علوم رایانش و فناوری اطلاعات
نشریه علمی انجمن کامپیوتر ایران
مجلد ۲۰، شماره ۲، ۱۴۰۱
صفحات ۲۶-۳۸
مقاله پژوهشی

مروری بر معیارهای محک آزمون و ارزیابی سامانه‌های پردازشی فوق‌سریع به همراه نصب، پیکربندی و تحلیل معیارهای منتخب و مطالعه موردی در سامانه پردازشی فوق‌سریع نمونه

احسان آریانیان[۱*]، زهرا معزکریمی[۲]

*نویسنده مسئول، دریافت: ۱۴۰۰/۰۷/۱۸، بازنگری: ۱۴۰۱/۰۷/۲۳، پذیرش: ۱۴۰۱/۰۹/۱۶

[۱] استادیار، دکتری مهندسی برق، پژوهشگاه ارتباطات و فناوری اطلاعات، تهران، ایران
[۲] پژوهشگر، دکتری مهندسی کامپیوتر، پژوهشگاه ارتباطات و فناوری اطلاعات، تهران، ایران

چکیده

در این مقاله، معیارهای محک رایج برای آزمون و ارزیابی سامانه‌های پردازشی فوق‌سریع بررسی شده است. بدین منظور، با مرور معیارهای مختلف رتبه‌بندی سامانه‌های پردازشی فوق‌سریع، روش‌های آزمون و ارزیابی معیارهای منتخب برای منابع پردازشی، حافظه، شبکه و دیسک معرفی و نحوه راه‌اندازی، پیکربندی اولیه و اجرای هریک شرح داده می‌شود. همچنین با راه‌اندازی یک سامانه آزمایشی، خروجی هریک از معیارها به تفکیک ارائه و تحلیل می‌شود. در انتها، به‌منظور نشان‌دادن کاربردپذیری و داشتن درک صحیحی از مقدار خروجی معیارها در عمل، تحلیل نتایج حاصل از معیارها برای یک سامانه پردازشی فوق‌سریع واقعی ارائه می‌گردد. از آن‌جا که معیارهای درنظر گرفته‌شده در این مقاله، از معتبرترین معیارهای ارزیابی هستند و نیز دستورات آماده‌سازی محیط و نصب و اجرای معیارها با جزئیات کامل ارائه شده است، از محتویات این مقاله می‌توان برای انتخاب معیار و نصب و راه‌اندازی آن برای آزمون و ارزیابی سامانه‌های پردازشی فوق‌سریع در ابعاد واقعی و به‌صورت عملیاتی استفاده کرد.

کلمات کلیدی: ابررایانه، سامانه‌های پردازشی فوق‌سریع، ارزیابی، رتبه‌بندی ابررایانه‌ها، آزمون محک

۱- مقدمه

در اوایل دهه ۹۰ [۱] دنیا با مفهوم جدیدی به نام ابررایانه آشنا شده بود که از قدرت پردازش بسیار بالاتری نسبت به رایانه‌های معمولی برخوردار بود. به‌تدریج ابررایانه‌ها به دلیل داشتن قدرت پردازشی بالاتر و توانایی حل مسائل در زمان کمتر، در پیشرفت‌های علمی و تحقیقاتی مانند هواشناسی، تغییرات آب‌وهوایی و شناسایی و گسترش صنعت نفت و گاز، نقش مهمی پیدا کردند.

با پیشرفت فناوری‌های مختلف، قدرت پردازشی ابررایانه‌ها و سامانه‌های پردازشی فوق‌سریع که از این به بعد آن را سامانه HPC می‌نامیم، نیاز به

معیارهای جدیدی برای تعریف داشت. در ابتدا از تعداد هسته‌های پردازشی موجود در یک سامانه HPC برای این منظور استفاده می‌شد. در تلاش‌های بعدی سعی شد تا این معیارها قدرت کلی یک سامانه HPC شامل قدرت پردازشی، سرعت حافظه، پهنای باند شبکه و سرعت دیسک را مورد سنجش قرار دهند [۲].

امروزه سامانه‌های HPC شامل گره‌های مختلفی می‌شوند که در کنار هم به حل مسائل پیچیده می‌پردازند به‌طوری که در نهایت تعداد زیادی از گره‌ها می‌توانند تلاشی برای حل یک مسئله واحد داشته باشند. با توجه به معماری سامانه‌های HPC و توانایی استفاده از قدرت هزاران گره، قدرت پردازشی یک سامانه HPC بسیار بیشتر از سیستم‌های معمول می‌باشد. معمولاً قدرت پردازشی یک سامانه HPC با معیار FLOPS که تعداد عملیات ممیز شناور در ثانیه است سنجیده می‌شود [۳]. همچنین معیارهای اختصاصی مختلفی برای سنجش هریک از منابع سیستم، توسعه یافته‌اند. با ارزیابی سامانه HPC بر اساس این معیارها، می‌توان به شناخت دقیق‌تری از آنها و جایگاه آن در بین دیگر سامانه‌های HPC دنیا دست یافت [۲].

سامانه‌های HPC از گره‌های مختلفی تشکیل می‌شوند که هر گره منابع مختلفی دارد. این گره‌ها و منابع در کنار هم برای ارائه نتیجه نهایی استفاده می‌شوند. برای سنجش قدرت یک سامانه HPC نیاز است تا همه این منابع به‌صورت جداگانه و همچنین به‌صورت کلی سنجیده شوند [۴]. به‌طور کلی منابع موجود در یک سامانه HPC را می‌توان به چهار دسته زیر تقسیم‌بندی کرد:

۱. پردازشی
۲. حافظه
۳. شبکه
۴. دیسک

برای هریک از این منابع، معیارهای مختلفی برای سنجش وجود دارند که می‌توان آن معیار را برای آزمون و ارزیابی منبع موردنظر پیکربندی کرد.

معیارهای موجود برای سنجش منابع مختلف سامانه HPC در جدول ۱ ارائه شده است.

جدول ۱. معیارهای سنجش منابع مختلف سامانه HPC

منبع مورد سنجش	معیار
پردازنده	LINPACK (HPL)[1], [2]
	DGEMM[3]
	NPB[4], [5]
	IAPCM[6]
	Graph500[7], [8]
	HPL-AI[9]
حافظه	STREAM[10]
	RandomAccess[11]
شبکه	b_eff[12]
	PTRANS[13]
	OMB[14]
دیسک	IOR[15]
	Mdtest[16]

در این مقاله به‌ازای هر منبع سامانه HPC، دو مورد از بنچمارک‌ها بر اساس نیازمندی، دقت نتیجه و سهولت در پیاده‌سازی انتخاب و شرح داده شده‌اند. در میان معیارهای پردازشی، مهم‌ترین معیار، معیار HPL می‌باشد که در معتبرترین رتبه‌بندی‌های جهان نظیر Top500 از آن استفاده می‌شود [۱۷]. همچنین DGEMM معیار دیگری است که با ارائه نتایجی نزدیک به HPL، پیاده‌سازی ساده‌تری نسبت به آن دارد. معیار NPB نسبت به بقیه کمتر شناخته شده است و متعلق به شرکت ناسا می‌باشد. این معیار در واقع از برنامه‌های دینامیک سیالات محاسباتی (CFD) نشأت گرفته شده است. در نسخه‌های بعدی، معیارهایی ازقبیل خواندن و نوشتن موازی به‌منظور سنجش سامانه HPC با برنامه‌هایی در نقاط مختلف و شبکه‌های پردازشی موازی به آن اضافه شده است. معیار IAPCM متعلق به کشور چین است و نسخه متن‌بازی از آن برای پیاده‌سازی وجود ندارد. معیار Graph500 هم برای سنجش سامانه‌های HPC خاص منظوره، که به‌طور تخصصی برای حل مسائل مربوط به گراف‌ها ساخته می‌شوند، مناسب است.

همچنین، معیارهایی برای سنجش پردازنده گرافیکی سامانه HPC وجود دارند. معیار HPL-AI از این دسته معیارها می‌باشد که با حل مسائل هوش مصنوعی، به سنجش پردازنده گرافیکی سامانه HPC می‌پردازد. با توجه به نیازمندی‌ها و همچنین با هدف شناخت بیشتر معیارهایی برای سامانه‌های HPC عمومی، دو معیار HPL و DGEMM برای شرح بیشتر در بخش منبع پردازشی انتخاب شدند.

گروه دیگری از معیارها، مربوط به سنجش منبع حافظه سامانه HPC می‌باشد. در این قسمت، دو معیار STREAM و RandomAccess وجود دارند که در رتبه‌بندی HPC Challenge[1] از این دو معیار استفاده می‌شود. در قسمت منبع ارتباطی و شبکه دو معیار b_eff و PTRANS در بیشتر رتبه‌بندی‌ها نظیر HPC Challenge و OpenBenchmarking[2] استفاده می‌شوند بنابراین در این مقاله هم به بررسی این دو معیار در بخش شبکه می‌پردازیم. در این بخش معیار OMB نیز وجود دارد که از آن برای سنجش سرعت شبکه بر روی بستر MPI استفاده می‌شود. این معیار، قابلیت سنجش سرعت بین دو گره مختلف در یک سامانه HPC را دارد. متأسفانه برای این معیار، مستندات کمی نسبت به دو معیار دیگر موجود است. در نهایت، در قسمت سنجش دیسک، معیار IOR را برای بررسی انتخاب می‌کنیم زیرا این معیار شامل معیار دیگر، یعنی Mdtest نیز می‌باشد و با پیکربندی آن، می‌توان هر دو معیار را به‌صورت همزمان استفاده کرد. شایان ذکر است که شرکت‌های بزرگ نظیر دل و آی‌بی‌ام عمدتاً از زیرمجموعه‌ای از همین ابزارها

1 https://icl.utk.edu/hpcc

2 https://openbenchmarking.org

برای نشان‌دادن توان پردازشی نسل‌های جدید پردازنده‌ها خود استفاده می‌کنند.

