilmiştir. Herbiri için, yüksel δ değerleri için alt-oyun tam dengesi olup olmadıklarını kontrol edin, eğer denge ise, bunu denge yapan en küçük δ değerini bulun.
- Ne olursa olsun, firma her zaman öder ve işçi de her zaman çalışır.
 - Herhangi bir t anında, işçi ancak ve ancak t anında ödeme aldıysa çalışır ve firma her zaman ödeme yapar.
 - $t = 0$ anında, firma öder ve işçi de ne olursa olsun çalışır. Diğer tüm $t > 0$ anlarında, firma ancak ve ancak işçi tüm önceki periyotlarda çalıştıysa ödeme yapar ve işçi de ancak ve ancak tüm önceki periyotlarda çalıştıysa çalışır.
 - $t = 0$ anında, firma öder ve işçi de ancak ve ancak ödeme yapıldıysa çalışır. Diğer tüm $t > 0$ anlarında, firma ancak ve ancak işçi firmanın ödeme yaptığı tüm önceki periyotlarda çalıştıysa ödeme yapar ve işçi de ancak ve ancak t anında ödeme yapıldıysa ve ödeme yapıldığı tüm önceki periyotlarda çalıştıysa çalışır.
 - İki durum var: İşe alınma ve işsizlik. Oyun işe alınma durumunda başlar. Bu durumda, firma öder ve işçi de ancak ve ancak o periyot ödeme yapıldıysa çalışır. Eğer işçi çalışmazsa, işsizlik durumuna geçeriz; aksi halde işe alınma durumunda kalırız. İşsizlik durumunda firma

ödeme yapmaz ve işçi de o periyot ne olursa olsun çalışmaz. İşsizlik durumunda $T > 0$ gün geçtikten sonra hep işe alınma durumuna geri dönüyoruz. (Yanıtınız tüm $T > 0$ 'ları da içermeli.)