CS 497sml: Week 8 Notes (Mar. 11, 13)
LinShi(linshi@uiuc.edu)

03/11

1 Mixed Strategy Nash Equilibrium(NE)

Definition: If there is a set of mixed strategies with the property that no player can benefit by changing her strategy while the other players keep their strategies unchanged, then that set of mixed strategies and the corresponding payoffs constitute the NE.
In the 2-player case, let $A(B)$ to be the matrix form of the expected cost for player 1(2). Mixed strategy (y^*, z^*) is a NE if

$$y^{*T} A z^* \leqslant y^T A z^*, \ \forall y \in Y$$

$$y^{*T} B z^* \leqslant y^{*T} B z, \ \forall z \in Z$$

Nash(1950) showed that there exist at least one NE for any finite noncooperative game. In general Nash equilibrium $\neq$ security strategy. For the case of 2 players, instead of linear programming, quadratic programming is needed to compute NE and in general non-linear programming is needed.

2 Sequential Games

So far we used normal form(matrix) to represent simultaneous games. While sequential games, where each player makes decision in order and knows all what happened prior to making a decision, is more easily represented by extensive form(tree) Each node indicate a *choice* of a player and edges

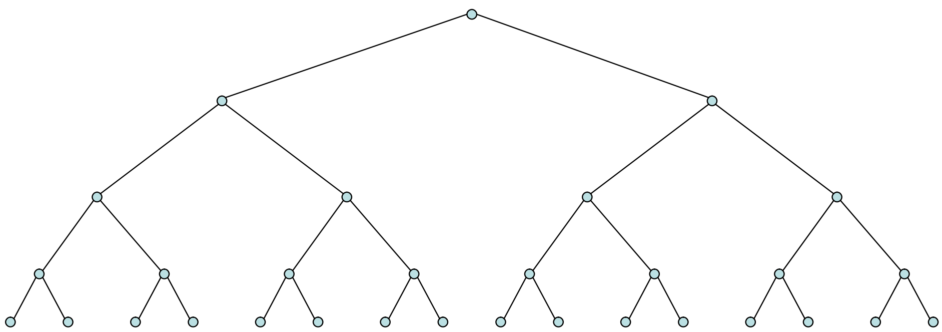

Figure 1: **Extensive Form Game**

indicate a *decision* made by a player.

3 Derivation of Dynamic Programming(DP): cost-to-go

Let $F = K+1$, $l_F(x_F)$ be the cost of the final state and $L^*_{k,F}(x_k)$ be the optimal cost of going from state x_k to x_F. We have:

$$\begin{aligned} L^*_{k,F}(x_k) &= \min_{u_k,\ldots u_K}[\sum_{i=k}^{K} l(x_i,u_i) + l_F(x_F)] \\ &= \min_{u_k}[l(x_k,u_k) + \min_{u_{k+1},\ldots,u_K}[\sum_{i=k+1}^{K} l(x_i,u_i) + l_F(x_F)]] \\ &= \min_{u_k}[l(x_k,u_k) + L^*_{k+1,F}(x_{k+1})] \end{aligned}$$

The last one is the backward DP equation.

4 Sequential Decision Making with Continuous X and U

Consider the simple case: transition equation $x_{k+1} = f(x_k, u_k)$, no nature, single player, $X \in \mathbf{R}^n$ and $U \in \mathbf{R}^m$, $m \leqslant n$. Also consider finite input set $|\mathbf{U}| < \infty$. Both $\mathbf{X}$ and $\mathbf{U}$ are closed and bounded.
First recall the *Lipschitz condition*:

- For continuous function $f(x) : \mathbf{R} \to \mathbf{R}$, $\exists C \to |f(x) - f(y)| \leqslant C|x-y|$, $\forall x, y$
  If f is differentiable, the above difference can be replaced by differential: $|f'(x)| \leqslant C$. ■

For our problem we considered the case that all the transition equation and cost function satisfies *Lipschitz condition*, i.e. $\exists \alpha_1, \alpha_2, \alpha_3 \in \mathbf{R}$ such that $\forall x, x' \in \mathbf{X}$, $\forall u, u' \in \mathbf{U}$:

$$\|f(x,u) - f(x',u')\| \leqslant \alpha_1(\|x-x'\| + \|u-u'\|)$$

$$\|l(x,u) - l(x',u')\| \leqslant \alpha_2(\|x-x'\| + \|u-u'\|)$$

$$\|l_F(x) - l_F(x')\| \leqslant \alpha_3\|x-x'\|$$

Now we are going to describe some schemes which under these conditions will be guaranteed to converge, how fast they will converge depends on those *Lipschitz constants.*

