

بکارگیری سیاست کمیسیون چندعامله در تخصیص ایستای منابع گرید با استفاده از عاملهای CBR

سعید صبامنیری^۱؛ بهروز معصومی^۲؛ محمدرضا میبدی^۳

چکیده

در این مقاله، یک سیاست کمیسیون چندعامله برای انتساب ایستای وظایف در سیستمهای محاسباتی ناهمگن گرید، ارائه گردیده است. از آنجائیکه سیاست مورد نظر از عاملهای CBR استفاده می‌کند، لذا در مدل بکارگرفته شده در این مقاله هر یک از عاملها از الگوریتم CBR-LA برای شرکت در همکاری استفاده می‌کنند. در این روش هر عامل از تجارب گذشته خود در حل مساله استفاده می‌کند. هدف این مقاله ارائه روشی جدید بر اساس معماری چندعامله و سیاستهای همکاری عاملها برای بهبود عملکرد روش اصلی می‌باشد. به منظور ارزیابی کارایی مدل پیشنهادی از شبیه‌سازی کامپیوتری و مقایسه‌ی نتایج بدست آمده از اجرای این الگوریتم با الگوریتم اصلی استفاده شده است که نتایج بدست آمده کارایی بهتری را برای مدل پیشنهادی نشان می‌دهد.

کلمات کلیدی

همکاری چندعامله، استنتاج مبتنی بر مورد، اتوماتای یادگیر، زمانبندی وظایف.

Applying Multi-Agent Committee Policy in Grid Static Resource Assignment Using CBR Agents

Saeed Saba Moniri; Behrouz Masoumi; Mohammad Reza Meybodi

ABSTRACT

In this paper a multi-agent committee policy is presented to static task assignment in heterogeneous computing systems. Since the policy uses CBR agents, so in the model applied in this paper each agent uses CBR-LA algorithm to have cooperation with the other agents. In this approach each agent uses its experiences to solve problems. Presenting a new approach based on the multi-agent architecture is the goal of this paper. In order to evaluating the recommended model's performance the computerized simulation had used together with comparing gained results from executing this algorithm with the base algorithm. The results illustrated that the new model outperforms the base algorithm.

KEYWORDS

Multi-Agent Cooperation, Case-Based Reasoning, Learning Automata, Task Scheduling.

^۱ دانشگاه آزاد اسلامی واحد صوفیان، Saba_Moniri@Hotmail.com

^۲ دانشکده مهندسی برق، کامپیوتر و فناوری اطلاعات، دانشگاه آزاد اسلامی قزوین، قزوین، bmasoumi@Qazviniau.ir

^۳ دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران، mmeybodi@aut.ac.ir

۱. مقدمه

در جهان بهم پیوسته‌ی امروزی بسیاری از صنایع، آزمایشگاهها، و مجموعه‌های نظامی، هر کدام مجموعه‌ای از کامپیوترهای بهم شبکه شده را استفاده می‌کنند که از لحاظ قدمت و نوع با هم متفاوت هستند. در چنین محیطهایی برنامه‌های کاربردی بزرگ یا مجموعه‌ای از برنامه‌های کاربردی وجود دارند که بایستی پردازش شوند. این بار کاری می‌تواند در سرتاسر این مجموعه‌ی منابع محاسباتی ناهمگن توزیع شده باشد تا بدین وسیله برخی از اهداف کارایی ماکزیمم شوند. برای انجام چنین کاری، طراح این سیستمها به منظور بهره‌برداری موثرتر از بستر ناهمگن این سیستمها، باید متدهایی را برای تعیین اینکه چه وظایفی از برنامه‌ی کاربردی به ماشینهای مختلف شبکه تخصیص داده شوند، در نظر بگیرد. گرید یک زیرساخت محاسباتی بزرگ می‌باشد که امکان به اشتراک گذاری منابع، بین نهادها و افراد را ارائه می‌کند [۲]. گریدهای محاسباتی دسترسی شفاف را برای منابع محاسباتی توزیع‌شده با مقیاس بزرگ فراهم می‌کنند و خودشان را جهت ایجاد اتحادیه‌ها برای به اشتراک گذاری و جمع‌آوری مجموعه بزرگی از منابع، قرض می‌دهند. گریدها از منابع ناهمگن، به‌همراه نیازمندیهای مدیریت منابع گوناگون تشکیل یافته‌اند [۳]. هدف از ایجاد چنین سیستمهایی اجرا نمودن برنامه‌های کاربردی قدرتمند مختلف از نظر محاسباتی است که نیازمندیهای محاسباتی متفاوتی دارند. گریدها نوعی از سیستمهای محاسباتی ناهمگن (HC) می‌باشند. در یک سیستم محاسباتی ناهمگن، منابع دارای امکانات متفاوت و وظایف دارای نیازمندیهای مختلفی هستند. در اینگونه سیستمها از ناهمگنی وظایف و منابع برای حداکثر کردن کارایی یا سودمندی سیستم استفاده می‌شود [۴ و ۵ و ۶ و ۷].

