دانشگاه پیام نور

دانشکده فنی و مهندسی

پایان‌نامه برای دریافت درجه کارشناسی ارشد

رشته مهندسی کامپیوتر – گرایش نرم افزار

گروه مهندسی کامپیوتر و فناوری اطلاعات

شتاب‌دهی الگوریتم‌های خوشه‌بندی با استفاده از مدل برنامه‌نویسی موازی ترکیبی روی پردازنده‌های گرافیکی

فاطمه نوری

استاد راهنما:

دکتر اکبر فرهودی‌نژاد

بهار ۱۳۹۳

بسم الله الرحمن الرحيم

با سپاس ایزد بی همتا را . . .

با تقدیر و تشکر شایسته از استاد فرهیخته و فرزانه جناب آقای دکتر فرهودی نژاد که با نکته های دلاویز و گفته های بلند، صحیفه های سخن را علم پرور نمود و راهنما و راه گشایم در اتمام و اکمال این پایان نامه بوده است.

سپاس بی پایان از زحمات پدر بزرگوار و مادر مهربانم ، آن دو فرشته ای که از خواسته هایشان گذشتند، سختی ها را به جان خریدند و خود را سپر بلای مشکلات و ناملایمات کردند تا من به جایگاهی که اکنون در آن ایستاده ام برسم .

از همسرم، اسطوره زندگیم و پناه خستگیم و پسرم ، امید بودنم سپاس گزارم.

چکیده :

در دنیای محاسبات مسائل چالش برانگیزی وجود دارند که حل آن‌ها در زمان معقول و با استفاده از یک پردازنده، دشوار است. با معماری پردازنده‌های چندهسته‌ای، زمان اجرای این کاربردها کاهش می‌یابد. پردازنده‌های گرافیکی با معماری چندهسته‌ای موازی، توان پردازشی بسیار بالایی را نسبت به پردازنده‌های مرکزی ارائه می‌دهند. این قابلیت موجب گسترش استفاده از آن‌ها در کاربردهای غیرگرافیکی شده است. توسعه استفاده از پردازنده‌های‌گرافیکی همه‌منظوره، گسترش زبان‌ها و ابزارهای برنامه‌نویسی موازی را به دنبال داشته است. محاسبات با کارایی بالا، زمانی امکان‌پذیر است که برنامه‌نویس روی سخت‌افزار این پردازنده‌ها، تخصیص وظایف بین آن‌ها، هماهنگ‌سازی و زمان‌بندی تمرکز کند. کودا، یک مدل برنامه‌نویسی موازی در راستای استفاده آسان از سخت‌افزار پردازنده‌های گرافیکی است. کودا، سلسله مراتبی از بلوک‌های نخ و حافظه مشترک را برای نوشتن برنامه‌های کاربردی بسیار موازی ارائه می‌دهد. هدف، استفاده از پردازش موازی به‌عنوان راه‌حلی کارآمد در افزایش کارایی محاسبات با حجم بالا است. در محاسبات داده–موازی، مانند خوشه‌بندی، که عملیات‌های محاسباتی بیشتر از عملیات‌های حافظه است می‌توان به حداکثر کارایی روی پردازنده‌های گرافیکی رسید. خوشه‌بندی، فرایند تجزیه و تحلیل آماری است که اشیاء مشابه را در مجموعه‌های همگن گروه‌بندی می‌نماید که با رشد داده‌ها و افزایش زمان انجام محاسبات، کارایی آن کاهش می‌یابد. خوشه‌بندی فازی، الگوریتمی تکراری است که گروه‌بندی داده‌ها را با استفاده از محاسبات خطی و جمع‌های تکراری، انجام می‌دهد؛ لذا انتخاب مناسبی برای موازی‌سازی به‌منظور شتاب‌دهی و افزایش کارایی است. در این تحقیق، تسریع محاسبات خوشه‌بندی فازی با استفاده از پردازنده‌گرافیکی و مدل برنامه‌نویسی موازی کودا، نسبت به پردازنده‌مرکزی بررسی می‌گردد. نتایج حاصل از پیاده‌سازی این الگوریتم با روش پیشنهادی نشان می‌دهد که زمان اجرا، از تسریع قابل توجهی برخوردار است.

واژه های کلیدی:

پردازنده‌های‌گرافیکی همه‌منظوره، خوشه‌بندی فازی، شتاب‌دهی به محاسبات، محاسبات موازی، مدل برنامه‌نویسی موازی کودا.

