فهرست مطالب

چکیده ۱

فصل اول: بیان مسئله و کلیات تحقیق ۲

۱-۱ مقدمه ۳

۱-۲ بیان مسئله ٤

۱-۳ اهداف تحقیق ۷

۱-٤ فرضیه ها ۸

۱-۵ متغیرها ۸

۱-٦ ضرورت تحقیق ۹

۱-۷ جنبه های نوآوری ۹

فصل دوم: پیشینه و ادبیات تحقیق **..... ۱۰**

۲-۱ مقدمه ۱۱

۲-۲ گرید محاسباتی ۱۲

۲-۳ تعاریف سیستم گرید ۱٤

۲-۳-۱ تعریف آی.بی.ام از محاسبات گرید ۱۵

۲-۳-۲ تعریف گلوبوس از گرید ۱٦

۲-۳-۳ تعریف صنعتی گرید ۱٦

۲-٤ توانمندی های سیستم گرید ۱٦

۲-٤-۱ استفاده بهینه از منابع ۱۷

۲-٤-۲ موازی سازی پردازنده ها ۱۸

۲-٤-۳ منابع و سازمان های مجازی ۱۹

۲-٤-٤ دسترسی به منابع اضافه ۱۹

۲-٤-۵ تقسیم و توازن بار ۲۰

۲-۵ نگاهی به اجزای گرید ۲۱

۲-۵-۱ رابط کاربر ۲۱

۲-۵-۲ اجزای امنیتی ۲۲

۲-۵-۳ مدیریت کنترل کار سیستم ۲۳
۲-۵-٤ زمان‌بندها ۲٤
۲-۵-۵ مدیریت داده ۲٤
۲-۵-٦ مدیریت منابع ۲۵
۲-٦ اصول ساخت گرید ۲٦
۲-٦-۱ قلمروهای مدیریتی چندگانه و استقلال آنها ۲٦
۲-٦-۲ ناهمگونی ۲٦
۲-٦-۳ مقیاس‌پذیری ۲۷
۲-٦-٤ پویایی و انعطاف‌پذیری ۲۷
۲-٦-۵ ساختار اصلی گرید ۲۷
۲-٦-٦ میان‌افزار هسته‌ای گرید ۲۷
۲-٦-۷ میان‌افزار سطح کاربر ۲۸
۲-٦-۸ برنامه‌های کاربردی گرید ۲۸
۲-۷ معماری گرید ۲۸
۲-۷-۱ لایه فابریک ۳۰
۲-۷-۲ لایه پروتکلهای منبع و اتصال ۳۱
۲-۷-۳ لایه سرویس های تجمعی ۳۱
۲-۷-٤ لایه برنامه های کاربر ۳۲
۲-۸ مثال های عملی از سیستم های گرید ۳۲
۲-۸-۱ پروژه SETI ۳۲
۲-۸-۲ پروژه NAREGI ۳۳
۲-۹ نتیجه گیری ۳٤

فصل سوم: مفاهیم و کارهای مرتبط ۳۵
۳-۱ مقدمه ۳٦
۳-۲ مروری بر الگوریتم‌ها و روش‌ها ۳۷
۳-۲-۱ زمان‌بندی چند سطحی پویا ۳۷
۳-۲-۲ اختصاص سریع‌ترین پردازنده به بزرگ‌ترین وظیفه ۳۷
۳-۲-۳ صف کارها با تکرار ۳۸

۳-۲-٤ الگوریتم اجتماع مورچگان تعادلی ۳۸
۳-۲-۵ الگوریتم ژنتیک در سیستم گرید ۳۹
۳-۲-٦ سیستم مبتنی بر عامل برای مدیریت منابع ٤۰
۳-۲-۷ روش پیوندی مورچگان به منظور زمان‌بندی وظایف مستقل ٤۱
۳-۲-۸ در اختیار گرفتن منابع به کمک الگوریتم یادگیری تقویتی ٤۲
۳-۲-۹ روش حراج دو طرفه پیوسته در پردازش گرید ٤۳
۳-۲-۱۰ متا زمان‌بندها برای زمان‌بندی برنامه‌های موازی ٤٤
۳-۲-۱۱ کلونی مورچگان به منظور زمان‌بندی پویای کارها در گرید ٤۹
۳-۳ تعریف رسمی زمان‌بندی درگرید ۵۰
۳-۳-۱ مدل زمان‌بندی متمرکز ۵۲
۳-۳-۲ مدل زمان‌بندی توزیع شده ۵٤
۳-۳-۳ مدل زمان‌بندی سلسله مراتبی ۵٦
۳-٤ نتیجه گیری ۵۷