ما برای نشان‌دادن خروجی‌های نمونه هر معیار، یک سامانه کمینه با دو گره راه‌اندازی کرده‌ایم. هرکدام از این گره‌ها دارای چهار هسته پردازشی با قدرت ۱/۸ گیگاهرتز و ۸ گیگابایت حافظه اصلی هستند. در ادامه مقاله به این سامانه برپاسازی‌شده، به‌اختصار سامانه آزمون می‌گوییم و خروجی آزمون‌های مختلف برای این سامانه ارائه خواهند شد.

نوآوری‌های اصلی این مقاله به‌صورت زیر می‌باشد:

۱) این مقاله برای اولین‌بار ابزارها و بنچمارک‌های موجود برای آزمون و ارزیابی سامانه‌های پردازش فوق‌سریع (ابررایانه) را در تمامی بخش‌های مختلف (۱) پردازشی، (۲) حافظه ذخیره‌سازی، (۳) شبکه ارتباطی و (۴) حافظه دیسک، به‌صورت یکپارچه مرور کرده است.

۲) نتایج اجرای آزمون‌ها را بر روی دو ابررایانه واقعی در قالب مطالعه موردی ارائه کرده است. یکی از این ابررایانه‌ها، ابررایانه آریا (متعلق به هدلینگ مدیس) به‌عنوان سامانه پردازش فوق‌سریع آزمون و دیگری ابررایانه سیمرغ به‌عنوان قوی‌ترین سامانه پردازش فوق‌سریع کشور می‌باشد که پژوهشگاه ارتباطات و فناوری اطلاعات با همکاری دانشگاه صنعتی امیرکبیر طراحی و راه‌اندازی کرده‌اند.

۳) در کنار شرح جامع معیارهای مطرح منتخب برای ارزیابی منابع مختلف سامانه‌های پردازشی فوق‌سریع، به‌صورت جامع به نحوه راه‌اندازی معیارهای محک منتخب به‌صورت گام به گام و ارائه خروجی هریک پرداخته است.

در ادامه، در بخش دوم، پژوهش‌های مرتبط مرور خواهند شد. در بخش سوم، مراحل آماده‌سازی سامانه برای اجرای معیارها توضیح داده می‌شود. سپس در بخش‌های ۴ الی ۷، معیارهای انتخاب‌شده برای هر منبع، بررسی و مراحل اجرای هریک توضیح داده می‌شود. در بخش ۸ نتایج به‌دست‌آمده از اجرای این معیارها در یک سامانه پردازشی فوق‌سریع نمونه واقعی بررسی می‌شوند. در نهایت، در بخش ۹ جمع‌بندی کلی از نتایج این مقاله و معضلات پیش رو ارائه می‌شود.

۲- کارهای مرتبط

نسخه اولیه معیار LINPACK در سال ۱۹۸۷ با هدف نشان‌دادن زمان اجرای حل یک معادله خطی ارائه گردید [۱۸]. این پیاده‌سازی اولیه، مسئله‌ای با ماتریسی به ابعاد ۱۰۰ را حل می‌کرد [۱] و بر اساس ابزار جبر خطی معرفی‌شده در [۱۹] نوشته شده بود. به‌مرور زمان، معماری سامانه‌های HPC پیچیده‌تر شد و نیاز بود تا با معیارهای پیچیده‌تری، قدرت پردازنده یک سامانه HPC سنجیده شود. بر این اساس معیار HPL شکل گرفت که در واقع نسخه پیچیده‌تری از معیار LINPACK و با قابلیت پردازش موازی بود [۲۰].

از آن‌جا که قدرت یک سامانه HPC تنها محدود به قابلیت پردازشی آن نیست، نیاز بود تا سنجش سامانه HPC فقط محدود به سنجش پردازنده نباشد و منابع دیگر نیز سنجیده شود. برای نمونه سه منبع پردازنده، حافظه و شبکه در [۱۳] درنظر گرفته شده است. در نتیجه برای سنجش دقیق، یک سامانه HPC، به تفکیک معیارهایی نظیر [۲۱]، [۲۲] و [۲۳] معرفی شده‌اند. همچنین تلاش برای رتبه‌بندی آنها [۲۴]، کمک شایانی به بهبود معیارهای مختلف کرده است [۲۵].

از طرفی با ورود معماری‌های جدید پردازنده به دنیای سامانه‌های HPC، نیاز برای سنجش سامانه HPC بر اساس معماری پردازنده نیز دیده شده است. در نتیجه می‌توان سامانه HPC را بر اساس معماری پردازنده انجام داد. بیشتر معیارهای موجود، به‌طور پیش‌فرض معماری پردازنده را X86 درنظر می‌گیرند اما برای نمونه می‌توان به معیارهایی برای سنجش سامانه‌هایی که بر پایه معماری آرم[1] طراحی شده‌اند اشاره کرد [۲۶]. امروزه ارزیابی سامانه‌های HPC را می‌توان از منظر نوع نیازمندی و هدف آنها نیز سنجید. برای نمونه در HPC-AI-500 [۲۷] سامانه‌های HPC که با هدف یادگیری عمیق[2] ساخته شده‌اند مورد سنجش قرار می‌گیرند. همچنین در [28] ابزارها و الگوریتم‌های مختلفی برای سنجش کارایی این سامانه‌ها بررسی شده است. در این مقاله، نیازمندی‌هایی که برای استفاده واقعی در سامانه‌های HPC وجود دارد شبیه‌سازی شده‌اند و بر اساس این نیازمندی‌ها، ابزارهای سنجش سامانه‌های HPC بررسی می‌شوند. در برخی از پژوهش‌ها نیز به بررسی کارایی سامانه‌هایی که به شکل شبکه‌ای طراحی شده‌اند پرداخته شده است. برای نمونه در [29] ابزارهایی برای سنجش کارایی این مدل سامانه‌های HPC مطرح شده است و هرکدام بر اساس منابع مختلف بررسی شده‌اند. در جدول ۲ خلاصه کارهای مرتبط و تمرکز هریک بر اساس اینکه تمرکز روی قدرت پردازشی، سایر منابع نظیر پردازنده، حافظه و شبکه است یا متمرکز بر معیارهای محک و ارائه آزمون‌هایی برای کاربردهای پراستفاده نظیر یادگیری ماشین و عمیق، هواشناسی و ... می‌باشد، ارائه شده است.

جدول ۲. دسته‌بندی کارهای مرتبط

	منبع	شرح مختصر
قدرت پردازش و سایر منابع سامانه پردازشی فوق‌سریع	[۱] [۱۸] [۲۰]	LINPACK
	[۱۹]	برنامه فرترن و اسمبلی برای سیستم‌های آی بی ام
	[۲۱]	RandomAccess
	[۲۳]	مجموعه ابزارهای HPC challenge
	[۲۴]	TOP500 (LINPACK)
	[۲۵]	بهبود قابلیت موازی‌سازی در OpenMPI
	[۲۶]	بررسی استفاده از معماری ARM در HPC
	[۲۹]	آزمون‌های محک شبکه‌های تورین
کاربرد محور	[۲۲]	معیار HPCG با تمرکز بر کاربرد
	[۲۷]	یادگیری ماشین و هوش مصنوعی

1 ARM

2 Deep Learning

[۲۸]	بررسی ابزارهای محک برای برنامه‌های کاربردی سامانه‌های پردازشی فوق‌سریع
[۴۳]	بررسی ابزارهای محک برای برنامه‌های کاربردی سامانه‌های پردازشی فوق‌سریع نظیر کاربردهای مولکولی، آب‌وهوا و غیره

نوآوری اصلی این مقاله نسبت به مقالات مرتبط، این است که علاوه بر درنظرگرفتن معیارهای محک اصلی برای منابع پردازشی (شامل CPU و GPU)، معیارهای محک مطرح برای حافظه، شبکه و دیسک نیز به‌صورت جامع و یکپارچه معرفی شده و نحوه نصب و راه‌اندازی هریک به همراه تحلیل نتایج ارائه شده است. همچنین مقادیر خروجی علاوه بر سامانه آزمون، بر روی یک سامانه پردازشی فوق‌سریع واقعی نیز ارائه شده است.

۳- آماده‌سازی محیط آزمون

در این بخش از مقاله به شرح نحوه آماده‌سازی محیط سامانه پردازشی فوق‌سریع به‌منظور اجرای ابزارهای موردنیاز برای آزمون و ارزیابی معیارهای مختلف می‌پردازیم. به‌منظور راه‌اندازی و پیاده‌سازی معیارهای موردنظر، نیاز به محیطی است تا بتوان معیارها را در آن بستر اجرا کرد. محیط موردنظر و اقدامات موردنیاز که در واقع همان نیازمندی‌های موردنظر برای آزمون است، شامل موارد زیر هستند:

- سیستم عامل CentOS نسخه ۷
- افزودن مخزن نرم‌افزاری EPEL به سیستم عامل
- دسترسی سیستم عامل به اینترنت برای دانلود ابزارها
- اتصال سیستم‌ها با هم بر بستر شبکه Ethernet
- نصب و پیکربندی OpenMPI در سیستم‌ها
- نصب و پیکربندی OpenBLAS در سیستم‌ها
- تعریف متغیرهای محیطی موردنیاز
- نصب ابزار برای دانلود معیارها

در ادامه هریک از موارد فوق شرح داده خواهد شد.