فهرست مطالب

عنوان **صفحه**

فصل ۱ مقدمه ۱

۱-۱ مقدمه ۲
۱-۲ تعریف مسأله و بیان سؤال‌های اصلی تحقیق ۳
۱-۳ سابقه و ضرورت انجام تحقیق ۶
۱-۴ فرضیه‌ها ۷
۱-۵ هدف‌ها ۷
۱-۶ کاربردها ۷
۱-۷ جنبه نوآوری تحقیق ۸
۱-۸ روش تحقیق ۸
۱-۹ مراحل انجام تحقیق ۹
۱-۱۰ ساختار پایان‌نامه ۹

فصل ۲ مفاهیم عمومی پردازنده‌های گرافیکی ۱۰

۲-۱ مقدمه ۱۱
۲-۲ معماری پردازنده‌های گرافیکی ۱۱
۲-۲-۱ فِرمی ۱۲
۲-۳ استفاده از پردازنده‌های گرافیکی در محاسبات ۱۵
۲-۳-۱ محاسبات ناهمگن ۱۶
۲-۳-۲ گرایش‌ها در محاسبات موازی ۱۸
۲-۳-۳ پردازنده‌های گرافیکی همه‌منظوره ۲۰
۲-۴ کودا ۲۳
۲-۴-۱ اهداف کودا ۲۵

۲-۴-۲ ویژگی‌ها و مزایای کودا ۲۷
۲-۵ معماری مدل برنامه نویسی موازی کودا ۲۸
۲-۵-۱ کرنل، نخ و روند اجرای آن‌ها در سخت افزار پردازنده‌های گرافیکی ۲۸
۲-۵-۲ سازمان‌دهی نخ‌های کودا در تار، بلوک و گرید ۲۹
۲-۶ جمع‌بندی ۳۱

فصل ۳ مروری بر کارهای انجام شده ۳۳

۳-۱ مقدمه ۳۴
۳-۲ تاریخچه پردازش موازی ۳۴
۳-۲-۱ عملکرد پردازنده‌های مرکزی ۳۵
۳-۲-۱-۱ موازی‌سازی در سطح دستور ۳۶
۳-۲-۱-۲ موازی‌سازی در سطح نخ ۳۸
۳-۲-۱-۳ موازی‌سازی در سطح اطلاعات ۳۹
۳-۳ ظهور پردازنده‌های گرافیکی همه‌منظوره ۴۰
۳-۳-۱ توسعه بهره‌وری از توان محاسباتی پردازنده‌های گرافیکی ۴۱
۳-۳-۲ پیدایش مدل‌های برنامه‌نویسی موازی ۴۳
۳-۳-۲-۱ میان‌افزار کودا ۴۳
۳-۴ کاربردهای اجرا شده روی پردازنده‌های گرافیکی ۴۵
۳-۴-۱ خوشه‌بندی ۴۶
۳-۴-۱-۱ هدفِ خوشه‌بندی ۴۸
۳-۴-۱-۲ انواع خوشه‌بندی ۴۹
۳-۴-۱-۳ الگوریتم‌های خوشه‌بندی پرکاربرد ۵۰
۳-۴-۱-۳-۱ خوشه‌بندی K-means ۵۱
۳-۴-۱-۳-۲ خوشه‌بندی فازی ۵۲
۳-۴-۱-۴ پیاده‌سازی‌های قبلی الگوریتم‌های خوشه‌بندی روی پردازنده گرافیکی ۵۴
۳-۵ جمع‌بندی ۵۵

فصل ۴ معرفی روش پیشنهادی ۵۶

۴-۱ مقدمه ۵۷
۴-۲ معرفی الگوریتم پیشنهادی ۵۷
۴-۲-۱ خوشه‌بندی C-means فازی ۵۸
۴-۲-۲ مراحل پیاده‌سازی الگوریتم خوشه‌بندی فازی ۵۹
۴-۳ پیاده‌سازی الگوریتم پیشنهادی مبتنی بر کودا ۶۰
۴-۳-۱ انتقال داده‌ها بین پردازنده‌مرکزی و گرافیکی در حافظه مشترک ۶۱
۴-۳-۲ کرنل‌های فاصله و عضویت ۶۴
۴-۳-۳ کرنل به‌روزرسانی مراکز خوشه‌ها ۶۴
۴-۳-۴ فراخوانی کرنل‌ها ۶۶
۴-۳-۵ محاسبه زمان اجرا ۶۷
۴-۴ جمع‌بندی ۶۸