عمل انتساب مجموعه‌ای از وظایف (متاوظیفه) به ماشینها و زمانبندی ترتیب اجرای آنها نداشت نامیده می‌شود و یکی از مسائل مهم در سیستمهای HC می‌باشد. مساله‌ی عمومی نگاشت بهینه وظایف به ماشینها در یک دنباله HC، بعنوان یک مساله NP-complete شناخته شده است [۸]. متاوظیفه بعنوان مجموعه‌ای از وظایف مستقل تعریف می‌شود که هیچ وابستگی داده‌ای بین وظایف وجود ندارد [۹]. مساله نگاشت می‌تواند بطور ایستا یا پویا انجام شود. در نگاشت ایستا فرض می‌شود که هر ماشین در هر لحظه یک وظیفه واحد را به ترتیبی که وظایف تخصیص داده می‌شوند، اجرا می‌کند و هدف از آن، مینیمم کردن زمان کل اجرای متاوظیفه است. اندازه‌ی متاوظیفه، و تعداد ماشینهای موجود در دنباله HC (Heterogeneous Computing)، ایستادگی از قبل شناخته شده است.

روشهای متنوعی برای حل مساله نگاشت در منابع مختلف ارائه شده‌اند. در مقاله‌ی [۳] نویسندگان آن ۳ بازی نظری تخصیص منبع در محیط گرید را مورد بررسی، مقایسه، و تحلیل قرار داده‌اند. در این روش فرض شده است که وظایف ارسالی به محیط، مستقل بوده و مهلت زمانی اجرا را در بر دارند. در مقاله منبع [۲] نیز یک میان افزار چندعامله برای گرید ارائه شده است که امکان استفاده از تکنولوژی عاملها را بعنوان مکانیزم اصلی برای مذاکره‌ی منبع، در فرایند ارسال کار در گرید را فراهم می‌کند. البته روشهای دیگری نیز بدون استفاده از حالت چندعامله و تئوری بازیها به حل این مساله پرداخته‌اند که از آن جمله می‌توان به الگوریتمهایی اشاره کرد که بر پایه تکنیکهای تئوری گراف عمل می‌کنند، یا الگوریتم simulated-annealing [۱۰]، MET [۱۱]، GA [۱۲]، A* [۱۳]، و اتوماتای یادگیر [۱۵] ازجمله این روش ها هستند. در تمام این مدلها تخصیص وظیفه، بدون استفاده از تجربیات گذشته، راه‌حل را می‌سازند. در [۵] یک مدل ترکیبی از تکنیکهای اتوماتای یادگیر و استنتاج مبتنی بر مورد به نام CBR-LA ارائه شده است که هدف عمده و ویژگی کلیدی آن کاهش تعداد تکرار و دستیابی نسبتاً سریع به پاسخ بر اساس تجارب گذشته می‌باشد.

چنین به نظر می‌رسید که استفاده نمودن از سیاستهای همکاری آقای Plaza که در [۱۵] مطرح شده‌اند، منجر به بهبود نتایج گردد. لذا در این مقاله قصد داشتیم چهارچوبی برای فراهم نمودن امکان تبادل اطلاعات و همکاری بین عاملها در راستای سیاست کمیسیون و حل مسائل مختلف نگاشت، ایجاد کنیم و امکان بهبود پاسخها را نسبت به روش پایه‌ای CBR-LA مورد ارزیابی قرار دهیم. علت انتخاب الگوریتم CBR-LA برای این منظور، استفاده نمودن آن از مکانیزم استنتاج مبتنی بر مورد (CBR) بود. از آنجائیکه سیاستهای مطرح شده برای همکاری عاملها در منبع [۱۵] بین عاملهای دارای پایگاه موارد (CB) مد نظر قرار گرفته است، لذا برای ارزیابی دقیق کاربرد نهایی عاملهای زمانبند درگیر در سیستم مدیریت منابع HC بایستی از روشی برای تشخیص راه‌حل استفاده کنند که از این مکانیزم بهره می‌برد.

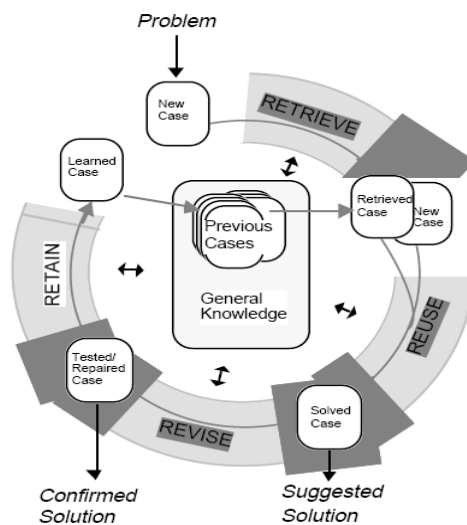
بنابراین، هدف اصلی این مقاله بکار گرفتن همکاری عاملها در یک محیط چندعامله به منظور ارائه مدلی بهبود یافته برای تخصیص ایستای وظایف می‌باشد. این روش با فراهم ساختن یک بستر همکاری برای عاملهای درگیر در حل مساله و نیز امکان استفاده عاملها از تجارب مشابه گذشته، عملکرد الگوریتم CBR-LA را بهبود می‌دهد. جهت نمایش میزان این بهبود، آزمایشهایی انجام گرفته اند که نشان می‌دهند مدل همکاری پیشنهادی (CoopMA-CBRLA)، پاسخهای تولیدی الگوریتم مذکور را تا حد قابل توجهی بهبود می‌دهد. در ادامه مقاله، در بخش ۲ مقدمه‌ای در مورد CBR و کاربردهای آن ارائه شده است. در بخش ۳ به معرفی استنتاج مبتنی بر مورد همکارانه (Coop-CBR) پرداخته شده است. در بخش ۴ تعریفی از مساله نگاشت وظایف انجام شده و بدنبال آن در بخش ۵ مدل همکاری چندعامله پیشنهادی ارائه گردیده و در بخش

۶ محیط شبیه‌سازی و نتایج آزمایشهای انجام شده در جهت نشان دادن کاهش معیار کارایی Makespan، ارائه می‌گردد. در نهایت بخش ۷ نتیجه‌گیری نهایی از کارهای انجام شده را ارائه می‌دهد.

