

## بررسی مدل سازی های شبیه ساز کلاسیم و ارزیابی آن

ندا ملکی<sup>۱</sup>، اسماعیل زینالی<sup>۲</sup>، محمد کاظم اکبری<sup>۳</sup>، مرتضی سرگلزایی جوان<sup>۴</sup>  
دانشجوی ارشد مهندسی دانشگاه آزاد اسلامی واحد قزوین، گروه کامپیوتر، قزوین، ایران  
nedamaleki87@gmail.com  
عضو هیئت علمی دانشگاه آزاد اسلامی واحد قزوین، گروه کامپیوتر، قزوین، ایران  
zeinali@qiau.ac.ir  
عضو هیئت علمی دانشگاه صنعتی امیرکبیر، تهران، ایران  
akbarif@aut.ac.ir  
دانشجوی دکتری مهندسی فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران  
msjavan@aut.ac.ir

چکیده - سرویس های کاربردی که روی مدل محاسباتی ابر قرار می گیرند، نیازهایی پیچیده از جمله آماده سازی، پیکربندی و استقرار دارند. ارزیابی کارایی سیاست های آماده سازی، مدل های بار کاری پویا و مدل های کارایی منابع با یک روش قابل تکرار و تحت سیستم های مختلف و نیازها و پیکربندی های متفاوت کاربران، کار بسیار مشکلی است. برای غلبه بر این چالش، ابزار شبیه سازی CloudSim را معرفی می کنیم که مدل سازی و شبیه سازی سیستم های محاسباتی ابر و محیط های آماده سازی کاربردها را مقدور می سازد. جهت ارزیابی این شبیه ساز، مقدار حافظه مصرفی و سربار حاصل از آن را بررسی می کنیم.  
کلیدواژه ها - شبیه سازی، مدل سازی، بارکاری پویا، آماده سازی.

امکان وجود ندارد که آزمایش های محک را در یک محیط قابل مقیاس، قابل تکرار و قابل اعتماد با استفاده از یک محیط ابر واقعی انجام داد. پس کلید حل این مشکلات استفاده از ابزار شبیه سازی است [3]. بخش های بعدی این مقاله به این شکل سازمان دهی می شود: در بخش ۲ شبیه ساز CloudSim را مورد بررسی قرار می دهیم و ویژگی ها و مدل سازی رفتارهای دنیای واقعی را بیان می کنیم. در بخش ۳ ارزیابی سربار و مقیاس پذیری CloudSim بیان خواهد شد و در بخش ۴ نتیجه کلی بیان می شود.

### ۱. مقدمه

نرم افزاری کلی است که می توانیم برای مدل کردن محیط های رایانش ابری و تست کارایی سرویس های کاربردی استفاده کنیم. این شبیه ساز مبتنی بر زبان جاوا است و یک شبیه ساز مبتنی بر رویداد<sup>۱</sup> است. یعنی موجودیت هایی که در آن تعریف می شود، از طریق فرستادن رویداد با یکدیگر ارتباط برقرار می کنند [1,2]. ارائه دهندگان سرویس ابر برای این که سرویسی را در سطح عالی به کاربران ارائه دهند با مسائلی از جمله تضمین کیفیت سرویس که می تواند زمان پاسخ کمتر و یا گذردهی بالاتر باشد، بهره وری کارایی منابع، بارهای کاری پویا مانند مدل کردن کاربردهای وبی چون سرویس دهنده-سرویس گیرنده و نقض سطح توافق ارائه سرویس<sup>۲</sup> مواجهه بوده اند. در این میان، مشکل تست کردن از جمله مهمترین مسائل است زیرا این