فصل ۵ آزمایش‌ها و ارزیابی نتایج ۶۹

۵-۱ مقدمه ۷۰
۵-۲ مجموعه آزمایشی ۷۰
۵-۲-۱ محیط پیاده‌سازی در اجرای آزمایش‌ها ۷۰
۵-۲-۲ مجموعه داده‌های مورد استفاده در آزمایش‌ها ۷۳
۵-۳ مراحل آزمایش ۷۳
۵-۴ معیارهای کارایی ۷۴
۵-۵ نتایج تجربی آزمایش ۷۴
۵-۶ جمع‌بندی و ارزیابی نتایج ۸۱

فصل ۶ نتیجه‌گیری و پیشنهادها ۸۲

۶-۱ نتایج حاصل از تحقیق ۸۳

۶-۲ نوآوری تحقیق..... ۸۴
۶-۳ پیشنهادها..... ۸۵
مراجع ۸۶

واژه‌نامه

واژه‌نامه فارسی به انگلیسی ۹۱
واژه‌نامه انگلیسی به فارسی ۹۴

فهرست شکل‌ها و جداول

عنوان **صفحه**

شکل ۲–۱ : نمونه‌ای از SM در کارت‌های گرافیک NvidiaG100 ۱۲
شکل ۲–۲ : نمونه‌ای از جی‌پی‌سی در سری NvidiaG100 ۱۲
شکل۲ –۳ : سخت‌افزار پردازنده‌های گرافیکی کلاس فرمی کنونی ۱۴
شکل ۲–۴ : جریان محاسبات ناهمگن ۱۸
شکل ۲–۵ : تاریخچه‌ی زبان‌های برنامه‌نویسی برای محاسبات پردازنده‌های گرافیکی ۲۱
شکل ۲-۶ : مقایسه تاریخی اوج کارایی برحسب گیگافلاپ و پهنای باند برای تسریع ۲۱
شکل ۲–۷ : گرایش‌های جستجو در گوگل برای پردازنده‌های گرافیکی، اف‌پی‌جی‌ای و پهن باند سلولی ۲۳
شکل ۲–۸ : جریان پردازش در کودا ۲۵
شکل ۲–۹ : سازمان نخ‌ها، بلوک نخ و گرید ۳۱
شکل ۳–۱ : موازی‌سازی در سطح دستور در پردازنده‌مرکزی ۳۶
شکل ۳–۲ : طبقه‌بندی فیلین و روش پردازش SIMD ۴۰
شکل ۳–۳ : مدل‌های برنامه‌نویسی معماری کودا ۴۴
شکل ۳–۴ : خوشه‌بندی نمونه‌های ورودی ۴۷
شکل ۳–۵ : خوشه‌بندی وسایل نقلیه ۴۷
شکل ۳–۶ : معیار تشابه بر اساس فاصله ۴۹
شکل ۳–۷ : طبقه‌بندی الگوریتم‌های خوشه‌بندی ۵۰
شکل ۳–۸ : مجموعه داده‌های پروانه‌ای در خوشه‌بندی ۵۳
شکل ۴–۱ : ارتباط بین میزبان، دستگاه و حافظه مشترک در انتقال داده‌ها ۶۲
شکل ۴–۲ : کد مربوط به کپی داده‌ها از پردازنده‌مرکزی به حافظه پردازنده‌گرافیکی ۶۲
شکل ۴–۳ : کد مربوط به تعریف متغیرها و ثابت‌ها ۶۳
شکل ۴–۴ :کرنل به‌روزرسانی مراکز خوشه‌ها ۶۵
شکل ۴–۵ : نحوه فراخوانی کرنل‌ها از بدنه اصلی برنامه ۶۶
شکل ۴–۶ : کد مربوط به محاسبه زمان ۶۷
شکل ۵–۱ : نمودار میله‌ای مقایسه زمان اجرای گروه داده‌های اول روی پردازنده مرکزی و گرافیکی ۷۵
شکل ۵–۲ : نمودار خطی روند تسریع زمان اجرا در گروه داده‌های اول ۷۵
شکل ۵–۳ : نمودار میله‌ای مقایسه زمان اجرای گروه داده‌های دوم روی پردازنده مرکزی وگرافیکی ۷۶
شکل ۵–۴ : نمودار خطی روند تسریع زمان اجرا در گروه داده‌های دوم ۷۶
شکل ۵–۵ : نمودار میله‌ای مقایسه زمان اجرای گروه داده‌های سوم روی پردازنده مرکزی وگرافیکی ۷۷
شکل ۵–۶ : نمودار خطی روند تسریع زمان اجرا در گروه داده‌های سوم ۷۷