۲. استنتاج مبتنی بر مورد و کاربردهای آن

تکنیک استنتاج مبتنی بر مورد (CBR) یک روش حل مساله مبتنی بر دانش است که بر پایه استفاده مجدد از تجربیات پیشین عمل می‌کند و از تحقیقات علوم شناختی^۱ پدیدار شده است [۱۶ و ۱۷]. در این روش فرض بر این است مسائل مشابه می‌توانند راه‌حلهای مشابهی داشته باشند. بنابراین، ممکن است مسائل جدید با روشهای تجربه شده در مسائل قبلی قابل حل باشند. برخلاف تکنیکهای سنتی مبتنی بر دانش، CBR بر روی تجربه حل مساله خاصی تمرکز می‌کند که برگرفته از موارد جمع‌آوری شده در پایگاه مورد^{۱۱} است. این موارد، تجربه خاصی را در یک دامنه حل مساله نشان می‌دهند. فعالیتهای اصلی در حین حل مسائل با این روش، در چرخه CBR، تشریح می‌شود که در شکل ۱ نشان داده شده است [۱۶ و ۱۷]. باید توجه داشت که CBR یک راه‌حل قطعی را پیشنهاد نمی‌کند بلکه فرضیات و نظراتی را برای عبور از فضای راه‌حل، ارائه می‌کند. برای مطالعه بیشتر می‌توان به منابع [۱۶ و ۱۷] مراجعه نمود.

تکنیک CBR در کاربردهایی مفید است که درک کاملی از فضای پاسخ مساله وجود نداشته باشد و نیز امکان تکراری بودن مسائل و مشابه بودن آنها وجود داشته باشد، و از طرفی نیز از لحاظ برخی معیارها همانند زمان، حل یک مساله اهمیت داشته باشد [۱۸ و ۱۹]. در سالهای اخیر، CBR قابلیتهای بالایی را در حوزه‌های مختلفی نظیر تصمیم‌گیری، پیشگویی، تشخیص عیب، برنامه‌ریزی، کنترل کیفیت/فرایند، پشتیبان تصمیم، محاسبات توزیع‌شده [۱۹] و بازیابی اطلاعات نشان داده است. با توجه به ویژگیهای مذکور مساله‌ی تخصیص وظایف می‌تواند بعنوان یکی از کاربردهایی مطرح باشد که امکان سود جستن از CBR برای آن امکانپذیر می‌باشد. در منبع [۵] این تکنیک برای حل مسائل در این کاربرد، مورد استفاده قرار گرفته و پس از انجام آزمایشهای مختلف و مقایسه با روشهای موجود چنین نتیجه‌گیری شده است که بکار بردن CBR و استفاده از تجارب قبلی، در کاهش تعداد تکرارهای مورد نیاز الگوریتم و بهبود نتایج بدست آمده از اجرای الگوریتمها موثر بوده و از لحاظ زمانی مقرون به صرفه است. در ادامه مقدمه و تاریخچه‌ای در مورد CBR همکارانه ارائه شده و سپس به معرفی مساله نگاشت وظایف و مدل همکاری پیشنهادی پرداخته می‌شود.



شکل (۱) چرخه‌ی استنتاج مبتنی بر مورد [۱۷ و ۱۹]

۳. استنتاج مبتنی بر مورد همکارانه (Coop-CBR)

استنتاج مبتنی بر مورد (CBR) و سیستمهای چندعامله (MAS) دو الگوی متفاوت در هوش مصنوعی هستند. CBR یک الگوی استنتاج است که بر اساس تجربیات موجود در پایگاه‌موارد یا موارد مشابه موجود، عمل استنتاج را انجام می‌دهد اما MAS یک الگوی جدید جهت سازماندهی برنامه‌های کاربردی هوش مصنوعی می‌باشد. به‌رحال یکپارچه‌سازی CBR و MASها در طی سالهای اخیر مورد توجه انجمن هوش مصنوعی واقع شده

است زیرا CBR به الگوهای MAS، توانایی یادگیری خودمختار از تجربیاتشان را می‌دهد. یعنی می‌توان گفت، دو ویژگی که همواره همراه عاملهای بالغ و کامل است، (خودمختاری و یادگیری) لذا CBR به الگوی سیستمهای چندعامله توانایی یادگیری خودمختار از تجارب را ارائه می‌کند [۲۰]. در مقایسه با سیستمهای استنتاج مبتنی بر مورد تک عامله، سیستمهای چندعامله مشترکاً یک توزیع از پایگاه موارد و/یا برخی جنبه‌های چرخه‌ی حیات CBR (بازیابی، استفاده‌ی مجدد، اصلاح، نگهداری) بین یک یا چند عامل را دارا می‌باشند [۲۱]. آقای Plaza در سال ۱۹۹۷ در منبع [۲۲] روشی را با عنوان یادگیری متناظر مشارکتی را مطرح کرده‌اند که هدف از این پروژه انجام مطالعاتی در رابطه با همکاری بین عاملها بود که در این روش، عاملها قابلیت یادگیری و حل مسائل را با استفاده از استنتاج مبتنی بر مورد دارا می‌باشند. با توجه به اینکه عاملها با هم همکاری می‌کنند و دارای پایگاه موارد هستند، لذا عنوان این روش "استنتاج مبتنی بر مورد همکاری" یا CBR Coop نامگذاری شده است.

