

Paralel rekenen met Linux-pc's

DOEKO HOMAN

Tot voor kort was er een vrij strakke scheiding tussen de wereld van pc's, en die van workstations en supercomputers. Pc's werkten onder Microsoft en Windows, workstations en supercomputers hebben UNIX als operating system. Pc's worden (onder meer) gebruikt voor tekstverwerking. Workstations en supercomputers voor (grootschalig) rekenwerk.

Door de opkomst van Linux, een UNIX-variant voor pc's, kan men het vertrouwde UNIX nu ook op pc's gebruiken. En met MPI (Message Passing Interface) erbij kan men meerdere pc's laten werken als computer. Paralel rekenen op pc's met Linux en MPI.

[Linux en MPI](#)
[Paralel rekenen](#)
[Paralel rekenen op pc's](#)
[IT-Lab opstelling](#)
[Werken met MPI](#)
[Account en cluster pc's](#)
[Enkele cijfers](#)
[Literatuur](#)

▲ Begin pagina ▲

Linux en MPI

Alweer een aantal jaren geleden is beschreven hoe pc's met MS DOS en het public domain-pakket PVM als parallele rekenmachine konden worden gebruikt. Ondertussen is MS-DOS naar een Windows-omgeving geïmplementeerd en is Linux met MPI binnen ieders bereik gekomen, zowel naast Windows op een pc, als 'standalone' op de pc, dat wil zeggen zonder MS DOS of Windows eraan.

Voor parallel rekenwerk kun je op eenvoudige wijze een aantal van deze Linux-pc's inrichten en gebruiken als parallele numbercruncher. Voor minder dan f 100,- kan een Linux-versie worden gekocht inclusief onder meer C en Fortran77 compilers. Daarnaast is via het public domain een versie van MPI verkrijgbaar. Daarmee is parallele verwerking van C en/of Fortran-programma's mogelijk.

▲ Begin pagina ▲

Paralel rekenen

Waarom parallel rekenen? Omdat dan de resultaten van het rekenwerk eerder beschikbaar kunnen zijn. Duurt de oplossing van een probleem bijvoorbeeld 2 dagen rekenen op je pc, en is het probleem goed parallel te programmeren, dan is met 2 van die pc's de oplossing na 1 dag rekenen beschikbaar. Grootschalig rekenwerk zal er wel altijd blijven, en de moderne supercomputers werken allemaal met vele processoren parallel. MPI is bijna altijd wel beschikbaar op dergelijke machines (maar mogelijk niet altijd het meest efficiënt).

▲ Begin pagina ▲

Paralel rekenen op pc's

Een eerste vereiste is de toegang tot meerdere pc's of een pc met meer processoren. En er moet regelmatig sprake zijn van rekenintensieve klussen. Voor kortdurend rekenwerk hoeft je de moeite niet te doen het programmeren, dan is met 2 van die pc's de oplossing na 1 dag rekenen beschikbaar. Grootschalig rekenwerk zal er wel altijd blijven, en de moderne supercomputers werken allemaal met vele processoren parallel. MPI is bijna altijd wel beschikbaar op dergelijke machines (maar mogelijk niet altijd het meest efficiënt).

▲ Begin pagina ▲

IT-Lab opstelling

Voor testwerk is in het IT-Lab van het RC een pc beschikbaar waarop de Red Hat Linux versie 5.2 geïnstalleerd is. Het systeem heeft een 166 Mhz processor en 128 MB geheugen en is puur bedoeld als 'reken-server'. C en Fortran 77 zijn aanwezig, evenals Telnet en FTP. Verwacht echter geen zaken als webbrowsers, luxe mailers (al kan mail verzenden wel), tekstopmaaksystemen en dergelijke. Hoewel dat allemaal wel beschikbaar is onder Linux, is het hier niet geïnstalleerd. We varen dit keer niet onder de vlag van het RC als centrum voor ICT, maar onder de oude vlag van 'Reken'centrum.

