University of Qom
Engineering Management and Soft Computing
JEMSC.QOM
Online ISSN: 2538-2675

Performance Analysis of Proxy-Based Object-Oriented Distributed Systems Using Game Theory

Hossein Sadr[1], Pyman Bayat[2] and Mozhdeh Nazari Solimandarabi[3]✉

1. Corresponding author, Assistance Prof. Department of Computer Engineering, Islamic Azad University of Rasht Branch, Rasht, Iran. Email: sadr@qiau.ac.ir
2. Assistance Prof. Department of Computer Engineering, Islamic Azad University of Rasht Branch, Rasht, Iran. Email: bayat@iaurasht.ac.ir
3. PHD Student, Department of Computer Engineering, Islamic Azad University of Rasht Branch, Rasht, Iran. Email: mozhdeh_nazary@yahoo.com

Article Info

Article type:
Research Article

Article history:
Received 2022 May 4
Received in revised form 2022 Jun 72
Accepted 2022 Jun 14
Published online 2022 Sep 16

Keywords:
Distributed Computing,
Game Theory,
Nash Equilibrium,
Object-Oriented Distributed System,
Proxy.

ABSTRACT

Recently, there has been a remarkable growth of research on the practical applications of game theory in networks, and in particular, the modeling of users' behavior in distributed and decentralized systems. Reducing the runtime of operations in these types of systems will increase their performance. In order to achieve this goal, the system can be implemented using an object-oriented approach, through which the client machine treats the srver machine as an object, and the communication between them is done only through a proxy. In these types of systems, users have a set of possible choices, and may choose personal benefits over the interest of the whole system and other users. Since in a distributed system, all users want to control their resource of choice, the use of game theory can be a good tool to evaluate the behavior of selfish nodes. In this paper, game theory is used to investigate the behavior of nodes in an object-oriented distributed system, in which the communication between the client machine and the server machine is established through a proxy. To understand the behavior of nodes in a distributed system,one-time games and infinitely-repeated games are studied, and finally, the behavior of one node against an object-oriented distribution system is analyzed. According to the results of this study, nodes defect and will be uncooperative in one-time games. But when there is a strategy of an infinitely-repeated game, the cooperation between nodes will depend on the discount factor, or the probability of the next stage

Cite this article Sadr, H., & Others (2022). Performance Analysis of Proxy-Based Object-Oriented Distributed Systems Using Game Theory. *Enginiiring Management and Soft Computing, 8* (1). 123-137. **DOI**: https://doi.org/10.22091/jemsc.2019.1307

© The Author(s)
DOI: https://doi.org/10.22091/jemsc.2019.1307
Publisher: University of Qom

JEMSC.QOM
شاپا الکترونیکی: ۲۶۷۵-۲۵۳۸

مدیریت مهندسی و رایانش نرم

دانشگاه قم

تحلیل و بررسی عملکرد سیستم‌های توزیع شده شئ‌گرا مبتنی بر پراکسی با استفاده از نظریه بازی‌ها

حسین صدر[۱]، پیمان بیات[۲] و مژده نظری سلیماندارابی[۳]

۱. نویسنده مسئول، گروه مهندسی کامپیوتر، واحد رشت، دانشگاه آزاد اسلامی، رشت، ایران. رایانامه: Sadr@qiau.ac.ir

۲. گروه مهندسی کامپیوتر، واحد رشت، دانشگاه آزاد اسلامی، رشت، ایران. رایانامه: Bayat@rashtiau.ac.ir

۳. باشگاه پژوهشگران جوان و نخبگان، واحد رشت، دانشگاه آزاد اسلامی، رشت، ایران. رایانامه: Mozhdeh_nazary@yahoo.com

اطلاعات مقاله

نوع مقاله: مقاله پژوهشی

تاریخ دریافت: ۱۴۰۱/۰۲/۱۲

تاریخ بازنگری: ۱۴۰۱/۰۳/۱۷

تاریخ پذیرش: ۱۴۰۱/۰۳/۲۴

تاریخ انتشار: ۱۴۰۱/۰۶/۲۵

کلیدواژه‌ها:
پراکسی،
تعادل نش،
سیستم توزیع شده شی‌گرا،
محاسبات توزیع شده،
نظریه بازی‌ها.

چکیده

اخیراً شاهد رشد قابل توجه تحقیقات در زمینه‌های کاربردی نظریه بازی‌ها در شبکه‌ها و بخصوص مدلسازی رفتار کاربران در سیستم‌های توزیع شده و نامتمرکز هستیم. کاهش زمان اجرای عملیات در این نوع سیستم‌ها منجر به بالا رفتن کارایی آن می‌شود. برای حصول این هدف، می‌توان آن را به صورت شئ‌گرا پیاده‌سازی کرد که ماشین مشتری با ماشین خدمتگزار به عنوان یک شئ برخورد کرده و ارتباط بین آن‌ها تنها از طریق پراکسی صورت می‌گیرد. کاربران در این نوع سیستم‌ها مجموعه‌ای از انتخاب‌های ممکن را پیش روی خود دارند و ممکن است در تصمیم‌گیری‌های خود سود شخصی را به منافع کلی سیستم و سایر کاربران ترجیح دهند. از آنجا که در یک سیستم توزیع شده تمامی کاربران می‌خواهند منابع مورد نظر خود را در اختیار گیرند، استفاده از نظریه بازی‌ها می‌تواند ابزار مناسبی برای بررسی رفتار گره‌های خودخواه باشد. در این مقاله با استفاده از تئوری بازی‌ها رفتار، گره‌ها در یک سیستم توزیع شده شئ‌گرا ارزیابی شده است که در آن ارتباط بین ماشین مشتری و خدمتگزار به وسیله پراکسی صورت می‌گیرد. برای درک رفتار گره‌ها در سیستم توزیع شده، بازی‌های تک‌مرحله‌ای و بازی‌هایی که بی‌نهایت بار تکرار می‌شوند، بررسی شده و در انتها نیز رفتار یک گره در مقابل یک سیستم توزیع شئ‌گرا مورد تحلیل و بررسی قرار گرفت. از نتایج نهایی این ارزیابی می‌توان گفت در بازی‌های تک‌مرحله‌ای گره‌ها غیرتعاونی بوده و با هم همکاری نمی‌کنند، اما زمانی که استراتژی بازی بی‌نهایت بار تکرار شود، همکاری بین گره‌ها به عامل تخفیف یا همان احتمال مرحله بعد بستگی خواهد داشت.

استناد: صدر، حسین؛ بیات، پیمان و نظری سلیمانداری، مژده (۱۴۰۱). «تحلیل و بررسی عملکرد سیستم‌های توزیع شده شئ‌گرا مبتنی بر پراکسی با استفاده از نظریه بازی‌ها». *مدیریت مهندسی و رایانش نرم، دوره ۸* (۱). صص: ۱۳۷-۱۲۳. https://doi.org/10.22091/jemsc.2019.1307

ناشر: دانشگاه قم

© نویسندگان.

۱) مقدمه

امروزه نیاز به منابع توزیع شده بیش از پیش افزایش یافته است و با توسعه استفاده از اینترنت این نیاز بیشتر احساس می‌شود. اینترنت و قابلیت‌های فعلی آن امکاناتی را ارائه می‌دهند که اگر به درستی مورد استفاده قرار گیرند، تأثیر و اهمیت زیادی روی زندگی بشر خواهند داشت که نتیجه آن وجود انواع مختلف سیستم‌های توزیع شده است (مسوس و همکاران، ۲۰۱۹).

