****

**Факултет: Мениджмънт и Маркетинг**

**Специалност: Бизнес информатика**

Доклад

***На Тема***

Oценка на производителността.

***Изготвил: Теодор Георгиев Пенев***

***Фак. № 115013***

***Учебна година: 2013/2014***

***Гр. Свищов***

**Съдържание:  
  
Мерки за производителност………………………................................2 стр.  
  
Принципи на подходите за оценка на производителността………..3 стр.   
  
Йерархичен подход за оценка на производителността………..…....7 стр.**

**Експериментални техники за оценка на производителността…...10 стр.**

**Използвана литература : ...................................................... 14 стр.**

Точнотооценяване на производителността е особено важно при високопроизводителнитекомпютри с висока цена, при които потребителите изискватмаксимално бързодействие за приложните си програми.

**Мерки за производителност**

Производителността (бързодействието) на компютърната система отразява скоростта на обработката на данните. Измерва се в брой операции (инструкции), изпълнени за единица време (секунда). Производителността на компютъра зависи главно от два фактора: технологията (елементната база) и компютърната архитектура. Тя обаче зависи и от много взаимно свързани фактори като конкретното приложение, обема на изчислител­ната работа, избрания алгоритъм, човешките усилия за оптимизиране на програмата, способността на компилатора за оптимизация, ефективнос­тта на операционната система, компютърната архитектура и др.

Производителите на големите компютри оценяват поизводителността на апаратната им част с различни мерни единици:

* MIPS- милиони инструкции за секунда;
* MOPS- милиони операции за секунда
* MFLOPS- милиони операции с числа с плаваща запетая за секунда.

За по-големи порядъци на производителността се използват G (гига) или T (тера) операции за секунда.

Традиционно фирмата производител оценява производителността на компютъра на машинно ниво, като тя представлява абсолютната мак­симална стойност, измервана в милиони инструкции за секунда МIРS (Millions of Instructions per Second) или милиони операции за секунда МОРS (Millions of Operations per Second). Тази производителност се на­рича *максимална (апаратна)* и представлява абсолютна оценка на про­изводителността, тъй като отразява апаратното бързодействие на машинно ниво. Измерва се при пълно натоварване на ресурсите на цент­ралния процесор при 50% операции сумиране и 50% операции умноже­ние. При научни и инженерни приложения, изискващи голям обем изчисления над числа с плаваща точка, абсолютната произ­водителност се измерва в милиони операции с плаваща запетая за секун­да MFLOPS. Аналогично при приложения за изкуствен интелект производителността се измерва в брой логически изводи за секунда LIPS (Logic Inferences per Second). В този случай оценките не отразяват непос­редствено апаратните възможности на компютъра, тъй като за реализа­цията на действия от високо ниво са необходими множество машинни операции.

**Принципи на подходите**

**за оценка на производителността**

При сравнение на производителностите на два компютъра трябва да се избягват следните похвати: използването на малки по обем еталон­ни програми, които се побират изцяло в кеша, използването на ограни­чен брой типове данни, неизползването на В/И операции, надценяването на ролята на конкретни инструкции в програмните ядра, сравняването на резултатите от обработката на експерименталните данни, получени по различни методи като например сравняването на експерименталните данни от симулацията на едната компютърна система с тези от програм­ния мониторинг на другата система, използването на компилатор с много добри (лоши) оптимизиращи функции. Интерес представлява изследване­то на влиянието на различните видове високоскоростна памет в процесо­ра (регистрови прозорци, множество регистрови файлове, стандартни регистрови файлове) върху производителността на компютъра при ши­рок спектър компютърни архитектури с различни системи инструкции.

Съществуват множество подходи за оценяване на производително­стта, стремежът при които е да се удовлетворят изискванията на всички заинтересовани страни от измерването на производителността (проек­танти и потребители). За потребителите е особено важно да знаят по какъв начин системната производителност се влияе от архитектурата, компи­латора, операционната система, периферията и др. По този начин те могат да се възползват от спецификата на архитектурните белези и да изберат оптималния стил на програмиране. На проектантите тази оценка ще по­могне при оптимизацията на апаратните средства и програмното осигуря­ване. За реални приложения се определя т.нар. *средна производителност*

на компютъра, която с значително по-ниска от максималната производителност и в общия случай представлява около 40% от максималната производителност.