### ۲. شبیه ساز CloudSim

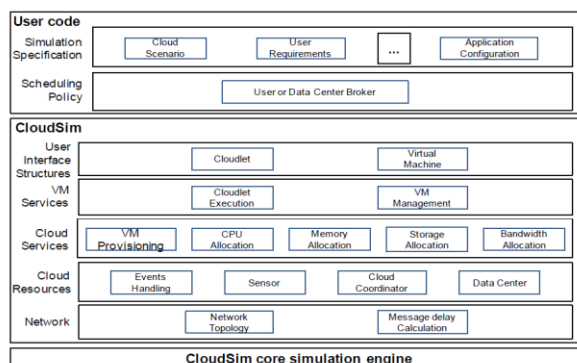
شبیه ساز Cloudsim، مدل سازی رفتار مولفه های سیستم مانند مراکز داده، ماشین های مجازی و سیاست های آماده سازی منابع را پشتیبانی می کند [1]. مولفه هایی مانند مرکز داده، کارگزار<sup>۳</sup> و سرویس اطلاعات ابر<sup>۴</sup> را به عنوان موجودیت در

<sup>۱</sup> - Event-driven

<sup>۲</sup> - Service Level Agreement (SLA)

<sup>۳</sup> - Broker

ایجاد مولفه های سیستمی مانند سرویس ها، میزبان ها و مراکز داده، کارگزار و ماشین های مجازی و یا قابلیت های ارتباط بین مولفه ها و مدیریت کلاک شبیه سازی را نام برد. به عبارت کامل تر، وظایف رسیدگی به رویدادها<sup>۷</sup> و ارسال پیام ها<sup>۸</sup> به عهده این لایه است.



شکل ۱: ساختار لایه ای CloudSim [1]

لایه بعدی GridSim است که مولفه های نرم افزاری سطح بالا را برای مدل کردن چندین زیرساخت گرید مانند شبکه ها، پروفایل های کاربران، ترافیک شبکه و مولفه های بنیادی گرید مانند منابع، مجموعه داده ها، سرویس اطلاعات گرید<sup>۹</sup> و تاثیرات بار کاری ارائه می دهد.

لایه سوم CloudSim است که به وسیله برنامه نویسی، قابلیت های هسته ای لایه GridSim را به ارث می برد. این لایه مدل سازی و شبیه سازی محیط های مرکز داده مبتنی بر ابر مانند: واسطه های مدیریت اختصاصی ماشین های مجازی، حافظه ها و پهنای باند را انجام می دهد. همچنین موجودیت های اصلی مثل ماشین های مجازی، میزبان ها، مراکز داده و کاربردها را در طی شبیه سازی، نمونه سازی و اجرا می کند. از جمله قابلیت های آن، ارائه میزبان ها به ماشین های مجازی بر اساس درخواست های کاربران، مدیریت اجرای کاربردها و نظارت پویای آن ها است. همه این کارها برای یک ارائه دهنده سرویس که می خواهد کارایی سیاست های مختلف را بررسی کند باید در بخش آماده سازی ماشین های مجازی<sup>۱۰</sup> از طریق برنامه نویسی انجام شود.

CloudSim تعریف می کنیم. در حال حاضر این شبیه ساز محیط های رایانش ابری واحد و یا متحد شده<sup>۵</sup> (مجموعه ای از چند محیط ابری که توسط ارائه دهندگان مختلف سرویس ابر تولید می شود). همچنین واسطه هایی را برای پیاده سازی سیاست ها و تکنیک های آماده سازی جهت تخصیص ماشین های مجازی تحت سناریوهای پیچیده در اختیار قرار می دهد. محققان بسیاری در زمینه آماده سازی منابع ابر و مدیریت انرژی کاری مراکز داده در سازمان هایی مانند آزمایشگاه HP در آمریکا از شبیه ساز CloudSim برای انجام کارهای خود استفاده می کنند.

## ۲.۳ مزایای CloudSim

قبل از CloudSim، برای شبیه سازی محیط های توزیع شده از شبیه سازهایی همچون GridSim [8]، OptoStrim، SimGrid و GangSim استفاده می کردند. اما به این دلیل که هیچ یک از این شبیه سازها قادر نبودند لایه های سرویس ابر IaaS، PaaS و SaaS را از هم تفکیک کنند و منابع مجازی سازی شده مورد نیاز ابر را مدل کنند، به سمت استفاده از CloudSim روی آورده ایم [1,6].