۳-۱- سیستم‌عامل CentOS 7

به دلیل محبوبیت و استفاده زیاد از سیستم عامل CentOS در سامانه‌های HPC مختلف، در این مقاله به‌منظور برپاسازی محیط آزمون و ارزیابی از سیستم عامل CentOS[1] نسخه ۷ و در نتیجه برای محیط این توزیع از دستورات ردهت استفاده شده است. انتخاب سیستم‌عامل یا توزیع موردنظر بسته به اهداف آزمون صورت می‌پذیرد و بنابر سیستم‌عامل انتخابی، پیکربندی‌ها نیز باید مختص آن انجام پذیرد.

۳-۲- مخزن نرم‌افزاری EPEL

مخزن EPEL[2] در واقع بسته‌های اضافی نرم‌افزاری در لینوکس هستند. این مخزن یک پروژه متن‌باز و یک مخزن قوی از شرکت مشهور فدورا[3] می‌باشد که سازنده بسته‌های نرم‌افزاری برای سیستم عامل لینوکس است. برای افزودن این مخزن نرم‌افزاری، دستور زیر را در محیط خط فرمان سیستم عامل اجرا می‌کنیم.

```
yum install -y epel-release
```

۳-۳- نصب و پیکربندی OpenMPI

بستر MPI[4] یک سیستم تبادل پیام قابل‌حمل استاندارد است. این سیستم توسط گروهی از محققان آکادمیک و صنعتی طراحی شده است تا بر روی گستره وسیعی از کامپیوترهای موازی مختلف کار کند. در واقع می‌توان گفت که MPI یک زبان با پروتکل ارتباطی مستقل است که در برنامه‌هایی که از کامپیوترهای موازی استفاده می‌کنند، استفاده می‌شود [۳۰]. این روش، مجموعه‌ای از ارتباطات را پشتیبانی می‌کند. بسیاری از معیارهای آزمون سامانه‌های HPC نیاز به بستری برای انتقال پیام دارند که از ابزار OpenMPI[31] برای انتقال اطلاعات استفاده می‌کنند. برای نصب ابزار OpenMPI دستور زیر را اجرا می‌کنیم.

```
yum install -y openmpi openmpi-devel
```

۳-۴- تعریف متغیرهای محیطی موردنیاز

برای اجرای ابزار OpenMPI نیاز است تا متغیرهای محیطی زیر در سیستم تعریف شوند. برای این کار فایلی با نام benchmark.sh در آدرس /etc/profile.d/ ایجاد می‌کنیم و دستورات زیر را در فایل می‌نویسیم.

```
PATH=$PATH:/usr/lib64/openmpi/bin
LD_LIBRARY_PATH="/usr/lib64/openmpi/lib/libmpi.so:$LD
_LIBRARY_PATH"
LD_LIBRARY_PATH="/usr/lib64/libopenblas.so:$LD_LIBRAR
Y_PATH"

export PATH
export LD_LIBRARY_PATH
```

همچنین برای اجرای برنامه‌ها بر بستر OpenMPI نیاز است تا گره‌ها به یکدیگر از طریق پروتکل ssh دسترسی داشته باشند. برای این کار ابتدا با دستور زیر کلید درست می‌کنیم.

```
ssh-keygen
```

سپس کلید ایجادشده را با استفاده از دستور زیر به گره‌های دیگر انتقال می‌دهیم. کپی کلید ایجادشده برای یک گره در گره‌های دیگر، باید به‌ازای تمام گره‌ها انجام شود.

```
ssh-copy-id <ANOTHER_NODE_IP>
```

۳-۵- نصب و پیکربندی OpenBLAS

معیار HPL برای اجراشدن، نیاز به کتابخانه OpenBLAS دارد که آن را با دستور زیر نصب می‌کنیم.

```
yum install -y openblas openblas-devel
```

1. https://www.centos.org
2. Extra Packages for Enterprise Linux
3. https://getfedora.org
4. Message Passing Interface

۳-۶- نصب wget

برای سهولت دانلود معیارهای مختلف، ابزار wget را به‌صورت زیر نصب می‌کنیم.

```
yum install -y wget
```

۳-۷- غیرفعالسازی دیواره‌آتش

از آن‌جا که معیارهای ارزیابی برای انتقال اطلاعات بین گره‌ها از بستر شبکه استفاده می‌کنند، فعال‌بودن دیواره‌آتش در گره‌ها باعث اختلال در ارتباط بین گره‌ها می‌شود. برای غیرفعالسازی دیواره‌آتش در سیستم باید دستورات زیر اجرا شود. باید توجه کرد که پس از اتمام آزمون، ضروری است تنظیمات مربوط به دیواره آتش به حالت اولیه بازگردانده شود.

```
systemctl stop firewalld
systemctl disable firewalld
```

در ادامه این مقاله به بررسی معیارهای موردنیاز برای سنجش هر منبع می‌پردازیم.

۴- منبع پردازشی

برای سنجش منابع پردازشی سامانه‌های HPC از معیارهای مختلفی استفاده می‌شود. هرکدام از این معیارها با استفاده از الگوریتم و راه‌حل خاص خود قدرت پردازشی این سامانه‌ها را می‌سنجند. منبع پردازشی یک سامانه HPC، شامل گره‌های مختلف پردازشی در سرورهای مختلف می‌باشد که کارکرد و معماری هریک از آنها می‌توانند متفاوت باشند. برای مثال یک سامانه HPC می‌تواند فقط از قدرت پردازشی CPU برای محاسبات استفاده کند که در این نوع پردازش، قدرت پردازشی هر هسته بسیار بالا است ولی تعداد هسته‌های هر CPU محدود می‌باشد. حال اگر از قدرت پردازشی GPU نیز در کنار CPU استفاده کند، می‌تواند از تعداد هسته‌های بسیار بالایی برای پردازش موازی استفاده کند [۳۲]. معیار موردنظر باید بتواند برای حل مسئله، از تمامی گره‌های پردازشی به‌صورت موازی استفاده کند. در ادامه به بررسی، نحوه آماده‌سازی و اجرای دو معیار سنجش قدرت پردازش در یک سامانه HPC می‌پردازیم.

۴-۱- معیار HPL

معیار LINPACK، معیاری برای سنجش قدرت محاسبه اعشاری یک سیستم است. این معیار را جک دونگارا در سال ۱۹۷۹ معرفی کرده است. در معیار LINPACK برای سنجش قدرت پردازشی سیستم، از حل یک معادله جبر خطی استفاده می‌شود [۳۳]. از نسخه نهایی این معیار در سایت TOP500 برای ساخت لیست قدرتمندترین ابررایانه‌های جهان استفاده می‌شود [۱۷].

پیاده‌سازی‌های اولیه معیار LINPACK قابلیت پردازش موازی نداشتند. به همین دلیل تلاش برای پیاده‌سازی معیار LINPACK به‌صورت پردازش موازی منجر به پیاده‌سازی HPLinpack گردید که به اختصار به پیاده‌سازی HPL معروف است. در این پیاده‌سازی که با زبان C انجام شده است، برای حل دستگاه معادله خطی از روش تجزیه ال یو[1] [17] استفاده می‌شود که قابلیت موازی‌سازی نیز دارد. پیاده‌سازی HPL برای انتقال پیام بین گره‌های مختلف از MPI استفاده می‌کند که نیاز است بر روی سیستم‌هایی که آزمون روی آن‌ها انجام می‌شود نصب باشد. همچنین مقدار عمق ماتریس ثابت نیست و می‌توان آن را به‌عنوان پارامتر ورودی، تغییر داد. این پیاده‌سازی بعد از هر اجرا موارد زیر را گزارش می‌دهد:

- Rmax: کارایی سیستم برای بزرگ‌ترین مسئله‌ای که LINPACK توانسته است روی یک سیستم آن را حل کند. این مقدار با GFLOPS گزارش می‌شود.
- Nmax: اندازه بزرگ‌ترین مسئله‌ای که LINPACK توانسته آن را حل کند.
- N1/2: اندازه مسئله‌ای که در نصف زمان Rmax توسط سیستم حل شده است.
- Rpeak: این مقدار حداکثر توان تئوری سیستم را نشان می‌دهد.

در ادامه، دستورات نصب و پیکربندی HPL ارائه خواهد شد. ابتدا با دستور زیر HPL را دانلود می‌کنیم.

```
wget 'https://www.netlib.org/benchmark/hpl/hpl-2.3.tar.gz'
```

پس از آن فایل دانلود شده را با دستور زیر از حالت فشرده خارج می‌کنیم، نام پوشه جدید را تغییر می‌دهیم و به پوشه ایجادشده می‌رویم.

```
tar xf hpl-2.3.tar.gz
mv hpl-2.3 hpl
cd hpl
```

سپس فایل دستورالعمل ساخت HPL برای ماشین Linux را با دستور زیر در پوشه‌ای که در آن هستیم، کپی می‌کنیم.

```
cp setup/Make.Linux_PII_CBLAS Make.linux64
```

در ادامه، در فایل کپی‌شده با نام Make.linux64، تغییرات زیر را در خصوص مشخص‌شده اعمال می‌کنیم.

```
line 64: Linux_PII_CBLAS → linux64
line 84: /usr/local/mpi → /usr/lib64/openmpi
line 85: -I$(MPdir)/include → -I/usr/include/openmpi-x86_64
line 86: $(MPdir)/lib/libmpich.a → $(MPdir)/lib/libmpi.so
line 95: $(HOME)/netlib/ARCHIVES/Linux_PII → /usr/lib64
line 96: → -I/usr/include/openblas
line 97: $(LAdir)/libcblas.a $(LAdir)/libatlas.a → $(LAdir)/libopenblas.so
line 169: /usr/bin/gcc → /usr/lib64/openmpi/bin/mpicc
line 176: /usr/bin/g77 → /usr/lib64/openmpi/bin/mpicc
```

بعد از آن در همان مسیر کنونی، با استفاده از دستور زیر فایل اجرایی HPL را می‌سازیم.

