MPI on a non-dedicated "cluster"

Markus Osterhoff

August 25, 2009

1 about me...

2 Simulation / Boundary Conditions

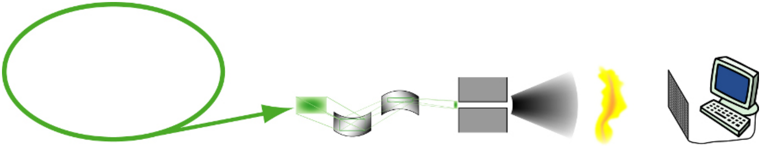
3 Solutions

about me...

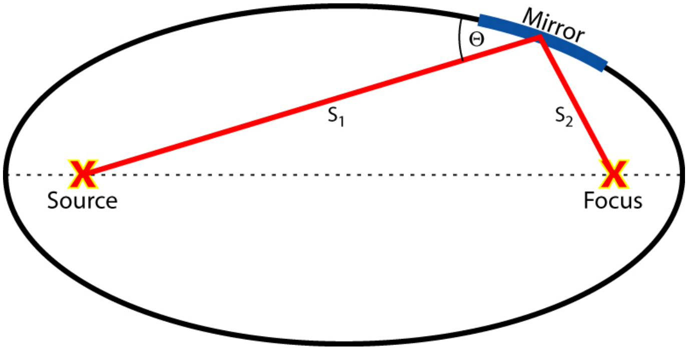
- Markus Osterhoff
- PhD student @ . . .

- Institute for x-ray physics, Göttingen
- European Synchrotron Radiation Facility, Grenoble

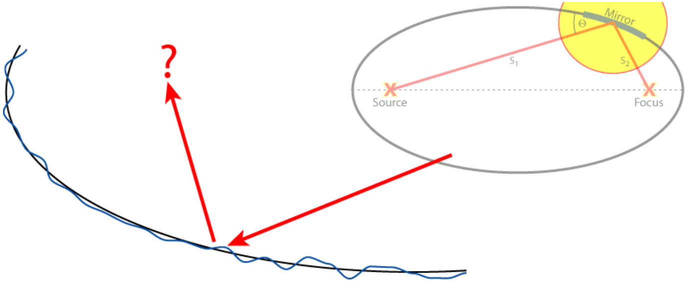
What is an X-Ray Experiment?



Mirror / Multilayer-Mirror



Mirror / Multilayer-Mirror



My Research Topic

Numerical simulation / optimization of MLM...

- to get a sharper focus
- to get more efficiency

Investigations on...

- effects of roughness / surface errors
- effects of diffusion / fabrication errors
- misalignment, vibration, ...

My Research Topic

Numerical simulation / optimization of MLM. . .

- to get a sharper focus
- to get more efficiency

Investigations on. . .

- effects of roughness / surface errors
- effects of diffusion / fabrication errors
- misalignment, vibration, . . .

The Past

- So far, geometrical calculations have been carried out. . .
- . . . , but they cannot account for volume diffraction
- or coherence, . . .

The Past

- So far, geometrical calculations have been carried out...
- ..., but they cannot account for volume diffraction
- or coherence, ...

The Present

- xms - X-ray Mirror Simulation
- written in C with MPI

- can simulate only "simple mirrors"
- (because IRP is building a new beamline @ Petra 3, Hamburg)

The Present

- xms - X-ray Mirror Simulation
- written in C with MPI

- can simulate only "simple mirrors"
- (because IRP is building a new beamline @ Petra 3, Hamburg)

The Future:

- after CSC, a major redesign (C++, . . .) is appropriate
- M and ML as

The Present

- xms - X-ray Mirror Simulation
- written in C with MPI

- can simulate only “simple mirrors“
- (because IRP is building a new beamline @ Petra 3, Hamburg)

The Future:

- after CSC, a major redesign (C++, . . .) is appropriate
- M and ML as

Parallelization

- A full wave-optical treatment (parabolic wave equation) needs ≈ 4 months (for reasonable geometry)
- parallelization needed!

- But PWE only scales logarithmically
  (solving a tri-band matrix equation)

Parallelization

- A full wave-optical treatment (parabolic wave equation) needs ≈ 4 months (for reasonable geometry)
- parallelization needed!

- But PWE only scales logarithmically (solving a tri-band matrix equation)

- idea: mixture of Monte Carlo methods + Phase Ray Tracing + Wave Optics inside the ML

Parallelization

- A full wave-optical treatment (parabolic wave equation) needs ≈ 4 months (for reasonable geometry)
- parallelization needed!

- But PWE only scales logarithmically
  (solving a tri-band matrix equation)

- idea: mixture of Monte Carlo methods + Phase Ray Tracing + Wave Optics inside the ML

MPI

MPI: Message Passing Interface

- MPI is a language-independent communications protocol used to program parallel computers. Both point-to-point and collective communication are supported.

MPI

MPI: Message Passing Interface

- MPI is a language-independent communications protocol used to program parallel computers. Both point-to-point and collective communication are supported.

  MPI "is a message-passing application programmer interface, together with protocol and semantic specifications for how its features must behave in any implementation."

MPI

MPI: Message Passing Interface

- MPI is a language-independent communications protocol used to program parallel computers. Both point-to-point and collective communication are supported.

  MPI "is a message-passing application programmer interface, together with protocol and semantic specifications for how its features must behave in any implementation."

  MPI's goals are high performance, scalability, and portability. MPI remains the dominant model used in high-performance computing today.