از جمله مزایای این شبیه ساز عبارت است از:

- در نظر گرفتن تاثیرات زمان.
- شبیه ساز کاملاً انعطاف پذیر و کاربردی.
- تست سیاست های مورد نظر در یک محیط قابل کنترل و قابل تکرار
- تنظیم گلوگاه های سیستم قبل از انجام استقرار روی ابر واقعی.

## ۲.۲ ساختار لایه ای CloudSim

ساختار لایه ای Cloudsim را در شکل ۱ مشاهده می کنید [1,2,3]. همانطور که ملاحظه می کنید در پایین ترین سطح، موتور شبیه سازی زمان گسسته SimJava<sup>۴</sup> قرار دارد که قابلیت های هسته ای مورد نیاز برای چارچوب های شبیه سازی سطح بالاتر یعنی GridSim و لایه های بالاتر را پیاده می کند. از جمله این قابلیت ها می توان صف بندی و پردازش رویدادها،

<sup>7</sup> - Event Handling

<sup>8</sup> - Message Passing

<sup>9</sup> - Grid Information Service (GIS)

<sup>10</sup> - Virtual Machines Provisioning

<sup>4</sup> - Cloud Information Service

<sup>5</sup> - Federated Clouds

<sup>6</sup> - SimJava Discrete Event

سرویس های کاربردی عمل می کند. از این رو مدل های سنتی آماده سازی کاربردها که عناصر کاربردی منفرد را به نودهای محاسباتی تخصیص می دهد، انتزاع محاسباتی را که به طور عادی به منابع ابر مرتبط می شوند به طور دقیق نشان نمی دهند. برای مثال یک میزبان را که یک هسته پردازشی واحد دارد را در نظر بگیرید، نیاز به ایجاد دو ماشین مجازی بر روی این میزبان به طور همزمان به چشم می خورد. حتی اگر ماشین های مجازی به طور معمول از یکدیگر جدا باشند (حافظه فیزیکی و حافظه جانبی مجزا برای هر کدام)، بازهم نیاز دارند تا هسته های پردازشی یا باس سیستم را با یکدیگر به اشتراک بگذارند. گرچه مقدار منابع سخت افزاری موجود برای هر ماشین مجازی توسط توان پردازشی و پهنای باند میزبان، محدود شده است. این فاکتور بحرانی باید هنگام فرایند آماده سازی ایجاد ماشین مجازی در نظر گرفته شود تا از ایجاد ماشین مجازی ای که توان پردازشی بیشتری را می طلبد و موجود نیست، جلوگیری کند.

به منظور شبیه سازی سیاست های آماده سازی مختلف تحت سطوح مختلف کارایی که از یکدیگر مجزا باشند، شبیه ساز CloudSim آماده سازی ماشین های مجازی را در دو سطح پشتیبانی می کند [1,3]: ابتدا در سطح میزبان و سپس در سطح ماشین مجازی. در سطح میزبان تعیین می کنیم که چه مقدار از توان پردازشی کل هر هسته به هر ماشین مجازی اختصاص خواهد یافت. در سطح ماشین مجازی، هر ماشین مجازی یک مقدار ثابتی از توان پردازشی موجود را به سرویس های کاربردی منحصر به فرد (واحد های وظیفه) که بر روی آن قرار گرفته است، تخصیص می دهد. در این شبیه ساز، هر واحد وظیفه را به عنوان یک انتزاع جزئی تر از یک سرویس کاربردی که در ماشین مجازی قرار گرفته است، در نظر می گیریم. در شبیه ساز CloudSim، در هر سطح دو سیاست SpaceShared و TimeShared وجود دارد. شکل زیر تفاوت بین این دو سیاست و تاثیر آنها بر کارایی سرویس کاربردی را نشان می دهد [1,2].

در لایه چهارم که آخرین لایه است و در بالاترین سطح قرار دارد، کاربر درخواست های خود از جمله کاربرد ها (که در این شبیه ساز با عنوان Cloudlet تعریف می شوند و با پارامترهایی چون تعداد دستورالعمل ها، مقدارحافظه مورد نیاز و غیره مدل می شود) و ماشین های مجازی مورد نیاز خود (که با قدرت پردازشی، تعداد هسته های پردازشی، مقدار حافظه مورد نیاز و غیره مدل می شود) را تعریف می کند.