[1] LU decomposition with partial row pivoting

```
make arch=linux64
```

سپس به‌صورت زیر به پوشه ساخته‌شده می‌رویم و HPL را در گره‌های مختلف اجرا می‌کنیم.

```
cd bin/linux64/
mpirun --allow-run-as-root --host
<NODE_1_IP>,<NODE_2_IP>,<NODE_3_IP> ./xhpl
```

در دستور فوق، باید مقادیر نمونه NODE_1_IP با نام و آی‌پی هر نود جایگزین گردند. در معیار HPL برای پیکربندی از فایل HPL.dat استفاده می‌شود. پس از ساخت معیار HPL این فایل با مقادیر مشخصی ساخته می‌شود که می‌توان آن را تغییر داد و سفارشی کرد. برای تغییر این پارامترها نیاز به درک دقیقی از الگوریتم و محاسبات ریاضی این معیار در هنگام اجرا وجود دارد. به‌منظور بررسی بیشتر مفاهیم ریاضی مرتبط و پارامترها می‌توان به [34] مراجعه کرد. همچنین از [35] می‌توان برای محاسبه پارامترها بر اساس اطلاعات سامانه HPC استفاده کرد. ابزار معرفی‌شده در [35] بر اساس تعداد گره‌های موجود، تعداد پردازنده و حافظه هریک از گره‌ها، پارامترها را طوری تعیین می‌کند تا بهترین نتیجه حاصل شود. چند نمونه از پارامترهای قابل تنظیم در HPL عبارت است از:

- تعداد مسئله‌هایی که در هر اجرا حل می‌شود (مقدار پیش‌فرض: ۳؛ حداکثر تعداد ممکن: ۲۰).
- ابعاد هریک از مسائلی که حل می‌شود (مقدار پیش‌فرض: ۳۰۰۰، ۶۰۰۰ و ۱۰۰۰۰).
- تعداد بلاک‌هایی که ماتریس به آن تقسیم می‌شود (مقدار پیش‌فرض: ۵).
- تعداد شبکه‌های پردازشی که برای حل این مسائل در نظر گرفته می‌شود (مقدار پیش‌فرض: ۲).

این معیار بعد از اجرا در سامانه آزمون، خروجی مانند شکل ۱ ارائه می‌دهد:

```
--------------------------------------------------------------------------------
||Ax-b||_oo/(eps*(||A||_oo*||x||_oo+||b||_oo)*N)=   0.00000000e+00 ..... PASSED
================================================================================
T/V                N    NB     P     Q               Time                 Gflops
--------------------------------------------------------------------------------
WR00L2C4           1     1     2     2               0.00             3.8746e-04
HPL_pdgesv() start time Wed Oct 14 02:18:59 2020

HPL_pdgesv() end time   Wed Oct 14 02:18:59 2020

--------------------------------------------------------------------------------
||Ax-b||_oo/(eps*(||A||_oo*||x||_oo+||b||_oo)*N)=   0.00000000e+00 ..... PASSED
================================================================================
T/V                N    NB     P     Q               Time                 Gflops
--------------------------------------------------------------------------------
WR00L2R2           1     1     2     2               0.00             3.9777e-04
HPL_pdgesv() start time Wed Oct 14 02:18:59 2020

HPL_pdgesv() end time   Wed Oct 14 02:18:59 2020
```

شکل ۱. نتایج آزمون HPL برای سامانه آزمون

همان‌طور که در تصویر شکل ۱ مشخص است مقدار قدرت پردازشی به‌ازای آزمون‌های مختلف بر مبنای GFlops در خروجی آمده است. خروجی نهایی برای سامانه HPC بهترین نتیجه‌ای است که این سامانه توانسته از آزمون‌های مختلف به‌دست بیاورد [۱۷]. برای نمونه اگر بیان شود قدرت پردازشی یک سامانه HPC برابر۵۰۰ ترافلاپس است، انتظار می‌رود عددی در محدوده ۵۰۰ ترافلاپس - بسته به توان واقعی آن- در نتیجه آزمون مشاهده شود. مقداری که از این طریق به‌دست می‌آید مقدار Rmax می‌باشد که سامانه به آن رسیده است.

علاوه بر مقدار Rmax که نشان‌دهنده قدرت واقعی سامانه HPC در محیط آزمایشگاه است و از طریق بنچمارک HPL به‌دست می‌آید، مقدار تئوری قدرت پردازشی یک سامانه HPC را نیز می‌توان محاسبه کرد. این مقدار، حداکثر توانی است که یک سامانه HPC در شرایط ایده‌ئال می‌تواند به آن برسد که Rpeak نامیده می‌شود و از طریق محاسبات ریاضی بر اساس تعداد و مدل و فرکانس پردازنده سامانه محاسبه می‌شود. برای محاسبه Rpeak از فرمول زیر استفاده می‌کنیم:

Rpeak = (Num of Cores * Processor Frequency * IPC)[36], [37]

در این معادله، IPC حداکثر تعداد عملیاتی است که هر هسته می‌تواند در یک چرخه[1] انجام دهد. برای مثال برای یک پردازنده اینتل با دو هسته که با فرکانس 1.5 GHz کار می‌کنند و مقدار IPC هر هسته برابر با ۴ است، مقدار Rpeak برابر با 12 Gflops است.

۲-۴- معیار DGEMM

معیار DGEMM همانند معیار LINPACK، برای سنجش قدرت پردازشی یک سیستم استفاده می‌شود. تفاوتی که این معیار با معیار LINPACK دارد در مسئله‌ای است که حل می‌کند. این معیار برای سنجش قدرت پردازشی یک سیستم از مسئله ضرب دو ماتریس استفاده می‌کند. این معیار در ابتدا دو ماتریس با اعداد تصادفی تولید می‌کند که عناصر ماتریس‌ها اعداد اعشاری با دقت مضاعف[2] هستند. سپس ضرب این دو ماتریس را محاسبه می‌کند و بر اساس مدت‌زمان طی‌شده، مقدار قدرت پردازشی سیستم را می‌سنجد. خروجی قدرت پردازشی سیستم بر اساس GFlops در این آزمون ارائه می‌شود. خروجی این معیار در سامانه آزمون در شکل ۲ ارائه شده است.

```
Begin of StarDGEMM section.
Scaled residual: 0.0135957
Node(s) with error 0
Minimum Gflop/s 32.598800
Average Gflop/s 33.860561
Maximum Gflop/s 34.602613
Current time (1619171436) is Fri Apr 23 14:20:36 2021

End of StarDGEMM section.
```

شکل ۲. نتایج آزمون DGEMM برای سامانه آزمون

همان‌طور که در شکل ۲ مشاهده می‌شود، معیار DGEMM سامانه تحت آزمون را چندین‌بار با داده‌های مختلف مورد سنجش قرار می‌دهد و در خروجی سه عدد را به‌عنوان بدترین، بهترین و میانگین قدرت پردازنده به‌عنوان نتیجه نشان می‌دهد.

با توجه به اینکه برای سنجش قدرت پردازشی موازی بین گره‌های مختلف، عواملی مانند شبکه ارتباطی بین گره‌ها، کارکرد ضعیف‌تر پردازنده‌ها در دمای بالا و عوامل سخت‌افزاری دیگری دخیل هستند، اگر تعداد گره‌های سامانه HPC دو برابر شود، معمولاً قدرت پردازشی آن افزایش صد درصدی نخواهد داشت. با این وجود نسبت به عوامل بیان‌شده قدرت پردازشی افزایش قابل قبولی خواهد داشت.

۵- حافظه

1 Cycle

2 Double-precision

پهنای باند و سرعت خواندن و نوشتن حافظه، پارامتر مهمی برای رتبه‌بندی یک سامانه HPC محسوب می‌شود. در محاسبات پردازشی با حجم بالایی از داده، سامانه HPC نیاز دارد تا محتوای موردپردازش را در حافظه ذخیره کند و سرعت بالای حافظه موجب افزایش سرعت کلی سامانه HPC می‌شود. معیارهای مهم برای سنجش حافظه این سامانه‌ها در ادامه شرح داده شده است.

۵-۱- معیار STREAM

معیار STREAM برای سنجش پهنای باند حافظه توسعه شده است. در این معیار، دادگان تستی ایجاد می‌شوند که حجم آن‌ها از مقدار حافظه نهان[1] پردازنده بسیار بیشتر می‌باشد و عملیات مختلفی بر روی این مجموعه دادگان انجام می‌شود. با توجه به حجم داده، به اجبار نیاز است که جابه‌جایی داده بین حافظه نهان و حافظه صورت پذیرد. در نهایت میزان پهنای باند حافظه بر اساس MB/sec گزارش می‌شود.

مجموعه دادگان آزمون ساخته‌شده برای سنجش پهنای باند باید دو شرط زیر را دارا باشند:

- اندازه هر مجموعه داده آزمون باید ۴ برابر اندازه آخرین لایه حافظه نهان پردازنده باشد.
- حداقل اندازه هر مجموعه داده آزمون باید یک میلیون رکورد باشد.

