

به نام خدا

الگوریتم های موازی (طراحی و تحلیل الگوریتم های موازی)

(مخصوص دانشجویان کارشناسی ارشد و دکتری کامپیوتر نرم افزار)

مؤلفان:

دکتر مهدی عفت پرور

مهندس مرضیه بهرامی

مهندس محمد اسمعیلی



الگوریتم های موازی	
سرشناسه : بهرامی، مرضیه، ۱۳۶۴- عنوان و نام پدیدآور : الگوریتم های موازی: (طراحی و تحلیل الگوریتم های موازی)/... مولفان مرضیه بهرامی، مهدی عفت پرور، محمد اسمعیلی. ناشر : سخنوران مشخصات نشر : تهران : سخنوران، ۱۳۹۶. مشخصات ظاهری : ۱۲۲ ص. شابک : ۹۷۸-۶۰۰-۴۵۵-۶۸۵-۹ معیت فهرست نویسی : فیپا موضوع : الگوریتم های موازی موضوع : Parallel algorithms شناسه بروده : عفت پرور، مهدی، ۱۳۶۱- شناسه افزوده : اسمعیلی، محمد، ۱۳۶۲- رده بندی کنگره : QA ۷۶/۶۱۷ ۳۹۰ الف ۷۹ رده بندی دیویی : ۰۰۴/۳۵ شماره کتابشناسی ملی : ۵۰۷۳۷۶۷	نویسنده : مرضیه بهرامی، مهدی عفت پرور، محمد اسمعیلی. ناشر : سخنوران صفحه آرا و طراحی جلد : سخنوران سر و ناشر چاپ : اول ۱۳۹۰ شمارگان : ۱۰۰۰ نسخه شابک : ۹۷۸-۶۰۰-۴۵۵-۶۸۵-۹
	کلیه حقوق این اثر محفوظ می باشد.
تهران، میدان انقلاب، خیابان کارگر شمالی، بعد از ادوارد براون، شماره ۱۴۰۷، طبقه اول تلفن: ۶۶۴۷۶۳۰۶ - ۰۹۱۹۳۶۱۶۶۱۳ (svnpub@gmail.com) ۰۹۱۳۰۶۱۷۲۸۳ سایت انتشارات: www.chapketab.com	قیمت : ۱۵۰۰۰ تومان

فهرست مطالب

پیشگفتار	۱
مقدمه	۲۵
۱. مدل دهی محاسبات ترتیبی	۲۷
۱.۱ مدل های چند پردازنده	۲۸
۱.۱.۱ توپولوژی شبکه	۳۱
۲.۱.۱ عملیات های اشیاء/متغیرهای اولیه	۳۸
۲.۱ مدل های کار-عمی	۴۰
۳.۱ تخصیص هزینه ها به الگوریتم	۴۳
۴.۱ نمونه سازی میان مدل ها	۴۴
۵.۱ مدل مورد استفاده در این فصل	۴۸
۲. روش های الگوریتمی موازی	۵۳
۱.۲ تقسیم و غلبه	۵۳
۲.۲ تصادفی سازی	۵۶
۳.۲ روش اشاره موازی	۵۸
۴.۲ دیگر روش ها	۶۰
۳. عملیات پایه در توالی ها، فهرست ها و درختان	۶۱

- ۱.۳ عملیات جمع/sums ۶۱
- ۲.۳ عملیات پوش/ scans ۶۲
- ۳.۳ چند پیشوندی و fetch-and-add (واکشی و افزودن) ۶۳
- ۴.۳ پرش اشاره گر ۶۴
- ۵.۳ رتبه بندی فهرست ۶۵
- ۶.۳ حذف زنجیره های تکراری ۶۹
- ۱.۶.۳ رویکرد استفاده از آرایه ای از پرچم ها ۶۹
- ۲.۶.۳ رویکرد ۲: درخت سازی ۷۰
۴. گراف ها ۷۳
- ۱.۴ گراف ها و نمایش های گراف ۷۴
- ۲.۴ جستجو اول سطح ۷۸
- ۳.۴ مؤلفه های همبند ۸۱
- ۱.۳.۴ انقباض گراف جفت تصادفی ۸۲
- ۲.۳.۴ انقباض گراف قطعی ۸۷
- ۳.۳.۴ نسخه های بهبود یافته مؤلفه های همبند ۹۱
- ۴.۳.۴ بسط درختان پوشا و درختان پوشا کمینه ۹۲
۵. مرتب سازی ۹۵
- ۲.۵ مرتب سازی مبنایی/پایه ای ۹۷