برای شبیه سازی نیاز است تا مفاهیم موجود در دنیای واقعی را مدلسازی کنیم. بنابراین مدلسازی این مفاهیم در شبیه ساز CloudSim را بیان می کنیم.

## ۲.۳. مدل سازی در CloudSim

### ۲.۳.۱. مدل سازی ابر

برای مدل کردن سرویس های سطح زیرساخت در شبیه ساز، می توانیم موجودیت مرکز داده را توسعه دهیم [1]. موجودیت مرکز داده تعداد زیادی از موجودیت ها مانند میزبان را مدیریت می کند. این میزبان ها طبق سیاست تخصیص ماشین های مجازی که توسط ارائه دهنده سرویس ابر تعریف می شود، به ماشین های مجازی تخصیص داده می شوند. در این شبیه ساز سیاست ماشین مجازی<sup>۱۱</sup> بیانگر سیاست های کنترل ماشین مجازی در رابطه با دوره زندگی آن مانند فراهم کردن میزبان برای ماشین مجازی، ایجاد و از بین بردن آن و مهاجرت این ماشین های مجازی می باشد. به طور مشابه یک یا چند سرویس کاربردی می تواند به این ماشین مجازی واحد تخصیص یابد. و این همان آماده سازی کاربردها در زمینه رایانش ابری است. در این شبیه ساز هر موجودیت نمونه ای از یک مولفه است. یک مولفه در CloudSim می تواند یک کلاس (انتزاعی یا کامل)، یا مجموعه ای از کلاس ها باشد که یک مدل (مرکز داده یا میزبان) را در CloudSim نشان می دهد.

### ۲.۳.۲. مدل کردن تخصیص ماشین های مجازی [1]

یکی از جنبه های کلیدی که زیرساخت محاسبات ابری را از زیرساخت محاسبات توری<sup>۱۲</sup> متمایز می سازد، استقرار سنگین ابزارها و تکنولوژی های مجازی سازی است. بنابراین برخلاف گرید، ابر از یک لایه اضافه تر به نام لایه مجازی ساز تشکیل می شود که به عنوان یک اجرا، مدیریت و محیط میزبانی برای

<sup>۱۱</sup> - VM Policy

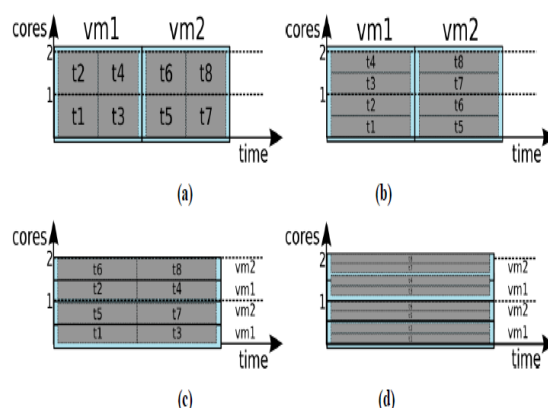
<sup>۱۲</sup> - Grid Computing

مفهومی<sup>۱۴</sup> است. در این مدل، هیچ موجودیت واقعی برای شبیه سازی موجودیت های شبکه مانند روترها و سوئیچ ها وجود ندارد. در عوض، تاخیر شبکه ای که یک پیام در مسیرش از نود مبدا (یک موجودیت مانند میزبان) به نود مقصد (مانند موجودیت کارگزار ابر) با آن مواجه می شود، بر اساس اطلاعاتی که در ماتریس تاخیر ذخیره شده است شبیه سازی می شود (جدول ۱).