Системната производителност на компютрите се влияе в значител­на степен от приложната област. Затова се дефинира понятието 3D *компютър,* при който призводителиостта се оценява като точка в тримерното пространство по три направления: бързодействие на централния проце­сор, латентност на системата за прекъсване и ефективност на В/И систе­ма. Бързодействието на централния процесор е от изключителна важност за инженерни и научни приложения с голям обем изчисления над числа с плаваща точка. Латентността на системата за прекъсване е критична за системи за управление в реално време, тъй като тя определя времето за отговор на компютъра за анализ на контролираните параметри на уп­равлявания обект и задействане на адекватна програма за осъществява­нето на обратната връзка за въздействие върху управлявания обект. Бързодействието на В/И система е от съществено значение за икономи­чески приложения, които изобилстват с интензивни обръщания към дис­ка за използване на бази данни и др.

Основен метод за определяне на компютърната производителност е изпълнението на еталонни програми (benchmarks). Клиентите разчитат на еталонните програми при закупува­нето на нов компютър, а програмистите – за избора на оптимален стил за програмиране.

Този метод осигурява полу­чаването както на качествени, така и на количествени оценки на произво­дителността. *Качествената оценка* засяга характеристиките на разработвания компютър и откриването на недостатъци и евентуални тес­ни места. Тя се прави още в етапа на проектиране въз основа на симула­ции с консервативни допускания.

Най-важната *качествена характеристика,* която трябва да бъде определена за даден компютър, е съвместимостта. Обхватът на съвмес­тимостта може да определи програмното осигуряване и периферните устройства, които могат да бъдат използвани, както и взаимстването на програми и файлове. Есте­ствено съществуването на съвместимост при езиците за програмиране, компилаторите и операционните системи е крайно необходимо, тъй като в противен случай потребителят трябва да разработва собствено при­ложно програмно осигуряване, което води до недопустими разходи на време и труд за постигането на реални резултати.

На основата на качествената оценка на производителността могат да бъдат предпочетени определени класове компютри в зависимост от архитектурата им. Едва след окончателното определяне на качествените характеристики може да се пристъпи към *количествени оценки.* Те включ­ват цената, продуктивността, надеждността и бързодействието. *Цената* включва не само средствата за закупуването на компютъра, но и разхо­дите за инсталирането, поддръжката, обучението на обслужващия пер­сонал и разработката на приложните програми. *Продуктивността* в общия случай се дефинира като отношение цена/производителност, но при нейната оценка задължително се включват диалоговите средства, висо­коскоростните графични средства, наличните мощни редактори и на­стройващи програми, както и средствата за документиране на резултатите.

Високата *надеждност* е от особено значение при програми, изиск­ващи голям обем изчислителна работа и съответно голямо машинно вре­ме. Добър показател за надеждността на компютъра е *средното време за отказ.*

Накрая се прави количествена оценка на бързодействието на ком­пютъра. Тя помага да се прецени полезността на компютъра за дадена фирма или организация, стимулира развитието на системното и прилож­ното програмно осигуряване, като дава възможност за ефективно изпол­зване на съществуващите архитектури.

Използването на неудачни еталонни програми обаче може да подведе както проектантите така и клиентите и потребителите. Ето защо оценката на производителността трябва да включва:

* Прецизна характеристика на изчислителния товар;
* Първоначално тестване с еталонни програми;
* Тестване с програми, апроксимиращи ежедневното натоварване на компютъра.

Поради относително простата си архитектура по-малките компютри могат да бъдат оценявани с прости еталонни програми (от вида на Whetstones). Производителността на тези компютри не се променя значи­телно при различните приложения. Мощните компютри със значителни архитектурни усъвършенствания и разширения са много по-чувствителни към характеристиките на конкретното приложение, тъй като производи­телността им зависи до голяма степен от степента на използване на архи­тектурните усъвършенствания на компютъра. С елементарните еталонни програми се определя максималната (апаратната) производителност.