اگر عامل CBR دارای تجارب خوبی در مورد مساله جدید باشد، روش CBR به تنهایی خوب عمل می‌کند ولی از آنجائیکه فضای مساله، بسته به دامنه‌ی مساله می‌تواند بطور دلخواه پیچیده باشد، اغلب چنین نیست. این موضوع یکی از دلایلی است که تلاشهایی در جهت ایجاد سیستمهایی با عاملهای CBR صورت گرفته است بطوریکه این عاملها می‌توانند با یکدیگر همکاری کنند و بنابراین مسائل را به شکلی توزیع شده حل کنند. این سیستمهای توزیع شده بایستی دارای چندین ویژگی دلخواه باشند، برای مثال معمولاً چندین عامل نیازمند CBهای جداگانه‌ای هستند. از آنجائیکه یک عامل از نمونه‌های زیادی برای تطبیق دادن با مساله‌ی جاری استفاده می‌کند، این موضوع شانس یک عامل را در یافتن مساله‌ای که همتای مساله‌ی جاری باشد، بالا می‌برد و نیز کارهای محاسباتی می‌توانند در بین چندین عامل توزیع شوند [۲۳].

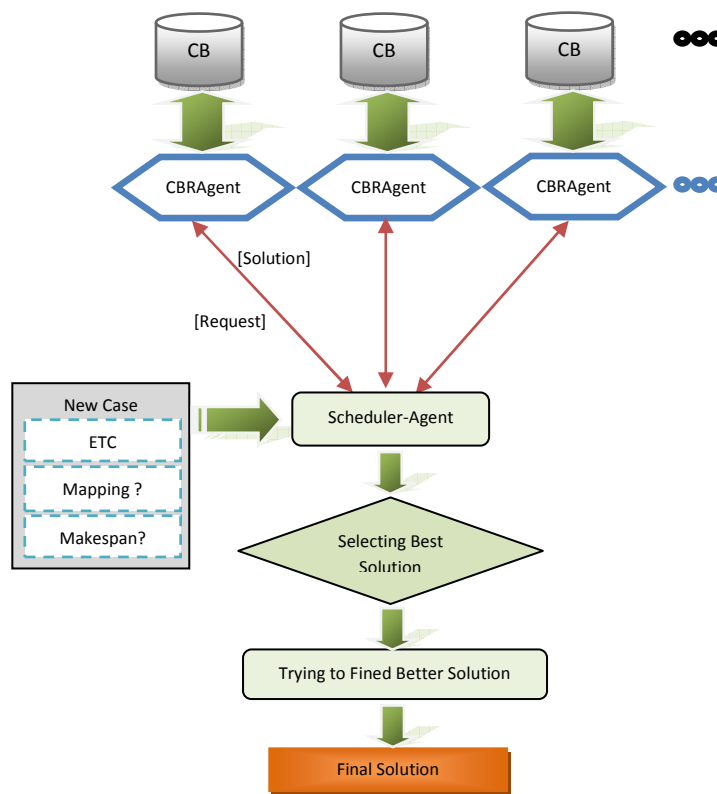
Plaza و McGinty در منبع [۲۴] دو نوع از سیستمهای CBR توزیع شده را تعریف می‌کنند، که عبارتند از: "سیستمهای چندعامله با یک پایگاه موارد" و "سیستمهای چندعامله با چندین پایگاه موارد". دسته‌ی اول، از یک پایگاه موارد برای تمام عاملها استفاده می‌کند و بدین صورت تعریف می‌شود: "سیستمی توزیع شده از عاملهای همکار بطوریکه هر عامل به یک (یا چند) منبع تجربیات گذشته متصل است تا مسائل جدید را حل کند". از طرف دیگر، سیستمهای چندعامله با چندین پایگاه موارد (دسته‌ی دوم) بیشتر شامل عاملهای خودمختاری هستند که از یک چارچوب همکاری برای بدست آوردن راه حل استفاده می‌کنند.

۴. ویژگیهای مسائل نگاشت ورودی

مساله تخصیص وظیفه‌رانی می‌توان بصورت نگاشت $\pi: S \rightarrow M$ از مجموعه S (وظایف) به مجموعه M (ماشینها) در نظر گرفت، بطوریکه شاخص هزینه-ی C بهبود یابد. C نشاندهنده شاخص هزینه می‌باشد و منظور از آن، مدت زمانی است که برای تکمیل اجرای یک متاوظیفه مورد نیاز است و این شاخص Makespan نامیده می‌شود. در ضمن، در هر تکرار الگوریتم، یک بردار به طول تعداد وظایف، نمونه‌ای از نگاشت π را نشان می‌دهد. هدف سیستم زمانبندی، یافتن نگاشتی از وظایف موجود در متاوظیفه به پردازنده‌های موجود در HC است، بطوریکه شاخص هزینه تعریف شده بهینه شود. مسائل جدید ورودی به سیستم زمانبند، شامل متاوظیفه و زمانهای تخمینی اجرای آنها بر روی هر یک از ماشینهای محیط HC می‌باشند. این زمانهای تخمینی ممکن است با استفاده از تکنیکهای محک‌زنی بدست آمده باشند [۲۵]. تخمین زمان اجرای مورد انتظار روی هر ماشین به ازای هر وظیفه، قبل از اجرا مشخص بوده و در ماتریسی به ابعاد $\begin{matrix} T & \times & M \\ ETC & & \end{matrix}$ بنام ETC^{iii} نگهداری می‌شود. ممکن است چنین فرض شود که مدخل $ETC(ti, mj)$ ، زمان مورد نیاز جهت انتقال فایل‌های قابل اجرا و داده مربوط به وظیفه ti از ماشین مبدا mj را در بر دارد [۵].