جدول ۱: ماتریس تاخیر [1]

0	40	120	80	200
40	0	60	100	100
120	60	0	90	40
80	100	90	0	70
200	100	40	70	0

برای مثال این ماتریس تاخیری را نشان می دهد که شامل ۵ موجودیت شبیه ساز CloudSim است. درهر نمونه از زمان، محیط CloudSim اندازه ماتریس  $n \times m$  را برای همه موجودیت های CloudSim که در حال حاضر در زمینه شبیه سازی فعال هستند را نگه می دارد. یک مدخل  $e_{ij}$  در ماتریس، بیانگر تاخیری است که یک پیام (رویداد) زمانی که از موجودیت  $i$  به موجودیت  $j$  در سطح شبکه منتقل می شود خواهد داشت. از طرفی گفتیم که CloudSim شبیه ساز مبتنی بر رویداد است که موجودیت ها/مدل های مختلف سیستم از طریق فرستادن رویداد با یکدیگر ارتباط برقرار می کنند. موتور مدیریت رویداد CloudSim از اطلاعات پوشیدگی<sup>۱۵</sup> شبکه بین موجودیتی<sup>۱۶</sup> برای تحریک تاخیرها در انتقال پیام به موجودیت ها استفاده می کند. این تاخیر به صورت واحدهای زمانی شبیه سازی مانند میلی ثانیه بیان می شود. این بدین معنی است که یک رویداد از موجودیت  $i$  به موجودیت  $j$  تنها زمانی که زمان شبیه سازی کلی فرا برسد یعنی  $t+d$ ، به وسیله موتور مدیریت رویداد به جلو<sup>۱۷</sup> خواهد رفت. که  $t$  زمان شبیه سازی است وقتی که پیام واقعا فرستاده شده است و  $d$  تاخیر شبکه بین موجودیت  $i$  و  $j$  است. دیگرام انتقال چنین تعاملی در شکل ۳ نشان داده شده است.



شکل ۲: تاثیرات سیاست های آماده سازی مختلف بر اجرای سرویس های کاربردی [۱]

در این شکل یک میزبان با دو هسته پردازنده، دو درخواست برای میزبانی ماشین مجازی را دریافت می کند. هر کدام از ماشین های مجازی به دو هسته نیاز دارند و می خواهند ۴ واحد وظیفه را اجرا کنند. وظایف  $t_1, t_2, t_3, t_4$  را ماشین مجازی ۱ و وظایف  $t_5, t_6, t_7, t_8$  را ماشین مجازی ۲ اجرا می کند. شکل (a) برای هر دوسطح سیاست SpaceShared و شکل (b) سیاست SpaceShared را برای ماشین های مجازی و سیاست TimeShared را برای کاربردها، شکل (c) سیاست TimeShared را برای ماشین های مجازی و سیاست SpaceShared را برای کاربردها و شکل (d) سیاست TimeShared را برای هر دو سطح به کار می گیرد.

### ۲.۳.۳. مدل کردن رفتار شبکه

مدل کردن توپولوژی شبکه جامع برای اتصال موجودیت های شبیه سازی شده رایانش ابری (مانند میزبان ها، حافظه ذخیره سازی و کاربران) یک مسئله مهم است زیرا تاخیرپیام ها به طور مستقیم روی سرویس هایی که رضایتمندی کاربران را فراهم می سازد، تاثیر می گذارد. یک کاربر پایانی با یک ارائه دهنده SaaS که از کیفیت سرویس ارائه شده راضی نیستند، ممکن است ارائه دهنده سرویس خود را تغییر دهد تنها به این دلیل که فریم ورک های شبیه سازی سیستم ابر تسهیلاتی را برای مدل کردن توپولوژی های شبکه ای و تولید و مدل کردن آنها فراهم نمی کند و این نیاز مهم آنها را برآورده نمی سازد [6]. شبکه بندی بین<sup>۱۳</sup> موجودیت های ابر مانند مراکز داده، ارائه دهندگان SaaS و کاربران پایانی بر اساس انتزاع شبکه های