هدف اصلی سیستم‌های کامپیوتری، پردازش و اجرای درخواست کاربران به گونه‌ای است که در کوتاه‌ترین زمان ممکن پاسخ خود را دریافت کنند و سیستم‌های توزیع شده راهکاری برای دستیابی به این هدف می‌باشند. هر سیستمی که روی مجموعه‌ای از ماشین‌ها بدون حافظه اشتراکی اجرا شده و برای کاربران به گونه‌ای اجرا شود که گویا روی یک کامپیوتر در حال اجرا می‌باشد، یک سیستم توزیع شده نامیده می‌شود (لونی و ژوسانگل، ۲۰۱۷). مدیریت صحیح و استفاده درست از منابع سیستم کامپیوتری توزیع شده در بالا بردن توان عملیاتی سیستم نقش بسزایی داشته و از اهمیت بالایی برخوردار است. استفاده از مفهوم شئ‌گرایی در این نوع سیستم‌ها نیز منجر به ارائه ویژگی‌های توزیعی مطلوب و کمینه کردن مشکلات این سیستم‌ها می‌شود. درواقع، شئ‌گرایی یک راهکار برای پیاده‌سازی سیستم توزیع شده است که بر اساس آن ارتباط بین مشتری و خدمتگزار تنها از طریق پراکسی صورت می‌گیرد (ماردن و شاما، ۲۰۱۸).

به بیان دیگر، به دلیل ورود وظایف در سیستم کامپیوتری توزیع شده بر مبنای الگوهای تصادفی و نامنظم و ناهمگن بودن توان پردازشی هر یک از منابع موجود در سیستم، یک منبع محاسباتی در یک نقطه از سیستم توزیع شده ممکن است دچار سربار شده باشد، در حالی که در نقطه دیگری از سیستم، منبعی به کار گرفته نشود و بیکار باشد (وستر، ۲۰۱۸).

درواقع، سیستم‌های توزیع شده می‌توانند منابعی را به کاربران پیشنهاد دهند تا به وسیله آن‌ها قادر به افزایش سرعت اجرای برنامه‌های یک کامپیوتر پیچیده و وابسته به زمان از طریق اجرای آن‌ها در پردازنده‌های موجود در شبکه باشند. گره‌هایی که وظایف خود را سریع انجام دهند نیز می‌توانند سودی را بر اساس قوانین شبکه دریافت کنند (مایلات، ۲۰۱۹). با در نظر داشتن این مسئله، این سؤال پیش می‌آید که برای کاربران (گره‌ها) بهتر است که وظایفشان را خودشان انجام دهند یا با سایر گره‌ها همکاری کرده و اجرای بخشی از برنامه‌ها را به سایر کاربران واگذار کنند (اسدی، آگاه و همکاران، ۲۰۱۸).

به منظور بررسی و تحلیل سیستم‌های توزیع شده در این مقاله از نظریه بازی‌ها به عنوان یک ابزار ریاضی استفاده شده است که می‌تواند چالش‌های موجود در سیستم‌های توزیع شده را مدیریت کند. نظریه بازی‌ها را می‌توان به عنوان علم مدلسازی و بررسی رفتار سیستم‌های تصمیم‌گیرنده تعریف کرد. هر بازی شامل مجموعه‌ای از بازیکنان، مجموعه‌ای از استراتژی‌های ممکن برای هریک از آن‌ها و بالاخره مجموعه از توابع سود برای هر بازیکن نسبت به استراتژی‌های بازیکنان بازی می‌شود (والن، بارتا و همکاران، ۲۰۱۷). یکی از اهداف این نظریه، پیش‌بینی پیش‌آمدهای محتمل برای بازی‌های تصمیم‌گیری است. غالباً به دنبال پاسخی برای این پرسش هستیم که با فرض عقلانیت برای بازیکنان، آن‌ها چه استراتژی برای حداکثر کردن سود خود در بازی با توجه به عکس‌العمل‌های ممکن برای سایر بازیکنان، انتخاب خواهند کرد؟

متداول‌ترین پاسخ به این سؤال، نقاط تعادل نش بازی‌ها هستند که در آن‌ها هیچ یک از بازیکنان به تنهایی نمی‌تواند با تغییر استراتژی، سود خود را افزایش دهد (کلمن ۲۰۱۶).

به بیان دیگر، نظریه بازی‌ها تلاش می‌کند تا رفتار ریاضی حاکم بر یک موقعیت استراتژیک (تضاد منافع) را مدلسازی کند. این موقعیت زمانی پدید می‌آید که موفقیت یک فرد وابسته به راهبردهایی است که دیگران انتخاب می‌کنند. هدف نهایی این دانش یافتن راهبرد بهینه برای بازیکنان است که استفاده از آن‌ها در سیستم‌های توزیع شده می‌تواند به مدیریت منابع و زمان‌بندی محاسبات کمک کند (ابراهام، دلو و همکاران، ۲۰۱۹).

در این مقاله حالت وجود دو گره در بازی استاتیک تک مرحله‌ای را مورد تحلیل و بررسی قرار دادیم و این نتیجه حاصل شد که در بازی‌های تک‌مرحله‌ای، استراتژی که گره‌ها انتخاب می‌کنند این است که با هم همکاری نداشته باشند، به این معنی که آن‌ها ترجیح می‌دهند به جای به اشتراک گذاشتن وظایفشان با سایر گره‌ها خودشان آن‌ها را انجام دهند. در ادامه، این مدل را با معرفی فاکتور تخفیف در بازی‌هایی که بی‌نهایت بار تکرار می‌شوند، توسعه دادیم. همچنین رابطه بین یک گره و یک سیستم توزیع شده را مدلسازی کردیم.

ادامه مقاله به صورت زیر سازماندهی شده است: در بخش دوم مفاهیم مرتبط با سیستم توزیع شده، شئ‌گرایی و نظریه بازی‌ها بیان شده است. در بخش سوم، رفتار گره‌ها در یک بازی تک‌مرحله‌ای بررسی شده و در بخش چهارم این مدل برای بازی‌هایی که بی‌نهایت بار تکرار می‌شوند، توسعه می‌یابد. در ادامه و بخش پنجم نیز رفتار متقابل بین یک گره و سیستم توزیع شده مورد تحلیل و بررسی قرار می‌گیرد. عملکرد استراتژی معرفی شده با مثالی در بخش ششم معرفی شده است. نتیجه‌گیری و مسیر تحقیقات آینده در بخش هفتم بیان شده‌اند.

۲) پیشینه پژوهش

در سال‌های گذشته تئوری بازی‌ها به طور قابل‌توجهی در تحلیل و بررسی سیستم‌های توزیع شده مورد استفاده قرار گرفته است. در یکی از تحقیقات انجام شده در این زمینه یک چارچوب یکپارچه برای بررسی بهره‌وری، بیشینه کردن کارایی و استراتژی هزینه در برای اختصاص کار در شبکه‌های توزیع شده موبایل معرفی شده است. در ادامه کوک و همکارانش یک مدل سلسله‌مراتبی از نظریه بازی‌ها در سیستم توزیع شده معرفی کردند اما تنها روی تأثیر کارهای غیرمشارکتی تأکید داشتند و با استفاده از یک تابع بهره‌وری جدید یک استراتژی بهینه را بر اساس تعادل نش معرفی کردند (کوک، سونگ و همکاران ۲۰۰۵). در ادامه از تئوری بازی‌ها به عنوان ابزاری برای مقایسه و تحلیل و بررسی مکانیسم‌های اختصاص منابع در شبکه‌های توزیع شده استفاده شد (خان و احمد ۲۰۰۶). اما با بررسی‌های انجام شده مشخص شده است که تاکنون بررسی روی عملکرد سیستم‌های توزیع شده شیءگرا با استفاده از تئوری بازی‌ها صورت نگرفته است. به همین دلیل هدف ما در این مقاله ۱) استفاده از بازی‌هایی که بی‌نهایت بار تکرار می‌شوند برای بررسی مشارکت کاربران در سیستم توزیع شده شیءگرا و ۲) ارائه انواع تجزیه و تحلیل‌ها با استفاده از نظریه بازی‌ها است که به خوبی می‌توانند در سیستم‌های توزیع شده به کار برده شوند.