این معیار برای سنجش پهنای باند حافظه از عملیات زیر استفاده می‌کند:

COPY: a(i) = b(i)
SCALE: a(i) = q*b(i)
SUM: a(i) = b(i) + c(i)
TRIAD: a(i) = b(i) + q*c(i)

در ادامه، مراحل ساخت و اجرای این معیار شرح داده خواهد شد. در سرور موردنظر پوشه STREAM را می‌سازیم و مسیر را به پوشه جدید تغییر می‌دهیم.

```
mkdir STREAM
cd STREAM
```

با استفاده از دستور wget دو فایل موردنیاز را از آدرس‌های مشخص‌شده دانلود می‌کنیم.

```
wget
'https://www.cs.virginia.edu/stream/FTP/Code/stream.c'
wget
'https://www.cs.virginia.edu/stream/FTP/Code/mysecond.c
'
```

با استفاده از دستورات زیر، فایل اجرایی ابزار آزمون را می‌سازیم و اجرا می‌کنیم.

```
gcc -O stream.c -o stream
./stream
```

```
-------------------------------------------------------------
Function    Best Rate MB/s  Avg time     Min time     Max time
Copy:           17580.2     0.009564     0.009101     0.011986
Scale:          17041.4     0.009667     0.009389     0.009911
Add:            18069.5     0.013820     0.013282     0.015275
Triad:          19209.1     0.013019     0.012494     0.016010
-------------------------------------------------------------
Solution Validates: avg error less than 1.000000e-13 on all three arrays
-------------------------------------------------------------
```

شکل ۳. نتایج آزمون STREAM برای سامانه آزمون

خروجی این معیار برای سامانه آزمون پس از اجرا در شکل ۳ ارائه شده است. همان‌طور که در تصویر خروجی شکل ۳ مشخص است این معیار، سرعت چهار آزمون مختلف را بر اساس MB/s نشان می‌دهد. خروجی نهایی این آزمون برای سامانه HPC مقدار Triad می‌باشد [۱۰].

۵-۲- آزمون RandomAccess

معیار مهم دیگر برای سنجش حافظه، سرعت دسترسی تصادفی در حافظه می‌باشد. در دسترسی تصادفی، سرعت دسترسی به یک خانه حافظه باید ثابت و بدون وابستگی به خانه حافظه قبلی باشد. همچنین به‌روزرسانی یک خانه حافظه باید با سرعت بالایی انجام شود تا پردازنده بتواند به حداکثر کارایی خود برسد [۴۱]. این معیار برای سنجش سرعت دسترسی تصادفی در حافظه مراحل زیر را انجام می‌دهد:

- ابتدا یک آدرس حافظه به‌صورت تصادفی تولید می‌شود.
- مقدار موجود در خانه حافظه تولیدشده خوانده می‌شود و به پردازنده انتقال داده می‌شود.
- مقدار خوانده‌شده در پردازنده به‌روزرسانی می‌شود. عملیات ریاضی برای به‌روزرسانی عبارتند از ADD، AND، OR و XOR.
- مقدار جدید در همان خانه حافظه نوشته می‌شود.
- در نهایت این معیار خروجی سنجش را با واحد GUPS[2] گزارش می‌دهد.

واحد GUPS نشان می‌دهد سیستم در یک ثانیه می‌تواند چند خانه حافظه تصادفی را به‌روزرسانی کند. در شکل ۴ نمونه‌ای از خروجی این معیار ارائه شده است. سرعت دسترسی به خانه‌های حافظه، بستگی به سرعت هریک از حافظه‌های موجود در سامانه دارد. بنابراین اضافه‌کردن حافظه جدید، تأثیر ثابتی در افزایش سرعت حافظه ندارد و سرعت کلی وابسته به سرعت حافظه‌هایی با کمترین سرعت است. همچنین تعداد محل‌های قرارگیری کارت حافظه، محدود است و نمی‌توان این تعداد را تا مقدار دلخواه اضافه کرد.

```
Begin of SingleRandomAccess section.
Node(s) with error 0
Node selected 2
Single GUP/s 0.510185
Current time (1619171434) is Fri Apr 23 14:20:34 2021

End of SingleRandomAccess section.
```

شکل ۴. نتایج آزمون RandomAccess برای سامانه آزمون

۶- شبکه

هر سامانه HPC شامل مجموعه‌ای از گره‌های مختلف است که در کنار یکدیگر می‌توانند یک مسئله محاسباتی را حل کنند. گره‌های سامانه برای انتقال اطلاعات بین یکدیگر از شبکه ارتباطی پیاده‌سازی‌شده در محیط سامانه HPC استفاده می‌کنند و اگر این بستر، پهنای باند و سرعت بالایی نداشته باشد، باعث کاهش کارایی کلی سیستم می‌شود. این کاهش کارایی در محاسباتی که به تبادل داده زیادی نیاز دارد بسیار چشم‌گیر خواهد بود [13]. در ادامه به بررسی معیارهای مهم برای سنجش شبکه ارتباطی در سامانه‌های HPC می‌پردازیم.

1 Cache

2 Giga Updates Per Second

۶-۱- آزمون PTRANS

انتقال اطلاعات بین هسته‌های مختلف می‌تواند در بین هسته‌های یک پردازنده باشد یا بین پردازنده‌های مختلفی که در گره‌های مختلف یک کلاستر قرار دارند بنابراین نیاز به معیاری است تا سرعت انتقال اطلاعات بین هسته‌های مختلف، پردازنده‌ها را در شرایط شامل حالت‌های زیر بسنجد:

- انتقال اطلاعات بین هسته‌های مختلف یک پردازنده
- انتقال اطلاعات بین هسته‌های مختلف دو پردازنده در یک گره
- انتقال اطلاعات بین هسته‌های مختلف دو پردازنده در دو گره مختلف یک کلاستر.

معیار PTRANS برای سنجش سرعت انتقال اطلاعات، یک ماتریس با ابعاد بزرگ تولید می‌کند و سپس با محاسبه ترانهاده آن ماتریس که نیاز به انتقال اطلاعات بین هسته‌های مختلف دارد، سرعت انتقال اطلاعات را می‌سنجد.

در شکل ۳ نمونه‌ای از خروجی قابل‌مشاهده برای سامانه آزمون ارائه شده است. این معیار ۵ بار عملیات آزمون را در سامانه اجرا می‌کند و به ازای هر اجرا، دو سطر به‌عنوان زمان کلی اجرا[1] و زمان پردازنده[2] در خروجی نشان می‌دهد. نتیجه خروجی را می‌توان در غالب حجم اطلاعات انتقال داده شده در هر ثانیه بر اساس GB/s مشاهده کرد.

```
TIME   M     N    MB  NB  P   Q     TIME   CHECK   GB/s   RESID
---- ----- ----- --- --- --- --- -------- ------ -------- -----
WALL   500   500  80  80   2   1     0.01 PASSED    0.357  0.00
CPU    500   500  80  80   2   1     0.00 PASSED    0.408  0.00
WALL   500   500  80  80   2   1     0.00 PASSED    0.357  0.00
CPU    500   500  80  80   2   1     0.00 PASSED    1.346  0.00
WALL   500   500  80  80   2   1     0.00 PASSED    0.357  0.00
CPU    500   500  80  80   2   1     0.00 PASSED    1.258  0.00
WALL   500   500  80  80   2   1     0.00 PASSED    0.357  0.00
CPU    500   500  80  80   2   1     0.00 PASSED    2.587  0.00
WALL   500   500  80  80   2   1     0.00 PASSED    0.357  0.00
CPU    500   500  80  80   2   1     0.00 PASSED    1.735  0.00

Finished    5 tests, with the following results:
    5 tests completed and passed residual checks.
    0 tests completed and failed residual checks.
    0 tests skipped because of illegal input values.
```

شکل ۳. نتایج آزمون PTRANS برای سامانه آزمون

۶-۲- معیار b_eff

معیار b_eff شامل مجموعه‌ای از آزمون‌ها برای سنجش تأخیر و پهنای باند در ارتباطات شبکه است. این معیار برای انتقال اطلاعات از الگوها و پیام‌هایی با اندازه‌های مختلف استفاده می‌کند. در ادامه، مراحل ساخت و اجرای این معیار شرح داده خواهد شد. در ابتدا در سرور موردنظر، پوشه b_eff را می‌سازیم و مسیر را به پوشه جدید تغییر می‌دهیم.

```
mkdir b_eff
cd b_eff
```

با استفاده از دستور wget فایل موردنظر را دانلود می‌کنیم.

```
wget 'https://fs.hlrs.de/projects/par/mpi//b_eff/b_eff.c'
```

در ادامه، با استفاده از دستور زیر فایل اجرایی را ساخته و اجرا می‌کنیم.

```
mpicc -O b_eff.c -DMEMORY_PER_PROCESSOR=1024 -o b_eff –lm
mpirun -n 4 –allow-run-as-root ./b_eff
```

مقدار خروجی برای این معیار پس از اجرا روی سامانه آزمون ۲.۹۶۹ MB/s می‌باشد. این معیار سرعت ارتباط بین گره‌های مختلف را بر اساس MB/s نشان می‌دهد. این معیار به نسبت معیار PTRANS اطلاعات کمتری ارائه می‌کند. در واقع معیار PTRANS، آزمون را پنج بار اجرا می‌کند و برای هر اجرا زمان دقیق CPU، WALL-clock و سرعت انتقال اطلاعات شبکه را در خروجی نمایش می‌دهد در حالی که معیار b_eff فقط یک عدد که نشان‌دهنده سرعت ارتباط بین گره‌ها است را به‌عنوان خروجی نمایش می‌دهد.