<sup>14</sup>- Conceptual Networking Abstraction

<sup>15</sup>- Latency Information

<sup>16</sup>- Inter-Entity

<sup>17</sup>- Forward

<sup>13</sup>- Inter-Networking

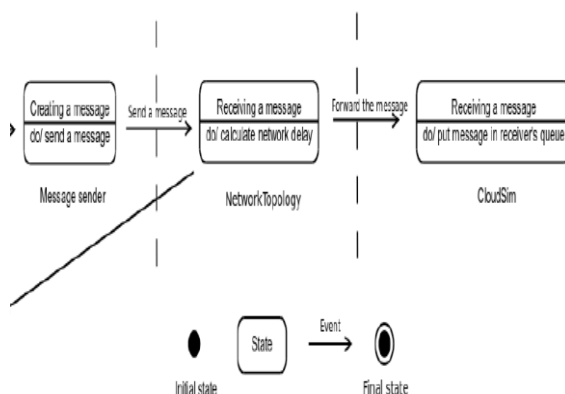
مصرفی باید این کلاس را توسعه دهیم. کاربران شبیه ساز باید متد `getPower()` از این کلاس را باز نویسی کنند که پارامتر ورودی آن بهره وری فعلی میزبان و پارامتر خروجی مقدار توان مصرفی فعلی است. این قابلیت ایجاد سیاست های آماده سازی توان آگاه را که نیاز به دانش در لحظه از توان مصرفی توسط مولفه های سیستمی ابر را دارد، قادر می سازد.

۲,۳,۵. مدل کردن بارهای کاری پویا

توسعه دهندگان نرم افزار و مشتریان ابر جهت ارائه سرویس به کاربران پایانی، کاربردهایی را که رفتار پویا با توجه به نیازهای الگوهای بار کاری، قابلیت دسترسی و مقیاس پذیری دارند، استقرار می دهند [2,3]. بنابراین یکی از نیازهای مهم در هر محیط شبیه سازی است که از مدل کردن الگوهای بار کاری پویا که به دلیل وجود این کاربردها یا همان مدل های SaaS به وجود می آیند، پشتیبانی کند. از جمله الگوهای بارکاری که وجود دارد الگوی بار کاری کامل<sup>۲۰</sup> و بار کاری تصادفی<sup>۲۱</sup> است که در این شبیه ساز با تغییر بهره وری ماشین های مجازی آن را مدل کرده ایم. یعنی بهره وری پردازنده ها را در بازه های زمانی مختلف تغییر می دهند تا بتوانند بار کاری حالت واقعی (بارکاری ای که در دنیای واقعی وجود دارد) را مدل سازی کنند. بنابراین پارامتر زمان را به عنوان ورودی می دهیم و در صد منبع محاسباتی مورد نیاز را محاسبه می کنیم.

### ۳. ارزیابی سربار و مقیاس پذیری CloudSim

تستی برای تحلیل سربار و مقیاس پذیری حافظه مصرفی شبیه ساز CloudSim انجام گرفته است که بیانگر بهره وری شبیه ساز است [1]. برای انجام این تست، از یک میزبان با دو Intel Xeon هر کدام چهار هسته ای با ۲,۲۷ گیگا هرتز توان پردازشی و یک حافظه اصلی ۱۶ گیگا بایت استفاده کرده ایم. همچنین از یک ماشین مجازی با سیستم عامل Ubuntu 8.04 برای اجرای تست ها استفاده کرده ایم. تنظیمات محیط شبیه سازی برای محاسبه سربار و مقیاس پذیری CloudSim که شامل کارگزار و مرکز داده می باشد، به این صورت است که ابتدا در یک مرکز داده و سپس در دو مرکز داده هر کدام با تعداد میزبان های متغیر از ۱۰۰۰ تا ۱۰۰۰۰۰، تست را انجام می دهیم. هر



شکل ۳: روند ارتباطات شبکه ای در CloudSim [6]

این روش شبیه سازی تاخیرات شبکه به ما یک راه واقعی و ساده برای مدل کردن ساختار شبکه ای کاربردی برای محیط شبیه سازی می دهد. به علاوه این رویکرد بسیار ساده تر و رساتر از مدل کردن کردن مولفه های شبکه ای پیچیده مانند روترها و سوئیچ ها برای پیاده سازی است.