  (wikipedia.org)

MPI

MPI: Message Passing Interface

- MPI is a language-independent communications protocol used to program parallel computers. Both point-to-point and collective communication are supported.

  MPI "is a message-passing application programmer interface, together with protocol and semantic specifications for how its features must behave in any implementation."

  MPI's goals are high performance, scalability, and portability. MPI remains the dominant model used in high-performance computing today.

  (wikipedia.org)

ESRF cluster

- Boundary condition: ESRF "cluster"

- not a real cluster, but

  40 × 4 cores

ESRF cluster

- Boundary condition: ESRF "cluster"
- not a real cluster, but

  40 × 4 cores
- used by staff + users

  (users = guests who carry out experiments at the beamlines)

  (24/7)

ESRF cluster

- Boundary condition: ESRF "cluster"
- not a real cluster, but

  40 × 4 cores
- used by staff + users

  (users = guests who carry out experiments at the beamlines)

  (24/7)

ESRF cluster

- so we have a non-dedicated bunch of computers...
- where suddenly a Matlab uses 70% of the memory,

ESRF cluster

- so we have a non-dedicated bunch of computers. . .
- where suddenly a Matlab uses 70% of the memory,
- where suddenly a mis-configured Condor starts 100 processes (on 4 cores),

ESRF cluster

- so we have a non-dedicated bunch of computers. . .

- where suddenly a Matlab uses 70% of the memory,
- where suddenly a mis-configured Condor starts 100 processes (on 4 cores),
- where suddenly the storage server is overloaded,
- . . .

ESRF cluster

- so we have a non-dedicated bunch of computers. . .

- where suddenly a Matlab uses 70% of the memory,
- where suddenly a mis-configured Condor starts 100 processes (on 4 cores),
- where suddenly the storage server is overloaded,
- . . .

comparison

overview of existing methods & xmls

cluster	. . . @home	condor	xmls

N × M real cluster

dedicated cluster with dedicated cores:

- fast bandwidth between nodes
- no bothering of other users
- not bothered by other users

N × M real cluster

dedicated cluster with dedicated cores:

- fast bandwidth between nodes
- no bothering of other users
- not bothered by other users

seti@home etc.

bunch of distributed PCs

- when online, contact master
- get rather large computing jobs
- submit when done

- aggregate results

seti@home etc.

bunch of distributed PCs

- when online, contact master
- get rather large computing jobs
- submit when done

- aggregate results

condor

job farming

- start same code with different parameters
- store results

- aggregate results

condor

job farming

- start same code with different parameters
- store results

- aggregate results

xmls

combination of both:

- `kernel` running on N nodes (using 1 core)
- interface via `shm` to cli/gui

xmls

combination of both:

- `kernel` running on N nodes (using 1 core)
- interface via `shm` to cli/gui

- master distributes parameters
- nodes ask master; load balancing
- after computation, nodes aggregate results (tree-like)

xmls

combination of both:

- `kernel` running on N nodes (using 1 core)
- interface via `shm` to cli/gui

- master distributes parameters
- nodes ask master; load balancing
- after computation, nodes aggregate results (tree-like)

- master does postprocessing + plotting

xmls

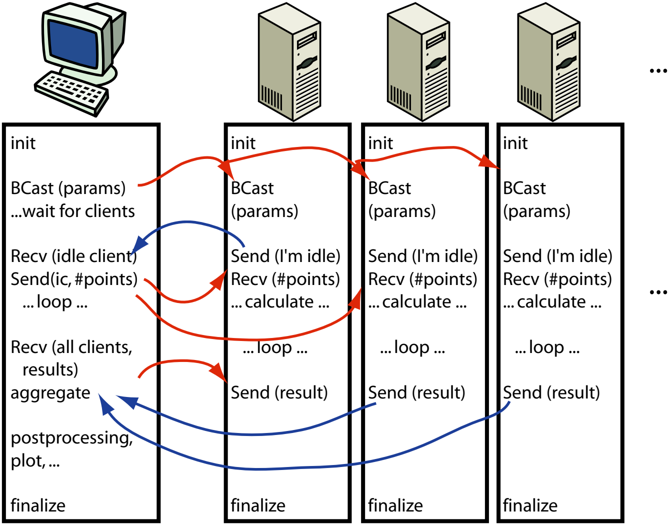
combination of both:

- `kernel` running on N nodes (using 1 core)
- interface via `shm` to cli/gui

- master distributes parameters
- nodes ask master; load balancing
- after computation, nodes aggregate results (tree-like)

- master does postprocessing + plotting

xmls



init

BCast (params)
...wait for clients

Recv (idle client)
Send(ic, #points)
... loop ...

Recv (all clients,
results)
aggregate

postprocessing,
plot, ...

finalize

init

BCast
(params)

Send (I'm idle)
Recv (#points)
... calculate ...

... loop ...

Send (result)

finalize

init

BCast
(params)

Send (I'm idle)
Recv (#points)
... calculate ...

... loop ...

Send (result)

finalize

init

BCast
(params)

Send (I'm idle)
Recv (#points)
... calculate ...

... loop ...

Send (result)

finalize

Comparison & Summary

cluster	...@home	condor	xmls
ideal case	large jobs	medium jobs	this case
dedicated machines	idle	dedicated	not dedicated, nor idle
low latency high load	high latency low load	no communication high load	medium lat. medium ld.

the end

Thank you all for your attendance.