در بخش ارتباطات، انتخاب ابزارهای شبکه مناسب همراه با طراحی و معماری دقیق و درست برای شبکه سامانه HPC بسیار مهم می‌باشد. تأثیر افزایش سرعت یک ابزار در شبکه، کاملاً بستگی به معماری آن شبکه و نقش آن ابزار در معماری دارد. برای مثال مسیریاب[3] مرکزی نقش بسیار مهمی در هدایت بسته‌های شبکه دارد که با افزایش سرعت آن، سرعت کلی شبکه افزایش مشهودی خواهد داشت.

۷- دیسک

سامانه‌های HPC که در حوزه کلان‌داده کار می‌کنند نیازمند فضای ذخیره‌سازی با حجم بسیار بالایی هستند. این فضای ذخیره‌سازی می‌تواند بین تمامی گره‌ها به‌صورت مشترک استفاده شود، یا هر گره فضای ذخیره‌سازی مختص خودش را داشته باشد یا از راهکارهای ترکیبی استفاده شود. با استفاده از معیار IOR می‌توان سرعت خواندن و نوشتن بر روی دیسک در حالت‌های اختصاصی و اشتراکی را سنجید.

۷-۱- معیار IOR

در بسیاری از مسائل کلان‌داده، حجم داده‌ای که قرار است پردازش شود بسیار بزرگ‌تر از حافظه سیستم‌ها می‌باشد و نیاز است تا این داده در دیسک ذخیره شود. بنابراین در پردازش‌های کلان‌داده نیاز است تا تعادلی بین سرعت پردازنده و دیسک‌ها برقرار باشد تا پردازنده زمان زیادی را منتظر خواندن اطلاعات از دیسک نباشد. معیار IOR سرعت نوشتن و خواندن از دیسک سخت را می‌سنجد. در این معیار، کارایی برای خواندن و نوشتن روی دیسک به‌صورت سریال و موازی سنجیده می‌شود. همچنین این معیار قابلیت سنجش فایل سیستم‌های مختلف از قبیل HDFS، S3، POSIX و غیره را دارد. این معیار برای انتقال اطلاعات، در حالی که فایل سیستم توزیع‌شده نباشد، از MPI استفاده می‌کند. در نتیجه نیاز است پیاده‌سازی از MPI بر روی گره‌های سیستم (تحت آزمون) نصب و پیکربندی شده باشد [۳۹]. مراحل ساخت و اجرای این معیار در ادامه شرح داده شده است.

ابتدا با دستور زیر IOR را دانلود می‌کنیم.

```
Wget 'https://github.com/hpc/ior/releases/download/3.3.0/ior-3.3.0.tar.gz'
```

1 Wall-clock
2 CPU-clock

3 Router

پس از آن فایل دانلودشده را با دستور زیر extract می‌کنیم، نام پوشه جدید را تغییر می‌دهیم و به پوشه ایجادشده می‌رویم.

```
tar xf ior-3.3.0.tar.gz
mv ior-3.3.0 ior
cd ior
```

با استفاده از دستورات زیر فایل اجرایی IOR را می‌سازیم.

```
./configure
Make
```

در انتها با دستور زیر IOR را اجرا می‌کنیم.

```
mpirun --allow-run-as-root --host
<NODE_1_IP>,<NODE_2_IP>,<NODE_3_IP> src/ior -t 1m -b
16m -s 16 -F
```

خروجی اجرای این معیار برای سامانه آزمون در شکل ۶ ارائه شده است.

```
Results:

access    bw(MiB/s)  IOPS       Latency(s)  block(KiB) xfer(KiB)  open(s)    wr/rd(s)   close(s)   total(s)   iter
------    ---------  ----       ----------  ---------- ---------  --------   --------   --------   --------   ----
write     66843      66887      0.024021    16384      1024.00    0.001973   0.428664   0.231959   0.428947   0
read      213003     213039     0.008259    16384      1024.00    0.000563   0.134586   0.047422   0.134609   0
remove    -          -          -           -          -          -          -          -          1.68       0
Max Write: 66842.70 MiB/sec (70089.65 MB/sec)
Max Read:  213002.73 MiB/sec (223349.55 MB/sec)
```

شکل ۴. نتایج آزمون IOR برای سامانه آزمون

همان‌طور که در شکل ۶ مشخص است این معیار، اطلاعات کاملی از پهنای باند[1] دیسک، تعداد عملیات در ثانیه[2]، مجموع تأخیر[3] عملیات، تعداد بلاک‌ها و فایل‌های بازشده به ازای خواندن و نوشتن ارائه می‌دهد. همچنین در انتها بیشترین مقدار سرعت نوشتن و خواندن در دیسک را بر اساس MiB/s نشان می‌دهد.

افزایش تعداد دیسک‌های مورداستفاده در سامانه HPC، باعث افزایش حجم حافظه موردددسترسی می‌شود اما این امر همیشه باعث افزایش سرعت دسترسی به دیسک نمی‌شود. افزایش سرعت دسترسی به دیسک وابسته به عوامل دیگری مانند نوع دیسک‌های استفاده‌شده و چگونگی پیاده‌سازی و پیکربندی دیسک‌ها در سامانه HPC است. برای مثال در معماری RAID افزایش دیسک می‌تواند سرعت دسترسی به اطلاعات را به مقدار قابل‌توجهی افزایش دهد.

۸- نتایج اجرا در یک سامانه HPC نمونه

سامانه‌های HPC بعد از پیاده‌سازی موفق، نیازمند ارزیابی دقیق منابع مختلف می‌باشند. در این ارزیابی‌ها معیارهای مختلفی که در این مقاله بررسی شدند، در سامانه HPC اجرا می‌شوند تا نتایج مورد انتظار به‌دست آید. در این فرایند، اگر مشکلی در سامانه HPC یا منابع آن وجود داشته باشد، مشخص می‌شود. به‌منظور نشان‌دادن اجرای معیارها در محیط‌های عملیاتی، این معیارها را علاوه بر سامانه آزمون -که یک محیط آزمایشی است- در یک سامانه HPC نمونه واقعی نیز اجرا کرده‌ایم. منابع موجود در هریک از محیط‌های آزمون و محیط واقعی در ادامه شرح داده شده است.

محیط آزمایشگاهی سامانه آزمون شامل دو ماشین مجازی است که هرکدام شامل ۲ هسته مجازی از پردازنده Intel i7-8565U و ۸ گیگابایت حافظه اصلی می‌باشند. این محیط فاقد پردازنده گرافیکی است. نتایج معیارهای مختلف برای سامانه آزمون به تفکیک در هر بخش ارائه گردید. در ادامه به ارائه نتایج برای سامانه HPC نمونه می‌پردازیم.

سامانه HPC نمونه تحت بررسی شامل ۱۰ گره پردازشی می‌باشد که هرکدام از این گره‌ها شامل دو پردازنده Intel 6248R و 325 GB حافظه می‌باشند. همچنین در این سامانه HPC از ۸ پردازنده گرافیکی NVIDIA A100 و ۶ پردازنده گرافیکی Nvidia RTX3090 استفاده شده است. با توجه به اینکه آزمون HPL مهم‌ترین معیار برای ارزیابی و رتبه‌بندی ابررایانه‌های جهان است، این معیار را به‌عنوان اولین معیار اجراشده بر روی سامانه HPC نمونه انتخاب کردیم. نتیجه پیکربندی و اجرای این معیار بر روی کلاستر با ۱۰ گره پردازشی سامانه HPC نمونه، برابر با 19.5 TFlops گردید. سپس معیار HPL را بر روی پردازنده‌های گرافیکی اجرا کردیم. نتیجه اجرای این معیار بر روی ۸ پردازنده گرافیکی NVIDIA A100 برابر با 82.8 TFlops و بر روی ۶ پردازنده گرافیکی NVIDIA RTX3090 برابر با 3.1 TFlops گردید. با نگاهی به رتبه‌بندی Top500[۴۰] مشاهده می‌شود که قدرت پردازشی ابررایانه اول این لیست برابر 442000 TFlops می‌باشد که تقریباً ۵۳۰۰ برابر سامانه HPC نمونه می‌باشد. ابررایانه دوم لیست Top500 با قدرت پردازشی 148000 TFlops نیز قدرتی معادل ۱۷۸۰ برابر سامانه HPC نمونه دارد.

پس از اجرای معیار HPL، سراغ معیار DGEMM رفتیم که این معیار هم برای سنجش قدرت پردازنده سامانه HPC استفاده می‌شود. پس از اجرای این معیار بر روی ۱۰ گره پردازشی، به نتیجه 8.89 GFlops رسیدیم.

سپس معیار STREAM را برای سنجش سرعت حافظه سامانه HPC اجرا کردیم. بر اساس نتیجه این معیار، سرعت حافظه استفاده‌شده در سامانه HPC نمونه برابر با 2.66 GB/s بوده است. معیار دیگر برای سنجش سرعت دسترسی به حافظه تصادفی در سامانه‌های فوق‌سریع، معیار RandomAccess می‌باشد که این معیار نیز بر روی سامانه HPC نمونه اجرا شد. پس از اجرای این معیار، نتیجه 0.1GUPs برای سرعت دسترسی به حافظه تصادفی در سامانه HPC نمونه ثبت شد. سپس برای سنجش پهنای باند شبکه ارتباطی در سامانه HPC نمونه، دو معیار PTRANS و b_eff را اجرا کردیم. نتایج به‌دست‌آمده برای این دو معیار به‌ترتیب برابر با 41.85 GB/s و 58 MB/s بوده است. در نهایت برای سنجش سرعت دیسک سامانه HPC نمونه، از معیار IOR استفاده کردیم. این معیار پس از نوشتن و خواندن فایل‌های با اندازه‌های مختلف در دیسک سامانه HPC نمونه، به حداکثر سرعت 1.3 GB/s برای نوشتن و 1.7 GB/s برای خواندن رسید. خلاصه نتایج به‌دست‌آمده از سامانه HPC نمونه در جدول ۳ آمده است. در این جدول، مقدار تئوری، بیشترین مقداری است که سامانه HPC از لحاظ تئوری می‌تواند به آن برسد[4] و مقدار عملی، مقداری است که سامانه HPC، در آزمون‌ها به آن رسیده است[5].