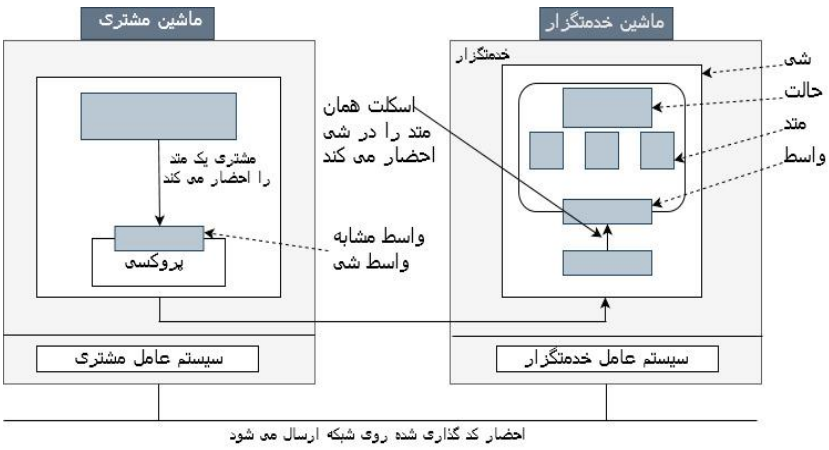
با توجه به این که تأکید این مقاله روی بررسی عملکرد سیستم‌های توزیع شده شئ‌گرا با استفاده از نظریه بازی‌ها است، در ابتدا به صورت خلاصه مفاهیم مرتبط با آن‌ها بیان شده و کارهای انجام شده در این حوزه مورد بررسی قرار می‌گیرند.

۳) سیستم‌های توزیع شده شیءگرا مبتنی بر پراکسی

تعریف‌های مختلفی از سیستم‌های توزیع شده ارائه شده است اما هیچ یک از آن‌ها بیانگر یک تعریف کامل و جامع از این نوع سیستم‌ها نبوده و همچنین در مواردی این تعریف‌ها با هم هماهنگ نیستند. کولوریس یک سیستم توزیع شده را به عنوان "یک سیستم که در آن مؤلفه‌های سخت‌افزاری و نرم‌افزاری در کامپیوترهای شبکه‌بندی شده تعبیه شده‌اند و تنها با استفاده از روش انتقال پیام با یکدیگر تعامل دارند" تعریف کرده است. تننبام یک سیستم توزیع شده را به این صورت تعریف می‌کند: " مجموعه‌ای از سیستم‌های کامپیوتری مستقل که برای کاربران به صورت یک کامپیوتر واحد دیده می‌شود" (لونی و ژوسانگل، ۲۰۱۷).

برای یک سیستم توزیع شده می‌توان چندین نوع معماری در نظر گرفت که عبارتند از معماری چندپردازنده‌ای، معماری مشتری-کارگزار و معماری شئ توزیع شده. با توجه به این که تأکید این مقاله روی شیءگرایی سیستم توزیع شده است، جزئیات مربوط به آن را در این بخش مرور می‌کنیم.

چالش‌های طراحان سیستم‌های توزیعی، طراحی نرم‌افزار و سخت‌افزار برای ارائه ویژگی‌های مطلوب سیستم توزیعی و کمینه کردن مشکلات این سیستم‌ها است. روش کلی‌تری برای طراحی سیستم توزیع شده، حذف تمایز بین مشتری و کارگزار و طراحی معماری سیستم به صورت معماری شیء توزیعی است (دنگ و لیانگ، ۲۰۱۹). در ساختار معماری شیء توزیعی قطعات اساسی سیستم اشیایی هستند که واسطی را برای مجموعه‌ای از سرویس‌ها فراهم می‌نمایند. اشیاء دیگر این سرویس‌ها را بدون هیچ‌گونه توزیع منطقی بین مشتری و سرویس‌دهنده فراخوانی می‌کنند. اشیا ممکن است در تعدادی از کامپیوترهای شبکه توزیع شوند و از طریق میان‌افزار باهم ارتباط داشته باشند. این میان‌افزار را می‌توان گذرگاه نرم‌افزار در نظر گرفت که مجموعه‌ای از سرویس‌ها را ارائه می‌کند که به اشیا اجازه می‌دهد با هم ارتباط برقرار کنند و به سیستم اضافه یا از آن حذف شوند. این میان‌افزار کارگزار درخواست شیء نام دارد و نقش آن برقراری واسط پیوسته بین اشیا است (تانبوم و استن، ۲۰۰۷).



شکل ۱. سازماندهی معمول یک شکل راه دور با پروکسی- سمت مشتری

جداسازی بین واسط‌ها و اشیائی که آن‌ها را پیاده‌سازی می‌کنند، یکی از مفاهیم کلیدی در سیستم‌های توزیعی است. این جداسازی ما را قادر می‌سازد تا (برای مثال) واسط را در یک ماشین قرار داده و خود شئ را روی ماشین دیگری قرار دهیم. این سازماندهی، که در شکل ۱ نشان داده شده، با نام شئ توزیعی شناخته می‌شود (تانبوم و استن، ۲۰۰۷).

وقتی مشتری با یک شیء توزیعی پیوند برقرار می‌کند، پیاده‌سازی واسط آن شئ که یک پروکسی نامیده می‌شود، در فضای آدرس ماشین مشتری باز می‌شود. پروکسی شبیه بنیان در سیستم‌های RPC است. تنها کار پروکسی این است که احضار متد را به صورت چند پیام کدگذاری کرده و بعد از دریافت پیام‌های پاسخ اجرای این متد در خدمتگزار، آن‌ها را کدگشایی و به مشتری تحویل دهد. شئ واقعی در ماشین خدمتگزار باقی می‌ماند، جایی که واسط آن کاملاً مشابه واسط ماشین مشتری است. درخواست‌های احضار متد که از طرف مشتری می‌آیند، ابتدا به یک بنیان خدمتگزار می‌رسند، که در آنجا به شکل مناسب برای تحویل به واسط شئ در ماشین خدمتگزار کدگشایی خواهند شد. بنیان خدمتگزار همچنین مسئول کدگذاری پاسخ خدمتگزار و برگرداندن آن به پروکسی سمت-مشتری نیز هست (تانبوم و استن، ۲۰۰۷). در سیستم توزیع شده پراکسی به عنوان یک رابط بین ماشین مشتری و ماشین خدمتگزار می‌باشد که پیام‌ها را بین آن‌ها مبادله می‌کند. اصل اساسی کار پروکسی دریافت درخواست مشتری، بررسی و آنالیز درخواست، ارسال به سرورهای موردنظر و دریافت پاسخ اصلی یا دستکاری‌شده و ارسال آن به مشتری است. پراکسی‌ها به‌صورت مستقل عمل کرده و فقط وظیفه انتقال را بر عهده دارند بدون آن که هیچ اطلاعاتی در خصوص بسته‌های ارسالی داشته باشند (لونی و ژاسنگل، ۲۰۱۷).

بنیان سمت-خدمتگزار اغلب اسکلت نیز نامیده می‌شود، چون فقط یک وسیله ساده است برای دسترسی میان‌افزار خدمتگزار به اشیا کاربر-تعریف در عمل، اغلب حتی کد کلاس‌های اسکلت کامل نیست و برنامه‌نویس باید آن‌ها را با توجه به نیازهای خود تکمیل کند (تانبوم و استن، ۲۰۰۷).