۵. مدل پیشنهادی برای همکاری عاملهای CBR

ما در این مقاله از مدل "سیستمهای چندعامله با چندین پایگاه موارد" برای همکاری بین عاملها استفاده می‌کنیم که هر عامل به یک CB مجهز بوده و تجارب متفاوتی را در مقایسه با دیگر عاملها دارا می‌باشد بدین معنی که روشهای مختلفی برای مقداردهی اولیه پایگاههای موارد عاملها بکار گرفته شده است. در این مدل هر یک از عاملها از متد یکسانی برای بازیابی و تطبیق راه-حل استفاده می‌کنند. این مدل در شکل ۲ نشان داده شده است. در این مدل از عملیات رای‌گیری به منظور همکاری عاملها با عامل زمانبند استفاده شده است که این سیاست همکاری در [۱۵] با عنوان سیاست کمیسیون (Committee Policy) توسط Plaza پیشنهاد شده است.



شکل (۲) مدل همکاری چندعامله پیشنهادی

در این مدل فرض شده است که وظایف متاوظیفه موجود در مورد جدید مستقل بوده و به تبادل داده و برقراری ارتباط نیازی ندارند. درضمن، زمان اجرای مورد انتظار وظایف مربوط به مساله جاری، در هر ماشین مشخص است و در قالب ماتریس ETC ارائه می‌گردد.

۱.۵. الگوریتم CBR-LA عاملها

این الگوریتم توسط آقای قنبری و همکارانش در سال ۲۰۰۴ ارائه شده است [۵]. در این الگوریتم از ترکیب استنتاج مبتنی بر مورد و اتوماتای یادگیر با ساختار متغیر استفاده شده است. استنتاج مبتنی بر مورد امکان حفظ و نگهداری تجارب مشابه قبلی و اتوماتای یادگیر ساختار متغیر امکان تطبیق موارد مشابه تجربه شده را به مساله جدیدالورود، فراهم می‌کند.

عملکرد این الگوریتم بدین صورت می‌باشد که ابتدا موارد مشابه به مساله جدید از پایگاه موارد بازیابی می‌شوند (تشخیص تشابه موارد با استفاده از مقدار فاصله اقلیدسی ETC‌های مسائل انجام می‌شود) و سپس راه‌حل موجود در آنها بوسیله یک مدل اتوماتای یادگیر به مساله‌ی جدید تطبیق داده می‌شود. در این گام به تعداد وظایف موجود در یک متاوظیفه (مجموعه‌ای از وظایف مستقل که فرایند تخصیص آنها به پردازنده‌های HC در قالب مساله‌ای جدید مطرح هستند) اتوماتای یادگیر تولید شده (اتوماتاهای بایاس شده) و در جهت بهینه کردن هرچه بیشتر راه‌حل نهایی اقدامهای مختلفی بر حسب تعداد پردازنده‌های موجود در محیط HC انجام می‌دهند و طی تکرارهای متوالی الگوریتم یادگیری، اقدامهای انجام شده خود را مورد بررسی قرار داده و بر حسب این بررسی‌ها بردارهای احتمال اقدامهای خود را تغییر می‌دهند. عملیات جستجوی فضای پاسخ توسط اتوماتاها تا زمانی ادامه می‌یابد که تعداد تکرار الگوریتم از مرز ۱۰۰۰۰ تکرار گذشته و یا طی ۱۵۰ تکرار متوالی پاسخ یکسانی تولید گردد.

در انتهای این مرحله به تعداد موارد بازیابی شده، راه‌حل‌های مختلفی (نگاشتهای مختلف) برای مساله‌ی جدید وجود دارد که از بین آنها راه‌حلی که دارای معیار کارایی مناسبی (Makespan کمتر) باشد، بعنوان راه‌حل نهایی انتخاب می‌شود. اگر راه‌حل نهایی با مورد اصلی‌اش تفاوت عمده‌ای داشته باشد، بعنوان مورد جدید در پایگاه موارد ذخیره می‌شود. شبه کد مربوط به اتوماتای یادگیر این الگوریتم در شکل ۳ و شبه کد الگوریتم کلی CBR-LA نیز در شکل ۴ نشان داده شده است [۵].