۱۰۰	هندسه محاسباتی
۱۰۱	۱.۶ نزدیکترین زوج
۱۰۵	۲.۶ پوش محدب مسطح
۱۰۵	۱.۲.۶ Quick Hull/پوش سریع
۱۱۰	۲.۲.۶ Merge Hull/ادغام پوش
۱۱۴	۱.۷. الگوریتم های عددی/رقمی
۱۱۴	۱.۷ عملیات هریسی
۱۱۶	۲.۷ تبدیل فوریه
۱۱۷	۸. خلاصه و مسائل تحقیقاتی
۱۱۸	۹. تعریف عبارات
۲۵	منابع و مآخذ

پیشگفتار

الگوریتم‌های موازی در علوم کامپیوتر، برخلاف الگوریتم‌های متوالی سنتی، الگوریتم‌هایی هستند که در آنها، هر بار قسمتی از برنامه روی پردازنده‌ای متفاوت اجرا می‌شود و در آخر برای کسب نتیجه مطلوب، نتایج کنار هم قرار می‌گیرند. بعضی از الگوریتم‌ها را می‌توان به آسانی به چنین قسمت‌هایی تقسیم کرد. به طور مثال، عمل بر روی اعداد از یک تا صد هزار برای تشخیص اعداد اول را، می‌توان با اختصاص دادن هر مجموعه‌ای از اعداد به هر پردازنده موجود و سپس گردآوری فهرست نتایج مطابقت بابت بندی کرد. برخی از الگوریتم‌ها برای اجرای مراحل بعد، نیاز به نتایج مراحل قبل دارند. اینگونه مسائل را مسائل ذاتاً متوالی می‌گویند. روش‌های عددی تکراری مانند روش نیوتون یا مسئله سه تن، نمونه‌هایی از الگوریتم‌های متوالی هستند. برخی از مسائل را خیلی دشوار می‌توان به صورت موازی درآورد حتی اگر راه حلی سنتی باشند. یکی از این نمونه‌ها جستجوی عمقی درخت است.

الگوریتم‌های موازی ارزشمندند زیرا اجرای عملیات محاسباتی بزرگ از طریق الگوریتم‌های موازی، به دلیل کارکرد پردازنده‌های مدرن، بسیار سریع‌تر از اجرای آنها با الگوریتم‌های متوالی است. ساخت یک کامپیوتر با یک پردازنده خیلی سریع بسیار سخت‌تر از ساختن یک کامپیوتر با تعداد زیادی پردازنده کندتر با توان عملیاتی یکسان است.

با این حال، برای سرعت الگوریتم‌های موازی نیز محدودیت‌های خاص نظری وجود دارد. قسمتی از هر الگوریتم موازی، متوالی است، از این رو هر الگوریتم

موازی یک نقطه اشباع دارد. بعد از آن نقطه اشباع اضافه کردن تعداد بیشتری پردازنده افزایش توان عملیاتی را در پی ندارد و تنها باعث بالا بردن هزینه و خسارات می شود. هزینه و پیچیدگی الگوریتم های موازی بر اساس حافظه و زمانی (تعداد سیکل های پردازنده) که مصرف می کنند تخمین زده می شود.

الگوریتم های موازی باید از جهت ارتباط بین پردازنده های مختلف نیز بهینه شوند. الگوریتم های موازی از دو راه با پردازنده ها ارتباط برقرار می کنند، حافظه مشترک، و رد و بدل کردن پیام.