۴) نظریه بازی‌ها

پیروزی در هر بازی تنها تابع یاری شانس نیست بلکه اصول و قوانین ویژه خود را دارد و البته هر بازیکن طی بازی چه بداند و چه نداند سعی می‌کند با به کارگیری آن اصول خود را به برد نزدیک کند و صد البته در این میان کسی پیروز میدان خواهد بود که بیش از دیگران از این اصول بهره گیرد. نظریه بازی‌ها مطالعه مدل‌های ریاضیِ درگیر و همکار در روشِ تصمیم‌گیری و انتخاب هوشمندانه و منطقی تصمیم‌گیرندگان در یک بازی است. این نظریه در تلاش است تا به وسیله ریاضیات، رفتار را در شرایط راهبردی یا در یک بازی برآورد کند که در آن‌ها موفقیت فرد در انتخاب کردن، وابسته به انتخاب دیگران است (کلمن، ۲۰۱۶).

نظریه بازی‌ها در مطالعه طیف گسترده‌ای از موضوعات کاربرد دارد. از جمله نحوه تعامل تصمیم‌گیرندگان در محیط رقابتی به شکلی که نتایج تصمیم هر عامل مؤثر بر نتایج کسب شده سایر عوامل باشد (والن، بارتا و همکاران، ۲۰۱۷). اصل اصیل نظریه بازی‌ها بر خِرَد رفتار بازکنان است. خرد به این معنا است که هر بازیکن تنها در پی بیشینه کردن سود خود بوده و هر بازیکن می‌داند که چگونه می‌تواند سود خود را بیشینه کند. بنابراین، حدس زدن رفتار ایشان که بر اساس نمودار هزینه-فایده است آسان خواهد بود (کلمن، ۲۰۱۶).

انواع مختلفی از بازی‌ها وجود دارد که می‌تواند در مسائل مختلف مورد استفاده قرار گیرند که توضیح خلاصه آن‌ها در ادامه آمده است. بازی متقارن–نامتقارن بازی است که نتیجه و سود حاصل از یک راهبرد تنها به این وابسته است که چه راهبردهای دیگری در بازی پیش گرفته شود؛ و مستقل از این موضوع است که کدام بازیکن این راهبرد را در پیش‌گرفته است. بازی‌های مجموع صفر، بازی‌هایی هستند که ارزش بازی در طی بازی ثابت می‌ماند و کاهش یا افزایش پیدا نمی‌کند. در این بازی‌ها، سود یک بازیکن با زیان بازیکن دیگر همراه است. بازی‌های تصادفی شامل عناصر تصادفی مانند ریختن تاس یا توزیع ورق هستند و بازی‌های غیرتصادفی بازی‌هایی هستند که دارای راهبردهایی صرفاً منطقی هستند. بازی‌های با آگاهی کامل، بازی‌هایی هستند که تمام بازیکنان می‌توانند در هر لحظه تمام ترکیب بازی را در مقابل خود مشاهده کنند. از سوی دیگر در بازی‌های بدون آگاهی کامل ظاهر و ترکیب کل بازی برای بازیکنان پوشیده است.

۵) تعادل نش

در یک سیستم اقتصادی تعادل به نقطه‌ای گفته می‌شود که در آن هیچ یک از طرفین معامله تمایل به تغییر نداشته باشند و با هرگونه تغییر شرایط بدتر شده و سیستم مجدداً به نقطه تعادل باز می‌گردد. تعادل نش همانند یک نمایه عمل بازی است که با فرض ثابت بودن بازی سایر بازیکنان، هر بازیکن با تغییر بازی خود شرایطش بدتر شود. یا به عبارت‌دیگر، نمایه عملی است که با فرض ثابت بودن بازی سایر بازیکنان هیچ بازیکنی انگیزه تغییر بازی خود را نداشته باشد (کلمن، ۲۰۱۶).

در تئوری بازیها، تعادل نش (به نام جان فوربز نش، که آن را پیشنهاد کرد) راه‌حلی از تئوری بازی‌ها است که شامل دو یا چند بازیکن است، که در آن فرض بر آگاهی هر بازیکن به استراتژی تعادل بازیکنان دیگر است و بدون هیچ بازیکنی که فقط برای کسب سود خودش با تغییر استراتژی یک‌جانبه عمل کند. اگر هر بازیکنی استراتژی را انتخاب کند هیچ بازیکنی نمی‌تواند با تغییر استراتژی خود در حالی که نفع بازیکن دیگر را بدون تغییر نگه داشته باشد عمل کند، سپس مجموعه انتخاب‌های استراتژی فعلی و بهره‌مندی مربوطه، تعادل نش را تشکیل می‌دهد. به همین ترتیب یک گروه از بازیکنان در تعادل نش است اگر هر یک در حال انجام بهترین تصمیم‌گیری باشند که آن‌ها می‌توانند، با توجه به تصمیمات دیگران داشته باشند (گوپتا و سومانی، ۲۰۰۵؛ بارتا و همکاران، ۲۰۱۷).

بر همین اساس و در ادامه، رفتار گره‌ها در یک سیستم توزیع شده مورد بررسی قرار می‌گیرد.گره‌ها تمایل دارند که وظایف خود را در سیستم توزیع شده انجام دهند، این که آن‌ها وظایف خود را به صورت فردی انجام دهند یا با سایر گره‌ها در انجام وظایف مشارکت کنند با استفاده از تعادل نش مورد بررسی قرار گرفته است. در واقع در ادامه قصد ما این است که از بازی‌هایی که بی‌نهایت بار تکرار می‌شوند برای توصیف مشارکت کاربران در سیستم توزیع شده شئ‌گرا استفاده کنیم و نقش گره‌ها را با استفاده از نظریه بازی‌ها در سیستم توزیع شده شئ‌گرا مورد تجزیه و تحلیل قرار دهیم.

۶) بررسی بازی استاتیک تک‌مرحله‌ای بین دو گره

یک بازی را در حالت معمولی می‌توان به صورت یک مجموعه با سه عضو به صورت $G = (N, S, U)$ تعریف کرد که در آن N تعداد بازیکنان، S_i فضای استراتژیک خالص برای بازیکن i است. استراتژی خالص، مقدار 0 را به همه حرکت‌ها اختصاص می‌دهد، به‌جز یکی از آن‌ها. در واقع این استراتژی کاملاً مشخص می‌کند که بازیکن باید چه عملی

را بازی کند. در حالت استراتژی‌های مختلط، بازیکنان عملیات را با احتمال مشخصی بازی می‌کنند. توزیع احتمال استراتژی خالص برای یک بازیکن برابر $\sigma_i(S_i)$ است. مجموعه‌ای مشترک از فضای استراتژی‌های همه بازیکنان به‌صورت $S = S_1 \times S_2 \times \dots \times S_{|N|}$ نشان داده می‌شود. با $S_{-i} = S/S_i$ فضای استراتژی خالص بازیکن i را نشان می‌دهیم. مجموعه استراتژی‌های انتخاب شده، پروفایل استراتژی $S = \{s_1, s_2, \dots\}$ را شکل می‌دهد. سود (مزیت) $u_i(s)$ نشان‌دهنده نتیجه بازیکن i با پروفایل استراتژی s است. به عنوان مثال برای دو بازیکن داریم: $U = \{u_1(s), u_2(s)\}$.