1 Bandwidth
2 IOPS (Input/output Operations Per Second)
3 Latency

4 peak
5 max

دلیل تفاوت میان مقدار تئوری و مقدار عملی، وابستگی‌های ارتباطی سخت‌افزاری و نرم‌افزاری می‌باشد که میان اجزای مختلف سامانه HPC وجود دارد. در تمامی سامانه‌های HPC موجود، به دلیل شرایط محیطی مانند گرمای محیط و نیز نیازمندی‌های ارتباطی بین گره‌های مختلف سامانه HPC، هیچ‌گاه نمی‌توان به حداکثر توان پردازشی که همان مقدار تئوری است رسید.

جدول ۳. نتایج به‌دست‌آمده از سامانه HPC نمونه

آزمون	مقدار تئوری	مقدار عملی	درصد کارایی
HPL (10* Intel 6248R)	46 TFlops	19.5 TFlops	٤٢
HPL (8* NVIDIA A100)	156 TFlops	82.8 TFlops	٥٣
HPL (6* NVIDIA RTX3090)	3.336 TFlops	3.1 TFlops	٩٢
DGEMM (10* Intel 6248R)	22.2 GFlops	8.89 GFlops	٤٠
STREAM	4 GB/s	2.66 GB/s	٦٦
RandomAccess	روشی برای محاسبه وجود ندارد	0.1GUPs	
PTRANS	100 GB/s	41.85 GB/s	٤١
b_eff	100 MB/s	58 MB/s	٥٨
IOR (Read)	2.5 GB/s	1.3 GB/s	٥٢
IOR (Write)	2 GB/s	1.7 GB/s	٨٥

جدول ۴. نتایج به‌دست‌آمده از محیط آزمایشگاهی

آزمون	مقدار تئوری	مقدار عملی	درصد کارایی
HPL (2* Intel i7-8565U)	7.1 GFlops	3.8 GFlops	٥٣
DGEMM (2* Intel i7-8565U)	3.7 MFlops	2.6 MFlops	٧٠
STREAM	2 GB/s	0.8 GB/s	٤٠
RandomAccess	روشی برای محاسبه وجود ندارد	0.04GUPs	
PTRANS	1 GB/s	0.7 GB/s	٧٠
b_eff	10 MB/s	4.5 MB/s	٤٥
IOR (Read)	1.5 GB/s	0.9 GB/s	٦٠
IOR (Write)	1 GB/s	0.8 GB/s	٨٠

۹- جمع‌بندی و نتیجه‌گیری

ما در این مقاله به بررسی معیارهای مهم در سنجش منابع مختلف یک سامانه HPC، نحوه راه‌اندازی، پیکربندی، اجرا و تحلیل معیارها پرداختیم. بدین منظور ارزیابی منابع سامانه HPC در چهاردسته پردازشی، حافظه، شبکه و دیسک درنظر گرفته شد. پس از شرح نحوه آماده‌سازی محیط برای اجرای معیارها، به بررسی معیارهای مطرح هریک از این چهار دسته و نحوه راه‌اندازی و اجرای آنها پرداخته شد. همچنین با راه‌اندازی یک سامانه کمینه، نتیجه اجرای هر معیار به تفکیک تحلیل گردید. در انتها به‌منظور نشان دادن کاربردپذیری و داشتن درکی از مقدار خروجی معیارها در عمل، تحلیل نتایج حاصل از معیارها برای یک نمونه واقعی سامانه HPC ارائه گردید.

با وجود اینکه معیارهای کلی برای سنجش منابع مختلف سامانه HPC وجود دارند، سنجش دقیق نیازمند شناخت کاملی از معماری سامانه HPC، منابع استفاده‌شده در آن و معیارهای موجود برای هر منبع است. طراحی و معماری کلی هر سامانه HPC می‌تواند نحوه اجرای معیارها برای سنجش قدرت یک سامانه HPC را تغییر دهد. برای مثال ساختار معماری اتصال گره‌های مختلف سامانه HPC باهم، نحوه تعریف هر گره (درنظرگرفتن سرور فیزیکی به‌عنوان یک گره[1] یا ایجاد چندین گره روی یک سخت‌افزار و استفاده از فناوری‌های مجازی‌سازی) و پیکربندی‌های مختلف باعث می‌شود تغییرات محسوسی در نتایج معیارها مشاهده شود. همچنین با توجه با اینکه ساختار سامانه‌های HPC در طول زمان، دچار تغییرات مختلفی شده است، نیاز است تا معیارهای سنجش آنها نیز تغییر کنند تا بتوانند سنجش دقیق و کاملی از این سامانه‌ها ارائه کنند. این تغییرات شامل حرکت نوع پردازش از تک‌هسته‌ای‌بودن به پردازش موازی، استفاده از واحد پردازش گرافیکی[2] و استفاده از معماری‌های مختلف پردازنده نظیر حافظه اختصاصی و حافظه مشترک برای هسته‌های مختلف می‌شود. در برخی از سامانه‌های HPC نیز از معماری اختصاصی در طراحی پردازنده استفاده شده است که ممکن است ساختار سنجش را به کلی تغییر دهد. بنابراین نمی‌توان از یک معیار و روش برای سنجش همه سامانه‌های HPC موجود استفاده کرد و نیاز است تا هر معیار بر اساس مشخصات و هدف ارزیابی و رتبه‌بندی مورد استفاده قرار گیرد. همچنین، با ظهور فناوری‌ها و روش‌های محاسباتی جدید نظیر رایانش کوانتومی، زیستی، نانو، حافظه‌ای و فوتونیک، سنجش و ارزیابی سامانه‌های فوق‌سریع نسل‌های آتی مستلزم طراحی و پیاده‌سازی معیارهای متناسب می‌باشد.

در انتها شایان ذکر است که برخی از معیارهای محک برای برنامه‌های پرکاربرد و معروف وجود دارند که از جمله می‌توان به NAMD و LAMMPS برای برنامه‌های مولکولی، LUXMark برای رندرینگ یا اجرای برنامه‌های مرتبط با پردازش تصویر یا آب‌وهوا برای یادگیری ماشین اشاره کرد. این آزمون‌ها عمدتاً به ازای کامپایلرهای مختلف (نظیر GNU، اینتل و GPI) یا پیکربندی‌های مختلف حافظه و پردازنده و غیره انجام می‌شوند. به‌صورت

1 Bare metal

2 GPU (Graphics Processing Unit)

کلی، بررسی‌های قبلی روی این دسته معیارها حاکی از آن است که کارایی با تغییر کامپایلر، سخت‌افزار و نوع کار تغییر می‌کند [۴۳]. با این وجود، بررسی این آزمون‌ها و ارتباط آنها با آزمون‌های محک عمومی‌تر، از مطالعات آتی درنظر گرفته‌شده در این پژوهش می‌باشد.

۱۰- مراجع

[1] J. J. Dongarra, P. Luszczek, and A. Petitet, "The LINPACK Benchmark: past, present and future," *Concurrency and Computation: Practice and Experience*, vol. 15, no. 9, pp. 803–820, Aug. 2003, doi: 10.1002/cpe.728.

[2] "HPL - A Portable Implementation of the High-Performance Linpack Benchmark for Distributed-Memory Computers." https://www.netlib.org/benchmark/hpl/ (accessed Sep. 26, 2021).

[3] P. Gepner, V. Gamayunov, and D. L. Fraser, "Effective Implementation of DGEMM on Modern Multicore CPU," *Procedia Computer Science*, vol. 9, pp. 126–135, 2012, doi: 10.1016/j.procs.2012.04.014.

[4] "NAS Parallel Benchmarks." https://www.nas.nasa.gov/software/npb.html (accessed Sep. 26, 2021).

[5] D. H. Bailey *et al.*, "The NAS parallel benchmarks---summary and preliminary results," in *Proceedings of the 1991 ACM/IEEE conference on Supercomputing - Supercomputing '91*, 1991, pp. 158–165. doi: 10.1145/125826.125925.

[6] ASC Community, "Supercomputer System Performance Evaluation Methods," in *The Student Supercomputer Challenge Guide*, Singapore: Springer Singapore, 2018, pp. 81–115. doi: 10.1007/978-981-10-3731-3_5.

[7] J. Ang, B. Barrett, K. Wheeler, and R. Murphy, "Introducing the graph 500," Sep. 2010.

[8] "Graph 500 | large-scale benchmarks." https://graph500.org/ (accessed Sep. 26, 2021).

[9] "HPL-AI Mixed-Precision Benchmark." https://hpl-ai.org/ (accessed Sep. 26, 2021).

[10] "MEMORY BANDWIDTH: STREAM BENCHMARK PERFORMANCE RESULTS."

https://www.cs.virginia.edu/stream/ (accessed Sep. 26, 2021).

[11] V. Aggarwal, Y. Sabharwal, R. Garg, and P. Heidelberger, "HPCC RandomAccess benchmark for next generation supercomputers," in *2009 IEEE International Symposium on Parallel & Distributed Processing*, May 2009, pp. 1–11. doi: 10.1109/IPDPS.2009.5161019.

[12] R. Rabenseifner and A. Koniges, "The Effective I/O Bandwidth Benchmark (b_eff_io)," Sep. 2000.