شکل ۵–۷ : نمودار میله‌ای مقایسه زمان اجرای گروه داده‌های چهارم روی پردازنده مرکزی‌وگرافیکی ۷۸
شکل ۵–۸ : نمودار خطی روند تسریع زمان اجرا در گروه داده‌های سوم ۷۸
شکل ۵–۹ : نمودار میله‌ای مقایسه کلی زمان اجرا روی پردازنده مرکزی و گرافیکی در داده‌های مورد آزمایش ۸۰
شکل ۵–۱۰: نمودار خطی درصد افزایش کارایی در اجرای موازی الگوریتم در داده‌های مورد بررسی در جداول ۸۰
جدول ۵–۱ :خصوصیات پردازنده‌گرافیکی در پیاده‌سازی الگوریتم ۷۲
جدول ۵–۲ : قابلیت‌های پردازشی پردازنده‌گرافیکی در پیاده‌سازی الگوریتم ۷۲
جدول ۵–۳ : مقایسه زمان اجرای برنامه روی پردازنده‌مرکزی و گرافیکی در گروه داده‌های اول ۷۵
جدول ۵–۴ : مقایسه زمان اجرای برنامه روی پردازنده‌مرکزی و گرافیکی در گروه داده‌های دوم ۷۶
جدول ۵–۵ : مقایسه زمان اجرای برنامه روی پردازنده‌مرکزی و گرافیکی در گروه داده‌های سوم ۷۷
جدول ۵–۶ : مقایسه زمان اجرای برنامه روی پردازنده‌مرکزی و گرافیکی در گروه داده‌های چهارم ۷۸
جدول ۵–۷ : زمان اجرای برنامه روی پردازنده‌مرکزی و گرافیکی در داده‌های مورد آزمایش ۷۹

فهرست واژگان اختصاری

ALU	Arithmetic and Logic Unit
ASLS	Application Specific Lightweight Scheduler
CDMA	Code Division Multiple Access
Cell BE	Cell Broadband Engine
CUDA	Compute Unified Device Architecture
ECC	Error Correction Code
EDA	Electronic Design Automation
E-Science	Electronic Science
FIFO	First In First Out
FPGA	Field Programmable Gate Array
FPU	Floating Point Unit
GPC	Graphics Performance Characterization
GPGPU	General Purpose Graphics processing unit
GPU	Graphics Processing Unit
HIC	Host Interface Card
HPC	High Performance Computing
IC	Integrated Circuit
IEEE	Institute of Electrical and Electronics Engineers
ILP	Instruction Level Parallelism
IPC	Instruction Per Clock
LHC	Large Hadron Collider
MoC	Model of Computation
MPI	Message Passing Interface
MT	Multi Threading
NMD	Nanoscale Molecular Dynamics
NASA	National Aeronautics and Space Administration
OpenCL	Open Computing Language
OpenGL	Open Graphics Library
OpenMP	Open Multi-Processing

PCI-E	Peripheral Component Interconnect Express
QPSK	Quadrature Phase Shift Keying
RMA	Remote Memory Access
SFU	Special function unit
SIMD	Single Instruction Multiple Data
SISD	Single Instruction Single Data
SMP	Stream Multiprocessor

فصل ۱ مقدمه

۱-۱ مقدمه

پردازنده‌های گرافیکی[۱]، سیستم‌های بسیار موازی می‌باشند که از صدها هسته پردازنده و هزاران نخ[۲] در حال اجرا و هم‌روند تشکیل شده‌اند و به‌دلیل توان محاسباتی بالا، بسیار سریع‌تر از پردازنده‌های مرکزی[۳] هستند(Brodtkorb,2012). امروزه، مدل‌های برنامه‌نویسی مقایس‌پذیر[۴] و همه‌منظوره[۵] برای نوشتن برنامه‌های بسیار موازی ارائه شده است. این مدل‌ها چند انتزاع کلیدی، سلسله مراتبی از بلوک‌های نخ و حافظه مشترک را ارائه می‌دهند و ثابت شده است که این مدل‌ها در برنامه‌نویسی پردازنده‌های گرافیکی چندهسته‌ای-چندنخی کاملا موفق عمل نموده و توسعه‌ی آن‌ها روی صدها هسته امکان‌پذیر است.

دانشمندان در شاخه‌های صنعتی و دانشگاهی در حال استفاده از مدل‌های برنامه‌نویسی موازی برای دستیابی به سرعت قابل توجه در برنامه‌های تولیدی و تحقیقاتی خود هستند. لذا به نظر می‌رسد بهترین راه افزایش کارایی در محاسبات با حجم بالا به کارگیری کاراتر پردازنده‌های گرافیکی است.