یک راه‌حل برای مسابقه‌های شامل دو بازیکن یا بیشتر تعادل نش است. این مفهوم روی بهترین پاسخ بازیکن بر اساس استراتژی‌های ممکن سایر بازیکنان تکیه می‌کند. در نتیجه بهترین پاسخ برای بازیکن iدر پروفایل استراتژی s_{-i}، استراتژی s_i خواهد بود در صورتی که (گوش، روی و همکاران، ۲۰۰۵):

رابطه ۱) $br_i(s_{-i}) = ar\ max_{s_i \in s_i} u_i(S_i, S_{-i})$

کاملاً مشخص است که اگر دو استراتژی به طور مشترک بهترین پاسخ باشند، هیچ کدام انگیزه‌ای برای تغییر استراتژی خود ندارند و این منجر به آسیب رساندن به دیگران می‌شود. در نتیجه می‌توانیم مفهوم تعادل نش را به این صورت تعریف کنیم: پروفایلی از استراتژی‌های خالص s تعادل نش است، اطلاعات مربوط به تمامی بازینان را حفظ کند (گوش، روی و همکاران، ۲۰۰۵).

رابطه ۲) $U_i(s_i^*, s_{-i}^*) \geq U_i(s_i, s_{-i}^*), \forall s_i \in S_i$

این بدان معنا است که در حالت تعادل نش هیچ کس نمی‌تواند استراتژی خود را تغییر داده و سودش را افزایش دهد. با استفاده از فرضیه نظریه بازی‌ها و تعادل نش هدف ما این است که رفتار گره را در سیستم توزیع شده شئ‌گرا شبیه‌سازی کنیم. مشکل این سیستم‌ها این است که کاربران همیشه مشتاق به همکاری نبوده و باید یک مکانیسم انگیزشی یا اقتدار مرکزی وجود داشته باشد که آن‌ها را وادار به همکاری کند (میشکوکی، فیلیپوکا و همکاران).

شرکت گره‌ها در سیستم توزیع شده شئ‌گرا را به عنوان یک بازی استراتژیک مدل‌سازی کردیم. بازیکنان در این بازی گره‌ها هستند که از طریق پراکسی با یکدگر ارتباط برقرار می‌کنند. هر دو بازیکن یک استراتژی دارند که یا در بازی مشارکت کنند و یا همکاری نکرده و به تنهایی وظایف خود را انجام دهند. در این نوع بازی‌ها بازیکنان قصد رقابت با هم را ندارند و تنها به دنبال همکاری با سایر بازیکنان هستند (اسدی، آگاه و همکاران، ۲۰۱۷). هر دو استراتژی برای هر دوی بازیکنان شناخته شده است. ما وظایفی که گره‌ها در بخش موازی و سریال دارند را نشان می‌دهیم. زمان مورد نیاز برای اجرا در بخش سریال برابر a و در بخش موازی برابر b است. ما این سناریو را با دو بازیکن که به ترتیب دارای $task_1$ و $task_2$ بودند، مورد بررسی قرار دادیم. به منظور راحتی فرض کردیم که هر دو گره وظایفی با اندازه یکسان دارند و زمان اجرای بخش موازی و سریال آن‌ها نیز برابر است. در بازی یک مکانیسم انگیزشی وجود دارد که به بازیکن‌هایی که وظایف خود را سریع انجام دهند جایزه می‌دهد که آن را با ثابت m نشان می‌دهیم. اگر هر دو بازیکن دفاع کنند، هر کدام وظیفه خود را انجام خواهند داد. اگر بازیکن شماره یک مشارکت کرده و بازیکن شماره دو دفاع کند، آنگاه بازیکن شماره یک باید وظیفه خود و نیمی از وظیفه بازیکن دو را انجام دهد. در این حالت زمان انجام وظایف بازیکن شماره دو کاهش یافته

و m را به عنوان جایزه دریافت می‌کند. جدول ۱ ماتریس باز پرداخت را برای یک بازی بر اساس سود و هزینه نشان می‌دهد.

جدول ۱.فرم استراتژی نرمال بازی

		بازیکن ۲: دفاع	بازیکن ۲: همکاری
بازیکن ۱	دفاع	-(a+b),-(a+b)	-(a+b/2)+,-(a+b+b/2)
	همکاری	-(a+b+b/2),-(a+b/2)+m	-(a+b/2+b/2)+m, -(a+b/2+b/2)+m

با استفاده از معادلات (۱) و (۲) مشخص است که تعادل نش برای دو بازیکن در حالت (دفاع، دفاع) است. این بدین معنی است که در بازی‌های استراتژی تک‌مرحله‌ای برای گره‌ها بهتر است که همکاری نکنند، زیرا همکاری با توجه با جایزه اختصاص یافته باعث ایجاد حالت تعادل کمتر از حد مطلوب می‌شود.

با بسط این مفهوم در سیستم توزیع شده شئ‌گرا می‌توان نتیجه گرفت که اگر در یک سیستم توزیع شده شئ‌گرا، دو گره وجود داشته باشد که هرکدام از آن‌ها وظیفه‌ای برای انجام دادن دارند، بر اساس تعادل نش بهتر است که آن‌ها با هم مشارکت نکرده و هر گره وظیفه خود را به صورت مجزا انجام دهد زیرا با مشارکت زمان صرف شده افزایش‌یافته و بر اساس تئوری نظریه بازی‌ها گره‌ای که مشارکت کرده می‌بازد و گره‌ای که مشارکت نکرده برنده بازی خواهد بود.

۷) بررسی بازی بی‌نهایت تکرار شونده بین دو گره

فرض می‌کنیم بازی که در مرحله قبل درباره آن صحبت کردیم، بی‌نهایت بار تکرار شود و برای هر تکرار t، خروجی $t-1$ بازی قبلی پیش از این که مرحله t^{th} شروع شود، باید مشاهده شود. ما از δ به عنوان فاکتور تخفیف استفاده کردیم تا بتوانیم یک بازی که بی‌نهایت بار تکرار می‌شود را به صورت بازی که بعد از تعداد تکرار تصادفی به پایان می‌رسد، تفسیر کنیم. به طور خلاصه، فاکتور تخفیف (کلمن، ۲۰۱۶) احتمال این که بازی به پایان برسد را نشان می‌دهد. در اینجا نشان خواهیم داد که بر خلاف بخش قبل که بازیکنان تمایل به همکاری نداشتند، همکاری بین بازیکن‌ها می‌توانند در تمامی مراحل بازی رخ دهد. اگر فرض کنیم که استراتژی بازیکن اول استراتژی حد آستانه است، این به معنی بازی همکارانه در مرحله اول است. در مرحله t^{th}، اگر خروجی تمامی مرحله‌های پیشین (همکاری، همکاری) باشد، بازیکن‌ها همکاری خواهند کرد، در غیر این صورت دفاع می‌کنند که این نشان‌دهنده همان بازی همکارانه است (ابراهام، آلویسی و همکاران ۲۰۱۱).

در صورتی که هر دو بازیکن خود را با استراتژی حد آستانه تطبیق دهند، خروجی به صورت زیر خواهد بود. فرض می‌کنیم که بازیکن اول خود را با استراتژی حد آستانه تطبیق داده و می‌خواهد زمان مناسب را به بازیکن دوم برای تطبیق با این استراتژی اعلام کند. اگر بازیکن دوم بخواهد بهترین پاسخ را به استراتژی بازیکن اول بدهد، باید همکاری نکند و بازیکن اول را نیز در مراحل بعدی بازی مجبور به دفاع کند. بازپرداخت مربوط به بازیکن دوم به صورت زیر محاسبه می‌شود:

$$V = \left[-\left(a+\frac{b}{2}\right)+m\right] - \sum_{i=1}^{i\to\infty}\delta^i(a+b) = -\left(a+\frac{b}{2}\right)+m-\frac{(a+b)\delta}{1-\delta}$$ رابطه ۳)

درصورتی‌که به صورت همکاری بازی کنند، بازپرداخت به‌صورت زیر خواهد بود:

$$V = -(a+b) + m + \delta V$$

یا

$$V = \frac{-\left(a+\frac{b}{2}+\frac{b}{2}\right)+m}{1-\delta}$$ رابطه ۴)