**Йерархичен подход за оценяване**

При оценяването на производителността на компютрите с успех се прилага йерархичен подход*,* при който информацията, получена на да­ден етап, подпомага анализа на следващия етап.

На най-ниското ниво на йерархията се изпълняват програми, които тестват базовите операции в компютъра – сумиране, умножение и др.От измерването на тези операции може да се определи времето за запълване на инструкционните конвейери, ефектът от използването на кеша, и др. Програмите, тестващи базовите операции в компютъра, дават възможност да се определи неговата мак­симална (апаратна) производителност.

На второто ниво в йерархията се използват колекции от **програмни ядра,** които представляват части от реални програми или са изкуствено синтезирани. Широко приложение са получили следните колекции про­грамни ядра:

• *Whetstones.* Представляват синтезирани еталонни програми за основна аритметика. Използват се за оценка на производителността на компютри с ниско и средно бързодействие.

• *Dhrystones.* Представляват синтезирани еталонни програми за оценка на операциите в системните програми. Използват се за оценка на производителността при нечислени приложения при малки и средни ком­пютри.

• *Linpack.* Представлява програмна библиотека за линейна алгеб­ра. Използва се за оценка на векторната и скаларната производителност при обработката на числа с плаваща точка от персоналните компютри до мощни компютърни платформи. Основните операции тук са умножение на вектор със скалар и сумиране на вектори.

• *Цикли на Livermore.* Това е колекция подпрограми, извлечени от най-често изпълняваните приложни програми в националната лаборато­рия „Lawrence Livermore" в САЩ. Тя се използва за оценка на производителносттана суперкомпютрите.

Понастоящем за различните приложни области са разработени широка гама еталонни програми като например за приложения с мултимедия, компютърна графика, предпечатна подготовка и т.н.

В общия случай на основата на програмни ядра се оценява само производителността на централния процесор. Много рядко с тях се оце­нява производителността на интегрираната система, тъй като поради изключително простата си структура те могат да формират погрешна представа за системната производителност при цялостния изчислителен товар. Друг недостатък при използването на такива относително прости програми е, че статистиките при индивидуалните оценки на производи­телността варират в широки граници и зависят до голяма степен от из­ползвания компилатор. На третото ниво при йерархичния подход за оценка на производи­телността на компютрите се използват базови подпрограми, извлечени от широк спектър приложения. Те тестват както централния процесор, така и входно-изходната система. На това ниво от йерархията резултатите от изпълнението на ета­лонните програми вече представляват интерес за крайния потребител, тъй като те дават информация не само за производителността, а също така и за паралелните структури и други оптимизиращи фактори.

На четвъртото ниво в йерархията се използват съкратени версии на основните програми в изчислителния товар. С тяхна помощ се оценяват взаимодействията между базовите подпрограми. Недостатък на използването на съкрате­ните версии на програмите е, че са облекчени изискванията за памет или вход – изход, поради което има опасност да се загуби оригиналната про­грамна структура.

На петото ниво от йерархията може да се получи най-пълна и точна картина на системната производителност при решаването на типични проблеми, но въпреки това съществуват и редица недостатъци.

Целта при най – високото, шестото ниво на йерархията, е да се из­следват насоките при формирането на бъдещия изчислителен товар, как­то и новите подходи при създаването на алгоритми и архитектури.

**Експериментални техники за**

**оценка на производителността**

Целта на проектанта на компютри е да създа­де компютърна система, която ще успее на пазара. Това изисква опти­мално използване на наличните хардуерни и софтуерни технологии. Основната цел е откриването и отстраняването на тесни места в проекта на компютъра. За тази цел проектантът на компютри разполага със след­ните алтернативи: изследване чрез аналитични модели като например веригите на Марков или да използва експериментални техники като на­пример симулация по следи (trace-driven simulation), симулация по съби­тия (event-driven simulation) или апаратен мониторинг на съществуващи прототипи.