فصل چهارم: الگوریتم‌های زمان‌بندی ۵۹
٤-۱ مقدمه ٦۰
٤-۲ فرم ریاضی مسئله ٦۱
٤-۳ تعاریف اولیه ٦۵
٤-٤ معرفی الگوریتم‌های اکتشافی ٦۷
٤-٤-۱ الگوریتم توازن بار خوش‌بینانه ٦۷
٤-٤-۲ الگوریتم حداقل زمان اجرا ٦۸
٤-٤-۳ الگوریتم حداقل زمان تکمیل ٦۹
٤-٤-٤ الگوریتم Min-min ۷۰
٤-٤-۵ الگوریتم Min-max ۷۱
٤-٤-٦ روش سافریج ۷۳
٤-٤-۷ الگوریتم A* ۷٤
٤-٤-۸ الگوریتم WQ ۷٤
٤-٤-۹ الگوریتم LJFR-SJFR ۷۵
٤-٤-۱۰ الگوریتم Duplex ۷٦

٤-٤-١١ الگوریتم Maxstd ٧٧
٤-٤-١٢ الگوریتم kPB..... ٧٨
٤-٥ مقایسه الگوریتم های اکتشافی ٧٩
٤-٦ معرفی الگوریتم‌های فرا اکتشافی ٨١
٤-٦-١ الگوریتم ژنتیک ٨٤
٤-٦-٢ الگوریتم شبیه سازی تبرید..... ٨٥
٤-٦-٣ الگوریتم جستجوی ممنوعه..... ٨٦
٤-٦-٤ الگوریتم بهینه‌سازی ازدحام ذرات ٨٧
٤-٧ هوش جمعی ٨٩

فصل پنجم: روش پیشنهادی و ارزیابی نتایج ٩٢
٥-١ مقدمه ٩٣
٥-٢ بستر شبیه‌سازی ٩٤
٥-٣ الگوریتم‌های اکتشافی ٩٦
٥-٣-١ زمان اجرا..... ٩٧
٥-٣-٢ زمان اتمام کل..... ٩٨
٥-٣-٣ زمان گردش وظایف ١٠١
٥-٣-٤ کارایی منابع ١٠٢
٥-٤ الگوریتم‌های فرا اکتشافی ١٠٦
٥-٤-١ ترکیب الگوریتم ژنتیک و ازدحام ذرات ١٠٦
٥-٤-٢ تنظیمات اولیه الگوریتم‌ها ١٠٩
٥-٤-٣ زمان اتمام کل..... ١١١
٥-٤-٤ زمان گردش وظایف ١١٢
٥-٤-٥ کارایی منابع ١١٤
٥-٥ مقایسه کارایی الگوریتم های تکاملی ١١٦
٥-٦ نتیجه گیری ١١٩
٥-٧ کارهای آینده ١٢٠