درنتیجه بازی همکارانه بهینه است اگر و تنها اگر:

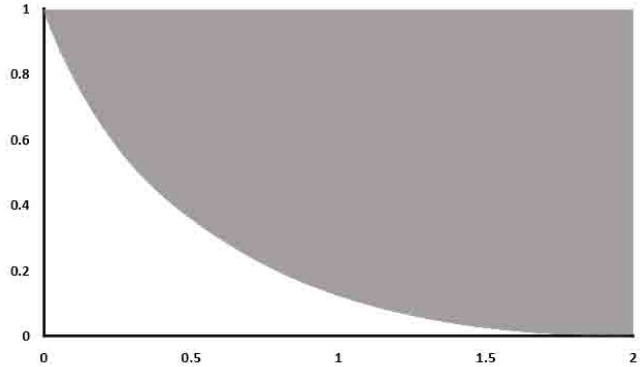
$$\frac{-a-b+m}{1-\delta} \geq -\left(a+\frac{b}{2}\right) + m - \frac{(a+b)\delta}{1-\delta}$$ رابطه ۵)

یا نابرابری زیر مورد نظر است:

$$\delta \geq \frac{1}{1+2m/b}$$ رابطه ۶)

به بیان‌دیگر، اگر خروجی تمامی مراحل پیشین به صورت (همکاری، همکاری) بوده باشد، استراتژی بهینه بازیکن دوم نیز همکاری است، اگر و تنها اگر نابرابری به دست آید. نتیجه این است که اگر هر دو بازیکن با استراتژی تطبیق یابند، خروجی بازی که بی‌نهایت بار تکرار می‌شود در تمامی مراحل می‌تواند همکاری باشد. این همکاری به فاکتور تخفیف بستگی دارد.

این مفهوم را می‌توان با سیستم توزیع شده شئ‌گرا با دو گره که چندین وظیفه برای انجام دارند، مقایسه کرد. بر اساس تعادل نش، اگر دو گره در یک سیستم توزیع شده از ابتدا برای انجام وظایف با هم مشارکت داشتند، استراتژی بهینه این است که تا پایان بازی (اتمام انجام وظایف) با هم همکاری کنند، زیرا همکاری بین دو گره باعث کاهش زمان اجرای وظایف در سیستم توزیع شده شئ‌گرا می‌شود. لازم به ذکر است که در این نوع سیستم‌های توزیع شده شئ‌گرا که دو گره در انجام وظایف با هم همکاری می‌کنند، پراکسی‌ها نقش اساسی در تبادل پیام بین آن‌ها خواهند داشت. در شکل ۲ ناحیه سایه زده شده مقدار فاکتور تخفیف را زمانی که استراتژی حد آستانه تعادل نش است، نشان می‌دهد. محور X نشان‌دهنده نسبت جایزه ای که قانون مرکزی به اشتراک گره‌ها می‌دهد به زمان صرف شده توسط یک گره برای اجرای بخش موازی است.



شکل۲. مقدار فاکتور تخفیف زمانیکه استراتژی آستانه تعادل نش است و به نسبتb/ m بستگی دارد. محور عمودی نشان دهنده δ و محور افقی نشانده b/ m است.

۸) بررسی بازی توسعه‌یافته بین یک گره و محیط توزیع شده شئ‌گرا

در این بخش به جای مدلسازی روابط بین گره‌ها، می‌خواهیم رابطه بین یک گره و سیستم توزیع شده را مورد بررسی قرار دهیم. زمانی که یک گره می‌خواهد در یک سیستم توزیع شده مشارکت کند، با پاداش‌ها یا الزامات خاصی مانند زمان هزینه ثابت برای پردازش وظایف برای سایر گره‌ها مواجه می‌شود. یک گره در صورتی می‌تواند بخشی از یک سیستم توزیع شده باشد که مقدار سود آن نسبت به زیانش بیشتر باشد. در واقع یک بازی همکارانه با چندین بازیکن را مورد بررسی قرار می‌دهیم درحالی‌که بازی می‌تواند بی‌نهایت بار تکرار شود (کوک سنگ و همکاران، ۲۰۰۵).

بازیکنان در این نوع بازی گره و سیستم توزیع شده شیء‌گرا می‌باشد. گره تمایل دارد که زمان صرف شده برای اجرای یک وظیفه مشخص را کاهش دهد و هدف شبکه نیز این است که اطمینان حاصل کند گره خود را در محیط شبکه قرار می‌دهد زیرا افزایش مشارکت گره در سیستم توزیع شده (مانند اختصاص منابع بیشتر یا پرداخت مبالغ بالاتر) می‌تواند باعث بهبود عملکرد سیستم توزیع شده شئ‌گرا شود (گوپتا و سومانی، ۲۰۰۵).

زمانی که گره وظایفی را برای اجرا داشته باشد، دو حالت ممکن پیش خواهد آمد. گره می‌تواند به صورت خودمختار عمل کرده و خودش به تنهایی وظیفه خود را اجرا کند یا در محیط توزیع شده مشارکت کرده و وظایفش را بین گره‌های موجود برای اجرا توزیع کند. زمانی که گره خودش وظایفش را انجام می‌دهد، هیچ تعهدی نسبت به سایر گره‌ها ندارد اما اگر به محیط توزیع شده ملحق شود از آن انتظار می‌رود که در اجرای وظایف دیگر نیز شرکت کند. وظایفی که گره ممکن است داشته باشد زمان اجرای یکسانی دارند و تفاوت آن‌ها تنها در این است که چه مقدار می‌توانند موازی شوند. کاملاً مشخص است که تمامی وظایف نمی‌توانند موازی اجرا شوند در نتیجه، گره وظایف را بین قسمت‌هایی تقسیم می‌کند که می‌توانند در محیط توزیع شده اجرا شوند و بقیه قسمت‌ها نیز توسط خود گره انجام می‌شوند.

$$T(\sigma) = a(\sigma) + \mathrm{b}(\sigma)$$ رابطه ۷)

در اینجا $T(\sigma)$زمان مورد نیاز برای اجرای وظیفه σ است. بسته به نوع وظیفه، آن وظیفه می‌تواند توسط گره $a(\sigma)$ انجام شود و بخش دیگر $b(\sigma)$ می‌تواند توسط سیستم توزیع شده شئ‌گرا موازی‌سازی شود. در ابتدای بازی گره نوع وظیفه را اعلام می‌کند. با توجه به قیمت اعلام شده توسط محیط، گره می‌تواند انتخاب کند که تنها باشد یا مشارکت کند. تابع عملکرد گره شامل زمانی است که گره صرف اجرای وظیفه می‌کند. تفاوت بین دو جایگزین یک گره، مدت زمانی است که برای اجرای یک وظیفه صرف می‌شود، در نتیجه تابع عملکرد برابر است با:

$$T_{No} = -t$$ رابطه ۸)

در اینجا t زمانی است که گره صرف اجرای وظیفه می‌کند. در واقع در صورتی گره در محیط توزیع شده مشارکت می‌کند که زمان صرف شده برای اجرای وظیفه زمانی که همکاری می‌کند، کمتر از حالتی باشد که خودش به تنهایی وظایفش را انجام می‌دهد.

زمانی که یک گره می‌خواهد مشارکت کند، سیستم توزیع شده شئ‌گرا نیازمند درگیری مشخص در توابع شبکه هنگام اجرای وظیفه است. گره با دو حالت ممکن پذیرش یا عدم پذیرش نیازمندی‌های شبکه مواجه است. در این حالت ما تعامل بین گره و محیط توزیع شده را در سطح انتزاعی نشان می‌دهیم (ابراهام آلویسی و همکاران، ۲۰۱۱).