We are ready to solve the DP equation:

$$L^*_{k,F}(x_k) = \min_{u_k}[l(x_k,u_k) + L^*_{k+1,F}(x_{k+1})]$$

Since we have a continuous domain, we need to discretize the domain and interpolation is naturally introduced into the solver. Formally:

$$L^*_{k,F}(x_k) = (1-\beta)L^*_{k,F}(x) + \beta L^*_{k,F}(x')$$

here x, x' are sample points, β is the measure of how far away x_k is to x if it falls inside the region of (x,x'). There are lots of other ways of doing interpolation, this is just the simplest one.

03/13

Approximation errors:
Let $\hat{L}^*(x)$=estimate of $L^*(x)$ and $\tilde{L}^*(x)$=cost obtained by executing a strategy using $\hat{L}^*$ instead of L^*(executing the iteration one more time). Given *Lipschitz constant* $\alpha_1, \alpha_2, \alpha_3$, then $\exists \epsilon(\alpha_1, \alpha_2, \alpha_3) > 0$ such that

$$\|L^*(x) - \hat{L}^*\| < \epsilon(\delta_x + \delta_u), \forall x$$

$$\|L^*(x) - \tilde{L}^*\| < \epsilon(\delta_x + \delta_u), \forall x$$

where δ_x, δ_u are the radius of largest empty balls of sample set

Quality Measures:
Now the question is how to sample X and U to minimize the error, especially if their dimensions are high. To tell the quality of sample sets, we use some quality measures. Dispersion and discrepancy are two of them.
Definition: for P, a set of samples in X:
Dispersion:

$$\delta(p) = \max_{x \in X} \min_{p \in P} \|x - p\|$$

here the norm is any L_p norm. Intuitively dispersion measures the radius of the largest empty ball under L_p norm. It has be proved that for n samples of a d dimension space:

$$\delta(P) \geqslant \frac{1}{2\lfloor n^{1/d} \rfloor}$$

The equality is achieved with even grid.
Discrepancy:

$$D(p) = \max_{R \in \mathbf{R}} \left| \frac{|P \cap R|}{|P|} - \frac{\mu(R)}{\mu(x)} \right|$$

here $|P|$ means number of elements of P, and μ is volume(area, etc) measure. $\mathbf{R}$ is the range space, usually it is the set of all axis-parallel-rectangles. The first term measure the ratio of samples found in the rectangle, and the second term measure the ratio of the size of the box. Thus the meaning of *discrepancy* is self-evident.

$2d$ examples:

- all samples are concentrating at one point. Bad in terms of both measures
- even grid. bad for discrepancy, and good for dispersion

Another concern about sampling is extensibility, i.e. strategy with the property that more samples can be added without recompute the existing samples.
Van der Corput Sequence:
Van der Corput provides an easy way to generate low-dispersion sequence with extensibility. For $1d$ case, we choose a base b, then for the n^{th} sample, we write n in base b as $\sum_{i=0}^{\log_b n} a_i b^i$, then the n^{th} sample is $\sum_{i=0}^{\log_b n} a_i b^{-i-1}$. The first 10 samples of base 2 case is listed in the follow table

index	binary	(.)reverse	sample
0	0	.0	.0
1	1	.1	.5
2	10	.01	.25
3	11	.11	.75
4	100	.001	.125
5	101	.101	.625
6	110	.011	.375
7	111	.111	.875
8	1000	.0001	.0625
9	1001	.1001	.5625

When adding a new sample, old samples are kept and low-dispersion property is maintained. The extension of Van de Corput sequence for higher dimensions is Halton sequence, each component is generated using the same method with relative prime bases respectively.

References

[1] Niederreiter, H. 1992. Random Number Generation and Quasi-Monte Carlo Methods. Society for Industrial and Applied Mathematics, Philadelphia.

[2] J. H. Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. Numer. Math., 2:84-90, 1960.