Begin

Find n closest cases to the new-case & put them in Θ ;

For each case in Θ Do

Set bias LA to the solution of the case ;

Resolve solution;

EndFor

Return the solution with minimum makespan;

End

شکل (۴) رویه کلی الگوریتم CBR-LA

While (true)

Begin

LA_i selects its action for all $1 \leq i \leq \tau$;

Evaluate the makespan ;

If Current-Makespan < ThePreviousMakespan **Then**

Set Reward for all LA_i that $1 \leq i \leq \tau$;

Else

Set Penalty for all LA_i that $1 \leq i \leq \tau$;

EndIf

If (no change occurs in makespan for ۱۰۰ consecutive iteration) **OR** (۱۰۰۰ iteration is over) **Then**

Exit ;

EndIf

End

شکل (۳) رویه کلی مدل اتوماتای یادگیر در CBR-LA

۶. محیط شبیه‌سازی و نتایج آزمایشها

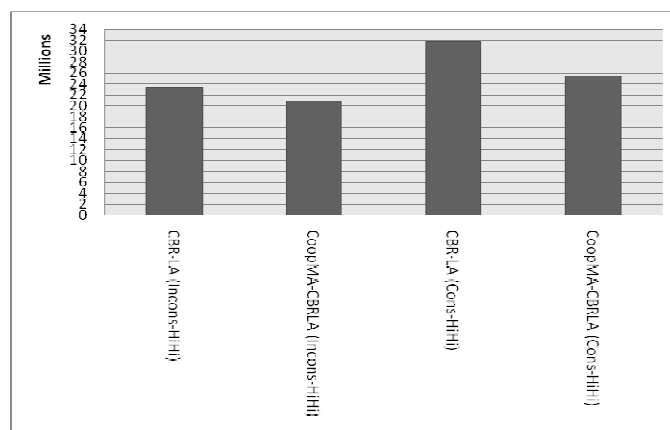
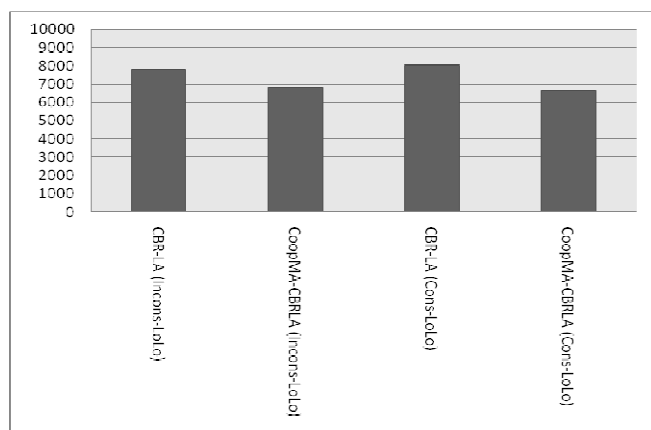
در این بخش، مجموعه‌ای از متاوظیفه‌ها در مقابل روش CBR-LA و تاثیر سیاست همکاری چندعامله در کاهش مقدار کارایی Makespan، نشان داده شده است. لازم به ذکر است که برای ایجاد ماتریس ETC از روش Baseline مورد استفاده در [۵] استفاده شده است. درضمن، هشت حالت مختلف برای ویژگیهای ماتریس ETC در این مطالعه بکار می‌رود: ناهمگنی وظیفه بالا یا پائین، ناهمگنی ماشین بالا یا پائین، و یک نوع از سازگاری (سازگار، ناسازگار).

در این مطالعه ماتریسهای ETC را با ابعاد گوناگون در نظر می‌گیریم بدین معنی که تعداد وظایف و تعداد ماشینهای مسائل ورودی را متغیر فرض می‌کنیم. در تولید ماتریسهای ETC، تعداد وظایف بین ۱۰۰ تا ۱۵۰ و تعداد ماشینها بین ۴ تا ۸ ماشین متغیر در نظر گرفته شده است تا بدین طریق تنوع بیشتری را در بین مسائل ایجاد کنیم. در ضمن قبل از آغاز آزمایشها، پایگاه موارد عاملها با ۱۰۰ مورد و با استفاده از روشهای Max-Min و GA مقادردهی اولیه شده‌اند و نیز تعداد موارد مشابه انتخابی از پایگاه موارد، ۳ مورد می‌باشد. البته لازم به ذکر است که نتیجه هر آزمایش به ازای هر نوع ناهمگنی، میانگین ۱۰ بار اجرا می‌باشد.

در آزمایشهای انجام شده، پاسخهای تولیدی الگوریتم پایه (CBR-LA) و الگوریتم پیشنهادی (CoopMA-CBRLA) را در ازای ارائه مسائل یکسان، مورد مقایسه قرار داده و تاثیر الگوریتم همکاری چندعامله پیشنهاد شده بر کاهش مقدار کارایی Makespan را مورد بررسی قرار می‌دهیم. نتایج این آزمایشها در جدول ۱ فهرست شده است. همانطوریکه در جدول ۱ مشخص است، نتایج بدست آمده از روش CBR-LA حداکثر در محیطهای سازگار با ناهمگنی بالای وظیفه و ماشین ۲۰/۲۴ درصد بهبود یافته‌اند که این موضوع بهبود قابل توجهی را در مورد معیار کارایی Makespan نشان می‌دهد. مقایسه‌ای از نتایج اجرای الگوریتمهای CBR-LA و الگوریتم پیشنهادی (CoopMA-CBRLA) در شکل‌های ۵ و ۶ و ۷ نشان داده شده‌اند.