منابع ١٢٢

فهرست شکل‌ها

شکل ۲-۱: لایه‌های انتزاع مختلف از یک سیستم گرید ۱۵
شکل ۲-۲: نحوه موازی‌سازی پردازنده‌ها در سیستم گرید ۱۸
شکل ۲-۳: منابع پراکنده به صورت یک سازمان مجازی واحد ۱۹
شکل ۲-٤: سیستم گرید از دید استفاده‌کنندگان ۲۲
شکل ۲-۵: جایگاه ماژول GSI در گرید ۲۳
شکل ۲-٦: موقعیت سرویس‌های MDS در گرید ۲٤
شکل ۲-۷: موقعیت زمان‌بندها در گرید ۲٤
شکل ۲-۸: موقعیت ماژول GASS در گرید ۲۵
شکل ۲-۹: بخش مدیریت منابع در گرید ۲٦
شکل ۲-۱۰: اجزای معماری چهار لایه‌ای گرید ۲۹
شکل ۲-۱۱: لایه‌های کلی معماری گرید ۳۰
شکل ۳-۱: ساختار کلی زمان‌بند گرید ۳۹
شکل ۳-۲: ساختار سیستم مبتنی بر عامل برای مدیریت منابع ٤۰
شکل ۳-۳: ساختار درختی به منظور مدیریت منابع ٤۱
شکل ۳-٤: زمان‌بندی کارها به صورت چند عامله در پردازش شبکه‌ای ٤۳
شکل ۳-۵: نحوه زمان‌بندی در روش FIFO ٤٤
شکل ۳-٦: رابطه میان متا زمان‌بند و کاربر و زمان‌بندهای محلی ٤۵
شکل ۳-۷: شمای کلی متا زمان‌بندها ٤۸
شکل ۳-۸: ساختار عمل زمان‌بندی در گرید (برنامه کاربردی و کارهای آن) ۵۱
شکل ۳-۹: معماری مدل زمان‌بندی متمرکز ۵۳
شکل ۳-۱۰: زمان‌بندی توزیع شده با ارتباط مستقیم ۵۵
شکل ۳-۱۱: مدل زمان‌بندی توزیع شده با ارتباط غیرمستقیم ۵٦
شکل ۳-۱۲: مدل زمان‌بندی سلسله‌مراتبی ۵۷
شکل ٤-۱: رویه ساخت ماتریس ETC ٦۱
شکل ٤-۲: شبه کد الگوریتم توازن بار خوشبینانه ٦۸
شکل ٤-۳: شبه کد الگوریتم حداقل زمان اجرا ٦۹
شکل ٤-٤: شبه کد الگوریتم MCT ۷۰

شکل ٤-٥: شبه کد الگوریتم Min-Min ٧١
شکل ٤-٦: شبه کد الگوریتم Max-Min ٧٢
شکل ٤-٧: شبه کد الگوریتم سافریج ٧٣
شکل ٤-٨: شبه کد الگوریتم WQ ٧٥
شکل ٤-٩: شبه کد الگوریتم SJFR-LJFR ٧٦
شکل ٤-١٠: شبه کد الگوریتم Duplex ٧٦
شکل ٤-١١: شبه کد الگوریتم Maxstd ٧٧
شکل ٤-١٢: شبه کد الگوریتم kPB ٧٨
شکل ٤-١٣: شبه کد الگوریتم ژنتیک ٨٥
شکل ٤-١٤: شبه کد الگوریتم جستجوی ممنوعه ٨٧
شکل ٤-١٥: شبه کد الگوریتم بهینه‌سازی ازدحام ذرات ٨٩
شکل ٤-١٦: فلوچارت الگوریتم بهینه‌سازی ازدحام ذرات ٩١
شکل ٥-١: میانگین هندسی زمان اتمام کل برای محیط‌های توزیع شده مختلف ١٠١
شکل ٥-٢: میانگین هندسی زمان گردش وظایف برای محیط‌های توزیع شده مختلف ١٠٢
شکل ٥-٣: میانگین مقادیر کارایی منابع برای محیط‌های توزیع شده مختلف ١٠٥
شکل ٥-٤: فلوچارت الگوریتم ترکیبی پیشنهادی ١٠٧
شکل ٥-٥: قسمتی از کروموزم مورد استفاده برای الگوریتم‌های تکاملی ١١٠
شکل ٥-٦: زمان اتمام کل برای حالات مختلف به واحد ثانیه ١١٢
شکل ٥-٧: زمان گردش وظایف برای حالات مختلف به واحد ثانیه ١١٤
شکل ٥-٨: درصد کارایی منابع در الگوریتم‌ها و محیط‌های مختلف ١١٥
شکل ٥-٩: مقایسهٔ سرعت همگرایی الگوریتم‌های تکاملی مختلف ١١٨