پردازش ها به مشترک نیاز به قفل بندی اضافه برای اطلاعات دارد، از این رو هزینه سیگرهای گذرگاه و پردازنده های اضافی را تحمیل می کند و همچنین باعث غیر موازی شدن قسمت هایی از الگوریتم می شود. پردازش از طریق انتقال پیام، از کانال ها و جعبه های پیام استفاده می کند اما این نوع ارتباط باعث افزایش هزینه انتقال روی گذرگاه، هزینه های برای صف و جعبه های پیام و تأخیر در پیام ها می شود.

در طراحی های چند پردازنده ای از گذرگاه های خاصی استفاده می شود تا بدین گونه از هزینه های تعاملات کاسته شود اما این پردازنده است که حجم ترافیک را تعیین می کند. مشکل دیگر الگوریتم های موازی تخمین موازنه درخور آنها است. برای مثال، بررسی تمام اعداد از یک تا صد هزار برای یافتن اعداد اول را می توان به راحتی بین پردازنده ها تقسیم کرد. اما در این روش ممکن است بعضی از پردازنده ها مجبور شوند بیشتر از بعضی دیگر کار کنند، در این صورت پردازنده هایی که کارشان به پایان رسیده است تا پایان کار دیگر پردازنده ها بی کار می مانند.

زیر مجموعه ای از الگوریتم های موازی، الگوریتم های توزیعی هستند که برای استفاده در محیط های محاسبات خوشه ای و محاسبات توزیعی طراحی شده اند،

که در این حیطه باید ملاحظات افزون بر الگوریتم‌های موازی «سنتی»، اعمال شود.

طراحی الگوریتم موازی

طراحی الگوریتم‌ها به راحتی و به دستورات مشخص محدود نمی‌شود. هدف ارائه چهارچوبی است که طی آن طراحی الگوریتم‌های موازی امکان‌پذیر شود. در این فرایند سعی بر ایجاد درکی شهودی است از آنچه که یک الگوریتم موازی خوب را تشکیل می‌دهد.

مشکلات موجود در طراحی الگوریتم‌های موازی

- بازده
- تناسب
- جزء بندی محاسبات
- تجزیه
- تکنیک‌های تجزیه تابعی
- موقعیت
- ارتباطات همگام و غیر همگام
- انباشتگی

طراحی‌های علمی

این روش طراحی را در چهار مرحله انجام می‌دهد.

- جزء بندی
- ارتباطات
- انباشتگی

• نقشه بندی

در دو مرحله اول، تمرکز ما روی تناسب و همزمانی است. در دو مرحله دیگر نیز تمرکز روی موقعیت، و دیگر مسائل مربوط به کارایی است.

جزء بندی

کارهای مربوط به محاسبات و داده‌هایی که روی آنها پردازش انجام می‌گیرد را به بخش‌های کوچک تقسیم کنید. مشکلات عملی مانند تعداد پردازنده‌ها در کامپیوتر مکرر در محاسبات نمی‌آید.

ارتباطات

ارتباطات لازم برای هماهنگ کردن اجرای امور مشخص می‌شوند. الگوریتم‌ها و ساختارهای مناسب ارتباطی بین تعیین می‌گردند.

انباشتگی

امور و ساختارهای ارتباطی تعریف شده در دو مرحله اول یک طرح با توجه به معیارهای زیر ارزیابی می‌شوند.

• نیازهای اجرایی

• هزینه‌های پیاده‌سازی

در صورت لزوم، کارها با هم ادغام می‌شوند برای:

• بهبود بخشیدن به کارایی

• کاهش هزینه‌های توسعه

نگاشت^۱ تطابق جزء جزء کردن کارها و تخصیص کارها به پردازنده‌ها)
برای هر پردازنده یک سری کار تعریف می‌شود و در این نگاشت موارد زیر
رعایت می‌شود:

- افزایش بهره‌برداری پردازنده‌ها - طوری که کارها به صورت متوازن بر روی
آن‌ها تقسیم شود
- کاهش هزینه‌های ارتباطی بین پردازنده‌ها

نگاشت را می‌توان به صورت ثابت یا در زمان اجرا توسط الگوریتم‌های توازن
بارگذاری انجام داد.