میان‌افزار کودا[۶]، یک راه‌حل جدید برای محاسبات موازی چندمنظوره[۷] است. کودا یک مدل برنامه‌نویسی موازی و محیطی نرم‌افزاری است که از توان محاسبات موازی پردازنده‌گرافیکی برای محاسبات غیرگرافیکی، در زمانی بسیار کم‌تر از زمان مورد نیاز پردازنده‌های مرکزی بهره می‌گیرد(Okitsu,2010). در سال‌های اخیر بسیاری از برنامه‌های کاربردی مقیاس‌پذیر، مانند حل ماتریس، مرتب‌سازی، جستجو و... روی این پردازنده‌ها و با تسریع چند برابری اجرا شده‌اند.

خوشه‌بندی[۸] داده‌ها در حوزه‌های بسیار متنوعی مانند داده‌کاوی[۹]، یادگیری‌ماشین، زمین‌شناسی و... کاربرد دارد. خوشه‌بندی فرایند تجزیه و تحلیل آماری است که در آن داده‌های مشابه در مجموعه‌های همگن به‌نام خوشه، گروه‌بندی می‌شوند(Tan,2006). در مجموعه داده‌های بزرگ، با

[1] GPU: Graphic Processor Unit
[2] Thread
[3] CPU: Central Processor Unit
[4] Scalable
[5] General Purpose
[6] CUDA: Compute Unified Device Architecture
[7] Multi purpose
[8] Clustering
[9] Datamining

افزایش تعداد خوشه‌ها، نیازهای محاسباتی افزایش می‌یابد که این امر موجب زمان‌بَر شدن تجزیه و تحلیل داده‌ها روی یک پردازنده سری می‌گردد.

خوشه‌بندی فازی[1] یک الگوریتم خوشه‌بندی تکراری است و معمولا بهتر از k-means عمل می‌کند(Jain,1998) زیرا در این روش از مجموعه‌های فازی استفاده می‌شود و هر نقطه ممکن است با درجه‌های عضویت مختلف به دو یا چند خوشه متفاوت تعلق داشته باشد. کاربردهای زیادی وجود دارند که از خوشه‌بندی فازی برای تولید پایگاه داده‌های فازی استفاده می‌کنند. این کاربردها عبارتند از قطعه‌بندی تصویر[2]، فشرده‌سازی تصویر، حس کردن از راه دور، تشخیص اشیا و داده‌کاوی.

خوشه‌بندی فازی مبتنی بر مدل کمترین مربعات خطای[3] استاندارد است و به‌دلیل تغییر مرکز خوشه‌ها، به‌ندرت از داده‌های یکسان در تکرارهای محاسبات استفاده می‌نماید. این ویژگی‌ها موجب می‌شوند که این الگوریتم انتخاب مناسبی برای موازی‌سازی و نگاشت به پردازنده‌های موازی، به‌منظور افزایش کارایی باشد.

در این بخش به تعریف مسأله و معرفی ایده خود خواهیم پرداخت.

۱-۲ تعریف مسأله و بیان سؤال‌های اصلی تحقیق

کلید موفقیت در محاسبات پردازنده‌های گرافیکی تا حدی به کارایی کلی آن در مقایسه با پردازنده‌های مرکزی بستگی دارد. امروزه، با توجه به حداکثر پهنای باند در تئوری و کارایی گیگافلاپ[4]ها، یک فاصله تقریباً هفت برابری(Cook Sh,2013) بین این دو وجود دارد. ریشه این فاصله کارایی در محدودیت‌های فیزیکی هر هسته و تفاوت معماری‌های این دو پردازنده است.

پردازنده‌های مرکزی در واقع یک پردازنده وان‌نیومن[5] سریال می‌باشند و برای اجرای سری عملیات‌ها به صورت ترتیبی بهینه‌سازی شده‌اند. یکی از عوامل اصلی کارایی پردازنده‌های مرکزی، فرکانس رو به افزایش آن‌ها است. با دو برابر شدن فرکانس، کارایی نیز دو برابر می‌گردد. گرایش به رشد فرکانس به‌صورت نمایی همیشه وجود دارد.

