۲-۲ بازی های تصادفی

بازی‌های‌تصادفی تعمیم فرآیندهای تصادفی مارکوف به‌حالت چندعامله بوده و می توانند به عنوان چارچوبی مناسب برای بررسی و تحقیقات یادگیری چند عامله استفاده شوند. این بازی ها توسعه‌ای از بازی‌های ماتریسی‌باچندین‌حالت بوده و به آنها بازیهای مارکوفی نیز گویند. دو مثال از بازی های ماتریسی درشکل (۱) آورده شده اند. در هر بازی دو بازیکن وجود دارند که بصورت سطری و ستونی اعمالشان نشان داده شده است. درایه های ماتریس همان پاداش بازیکنان است. در بازی اول که یک بازی با جمع صفر است پاداش بازیکن دوم منفی پاداش اولی است. در حالت کلی ماتریس پاداش برای بازی های دونفره با دو ماتریس نشان داده می‌شود. هر حالت دربازی اتفاقی می تواند بصورت یک بازی ماتریسی بیان شود. در هر حالت از بازی بازیکنان پس از بازی و دریافت پاداش با توجه به عمل گروهی انجام شده به حالت دیگری ازبازی می‌روند. در یک بازی تصادفی، اعمال بازیکنان همزمان صورت می گیرد. بطور کلی تعریف رسمی بازی های تصادفی بصورت زیر است[18]:

تعریف۲. یک بازی تصادفی به صورت‌چندتایی $\langle n, S, A_{1..n}, T, R_{1..n}\rangle$ بیان میشود که در آن n تعداد عامل ها، S مجموعه حالات، A_i مجموعه اعمال هر عامل i (در فضای اعمال گروهی $A_1 \times A_2 \times \ldots \times A_n$)، T تابع انتقال $S\times A\times S \rightarrow [0,1]$ و r تابع پاداش برای عامل i ام $S\times A\rightarrow \Re$ است.

در یک بازی تصادفی کاهش پذیر، هدف یک بازیکن بیشینه کردن مجموع پاداش کاهش یافته با پارامتر کاهش $\gamma \in [0;1)$ است. اگر π_iاستراتژی بازیکن i باشد بازای یک وضعیت اولیه s، بازیکن i سعی در بیشینه نمودن رابطه زیردارد:

$$V(s,\pi^1,\pi^2,\ldots,\pi^n)\equiv\Sigma_{t=0}^{\infty}\gamma^t E(r_t\left|\pi^1,\pi^2,\ldots,\pi^n,s_0=s\right.) \quad (۳)$$

	T	L
T	1,-1	-1,1
L	-1,1	1,-1

	L	R
L	6,6	2,7
R	7,2	0,0

شکل۱. دو نوع بازی ماتریسی به نامهای تطابق سکه ها (سمت چپ) و بازی Chicken (سمت راست)

با نگاه به (۳)و(۱) تشابه با مدل MDP دیده می شود منتها در اینجا چندین‌عامل وجود دارند ورفتن به وضعیت بعدی و دریافت پاداش به عمل گروهی عاملها بستگی دارد. در حالت یادگیری تقویتی چند عامله، بیشینه نمودن سودمندی(پاداش) مورد انتظار هر عامل در بازی های مجموع کلی ممکن نبوده و هدف پیدا کردن سیاست متعادل در بازی های مارکوف است. از جمله این سیاست ها می توان سیاست تعادل نش را نام برد. به عبارت دیگر پیدا کردن یک سیاست متعادل به عنوان یک راه حل برای بازی های اتفاقی محسوب می شود.

۲-۲ استراتژی های تعادل

در بازی های با جمع کلی، راه‌حل مبنا تعادل نش است[18]. نظریه‌ی تعادل‌های نش می‌گوید که در هربازی به فرض آنکه بازی کنان معقولانه استراتژی های خود را انتخاب کرده و به دنبال بدست آوردن حداکثر بهره(سود) از بازی باشند، حداقل یک استراتژی برای بدست آوردن بهترین نتیجه برای هر بازیکن قابل انتخاب است که اگر بازیکن راهکار دیگری به غیر از آن را انتخاب کند، نتیجه‌ی بهتری بدست نخواهد آورد[19].

تعریف۳. در یک بازی تصادفی Γ، یک تعادل نش بصورت چندتایی استراتژی های $(\pi_1,\ldots,\pi_n)$ تعریف می شود بطوریکه برای هر $s \in S$ و $i = 1,\ldots,n$ داریم :

$$V^i(s,\pi_*^1,\pi_*^2,\ldots,\pi_*^n) \geq V^i(s,\pi_*^1,\ldots,\pi_*^{i-1},\pi_*^i,\pi_*^{i+1},\ldots,\pi_*^n) \quad (۴)$$
$$for\ all\ \pi^i \in \Pi^i$$

که در آن Π^i مجموعه استراتژی های قابل دسترس برای عامل i است. استراتژی هایی که یک تعادل نش را می سازند می توانند استراتژی های ایستا باشند. قضیه زیر نشان می دهد که حداقل یک تعادل دراستراتژی های ایستا وجود دارد :

قضیه ۱. (Fink 1964) هربازی‌اتفاقی‌کاهش پذیر n نفره‌حداقل یک‌نقطه تعادل‌نش‌دراستراتژی‌های ایستا را داراست[20].