۲,۳,۴. مدل کردن مصرف توان مرکز داده

به دلیل ارتباطات زیاد تعداد میزبان های موجود در مراکز داده جهت ارائه سرویس های ابری و همچنین وجود سیستم های خنک کننده برای خنک کردن این میزبان ها، مقدار زیاد انرژی الکتریکی برای تامین این سیستم ها موجب افزایش هزینه های عملیاتی می شود [1,7]. بنابراین فقدان تکنیک های آماده سازی توان آگاه ممکن است موجب حرارت زیادی در منابع (سرورهای محاسباتی و ذخیره سازی) شود. موضوع مرتبط دیگر، انتشار گاز CO2 است که ممکن است برای محیط فیزیکی به دلیل تاثیرات گلخانه ای مضر باشد [7]. بنابراین شبیه ساز CloudSim مدل ها و موجودیت های پایه ای را برای ارزیابی و اعتبار الگوریتم ها و تکنیک های آماده سازی توان آگاه ارائه می دهد و برای تسهیل این کار، تعدادی توسعه صورت گرفته است. به عنوان مثال شی PE<sup>۱۸</sup> را طوری توسعه داده اند که شامل شی ای به نام مدل توان<sup>۱۹</sup> است که توان مصرفی به ازای هر میزبان ابر را مدیریت می کند. برای پشتیبانی از مدل کردن و شبیه سازی مدل های توان مصرفی مختلف مانند DVFS یک پیاده سازی انتزاعی به نام مدل توان است. برای شبیه سازی مدل توان

<sup>20</sup> - Full Workload Pattern

<sup>21</sup> - Stochastic Workload Pattern

<sup>18</sup> - Processing Element

<sup>19</sup> - Power Model

همچنین نتایج نشان می دهد که زمان مورد نیاز برای نمونه سازی یک میلیون میزبان، در حدود ۱۲ ثانیه است. بنابراین با مشاهده این نتایج در می یابیم که CloudSim قادر است از یک محیط شبیه سازی شده مقیاس وسیع با میزان سربرار ناچیز و حافظه کم، پشتیبانی کند.

#### ۴. نتیجه گیری

در این مقاله به بررسی شبیه ساز CloudSim و ویژگی های آن پرداختیم. همچنین مدل سازی رفتار های دنیای واقعی را در این شبیه ساز بیان کردیم و نشان دادیم که نه تنها سربرار حافظه مصرفی این شبیه ساز بسیار ناچیز است، بلکه دقت شبیه سازی بسیار بالایی دارد و برای تست الگوریتم ها و ارزیابی آنها، ابزار بسیار مناسبی می باشد. گرچه این شبیه ساز قابلیت ارتقا جهت در برگرفتن مشخصه های مهمی چون مدل کردن کاربردهای چند لایه ای، الگوهای بار کاری مختلف و غیره را نیز دارد و در این زمینه می توان تحقیقات گسترده ای انجام داد. از جمله کارهایی که تاکنون در زمینه انجام گرفته است، شبیه ساز کارهایی که تاکنون در زمینه انجام گرفته است، شبیه ساز CloudAnalyst [4] و CloudReport [5] است که دارای واسطه گرافیکی هستند و برای تست شبکه های اجتماعی و تجاری بسیار مناسب هستند.

#### مراجع

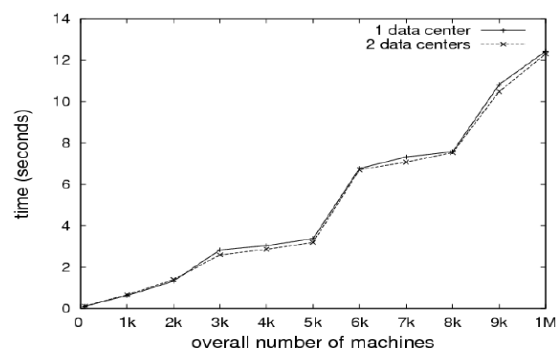
[1] R. Calheiros, R. Ranjan, A. Beloglazov, F. De Rose and R. Buyya, "CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms", IEEE 2011, 1-24 pages.