[1] Fuzzy Clustering
[2] Image segmentation
[3] Least square error
[4] Giga Flops
[5] VonNeuman

به‌هرحال، دردهه ۲۰۰۰(Brodtkorb,2012)، این افزایش به دلیل محدودیت‌های پردازنده‌های مرکزی متوقف شد. از آنجا که مصرف توان پردازنده‌های مرکزی به مکعب فرکانس[1] بستگی دارد، چگالی توان در حال نزدیک شدن به توان مرکز راکتور هسته‌ای بود. عدم توانایی در سرد کردن این‌گونه تراشه‌ها به‌نحو مؤثر، گرایش به رشد نمایی فرکانس را درست قبل از ۴.۰ گیگاهرتز متوقف نمود. هم‌زمان با برخورد کردن پردازنده‌های مرکزی به سقف کارایی سریال، گرایش به استفاده از پردازنده‌های گرافیکی، به‌طور گسترده و نمایی شروع به رشد نمود.

از آنجا که محاسبه رنگ یک پیکسل در صفحه نمایش می‌تواند مستقل از همه پیکسل‌های دیگر انجام شود، موازی‌سازی[2] یک روش طبیعی برای افزایش کارایی پردازنده‌های گرافیکی است. لذا به‌نظر می‌رسد، بهترین راه افزایش کارایی در محاسبات با حجم بالا به‌کارگیری کاراتر پردازنده‌های گرافیکی است.

پردازنده مرکزی به لحاظ دارا بودن برخی از خصوصیات فنی نظیر حافظه‌کش[3] و انشعاب‌های زیاد، توانایی بالایی برای انجام محاسبات سری را دارد. برعکس، پردازنده گرافیکی به دلیل توانایی‌های بالایی که در انجام محاسبات شناور دارد برای محاسبات موازی[4] بسیار مناسب است و می‌تواند آن‌ها را بسیار سریع‌تر انجام دهد(Brodtkorb,2010).

این موضوع موجب گسترش کاربردهای این پردازنده‌ها در حوزه‌هایی فراتر از بازی‌های رایانه‌ای شده و برای انجام کارها و مقاصدی به‌غیر از نگاشت پس‌زمینه و رندر کردن بازی‌های رایانه‌ای سه‌بعدی مورد استفاده قرار می‌گیرد. کاربردها و قابلیت‌هایی که نه‌تنها برای شتاب دادن به فرآیند ایجاد گرافیک‌هایی با کارآیی بالا موثرند، بلکه در انجام بسیاری از کارهای سنگین نیز بسیار خوب عمل کرده و از سرعت و دقت کافی برخوردارند(Tahmasebi,2012).

قدرت سه‌تا از پنج ابررایانه سریع جهان عمدتا به دلیل پردازنده‌های گرافیکی است(Jin,2011). این پردازنده‌ها هرگز با هدف محاسبات با کارایی بالا[5] طراحی نشدند، اما با این حال راه خود را در میان پردازنده‌های قوی باز کرده و در کمتر از ۱۰ سال این بازار را کاملا در دست گرفتند.

[1] Frequency cubed
[2] Parallelism
[3] Cache memory
[4] Parallel Computing
[5] HPC:High Performance Calculation

در اواخر سال ۲۰۰۶(Houzet,2009)، شرکت انویدیا[1]، اولین پردازنده‌گرافیکی همه‌منظوره با نام G80 را راه‌اندازی کرد. به منظور دستیابی به بالاترین کارایی، برنامه‌نویس مجبور است برنامه‌ها را طوری بنویسد که به بهترین شکل از معماری سخت‌افزار بهره بگیرد. این‌کار با استفاده از مدل‌های برنامه‌نویسی که جزئیات معماری سخت‌افزار از قبیل ارتباطات درون-پردازنده‌گرافیکی، زمان‌بندی وظایف و همگام‌سازی را پنهان می‌کنند، ساده‌سازی می‌شود.

امروزه کودا یک مدل برنامه‌نویسی موازی مقیاس‌پذیر و همه‌منظوره برای نوشتن برنامه‌های کاربردی بسیار موازی است(TungYang,2011). در حقیقت کودا، معماری دستگاه یکپارچه محاسباتی است و یک پلتفرم نرم‌افزاری برای محاسبات موازی حجیم با کارایی بالا در پردازنده‌های گرافیکی قدرتمند محسوب می‌شود که به‌صورت توابع مختلف در زبان‌های برنامه‌نویسی برای سازندگان نرم‌افزار قابل دسترس است. برنامه‌نویسان در نسخه خاصی از زبان قدرتمند سی که دارای ضمیمه اِنویدیا است، برنامه‌های خود را می‌نویسند و بعد از کامپایل، قادرند آن‌ها را روی پردازنده‌های گرافیکی اجرا کنند.