بازیکن دوم، محیط توزیع شده شئ‌گرا، شامل N گره همگن با ویژگی‌های محاسباتی یکسان است. از تأثیر شبکه روی هزینه نهایی محیط توزیع شده شئ‌گرا صرفنظر شده و فرض شده است که هیچ تحرکی در شبکه وجود ندارد. هدف اصلی سیستم توزیع شده شئ‌گرا این است که گره را مجبور به مشارکت کند که پیرو آن زمان کلی صرف شده کاهش خواهد یافت. مشارکت گره برای سیستم توزیع شده شئ‌گرا از اهمیت بالایی برخوردار است زیرا باعث افزایش ظرفیت محاسباتی می‌شود (ابراهام آلویسی و همکاران، ۲۰۱۱).

با توجه به این که ما روی زمان صرف شده تمرکز کرده‌ایم، سطح مشارکت یک گره می‌تواند به وسیله زمانی که آن گره صرف اجرای وظیفه می‌کند اندازه‌گیری شود. در نتیجه تابع عملکرد سیستم توزیع شده برابراست با:

$T_{Ne} = c$ رابطه ۹)

که در اینجا c برابر زمانی است که گره صرف انجام وظایف سیستم توزیع شده شئ‌گرا می‌کند. در اینجا ما یک بازی با یک گره صادق را انجام می‌دهیم که یا وظایف خود را به صورت مختارانه انجام می‌دهد یا در محیط توزیع شده شئ‌گرا مشارکت کرده و وظایف سایر گره‌های سیستم توزیع شده شئ‌گرا را نیز انجام می‌دهد. اگر گره تصمیم بگیرد که وظیفه را خودش انجام دهد، آنگاه زمان صرف شده برای اجرای یک وظیفه مشخص برابر $T(\sigma) = a(\sigma) + b(\sigma)$ است و اگر گره در سیستم توزیع شده مشارکت کند زمان آن به صورت زیر خواهد بود:

$\frac{T(\sigma)-a(\sigma)}{N} + a(\sigma)$ رابطه ۱۰)

اگر یک گره از گره‌های موجود در سیستم توزیع شده شئ‌گرا استفاده کند باید در اجرای وظایف آن‌ها نیز مشارکت کند. ما این بازی را به صورت یک بازی توسعه‌یافته مدلسازی کردیم که ساختار آن به این صورت است: محیط توزیع شده شئ‌گرا به گره پیشنهاد می‌دهد که وظیفه را در جهت پاداش محاسباتی c انجام دهد و گره می‌تواند بپذیرد یا درخواست را رد کند. استراتژی بهینه برای گره مشهود است.

اگر داشته باشیم: $If\ c \le c_0 = \frac{T(\sigma)-a(\sigma)}{n} - a(\sigma)$ آنگاه گره از محیط توزیع شده شئ‌گرا برای اجرای وظیفه استفاده می‌کند، در غیر این صورت خودش وظیفه را انجام می‌دهد که عملکرد آن به‌صورت زیر خواهد بود.

$U_{N0} = max\ (\frac{-T(\sigma)-a(\sigma)}{N} - a(\delta) - C, -T(\delta))$ رابطه ۱۱)

اگر $c > c_0$ باشد آنگاه گره بدون توجه به سیستم توزیع شده شئ‌گرا وظیفه را انجام می‌دهد و شبکه چیزی را به دست نخواهد آورد آنگاه عملکرد شبکه به صورت زیر خواهد بود:

$U_{Ne} = \begin{cases} 0, & \text{if } c>c_0 \\ c, & \text{if } c\le c_0 \end{cases}$ رابطه ۱۲)

استراتژی بهینه سیستم توزیع شده شئ‌گرا این است که به سهم c_0 نیاز داشته باشد آنگاه گره در سیستم توزیع شده شئ‌گرا شرکت خواهد کرد. این تعادل نش بازی است و یک حالت پارتو-بهینه یا بهینه اجتماعی است. سؤال پیش‌آمده این است که سیستم توزیع شده شئ‌گرا چه طور باید تشخیص دهد که چه هزینه‌ای را به یک گره مشخص باید پیشنهاد دهد. پاسخ این سؤال به وظیفه‌ای که گره باید انجام دهد بستگی دارد و محیط توزیع شده شئ‌گرا می‌تواند هزینه‌های مختلفی را بر اساس انواع مختلف وظایف پیشنهاد دهد (میسکوکی فیلیپوکا و همکاران).

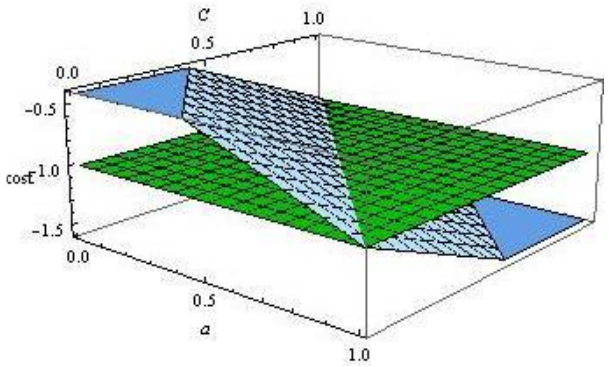
۹) بررسی عملکرد استراتژی معرفی شده

فرض می‌کنیم که شبکه توزیع شده شامل ۱۰۰ گره است و در زمان k گره i دارای وظیفه σ برای اجرا است. این وظیفه می‌تواند در قسمت‌های موازی b یا سریال a تقسیم‌بندی شوند. پس از شناختن وظیفه، مسئولیت شبکه توزیع شده شیء‌گرا پیشنهاد هزینه‌ای است که بتواند سیستم توزیع شده و گره را راضی نگه دارد. بدون هیچ دانش قبلی این کار می‌تواند سخت و ابهام برانگیز باشد. اما اگر شبکه توزیع شده شیء‌گرا دارای قابلیت یادگیری باشد، می‌تواند بهترین هزینه را برای راضی ساختن گره پیشنهاد کند. در جدول ۲ هزینه‌های مختلف تعادل نش پیشنهاد شده به وسیله سیستم توزیع شده شیء‌گرا بر اساس نوع وظیفه نشان داده شده است. فرض بر این است که اجرای کل وظیفه به یک واحد زمانی نیاز دارد.

هزینه‌های پیشنهاد شده توسط تعادل نش یکسان در شکل ۳ نشان داده شده است که در آن محور عمودی هزینه‌ای است که یک گره برای اجرای یک وظیفه باید پرداخت کند. سطح سبز نشان دهند زمان تلف شده است زمانی که گره وظیفه را خودش به تنهایی انجام می‌دهد و سطح آبی نشان‌دهنده زمان تلف شده برای انجام وظیفه در حالتی است که گره با سیستم توزیع شده مشارکت کرده و وظیفه در سیستم توزیع شده شیء‌گرا اجرا می‌شود. در واقع این سطح به مقداری که یک وظیفه می‌تواند موازی‌سازی شود و مقدار سهم c که سیستم توزیع شده از گره درخواست می‌کند، بستگی دارد.

جدول ۲. مقادیر مختلف تعادل نش پیشنهادشده توسط سیستم توزیع شده شیء‌گرا

هزینه پیشنهادی	بخش موازی	بخش سریال	نوع وظیفه
۰/۱۰۹	۰/۹	۰/۱	σ_0
۰/۵۰۵	۰/۵	۰/۵	σ_1
۰/۹۰۱	۰/۱	۰/۹	σ_2



شکل۳. هزینه گره‌ها وابسته به مقادیر c و k (محور x نشان‌دهنده a، محور y نشان‌دهنده cost و محور z نشان‌دهنده C است).