Основен е проблемът за оптимиране на съотношението между софтуера и хардуера в компютърната система. Възможно ли е да се създа­де високопроизводителна система от несложен хардуер и несложни ком­пилатори? Дебатът RISC/CISC, воден през последните години, е показателен за тази алтернатива.

Експерименталните подходи за оценка на качествените и количе­ствените аспекти на производителността на компютърните системи включват изпълнение на програми както от реални, така и от виртуални (симулирани) апаратни средства. В случаите на симулиран хардуер, по­ведението на системата се формира на основата на получените статис­тики, отразяващи динамиката на изчислителния процес.

Симулацията (имитационното моделиране) представлява основна част от експерименталните изследвания на компютърните архитектури и заема важно място в откриването на тесните места и постигането на оп­тимална производителност.

*Симулацията по следи* дава възможност да се акумулират данни, без да се намалява ефективността на моделираната система. Тя също осигурява възможност лесно да се променят моделираните системни па­раметри и изискванията за получаваните данни. Методът може да се из­ползва с лекота за получаване на данни при съществуващи продукти или експериментални прототипи. Този вид симулация обаче има два основ­ни недостатъка: ниска скорост на натрупване на статистиките и непер-фектността на модела. Фактът, че данните се натрупват няколко пъти по-бавно от бързодействието на изследваната система, е показателен, че не е възможно да се добие пълна представа за реалните събития. Въпреки това този вид симулация е добра първа стъпка за детайлното изследване на системата. По-нататък, когато се обединят и други експериментални техники, симулационният модел може да бъде потвърден, което пък оси­гурява механизма за даване на бърз отговор в коя посока да се оптимизи­ра системата.

При наличие на реален хардуер могат да се използват три метода: софтуерен мониторинг, фърмуерен мониторинг и апаратен мониторинг.

*Програмният мониторинг* е приложим както при реални компютри, така и при експериментални прототипи. И в двата случая е полезен за проверка на адекватността на симулационния модел. Така сравняването на данните, получени при програмния мониторинг, със симулационните резултати могат да доведат до разработването на по – добра методология за симулация. Основни недостатъци са ниската скорост на събирането на тези данни, както и липсата на гъвкавост (невъзможността да се променят параметрите на изследваната система) за разлика от симулационните методи.

*Апаратният мониторинг* както и програмния мониторинг може да се прилага при съществуващи машини и експериментални прототипи. И в двата случая цената е висока и необходимото време е голямо. Пре­димството е високата достоверност на информацията.

*Фърмуерният мониторинг* (измерване чрез изпълнение на микропрограми) заема средно място между програмния и апаратния монито­ринг. Както при апаратния мониторинг устройството за мониторинг не влияе на достоверността на резултатите. От друга страна подобно на програмния мониторинг не е скъп и има възможност за лесна промяна.

В резюме може да се обобщи, че при изследването на компютърни­те архитектури е наложително използването на експериментални техни­ки както в етапа на проектирането така и след създаването на реалната машина. Поради високата сложност на компютърните системи е необхо­димо да се провери първо фундаменталната концепция чрез екстензивна симулация, след което да се създаде и тества прототип на системата, при което се потвърждава рационалността и работоспособността на концеп­цията и се събира информация, полезна при следващата итерация от раз­работката на модела. Полезността на един метод (например симулацията) може да бъде подобрена като се съчетаят изводите, получени при използването на този метод с резултатите, получени при приложението на друг метод (например изпълнението на програмата от прототип).

С цел да се изследва взаимодействието и влиянието на различните фактори в компютърната система, е необходимо експерименталното из­следване на компютърната производителност да се извърши на няколко йерархични нива, като се оползотворяват максимално предимствата на всяко ниво и се отчитат съответните ограничения. При това се обобща­ват резултатите от всички нива за формиране и доказване на общата тен­денция. Този подход дава възможност за откриване на „тесни места" в сложните компютърни системи, които възпрепятстват оптималното из­ползване на наличната технология.

**Използвана литература :**

[http://programist-bg.com](http://programist-bg.com/)

[http://www.referati.org](http://www.referati.org/)