مراحل پردازش کودا به ترتیب کپی داده‌ها از حافظه اصلی به حافظه پردازنده‌گرافیکی، دستور انجام پردازش توسط پردازنده‌مرکزی به پردازنده‌گرافیکی، انجام پردازش موازی در هسته‌های پردازنده‌گرافیکی و کپی نتایج از حافظه پردازنده‌گرافیکی به حافظه اصلی می‌باشد(Cai,2012). با توجه به توانایی پردازنده‌های گرافیکی در پردازش موازی در این تحقیق به دنبال یافتن راه‌کاری مناسب، جهت تسریع[2] پردازش و موازی‌سازی یکی از الگوریتم‌های خوشه‌بندی پرکاربرد در داده‌کاوی و توسعه آن به سایر الگوریتم‌های دارای محاسبات سنگین هستیم.

که در این راستا سؤال‌های زیر مطرح می‌شود:

۱) آیا پردازنده‌های گرافیکی همه‌منظوره واحد، می‌توانند در محاسبات با کارایی بالا مؤثر باشند؟
۲) چه راه‌کارهایی برای استفاده بهتر از پردازنده‌های گرافیکی همه‌منظوره وجود دارد؟
۳) آیا راه‌کاری برای پردازش موازی الگوریتم‌های خوشه‌بندی با استفاده از پردازنده‌های گرافیکی وجود دارد؟

[1] NVIDIA
[2] Speedup

۱-۳ سابقه و ضرورت انجام تحقیق

یکی از موضوعات مهمی که بشر در دستیابی به یافته‌های جدید با آن مواجه است مبحث محاسبات سنگین (پیچیده) یا علوم الکترونیکی است. برای اینکه ماشین‌های محاسباتی بتوانند از پس این نوع محاسبات برآیند نیازمند پردازشگرهای بسیار قوی‌تر و فراتر از پردازنده‌های مرکزی امروزی می‌باشند.

در دهه گذشته پردازنده‌های گرافیکی به‌منظور پردازش گرافیک طراحی شده بودند اما در حال حاضر فناوری خاصی با نام جی‌پی‌جی‌پی‌یو[1] در حال تکمیل شدن است که کاربرد آن، استفاده از توان عملیاتی و محاسباتی هسته‌های قدرتمند گرافیکی در بخش‌هایی غیر از محاسبات گرافیکی است که با ابداع میان‌افزارهایی همانند کودا استفاده از پردازنده‌های گرافیکی کاراتر شده است.

معماری‌های پردازشی سلسله مراتبی و سریال به محدودیت‌های فیزیکی خود نزدیک شده‌اند. چنان‌که از سال ۲۰۰۳(Nvidia,2012) به این سو افزایش توان محاسباتی، دیگر مانند گذشته با افزایش فرکانس کلاک هسته پردازنده‌ها امکان‌پذیر نیست.

سیستم‌های پردازش موازی راه‌حلی کارآمد، مقرون به‌صرفه و مقیاس‌پذیر برای فراهم نمودن توان محاسباتی مورد نیاز اکنون و آینده هستند. این خاصیت توازی چیزی است که نقطه قوت فناوری پردازنده‌های گرافیکی محسوب می‌شود.

پردازنده‌های گرافیکی مدرن با معماری موازی خود پردازنده‌های بسیار سریعی به شمار می‌روند، در عین‌حال با قیمت و توان مصرفی کمتری عرضه می‌شوند و جهت پیاده‌سازی الگوریتم‌ها و برنامه‌هایی همچون پروژه‌های برخورد در شتابدهنده‌ها ، اخترفیزیک یا هواشناسی و... یک راه‌کار اقتصادی و کارآمد به شمار می‌روند.

البته برای استفاده آسان‌تر کاربران، باید این فناوری طوری باشد که برنامه‌نویس بدون نیاز به فراگیری واسط‌های برنامه‌نویسی گرافیکی به کمک کتابخانه‌های موجود برای این‌کار، بار پردازشی برنامه خود را به‌سادگی از پردازنده‌مرکزی به پردازنده‌گرافیکی منتقل کند.

[1] GPGPU: General Purpose Graphic Processor Unit

۱-۴ فرضیه‌ها

۱) پردازنده‌های گرافیکی بستر مناسبی برای پردازش موازی هستند.
۲) پردازش موازی الگوریتم‌های خوشه‌بندی با استفاده از مدل برنامه‌نویسی موازی کودا در بستر پردازنده‌گرافیکی امکان‌پذیر است.
۳) زمان اجرای محاسبات الگوریتم خوشه‌بندی فازی به‌صورت موازی و با استفاده از پردازنده‌گرافیکی نسبت به پردازنده‌مرکزی، سریع‌تر است.