۱۰) نتیجه‌گیری و پیشنهادها

کاهش زمان اجرای عملیات در هر سیستمی منجر به بالا رفتن کارایی آن می‌شود و هدف اصلی سیستم‌های کامپیوتری نیز پردازش و اجرای درخواست کاربران است تا پاسخ‌های خود را در کمترین زمان ممکن دریافت کنند. یکی از مهم‌ترین انواع سیستم‌های کامپیوتری، سیستم‌های توزیع شده می‌باشند که امروزه با توجه به فراگیر شدنشان با چالش‌هایی مواجه هستند. برای فائق آمدن بر این چالش‌ها و بهره‌برداری از کلیه ویژگی‌های سیستم توزیع شده می‌توان آن را به صورت شئ‌گرا پیاده‌سازی کرد که ماشین مشتری با ماشین خدمتگزار به عنوان یک شئ برخورد کرده و ارتباط بین آن‌ها تنها از طریق پراکسی صورت می‌گیرد که منجر به افزایش کارایی آن‌ها می‌گردد. مهم‌ترین چالش پیشروی سیستم‌های توزیع شده شئ‌گرا نحوه تعامل گره‌ها است. در این نوع سیستم‌ها گره‌های مختلف وظایف مختلفی را بر عهده دارند، سیستم توزیع شده تمایل دارد که گره‌ها هنگام انجام وظایف با یکدیگر مشارکت داشته باشند زیرا این عمل باعث افزایش کارایی سیستم می‌شود، اما گره‌ها نیز تمایل به انجام خودمختارانه وظایف دارند. یکی از مسائلی که می‌توان آن را به صورت بازی مدلسازی کرد نحوه برخورد گره‌ها با یکدیگر در یک سیستم توزیع شده شئ‌گرا است که جزء بازی‌های همکارانه به حساب می‌آید.

در این مقاله رفتار گره‌ها در سیستم توزیع شده با معرفی بازی‌های ایستا تک‌مرحله‌ای مورد تجزیه و تحلیل قرار گرفته است که در آن بازیکنان، گره‌ها در سیستم‌های توزیع شده شئ‌گرا هستند. در این نوع بازی‌ها نشان داده شد که تعادل نش بازی به بازیکنان پیشنهاد می‌دهد که باهم همکاری نداشته باشند. در ادامه بیان شد که اگر بازی بی‌نهایت بار تکرار شود و احتمالی برای مرحله بعدی وجود داشته باشد، کاربران می‌توانند با هم همکاری داشته باشند. همچنین از یک رویکرد متفاوت که در آن بازیکنان گره و سیستم توزیع شده است نیز در این مقاله استفاده شده است و نشان داده شد که برای یک گره بهتر است که با شبکه توزیع شده برای انجام وظایف همکاری کند. به علاوه، نشان داده شد که سیستم توزیع شده چه مقدار باید به گره پیشنهاد دهد تا آن را وادار به همکاری کند و آن مقدار به نوع وظیفه بستگی دارد.

در راستای مطالعات انجام‌شده در این تحقیق، می‌توان مدل معرفی شده را برای n بازیکن همگن توسعه داد (به بیان دیگر، شبکه توزیع شده با n گره را مورد بررسی قرار داد). همچنین در این مقاله تنها عملکرد سیستم‌های توزیع شده شئ‌گرا مورد بررسی قرار گرفته است که می‌توان سایر ویژگی‌های سیستم توزیع شده شئ‌گرا را مانند نرخ ایجاد وظیفه بین گره‌ها و کارایی محیط شبکه نیز مورد بررسی قرار داد.

منابع

Abraham, I., L. Alvisi and J. Y. Halpern (2011). "Distributed computing meets game theory: combining insights from two fields." Acm Sigact News 42(2): 69-76. DOI:10.1145/1998037.1998055

Abraham, I., D. Dolev and J. Y. Halpern (2019). "Distributed Protocols for Leader Election: A Game-Theoretic Perspective." ACM Transactions on Economics and Computation (TEAC) 7(1): 4. DOI:10.1007/s11227-021-03803-7

Asadi, M., A. Agah and C. Zimmerman (2017). "Applying Game Theory in Securing Wireless Sensor Networks by Minimizing Battery Usage." Game Theory: Breakthroughs in Research and Practice: Breakthroughs in Research and Practice: 337. DOI:10.1007/s11227-021-03803-7

Asadi, M., A. Agah and C. Zimmerman (2018). Applying Game Theory in Securing Wireless Sensor Networks by Minimizing Battery Usage. Game Theory: Breakthroughs in Research and Practice, IGI Global: 337-352. DOI: 10.4018/978-1-4666-4789-3.ch004

Colman, A. M. (2016). Game theory and experimental games: The study of strategic interaction, Elsevier. DOI:10.1016/j.automatica.2018.10.041

Deng, Z. and S. Liang (2019). "Distributed algorithms for aggregative games of multiple heterogeneous Euler–Lagrange systems." Automatica 99: 246-252. DOI:10.1016/j.automatica.2018.10.041

Ghosh, P., N. Roy, S. K. Das and K. Basu (2005). "A pricing strategy for job allocation in mobile grids using a non-cooperative bargaining theory framework." Journal of Parallel and Distributed Computing 65(11): 1366-1383. DOI:10.1016/j.jpdc.2005.05.013

Gupta, R. and A. K. Somani (2005). Game theory as a tool to strategize as well as predict nodes' behavior in peer-to-peer networks. Parallel and Distributed Systems, 2005. Proceedings. 11th International Conference on, IEEE. DOI:10.1109/ICPADS.2005.157

Khan, S. U. and I. Ahmad (2006). Non-cooperative, semi-cooperative, and cooperative games-based grid resource allocation. Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International, IEEE. DOI:10.1109/IPDPS.2006.1639358

Kwok, Y.-K., S. Song and K. Hwang (2005). Selfish grid computing: game-theoretic modeling and NAS performance results. Cluster Computing and the Grid, 2005. CCGrid 2005. IEEE International Symposium on, IEEE. DOI:10.1109/IPDPS.2006.1639358

Looney, J. and T. Schlossnagle (2017). "Distributed Systems Reasoning." doi:10.1111/ecin.13178

Mailath, G. J. (2019). "Modeling Strategic Behavior: A Graduate Introduction to Game Theory and Mechanism Design." World Scientific Books. doi:10.1016/j.arcontrol.2022.02.002

Marden, J. R. and J. S. Shamma (2018). "Game-theoretic learning in distributed control." Handbook of Dynamic Game Theory: 511-546. DOI:10.1109/TVT.2019.2902318

Messous, M. A., S.-M. Senouci, H. Sedjelmaci and S. Cherkaoui (2019). "A Game Theory Based Efficient Computation Offloading in an UAV Network." IEEE Transactions on Vehicular Technology. DOI:10.1109/TVT.2019.2902318

Mishkovski, I., S. Filiposka, D. Trajanov, A. Grnarov and L. Kocarev "Using Game Theory to Analyze Distributed Computing Systems." DOI:10.1109/TVT.2019.2902318

Tanenbaum, A. S. and M. Van Steen (2007). Distributed systems: principles and paradigms, Prentice-Hall. DOI:10.1109/TVT.2019.2902318

Valone, T. J., Z. Barta, J. Börner, J.-C. Cardenas, L.-A. Giraldeau, H. Kokko, J. A. Oldekop, D. Pauly, D. Rustagi and W. J. Sutherland (2017). "Insights from Game Theory." Investors and Exploiters in Ecology and Economics: Principles and Applications: 97. DOI:10.1109/TVT.2019.2902318

Webster, T. J. (2018). Introduction to game theory in business and economics, Routledge. DOI:10.1109/TVT.2019.2902318