پردازش موازی چیست؟

اجرای برنامه و محاسبات به صورت سریال
اجرای محاسبات بر روی یک پردازنده
تقسیم محاسبات به یک سری محاسبات جزا
اجرای عملیات بصورت جداگانه و پشت سر هم
اجرای فقط یک عملیات در هر لحظه زمانی

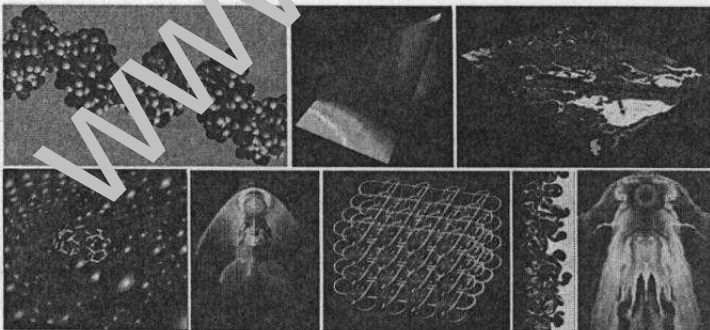
موارد استفاده پردازش موازی در علوم و مهندسی

علاوه بر موارد استفاده متعدد پردازش موازی در فعالیت‌های زیرمجموعه این موضوع
در علوم و مهندسی نیز کاربردهای فراوانی دارد. برخی از موارد کاربرد پردازش

موازی در علوم و مهندسی که معمولاً از آنها به عنوان محاسبات پیشرفته یاد می‌شود،

عبارتند از:

- محاسبات هواشناسی و محیط زیست
- فیزیک محاسباتی (کاربردی، هسته‌ای، ذره‌ای، ماده چگال و...) ...
- علوم زیستی، بیو تکنولوژی و ژنتیک
- شیمی محاسباتی (دینامیک مولکولی و...)
- محاسبات زمین‌شناسی و زلزله‌سنجی
- محاسبات در مهندسی مکانیک (دینامیک سیالات محاسباتی، مکانیک محاسباتی و...)
- محاسبات در مهندسی برق و الکترونیک (طراحی مدار، میکروالکترونیک و...) ...)
- گرافیک کامپیوتری و انیمیشن
- ریاضیات محاسباتی و رمزنگاری



چرا از پردازش موازی استفاده کنیم؟

عمده دلایل استفاده از پردازش موازی عبارتند از:

- کاهش زمان (و هزینه) اجرای محاسبات

- انجام محاسبات بزرگتر و حل مسائل با مقیاس بزرگتر
- فراهم نمودن امکان انجام همزمان محاسبات
- استفاده از منابع از راه دور (بعنوان مثال به دو پروژه زیر توجه فرمایید):
- SETI@home uses over 330,000 computers for a compute power over 528 TeraFLOPS (as of August 04, 2008)
- Folding@home uses over 340,000 computers for a compute power of 4.2 PetaFLOPS (as of November 4, 2008)

■ رسیدن به محدودیت های پردازش سریال

■ سرعت انتقال داده (سرعت نور)

■ محدودیت در دسترسی و بیناتوری کردن بوردها

■ محدودیت های مالی و اقتصادی

ساختار حافظه در رایانه های مدرن

حافظه مشترک

مشخصات کلی:

تمامی پردازنده ها دسترسی به حافظه به عنوان یک فضای یکپارچه و آدرس دهی یکسان دارند.

پردازنده های مختلف مستقلا عمل کرده ولی از حافظه مشترک استفاده می کنند.

هر تغییر در حافظه توسط یک پردازنده، بلافاصله توسط تمامی پردازنده ها قابل رویت می باشد.

به طور کلی رایانه های با حافظه مشترک را می توان به دو دسته کلی بر مبنای دسترسی به حافظه تقسیم نمود:

۱- دسترسی یکنواخت به حافظه