▲ Begin pagina ▲

Werken met MPI

Op de pc in het IT-Lab is ook het bovengenoemde pakket MPI geïnstalleerd. Het betreft de public domain-versie die beschikbaar is via anonymous FTP op <ftp.mcs.anl.gov/pub/mpl/mpich-1.1.2.tar.gz> (Deze versie is ook als tar-file op de machine van het IT-Lab aanwezig). De installatie hiervan op een Linux-pc is recht door zee. Van de ongeveer 50 MB diskruimte die nodig is bij de installatie wordt uiteindelijk door een werkende MPI-versie nog geen 4 MB in beslag genomen. Een installation guide en user guide zijn ook aanwezig.

Wie een Linux-pc heeft en toegang wil tot een pc om wat testen met parallel rekenen te doen, moet allereerst het pakket MPI op de pc geïnstalleerd hebben. Om te testen of MPI correct werkt, kan volstaan worden met de eigen pc. Op die pc met slechts 1 CPU zal weliswaar geen tijdswinst geboekt worden door MPI te gebruiken, maar wel is dan duidelijk of de zaak goed geconfigureerd is. Aanpassingen zijn nodig aan de `~/.rhosts` file en aan `/etc/hosts.allow` en `/etc/inetd.conf`, want de 'rsh' zal moeten werken. Men moet zich echter goed realiseren dat al te ruimhartig toelaten van deze faciliteiten de security benadeelt (en dus de kans op inbraak vergroot).

Bij MPI is een behoorlijk uitgebreide testset aanwezig, maar enkele tests daarin zullen fout gaan bij een te krap geheugen. (Een 32 MB-geheugen zal geen 16 MB lengte communicatie kunnen verwerken). Ook andere programma's kunnen dan gaan 'hangen'. Draai de complete testset dus zoveel mogelijk stand-alone.

▲ Begin pagina ▲

Account en cluster pc's

Heeft men een correct werkende MPI op de eigen pc, dan kan de pc in het IT-Lab gebruikt worden als testmachine voor parallelle verwerking. Een account op die machine is aan te vragen door een e-mail te sturen naar d.h.homan@rc.rug.nl onder vermelding van de gewenste username en het IP-adres van de eigen pc. Het eenvoudigst is dan om op de machine van het IT-Lab dezelfde directory-structuur te hebben als op de eigen pc. Ook zal het MPI-programma net zo op de IT-Lab machine beschikbaar moeten zijn als op de eigen pc. En aan MPI zal bekend gemaakt moeten worden dat er meerdere pc's beschikbaar zijn (dat wil zeggen, de file `/usr/local/mpl-1.1.2/util/machines/machines.LINUX` zal de naam van de IT-Lab pc moeten bevatten).

▲ Begin pagina ▲

Enkele cijfers

Om het effect van parallel rekenen nog eens inzichtelijk te maken, volgen hier enkele cijfers van een uitstekend parallel programma (in dit geval: cpi- een C-programma voor de benadering van het getal pi), gedaan op de pc op de werkplek (100 Mhz) en de pc in het IT-Lab (166 Mhz). Het rekenwerk wordt gelijk over alle processen verdeeld.

Aantal parallelle processen		Wall clock (turn-around time
Werkplek	IT-Lab	(seconds)

1	0	89
1	1	45
2	1	59
2	2	45

Omgekeerd, als het 'master process' op de IT-Lab machine draait:

Aantal parallele processen		Wall clock (turn-around time
IT-Lab	Werkplek	(seconds)
1	0	54
1	1	44
2	1	36
2	2	44
3	2	35

Deze getallen weerspiegelen uitstekend de snelheden van de CPU's op de beide machines waarbij de theoretisch beste turnaround tijd (bijna 35 seconden bij 5 resp. 3 processen) al praktisch bereikt is bij 3 resp. 2 processen.

Literatuur

- William Gropp, Ewink Lusk, Anthony Skjellum: Using MPI (MIT Press)
- Kai Dupke, Ralph Huelselbusch: LINUX als Supercomputer (magazin f?fessionelle Informationstechnik, Februar 1999, 52)