[2] Krishnadhyan Das, "Extension of cloudsims: cloud computing simulator", A Thesis Submitted in partial fulfillment of the requirements for the degree of Master of Technology under the guidance of Prof. Purushottam Kulkarni and Prof. Umesh Bellur.

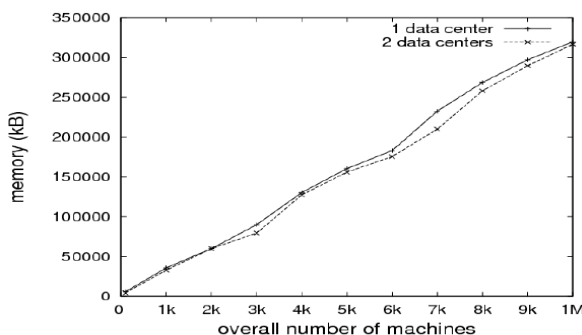
[3] R. Buyya, R. Ranjan, and R.N. Calheiros, "Modeling and simulation of scalable cloud computing environments and the cloudsims toolkit: Challenges and opportunities", In High Performance Computing & Simulation, 2011. HPCS'09. International Conference on, pages 1{11. IEEE, 2011.

[4] B. Wickremasinghe, R. Calheiros, and R. Buyya. "CloudAnalyst: A CloudSim-based Visual Modeller for Analysing Cloud Computing Environments and Applications", Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications (AINA 2010), Perth, Australia, April 20-23, 2010.

آزمایش ۳۰ بار تکرار شده است تا به بازه اعتماد<sup>۲۲</sup> ۹۵٪ دست یابیم. برای تست حافظه، مصرف حافظه فیزیکی کلی مورد نیاز برای نمونه گیری و بارگیری کامل محیط CloudSim، پروفایل شده است. برای تست سربرار، تاخیر کلی در مقداردهی کردن محیط شبیه ساز به عنوان تفاوت زمانی بین این رویدادها محاسبه شده است: الف) زمانی که در آن به ماشین مجازی جاوا<sup>۲۳</sup> فرمان داده می شود تا چارچوب CloudSim را بارگیری کند و ب) زمانی که در آن موجودیت های CloudSim به طور کامل مقداردهی اولیه می شوند و برای پردازش رویدادها آماده می باشند.



شکل ۴ (a): ارزیابی سربرار



شکل ۴ (b): ارزیابی حافظه مصرفی

شکل ۴ (a) مقدار متوسط زمانی را که برای راه اندازی شبیه سازی مورد نیاز است، بر حسب تابعی از چندین میزبان نشان می دهد. شکل ۴ (b) نیز مقدار حافظه ای که برای انجام تست ها به طور موفقیت آمیز مورد نیاز بود را نشان می دهد. نتایج نشان می دهد که سربرار به طور خطی با اندازه سیستم رشد نمی کند. در عوض می بینیم که به صورت مرحله ای، زمانی که تعداد خاصی میزبان استفاده می شوند، رشد می کند.

<sup>22</sup> - Confidence Interval

<sup>23</sup> - Java Virtual Machine(JVM)

[5] R. N. Colheiro, "CloudReport: A CloudSim-based Tool for Modelling and Analysis of Large Scale Cloud Computing Environments", MEDC Project by Rajkumar Buyya project supervisor Report, 22 may 2011.

[6] A. Medina, A. Lakhina, I. Matta, J. Byer, "BRITE: An Approach to Universal Topology Generation", Proceedings of the 9th International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS 2001), Cincinnati, OH, USA, 15-18 August 2001.

[7] Anton Beloglazov, Rajkumar Buyya, Young Choon, "A Taxonomy and Survey of Energy-Efficient Data Centers and Cloud Computing systems", LEE, ALBERT ZOMAYA.

[8] R. Buyya and M. Murshed, "GridSim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing", Concurrency and Computation: Practice and Experience, 14(13-15), Wiley Press, Nov.-Dec., 2002.