۱-۵ هدف‌ها

هدف اصلی در این پژوهش، حصول نتایج کاراتر و تسریع پردازش یکی از الگوریتم‌های خوشه‌بندی، با استفاده بهتر از توان پردازشی پردازنده‌گرافیکی است. برای دستیابی به آن، اهداف زیر دنبال خواهد شد:

۱) استفاده از پردازنده‌های گرافیکی در محاسبات با کارایی بالا.
۲) تمرکز بر روی پردازش کارآمد و کم‌مصرف.
۳) پشتیبانی از پردازش ناهمگن، به‌نحوی که برنامه‌ها هم‌زمان از پردازنده‌مرکزی و پردازنده‌گرافیکی بهره ببرند.
۴) تسریع پردازش الگوریتم خوشه‌بندی فازی با استفاده از مدل برنامه‌نویسی موازی کودا در بستر پردازنده‌گرافیکی.

۱-۶ کاربردها

با توجه به گسترش فناوری پردازنده‌های گرافیکی همه‌منظوره با استفاده از واسط‌های برنامه‌نویسی و بهره‌گیری از روش برنامه‌نویسی موازی و پیاده‌سازی خودکار جریان طراحی (ایجاد انتزاع در سطح سخت‌افزار و کاربرد)، امروزه استفاده از توان پردازشی پردازنده‌های گرافیکی در محاسبات سنگین امکان‌پذیر می‌باشد. این امر موجب شده تا در بسیاری از حوزه‌ها مثل فراگیری ماشین، هوش‌مصنوعی، اکتشاف نفت، پردازش‌تصویر، جبرخطی، آمار، بازسازی سه‌بعدی و حتی

تعیین قیمت در بازار سهام به یکی از پرکاربردترین زمینه‌های تحقیقاتی و صنعتی تبدیل گردد(Davendra,2012). در این تحقیق کاربرد این پردازنده‌ها در تسریع پردازش الگوریتم‌های دارای محاسبات سنگین و پیچیده (همانند الگوریتم‌های خوشه‌بندی پرکاربرد در داده‌کاوی) بررسی می‌شود.

۱-۷ جنبه نوآوری تحقیق

مهم‌ترین هدف این تحقیق، استفاده هم‌زمان از پردازنده‌گرافیکی و پردازنده‌مرکزی در محاسبات ناهمگن است . با توجه به عملکرد متفاوت هر یک از این پردازنده‌ها، از این روش استفاده شده است تا بتوان از توان محاسباتی هر دو پردازنده بهره گرفت. سعی شده است تا یک الگوریتم کاربردیِ محاسبات-محور انتخاب شود.

الگوریتم خوشه‌بندی فازی، به‌دلیل حجم بالای محاسبات در برخی کاربردها عملکرد خوبی ارائه نمی‌دهد و کارایی آن در اجرای سری، کاهش می‌یابد. این الگوریتم علاوه بر محاسبات-محور بودن، تمام ویژگی‌های مورد نیاز موازی‌سازی را دارد. در این تحقیق، پیاده‌سازی الگوریتم خوشه‌بندی فازی به‌صورت موازی و با بهره‌گیری از سخت‌افزار پردازنده‌گرافیکی انجام می‌شود. در این روش اجرای محاسبات نسبت به پردازنده‌مرکزی به‌صورت سریال، سریع‌تر انجام شده و لذا در داده‌های با حجم زیاد، این تسریع موجب بهبود کارایی الگوریتم خواهد شد.

۱-۸ روش تحقیق

روش انجام تحقیق در بخش اول به صورت کتابخانه‌ای، از طریق مطالعه مقالات و تحقیقات علمی، گزارش‌های فنی، کتب، پایان‌نامه‌ها و پروژه‌های تحقیقاتی صورت گرفته و مطالب موجود در سایت‌های اینترنتی علمی و فنی در زمینه موازی‌سازی با پردازنده‌های گرافیکی بوده و سپس در بخش دوم، به صورت آزمایشگاهی از طریق پیاده‌سازی و مقایسه روش‌های موجود و روش پیشنهادی جهت بررسی تسریع پردازش الگوریتم خوشه‌بندی فازی با استفاده از مدل برنامه‌نویسی موازی کودا در بستر پردازنده‌گرافیکی می‌باشد.