[13] P. Luszczek *et al.*, "Introduction to the HPC Challenge Benchmark Suite," Sep. 2004.

[14] S. Potluri, "The MVAPICH Project: Evolution and Sustainability of an Open Source Production Quality MPI Library for HPC." figshare, Sep. 2013. doi: 10.6084/m9.figshare.790743.v3.

[15] "IOR and mdtest." https://github.com/hpc/ior (accessed Sep. 26, 2021).

[16] "MDTest - Lustre Wiki." https://wiki.lustre.org/MDTest (accessed Sep. 26, 2021).

[17] "The Linpack Benchmark | TOP500." https://www.top500.org/project/linpack/ (accessed Sep. 26, 2021).

[18] J. J. Dongarra, "The LINPACK Benchmark: An explanation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 297 LNCS, pp. 456–474, 1987, doi: 10.1007/3-540-18991-2_27.

[19] LawsonC. L., HansonR. J., KincaidD. R., and KroghF. T., "Basic Linear Algebra Subprograms for Fortran Usage," *ACM Transactions on Mathematical Software (TOMS)*, vol. 5, no. 3, pp. 308–323, Sep. 1979, doi: 10.1145/355841.355847.

[20] R. F. Barrett, T. H. F. Chan, E. F. D'Azevedo, E. F. Jaeger, K. Wong, and R. Y. Wong, "Complex version of high performance computing LINPACK benchmark (HPL)," *Concurrency and Computation: Practice and Experience*, vol. 22, no. 5, pp. 573–587, Apr. 2010, doi: 10.1002/CPE.1476.

[21] P. Luszczek and J. Dongarra, "Analysis of various scalar, vector, and parallel implementations of RandomAccess *," 2010.

[22] J. Dongarra, M. A. Heroux, and P. Luszczek, “A new metric for ranking high-performance computing systems,” *National Science Review*, vol. 3, no. 1, pp. 30–35, Mar. 2016, doi: 10.1093/NSR/NWV084.

[23] J. Dongarra and P. Luszczek, “HPC challenge: Design, history, and implementation highlights,” *Contemporary High Performance Computing: From Petascale toward Exascale*, pp. 13–30, Jan. 2013, doi: 10.1201/9781351104005-2.

[24] H. W. Meuer, “The TOP500 Project. Looking Back over 15 Years of Supercomputing Experience,” vol. 31, no. 2, pp. 122–132, Jun. 2008, doi: 10.1515/PIKO.2008.0022.

[25] V. Turchenko, L. Grandinetti, G. Bosilca, and J. J. Dongarra, “Improvement of parallelization efficiency of batch pattern BP training algorithm using Open MPI,” *Procedia Computer Science*, vol. 1, no. 1, pp. 525–533, 2010, doi: 10.1016/J.PROCS.2010.04.056.

[26] D. Yokoyama, B. Schulze, F. Borges, and G. Mc Evoy, “The survey on ARM processors for HPC,” *The Journal of Supercomputing 2019 75:10*, vol. 75, no. 10, pp. 7003–7036, Jun. 2019, doi: 10.1007/S11227-019-02911-9.

[27] Z. Jiang *et al.*, “HPC AI500: A Benchmark Suite for HPC AI Systems,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11459 LNCS, pp. 10–22, Dec. 2018, doi: 10.1007/978-3-030-32813-9_2.

[28] T. Muhammed, R. Mehmood, A. Albeshri, and F. Alsolami, “HPC-Smart Infrastructures: A Review and Outlook on Performance Analysis Methods and Tools,” *EAI/Springer Innovations in Communication and Computing*, pp. 427–451, 2020, doi: 10.1007/978-3-030-13705-2_18.

[29] A. Snavely, G. Chun, H. Casanova, R. F. van der Wijngaart, and M. A. Frumkin, “Benchmarks for grid computing,” *ACM SIGMETRICS Performance Evaluation Review*, vol. 30, no. 4, pp. 27–32, Mar. 2003, doi: 10.1145/773056.773062.

[30] E. Gabriel *et al.*, “Open MPI: Goals, Concept, and Design of a Next Generation MPI Implementation,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3241, pp. 97–104, 2004, doi: 10.1007/978-3-540-30218-6_19.

[31] L. Dagum and R. Menon, “OpenMP: an industry standard API for shared-memory programming,” *IEEE Computational Science and Engineering*, vol. 5, no. 1, pp. 46–55, 1998, doi: 10.1109/99.660313.

[32] “GEO ExPro - Supercomputers for Beginners - Part III. GPU-Accelerated Computing.” https://www.geoexpro.com/articles/2016/03/supercomputers-for-beginners-part-iii-gpu-accelerated-computing (accessed Sep. 27, 2021).

[33] “LINPACK benchmarks - Wikipedia.” https://en.wikipedia.org/wiki/LINPACK_benchmarks (accessed Sep. 27, 2021).

[34] “HPL Tuning.” https://www.netlib.org/benchmark/hpl/tuning.html (accessed Sep. 27, 2021).

[35] “How do I tune my HPL.dat file? - Advanced Clustering Technologies.” https://www.advancedclustering.com/act_kb/tune-hpl-dat-file/ (accessed Sep. 27, 2021).

[36] R. P. Brent, “The LINPACK benchmark on the Fujitsu FAP 1000,” in *[Proceedings 1992] The Fourth Symposium on the Frontiers of Massively Parallel Computation*, pp. 128–135. doi: 10.1109/FMPC.1992.234897.

[37] “Frequently Asked Questions | TOP500.” https://www.top500.org/resources/frequently-asked-questions/ (accessed Sep. 27, 2021).

[38] “RandomAccess Rules.” https://icl.utk.edu/projectsfiles/hpcc/RandomAccess/ (accessed Sep. 26, 2021).

[39] “IOR Introduction.” https://ior.readthedocs.io/en/latest/intro.html (accessed Sep. 26, 2021).

[40] “November 2021 | TOP500.” https://top500.org/lists/top500/2021/11/ (accessed Dec. 01, 2021).

[41] https://www.dell.com/support/kbdoc/de-de/000133009/hpc-synthetic-benchmark-performance-using-2nd-generation-intel-xeon-scalable-processors-stream-hpl-and-hpcg?lang=en

[42] V. Aggarwal, Y. Sabharwal, R. Garg and P. Heidelberger, "HPCC RandomAccess benchmark for next generation supercomputers," 2009 IEEE International Symposium on Parallel & Distributed

Processing, 2009, pp. 1-11, doi: 10.1109/IPDPS.2009.5161019.

[43] Florida S. Uni.: https://acct.rcc.fsu.edu/doc/hpc-benchmarks

دکتر احسان آریانیان دارای مدرک دکترای مهندسی برق الکترونیک دیجیتال از دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران) و عضو هیأت علمی و رئیس پژوهشکده فناوری اطلاعات پژوهشگاه ارتباطات و فناوری اطلاعات است. وی دانش آموخته برتر دانشگاه علم و صنعت در مقطع تحصیلی کارشناسی و همچنین دانش آموخته برتر دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران) در مقاطع تحصیلی کارشناسی ارشد و دکتری می‌باشد. وی هم اکنون همزمان با تحقیق و پژوهش در زمینه های علمی و فناوری و همچنین تدریس دروسی نظیر رایانش ابری، ریزپردازنده پیشرفته، شبکه های مخابرات داده و شبکه‌های کامپیوتری پیشرفته در دانشگاه، به عنوان مجری، همکار و ناظر در پروژه های بنیادین، کاربردی و راهبردی در پژوهشگاه ارتباطات و فناوری اطلاعات مشغول به فعالیت می‌باشد. ایشان ضمن تدریس دروس مختلف دانشگاهی، هدایت و داوری پایان نامه در دوره کارشناسی ارشد و دکتری را عهده دار هستند. کتاب‌های «شبکه‌های بی سیم»، «اصول مجازی سازی و رایانش ابری به انضمام بسترهای ابری داده های عظیم»، «استانداردهای فنی و ابعاد حقوقی رایانش ابری در ایران»، «رگولاتوری کلان داده‌ها»، «اصول طراحی و ارزیابی زیرساخت مراکز داده HPC» و «پایگاه‌های داده پیشرفته» از مهمترین کتاب هایی هستند که تا کنون از ایشان به چاپ رسیده‌اند. همچنین مقاله‌های معتبر زیادی در زمینه رایانش ابری و پردازش موازی توسط ایشان در کنفرانس‌ها و ژورنال-های معتبر دنیا به چاپ رسیده است. زمینه‌ها و علایق پژوهشی دکتر آریانیان شامل حوزه‌های رایانش ابری، کلان‌داده‌ها، شبکه‌های کامپیوتری، پردازش موازی، زیرساخت فعال و غیرفعال مراکز داده و سکوهای فناوری اطلاعات می‌باشد.

دکتر زهرا معزکریمی دانش‌آموخته مهندسی نرم-افزار از دانشگاه تهران می‌باشد. ایشان از سال ۲۰۱۱ به عنوان محقق و توسعه‌دهنده لایه سیستم‌عامل و پلتفرم با تمرکز بر حوزه‌های مجازی‌سازی، رایانش-ابری، اینترنت اشیاء، کلان داده و زنجیره بلوکی در پژوهشگاه ارتباطات و فناوری اطلاعات فعالیت می-نماید. علاقمندی ایشان روش‌های صوری و منطق معرفتی، انتشار اطلاعات در شبکه‌های پیچیده و پلتفرم‌های فناوری می‌باشد. دستاوردهای علمی ایشان در قالب کتاب، مقالات ژورنالی و کنفرانسی داخلی و بین‌المللی به چاپ رسیده است.