فهرست جداول

جدول ٤-١: مقادیر پیشنهادی برای $Rmach$ و $Rtask$ ٦٢
جدول ٤-٢: یک نمونه ماتریس ETCبا مقادیر $Rmach$ و $Rtask$ پایین ٦٢
جدول ٤-٣: ماتریس ETC ناسازگار، ناهمگنی وظایف و ماشین‌ها بالا ٦٤
جدول ٤-٤: ماتریس ETC سازگار، ناهمگنی وظایف پایین، ناهمگنی ماشین‌ها بالا ٦٥
جدول ٤-٥: ماتریس ETC سازگار جزئی، ناهمگنی وظایف و ماشین‌ها پایین ٦٥
جدول ٤-٦: مقایسه الگوریتم‌های اکتشافی ٨٠
جدول ٤-٧: نگاهی به آخرین کارهای مرتبط ٨٣
جدول ٥-١: پارامترهای مدل براون و همکاران ٩٥
جدول ٥-٢: چند حالت مختلف از یک محیط توزیع شده ٩٧
جدول ٥-٣: زمان اجرای الگوریتم‌های زمان‌بندی مختلف ٩٨
جدول ٥-٤: زمان اتمام کل برای الگوریتم‌ها و حالات مختلف محیط ٩٩
جدول ٥-٥: زمان گردش وظایف برای الگوریتم‌ها و حالات مختلف محیط ١٠٢
جدول ٥-٦: میزان کارایی منابع برای الگوریتم‌ها و حالات مختلف محیط ١٠٣
جدول ٥-٧: توضیح پارامترهای الگوریتم ترکیبی پیشنهادی ١٠٩
جدول ٥-٨: تنظیمات اولیه مربوط به الگوریتم‌های مورد استفاده ١١٠
جدول ٥-٩: زمان اتمام کل الگوریتم‌ها و حالات مختلف محیط گرید ١١١
جدول ٥-١٠: زمان گردش وظایف الگوریتم‌ها و حالات مختلف محیط گرید ١١٣
جدول ٥-١١: درصد کارایی منابع برای الگوریتم‌ها و حالات مختلف ١١٥
جدول ٥-١٢: توصیف توابع آزمون مختلف برای بررسی سرعت همگرایی ١١٧

چکیده

گرید محاسباتی مجموعه‌ای از منابع و امکانات تحت پوشش یک شبکه کامپیوتری است. این سیستم بصورت جغرافیایی پراکنده شده و تحت مدیریت سازمان‌های مختلف است. منابع یک گرید محاسباتی، پویا و در دامنه‌های مدیریتی مختلفی قرار دارند. بنابراین زمان‌بندی کارها و مدیریت منابع، موضوعات حیاتی در گرید محاسباتی هستند. طبیعت پویا و ناهمگن منابع گرید و همچنین نیازهای مختلف برنامه‌های کاربردی روی آن، باعث پیچیدگی فرآیندی زمان‌بندی می‌شود. برای مدیریت چنین سیستم پیچیده‌ای، نمی‌توان از رویکردهای متداول مدیریت منابع که سعی می‌کنند کارایی را در کل سیستم بهینه کنند، استفاده کرد. بنابراین تاکنون روش‌های متفاوت زیادی برای حل این مسئله پیشنهاد شده است. در این تحقیق، به بررسی گرید محاسباتی و تعداد زیادی از الگوریتم‌های اکتشافی برای زمان‌بندی وظایف در این محیط می‌پردازیم. روش پیشنهادی بر پایه ترکیب دو الگوریتم بهینه‌سازی ازدحام ذرات و ژنتیک شکل گرفته است. برای شبیه‌سازی محیط گرید از ساختار ماتریس زمان مورد انتظار استفاده می‌شود. نتایج بدست آمده نشان می‌دهد که روش جدید کارایی بهتر و زمان پاسخ کمتری را ارئه می‌دهد.