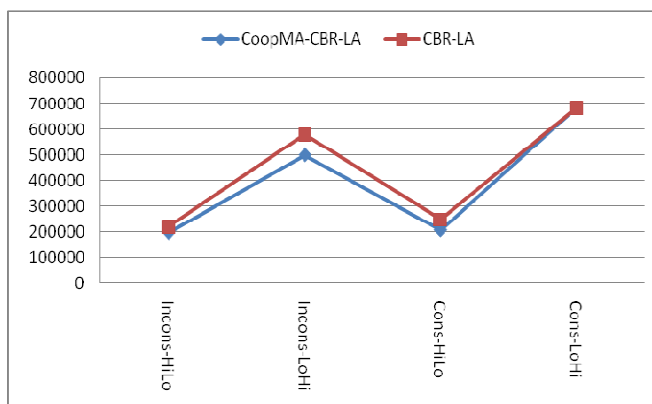
جدول (۱) نتایج بدست آمده از اجرای الگوریتمهای CBR-LA و CoopMA-CBRLA برای مسائل یکسان

Consistency	Task Heterogeneity	Machine Heterogeneity	CBR-LA	CoopMA-CBRLA	Reduction in Makespan (in terms of percentage)
Incons.	High	High	۲۳۲۹۹۲۲۲	۲۰۸۵۱۱۷۲	۱۰.۵۰
	High	Low	۲۱۸۹۳۰	۱۹۵۵۶۵	۱۰.۶۷
	Low	High	۵۷۹۴۸۶.۵	۴۹۹۶۸۹.۵	۱۳.۷۷
	Low	Low	۷۸۰۲.۵	۶۸۲۳	۱۲.۵۵
Cons.	High	High	۳۱۷۸۷۳۲۸	۲۵۳۵۲۸۹۷	۲۰.۲۴
	High	Low	۲۴۷۸۰۷.۵	۲۰۷۸۵۰	۱۶.۱۲
	Low	High	۶۸۳۰۰.۴	۶۸۲۷۱۶	۰.۰۴
	Low	Low	۸۰۵۸.۵	۶۶۴۳	۱۷.۵۶۵۳



شکل (۶) مقایسه نتایج الگوریتم پیشنهادی به ازای مسائل با ناهمگنی پایین (LoLo) در محیطهای سازگار و ناسازگار

شکل (۵) مقایسه نتایج الگوریتم پیشنهادی و الگوریتم CBR-LA به ازای مسائل کاملاً ناهمگن (HiHi) در محیطهای سازگار و ناسازگار



شکل (۷) مقایسه پاسخهای روش پیشنهادی با پاسخهای روش CBR-LA به ازای مسائل با ناهمگنی HiLo و LoHi (در حالت سازگار و ناسازگار)

۷. نتیجه

در این مقاله یک مدل جدید همکاری چندعامله شامل عاملهای CBR، ارائه شد که با استفاده از سیاست همکاری کمیسیون (Committee Policy) در جهت تخصیص وظایف در سیستم محاسباتی ناهمگن گرید با یکدیگر همکاری کرده و راهحلی نزدیک به بهینه را برای مسائل ورودی ارائه می‌کنند. مسائل ورودی از لحاظ ابعاد (تعداد وظایف و تعداد پردازنده‌های محیط HC) متفاوت در نظر گرفته شدند تا طیف متنوعی از مسائل در آزمایشها مورد بررسی قرار گیرند. به جهت نوع همکاری، الگوریتم استفاده شده در هر عامل برای حل مساله CBR-LA می‌باشد. هر عامل به یک پایگاه موارد مجهز است که امکان بررسی تجارب قبلی را برای عامل فراهم می‌نماید. برای ارزیابی کارایی روش پیشنهادی، نتایج حاصل از اجرای این روش با الگوریتم CBR-LA مورد مقایسه قرار گرفته است. با استفاده از مدل پیشنهادی، آزمایشها کاهش قابل توجهی را در مقادیر Makespan در مقایسه با روش CBR-LA نشان دادند که این موضوع، اثربخشی قابل توجهی در تولید پاسخهای نزدیک به بهینه است.