۲-۳ حل بازی های تصادفی

در بازی های اتفاقی بر خلاف MDP ها یک راه حل بهینه ای که مستقل از عاملهای دیگر باشد محتمل نیست. لذا نسبت به حالت تک عامله مساله مشکل تر است. همچنین ممکن است که چندین نقطه تعادل وجود داشته و هماهنگی عاملها برای توافق کار مشکلی است. مفهوم راه حل‌دربازی‌های‌اتفاقی به معنای‌پیداکردن نقطه تعادل منحصر به فرد می باشد. الگوریتمهای مختلفی برای حل این بازی های پیشنهاد شده اند که در یک رده بندی می توان به روشهایی که از تئوری بازی ها استفاده می کنند و روشهای مبتنی بر یادگیری تقویتی اشاره نمود. در روشهای مبتنی بر تئوری بازی ها، نیاز به مدل محیط داشته و توابع T و R نیزبایستی مشخص باشند. هدف این الگوریتمها محاسبه مقدارتعادل‌بازی (پاداش مورد انتظار کاهش یافته برای هر عامل) است، لذا نیاز به فرضیات قوی تری از رفتار عاملها دارند. درمقابل، الگوریتمهای مبتنی بر یادگیری تقویتی چند عامله فرض می کنند محیط ناشناخته بوده و مشاهدات T و R از طریق تجربی قابل مشاهده هستند. هدف این‌الگوریتمها پیدا کردن سیاست تعادل بوده و معمولا نیازهای کمتری را درباره رفتار عاملهای دیگر دارند. یک چارچوب‌کلی برای یادگیری Q چند عامله در شکل ۲ نشان داده شده است. در قالب کلی، الگوریتم های مختلف یک ورودی تابع انتخاب تعادل را دریافت می کنند تا بتوانند با توجه به بردار ماتریسی نظیر $Q=(Q_1,..Q_n)$ تابع ارزش V رامحاسبه نمایند. Hu و Wellman از تابع با نام $Nash\text{-}Q$ برای محاسبه مقدار V طبق رابطه (۵) استفاده نمودند[8]. Qioوهمکاران نیز از مکانیزم چانه زنی[i] (۶) [21]وMeiping Song نیز از الگوریتم Pareto-Q استفاده کردند [11].

$$V_i(s) \in Nash_i(Q_1(s),...,Q_n(s))$$
$$Nash_i(Q_i(s)) = \pi^1(s)...\pi^n(s).Q_i(s) \quad (۵)$$

$$V_i(s) \in NBS_i(Q_1(s),...,Q_n(s))$$
$$NBS_i(Q_i(s)) = Max_{\vec{a}}(Q_1(s,\vec{a}) \times ..., Q_n(s,\vec{a})) \quad (۶)$$

Multi-agent Q-learning (Stochastic Game, α, γ, M)

Inputs: discount factor γ ,learning rate α, M total training time

Output : state-value functions, action–valueV_i* function Q_i*

Initialize: s, $a_1 \ldots a_N$, $Q_1..Q_N$

1. *for* $k=1$ *to* M
2. simulate actions $a_1,...,a_n$ in state s
3. observe rewards $R_1,...,R_n$ and next state s'
4. *for* $i = 1$ *to* N
   (a) compute $V_i(S')$
   (b) $Q_i(s,\vec{a}) \leftarrow (1-\alpha)Q_i(s,\vec{a}) + \alpha[(1-\gamma)R_i + \gamma Vi(s')]$
5. agents choose action $a'_1, ..., a'_N$
6. $s = s'$,$a_1 = a'_1$,..., $a_N = a'_N$
7. decay α

شکل۲. الگوریتم یادگیری Q چند عامله

با توجه به پیچیدگی بالای محاسبات در هر یک ازالگوریتمهای محاسبه تعادل نش در بازی های اتفاقی، به نظر می رسد استفاده از الگوریتمهای مبتنی بر اتوماتای یادگیر با توجه به سادگی و پیچیدگی محاسباتی کم این الگوریتمها بتوانند کارایی مطلوبی را از خود ارائه دهند. با توجه به این موضوع در بخش بعدی مدلی مبتنی بر اتوماتای یادگیر ارائه و نتایج ارائه گردیده اند.

[i] Nash Bargaining