کلمات کلیدی: گرید محاسباتی[1]، زمان‌بندی[2]، الگوریتم ژنتیک[3]، بهینه‌سازی ازدحام ذرات[4]

[1] Computational grid
[2] Scheduling
[3] Genetic Algorithm (GA)
[4] Particle Swarm Optimization (PSO)

فصل اول:

بیان مسئله و کلیات تحقیق

۱-۱ مقدمه

مسئله زمان‌بندی وظیفه‌ها یک مسئله عمومی و پایه‌ای می‌باشد که به علت کاربردهای فراوان در علوم مختلف به عنوان یکی از مسائل مهم در دنیای امروز به شمار میرود. در این مسئله مایل هستیم تا منابع را بصورت بهینه به وظیفه ها تخصیص دهیم، لذا بایستی اهدافی نظیر کمترین زمان اجرا و کمترین تاخیر زمانی را برآورده کنیم. با وجود سادگی در بیان صورت این مسئله، زمان اجرایی الگوریتم آن NP-Hard بوده و تاکنون راه حل بهینه و قطعی برای آن ارایه نشده است. در روش‌های سنتی می‌توان از راه حلی چندهدفه جهت یافتن زمان‌بندی مناسب وظایف بلادرنگ نرم برروی سیستم‌های چندپردازشی ناهمگن درمحیط ایستا به صورت کاملا متصل و بدون ساختار استفاده نمود. غالب این روش‌ها، ازدیدگاه الگوریتمی به این مسئله نگاه می کنند. در این تحقیق به منظور ارایه‌ی دیدگاه سیستمی، بایستی دیدگاه‌های مختلف در خصوص این نوع طراحی بررسی شوند. ماهیت مسئله فوق غیرخطی است؛ لذا نمایش فضای حالت غیرخطی مسئله می‌تواند نمایانگر مناسبی از مسئله باشد. بعد از آن، با استفاده از خواص ماتریس‌ها و اعمال تبدیلات مناسب، مسئله زمان‌بندی وظیفه‌ها به فرم فضای حالت سوئیچینگ خطی به همراه چند قید غیر خطی تبدیل می‌شود. با استفاده از این نمایش جدید، الگوریتم‌هایی برای زمان‌بندی وظایف در دو محیط ایستا و پویا ارایه شده‌اند. در محیط ایستا دو الگوریتم HS و RT به منظور تعیین بردارهای کنترل پیشنهاد شده‌اند. نتایج این دو الگوریتم با الگوریتم معروف HEFT مورد مقایسه قرار گرفته است. نتایج فرآیند مقایسه، کارآیی قابل قبول روش پیشنهادی را نشان می‌دهد. در محیط پویا از مدل توسعه یافته زمان‌بندی وظایف استفاده شده است. نتایج مقایسات این روش با الگوریتم مرتبط نشان دهنده برتری ۱۸٪ روش پیشنهادی (LS3) می‌باشد. گسترش روز افزون تکنولوژی، به دنبال خود افزایش نیازهای منابع اطلاعاتی و محاسباتی را در پی داشته است .با وجود کارآیی بالای اَب ابر کامپیوترها، هزینه بالای آنها منجر شده است تا رویکردجدیدی به نام گرید در شبکه های کامپیوتری ایجاد شود که به سرعت در حال پیشرفت است. در محیط گرید زمان‌بندی و مدیریت موثر منابع امری حیاتی و مهم محسوب

می شود. زمان‌بندی باعث می شود که از یک طرف کاربران وبرنامه های کاربردی به بهترین نحو سرویس بگیرند و از طرف دیگر از کلیه منابع گرید، بهره برداری بهینه و حداکثر صورت گیرد. با توجه گستردگی و پویایی فضای گرید، الگوریتم‌های قطعی نمی‌توانند کارآیی مناسبی برای حل مسئله زمان‌بندی داشته باشند. این امر محققان را بر آن داشته تا الگوریتم‌های ابتکاری را برای این مسئله تجربه کنند. از میان آنها الگوریتم ژنتیک بخاطر اینکه در یک لحظه می‌تواند فضای مسئله را از چندین جهت مختلف و به صورت سراسری جستجو می کند، بهترین روش ابتکاری به شمارمی رود .به همین دلیل سبب شده است که برای حل بسیاری از مسائل بهینه سازی مورد استفاده قرار گیرد. نقطه ضعف الگوریتم ژنتیک این است که همگرایی آن به سمت بهینه سراسری است و در جستجوی محلی کارآیی چندان خوبی ندارد اما با ترکیب آن با الگوریتم‌های جستجوی محلی می‌توان این نقطه ضعف را برطرف نمود. بنابراین با ترکیب امتیازات این دو الگوریتم می‌توان، زمان‌بندی وظایف گریدهای محاسباتی را بهبود بخشید. تاکنون الگوریتم‌های ابتکاری زیادی در رابطه بانحوه زمان‌بندی وظایف در محیط گرید ارائه شده است که هدف بیشتر این الگوریتم‌ها، بهینه سازی یکی از پارامترهای کیفیت سرویس می‌باشد.