۸- مراجع

- [۱] S. K. Pal, S. C. K. Shiu, *Foundations of Soft Case-Based Reasoning*, Wiley Series on Intelligent Systems, A John Wiley & Sons INC. Publication, ۲۰۰۴.
- [۲] L. N. Nassif, M. Ahmed, J. M. S. Nogueira, R. Impey, "Negotiation Process for Resource Allocation in Grid Using a Multi-agent System", In *Proceedings of MATA*, pp. ۱۵۸-۱۶۷, ۲۰۰۴.
- [۳] S. Ullah Khan, I. Ahmad, "Non-Cooperative, Semi-Cooperative, and Cooperative Games-Based Grid Resource Allocation", In *Proceedings of IPDPS*, ۲۰۰۶.
- [۴] S. Ali, T. D. Braun, H. J. Siegel, A. A. Maciejewski, N. Beck, L. B'ol'oni, M. Maheswaran, A. I. Reuther, J. P. Robertson, M. D. Theys, B. Yao, "Characterizing Resource Allocation Heuristics for Heterogeneous Computing Systems", *Advances In Computers*, Vol. ۶۳, DOI ۱۰.۱۰۱۶/S۰۰۶۵-۲۴۵۸(۰۴)۶۳۰۰۳-۸, Elsevier Inc., ۲۰۰۵.
- [۵] S. Ghanbari, M. R. Meybodi, K. Badie, "A Case-Based Recommender for Task Assignment in Heterogeneous Computing Systems", *Proceedings of the Fourth IEEE International Conference on Hybrid Intelligent Systems*, pp. ۱۱۰-۱۱۵, ۲۰۰۴.
- [۶] R. M. Chen, Y. M. Huang, "Multiconstraint Task Scheduling in Multi-Processor Systems by Neural Networks", *Proceedings of the ۱۰th IEEE Conference on Tools with Artificial Intelligence*, pp. ۲۸۸-۲۹۴, ۱۹۹۸.
- [۷] J.K. Kima, S. Shibleb, H. J. Siegelb, A. A. Maciejewskib, T. D. Braunb, M. Schneiderb, S. Tidemanc, R. Chittac, R. B. Dilmaghanib, R. Joshib, A. Kaulb, A. Sharnab, S. Sripadab, P. Vangarib, S. S. Yellampallie, "Dynamically Mapping Tasks With Priorities and Multiple Deadlines in a Heterogeneous Environment", *Journal of Parallel Distributed Computing*, pp. ۶۷۱۵۴ - ۱۶۹, ۲۰۰۷.
- [۸] D. Fernandez-Baca, "Allocating Modules to Processors in a Distributed System," *IEEE Transaction on Software Engineering*, Vol. ۱۵, pp. ۱۴۲۷-۱۴۳۶, ۱۹۸۹.
- [۹] T. D. Braun, H. J. Siegel, N. Beck, "A Comparison of Eleven Static Heuristics for Mapping a Class of Independent Tasks onto Heterogeneous Distributed Computing Systems," *Journal of Parallel and Distributed Computing*, Vol. ۶۱, pp. ۸۱۰-۸۳۷, ۲۰۰۱.
- [۱۰] M. Coli, P. Palazzari, "Real Time Pipelined System Design through Simulated Annealing", *Journal of Systems Architecture*, Vol. ۴۲, pp. ۴۶۵-۴۷۵, ۱۹۹۶.
- [۱۱] R. Armstrong, D. Hensgen, T. Kidd, "The Relative Performance of Various Mapping Algorithms is independent of sizable variances in run-time predictions", *7th IEEE Heterogeneous Computing Workshop*, pp. ۷۹-۸۷, ۱۹۹۸.
- [۱۲] L. Wang, H. J. Siegel, V. P. Roychowdhury, A. A. Maciejewski, "Task Matching and Scheduling in Heterogeneous Computing Environments Using a Genetic-Algorithm-based Approach", *Journal of Parallel Distributed Computing*, Vol. ۴۷, pp. ۱-۱۵, ۱۹۹۷.
- [۱۳] K. Chow, B. Liu, "On Mapping Signal Processing Algorithms to a Heterogeneous Multiprocessor System", *International Conference on Acoustics, Speech, and Signal Processing*, Vol. ۳, pp. ۱۵۸۵-۱۵۸۸, ۱۹۹۱.
- [۱۴] R. D. Venkataramana, N. Ranganathan, "Multiple Cost Optimization for Task Assignment in Heterogeneous Computing Systems Using Learning Automata", *IEEE ۸th Heterogeneous Computing Workshop*, pp. ۱۳۷, ۱۹۹۹.
- [۱۵] E. Plaza, S. Ontañón, "Ensemble Case-Based Reasoning: Collaboration Policies for Multiagent Cooperative CBR", In *ICCBR*, pp. ۴۳۷-۴۵۱, ۲۰۰۱.
- [۱۶] R. Bergman, "Engineering Applications of Case-Based Reasoning", *Journal of Engineering Applications of Artificial Intelligence*, Vol. ۱۲, pp. ۸۰۵, ۱۹۹۹.

- [17] A. Aamodt, E. Plaza, "*Case-Based Reasoning: Foundational Issues*", Methodological Variations and System Approaches AI Communications, IOS Press, Vol. 7: 1, p. 39-59, 1994.
- [18] J. L. Kolodner, D. B. Leake, "A Tutorial Introduction to Case-Based Reasoning, in *Case-Based Reasoning Experiences*", Lessons and Future Directions, MIT Press, pp. 31-66, 1996.
- [19] R. S. Freeman, R. DiGiorgio, "Assessing Alternative Technologies for the Cost-Effective Computation of Derivatives", Applied Artificial Intelligence, pp.491-503, 1997.
- [20] S. Ontañón, Enric Plaza, "*Case Exchange Strategies in Multiagent Learning*", Lecture Notes in Artificial Intelligence, Springer Verlag, Vol. 2430, pp.231-244, 2002.
- [21] O. A. Brevik, "A Survey of Multi-Agent Case-Based Reasoning", 2000.
- [22] E. Plaza, J. L. Arcos, F. Martin, "*Cooperative Case-Based Reasoning*", Published in G. Weiss (Ed.), Distributed Artificial Intelligence meets Machine Learning, Lecture Notes in Artificial Intelligence, Springer Verlag, 1997.
- [23] T. Stoyanov, "*Cooperative Distributed CBR*", Technical Report, 2007.
- [24] E. Plaza, L. McGinty. "*Distributed case-based reasoning*", Knowledge Engineering Rev., Vol. 20, No. 3, pp. 261-268, 2005.
- [25] A. A. Khokhar, V. K. Prasanna, M. E. Shaaban, C. L. Wang, "*Heterogeneous Computing: Challenges and Opportunities*", IEEE Computing, Vol.26, pp.18-27, 1993.

ⁱ Cognitive Science

ⁱⁱ Case Base

ⁱⁱⁱ Expected Time to Compute

^{iv} Cooperative Multiagent CBR-LA