۱-۲ بیان مسئله

گرید محاسباتی یک زیرساخت سخت‌افزاری و نرم‌افزاری است که دسترسی به قابلیت‌های سطح بالای محاسباتی را به صورت مطمئن، سازگار، فراگیر و ارزان فراهم می-کند. این یک محیط اشتراکی است که با استقرار سرویس‌های ماندگار و استاندارد پیاده سازی شده است. این سرویس‌ها ایجاد و اشتراک منابع توزیع شده را پشتیبانی می کنند. امروزه افزایش کارایی گرید یک مسئله است. جهت افزایش کارایی گرید به یک زمان‌بندی درست و کارآمد احتیاج است. متأسفانه طبیعت پویای منابع گرید و همچنین تقاضاهای مختلف کاربران، باعث پیچیدگی مسئله زمان‌بندی گرید شده‌اند. پویایی کارایی منابع ناشی از ناهمگونی، خودمختاری و اشتراکی بودن منابع گرید است. هدف مسئله زمان‌بندی گرید، انتساب بهینه کارها به منابع است. مطالعه در زمان‌بندی گرید از این جهت حایز اهمیت است

که سیستم‌های دنیای واقعی معمولاً به طور فیزیکی یا عملیاتی، توزیع شده و ناهمگون می‌باشند. سیستم‌هایی مثل کنترل ترافیک هوایی، نجوم، پزشکی، زیست شناسی، نظامی و موبایل از این دسته هستند. بویا در سال ۲۰۰۱ یک چارچوب اقتصادی برای گرید معرفی کرد. به این ترتیب که کاربران گرید به ازای استفاده از منابع، به دارندگان آن منابع هزینه مالی پرداخت می کنند. این چارچوب باعث ایجاد انگیزه در دارندگان منابع می شود تا منابع خود را به اشتراک بگذارند. مدل اقتصادی بویا باعث اهمیت هزینه و سود اقتصادی از دید کاربران و دارندگان منابع می‌شود. به طوری که از آن پس، در بیشتر الگوریتم‌های زمان‌بندی، هزینه و سود اقتصادی در تابع هدف زمان‌بند مدنظر قرار داده می‌شد؛ اما مشکل این الگوریتمها اینجاست که برای بهبود زمان اتمام کار و هزینه اجرای کار، زمان‌بند توابع هدف مختلف و سیاست‌های مجزایی را بکار می‌گیرد. برای بهبود زمان‌بندی در گرید روش‌های مکاشفه‌ای با الهام از قوانین طبیعی پیشنهاد شده‌اند. این روش‌ها عبارتند از الگوریتم ژنتیک[۱]، سرمایش شبیه‌سازی شده[۲]، جستجوی ممنوعه[۳]، تئوری بازی[٤] و روش‌های ترکیبی آنها. ثابت شده، الگوریتم ژنتیک هم در بهبود زمان و هم در بهبود هزینه، نسبت به تئوری بازی راه حل بهینه‌تری تولید می‌کند. دلیل آن هم محدودیت‌های موجود در تئوری بازی غیر مشارکتی است. محققان نشان داده‌اند که سرعت اجرای الگوریتم ژنتیک از همه الگوریتم‌های فوق بیشتر است و لذا الگوریتم ژنتیک را فقط برای بهبود زمان اتمام کارها بکار گرفته‌اند. البته در الگوریتم ژنتیک برای بهبود زمان اتمام و هزینه اجرای کارها به صورت مجزا، بکار گرفته شده است. هیچکدام از مراجع برای مواقعی که اهمیت وزن هزینه و زمان برای کاربر متفاوت باشد روشی ارائه نکرده اند. مشکل دیگر الگوریتم‌های فوق این است که راه‌حل اولیه در آنها یک نگاشت تصادفی از وظایف به منابع است. از آنجا که ارزانی منبع و سرعت منبع دو فاکتور متضاد هستند یا به عبارت دیگر منبع سریع معمولاً گران است، زمان‌بند باید در انتساب وظایف به منابع نظر کاربر را اعمال کند. مثلاً اگر کاربر نگران هزینه اجرای کار خود باشد وظایف آن کار را روی یک منبع ارزان منتسب کند و اگر کاربر نگران زمان اتمام کار خود باشد، زمان‌بند باید وظایف آن کار را روی یک منبع سریع منتسب کند. زمان‌بند ما به کاربر آزادی بیشتری برای اعمال نظرش می دهد، به این صورت که کاربر می تواند ۷۰ %

نگران هزینه و ۳۰ %نگران زمان اتمام کار خود باشد. ما وزن زمان و وزن هزینه کارهای کاربر را با W_c و W_t و استفاده از دو عدد بین ۰ و ۱ که مجموع آنها ۱ است مدل کرده‌ایم. این دو مقدار توسط کاربر مشخص شده و در تابع هدف مسئله زمان‌بند تاثیر داده شده‌اند. به دلیل ماهیت زمان‌بندی در گرید، ما از الگوریتم غیر قطعی ژنتیک برای بهینه‌سازی زمان‌بندی استفاده کرده‌ایم. در استفاده از الگوریتم ژنتیک دو فاکتور زمان و هزینه به همراه وزن آنها مدنظر قرار داده شده‌اند همچنین راه‌حل اولیه در الگوریتم ژنتیک پیشنهادی یک راه حل تقریباً بهینه است نه یک راه حل تصادفی. موضوع همکاری منابع توزیع شده جغرافیایی برای حل مسائل بزرگ، گرید محاسباتی نام گرفته است. زمان‌بندی در گرید محاسباتی در سیستم‌های نظامی، موبایل و سیستم‌های کنترل آزمایشگاهی پزشکی که طبیعتاً نامتمرکز هستند، از اهمیت ویژه‌ای برخوردار است. زمان‌بندی در گرید یک مسئله غیر قطعی است پس نمی‌توان از الگوریتم‌های قطعی برای بهبود زمان‌بندی استفاده کرد. روش‌های غیرقطعی برای بهبود زمان‌بندی در گرید عبارتند از الگوریتم ژنتیک، جستجوی ممنوعه، سرمایش شبیه سازی شده، تپه نوردی و روش‌های جستجوی مکاشفه ای دیگر. روش‌های زمان‌بندی موجود، الگوریتم‌های غیر قطعی را یا برای کاهش زمان اتمـام به کار می‌برند یا برای کاهـش هزینه اجرا و در حقیـقت کاربـر نمی‌تواند اهمیت دو فاکتورموجود در زمان‌بندی گرید را مشخص کند. در زمان‌بندی گرید دو فاکتور زمان اتمام و هزینه اجرای کار را می‌بایست بطور همزمان مد نظر قرارداد. نسبت اهمیت کاهش زمان اتمام و هزینه اجرایی بهتر است توسط کاربر مشخص شود. در این تحقیق یک الگوریتم بر پایه الگوریتم ژنتیک برای زمان‌بندی وظایف مستقل از هم در گرید محاسباتی پیشنهاد شده است که در آن کاربر می تواند اهمیت زمان اتمام و هزینه اجرای کارهای خود را مشخص کند و زمان‌بندی وظایف کارهای کاربر بر طبق این دو فاکتور صورت می گیرد. امروزه فناوری گرید محاسباتی، به عنوان مدل و رهیافتی نو برای حل مسائل پیچیده و بزرگ در علوم مهندسی، صنعت و تجارت مطرح شده است و هر روز بر تعداد برنامه‌های کاربردی متفاوتی که از زیر ساخت گرید به منظور تأمین نیازهای محاسباتی، ذخیره سازی اطلاعات و سایر موارد بهره برداری می‌کنند، افزوده می‌شود بنابراین در چنین محیط پیچیده‌ای زمان‌بندی و