Computing exactly with unsafe resources: fault tolerant exact linear algebra and cloud computing

Majid KHONJI and Clément PERNET, joint work with Jean-Louis ROCH and Thomas STALINSKY

INRIA-MOAIS, Grenoble Univ. France

SAGE Days 16, June 23, 2009

fault tolerant linear algebra

Majid Khonji and Clément Pernet

nroduction

Distributed and Cloud computing Fault tolerance

Parallel Exact linear algeb

Chinese Remainde

code

Principle Correction

verview of the omputation cheme

Further

mprovement

Ordering, sign Early termination Updated computati

Application to computing

Softwares

Outline

Inroduction

Distributed and Cloud computing

Fault tolerance

Parallel Exact linear algebra

Chinese Remainder and Error correcting code

Principle

Correction

Overview of the computation scheme

Further improvements

Ordering, sign

Early termination

Updated computation scheme

Application to computing Determinants

Softwares

Live demo

Perspectives

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and

Fault tolerance

Parallel Exact li

Chinese

emaind

ode

rinciple orrection

verview o

urther

mprovem

Ordering, sign Early terminati

Updated comput scheme

Application to

Softwares

ive demo



Outline

Inroduction

Distributed and Cloud computing

Fault tolerance

Parallel Exact linear algebra

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction



High performance mathematics computing

Numerous application areas

Number theory:
• computing tables of elliptic curves, modular forms.

testing of conjectures,

Crypto: ► Algebraic attacks

Search for big primes

Graph theory: testing conjectures (graph isomorph.,...) Your research pb here!

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

ault tolerance

Parallel Exact linear

hinese

rror correcti

Principle

Correction

scheme

Further

mprovemen

Ordering, sign
Early termination
Updated computation

Application to computing

Softwares

Parallel computations

(Re-)birth of parallel computing

Personal computers ⇒multi/many cores

- End of the race for CPU clock-frequency
- ► Multi-core: 2, 4, 6, ...
- Many-cores: near future, already in GPU's
- ▶ multi-GPU

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Fault tolerance

Parallel Exact linear algebra

ninese lemainde

code Principle

Principle Correction

verview of the omputation cheme

Further

mprovement

Ordering, sign Early termination Updated computatio

Application to computing

Softwares

Parallel computations

(Re-)birth of parallel computing

Personal computers ⇒multi/many cores

- End of the race for CPU clock-frequency
- Multi-core: 2, 4, 6, ...
- Many-cores: near future, already in GPU's
- multi-GPU

HPC ⇒Global computing

- servers
- grids, clusters,
- volunteer computing, Peer to Peer
- cloud computing
- ⇒Shift towards massively parallel computations

INTERNET

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Fault tolerance

Chinese

code Principle

Principle Correction

Overview of the computation scheme

Further

Ordering, sign
Early termination
Updated computation

Application to computing Determinants

oftwares ive demo



Volunteer and Peer to peer computing

Well known successful projects:

- Mersenne Prime search
- ▶ SETI@Home
- Folding@Home/BOINC :
 - ⇒1 Petaflops (670 000 PS3s) in 2007
- **.**..
- ⇒What level of trust?

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

ault tolerance

Parallel Exact linear alge

Chinese

Error corr

ode

Principle Correction

Overview of the omputation cheme

Further

improveme

Ordering, sign
Early termination

scheme

Application to computing Determinants

Softwares

[Above the clouds, a Berkeley view on Cloud Computing, Feb 09]
Difficult agreement on the definition
Based on

- Software As A Service
- Computing centers providing ressources in a pay as you go fashion

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Fault tolerance

Parallel Exact line

Chinese Remaind

code

Principle Correction

verview of the omputation cheme

Further

nprovements

Ordering, sign
Early termination
Updated computation

Application to computing

Softwares Live demo

Parenactivos

reispectives



[Above the clouds, a Berkeley view on Cloud Computing, Feb 09] Difficult agreement on the definition Based on

- Software As A Service
- Computing centers providing ressources in a pay as you go fashion

The interesting thing about Cloud Computing is that we've redefined Cloud Computing to include everything that we already do... I don't understand what we would do differently in the light of Cloud Computing other than change the wording of some of our ads.

Larry Ellison

It's stupidity, It's worse than stupidity: it's a marketing hype campaign.

R. Stallman

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Distributed and Cloud computing

auit toierance Parallel Exact linear alo

Chinese

emainder ar ror correctir de

Principle Correction

Overview of the omputation cheme

Further mproveme

Ordering, sign Early termination Updated compu

Application to computing

Softwares

Interest:

- illusion of infinite computing ressources
- no up-front commitment by cloud users
- processor by the hour, storage by the day

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Distributed and Cloud computing



Interest:

- illusion of infinite computing ressources
- no up-front commitment by cloud users
- processor by the hour, storage by the day
- ⇒for massively parallel computations:

New cost associativity:

100's of computers for $1h \equiv 1$ computer for 100h

⇒What level of trust

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Distributed and Cloud computing



Fault tolerance

algebra Majid Khonji and Clément Pernet

fault tolerant linear

Inroduction

Distributed and Cloud computing

Fault tolerance

Parallel Exact linear algebra

nainder an or correctin

Principle

Correction

omputation cheme

Further

nprovement

ordering, sign arly termination

arly termination Ipdated computa

Application to computing

Softwares

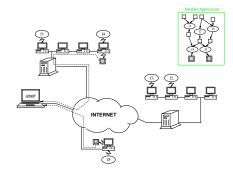
Perspectives

Need to deal with faults of different types

- ► Fail stop (crash failures)
- Network congestion
- Malicious attack
- ⇒Model of the Byzantine failure (not consistently failing)

Global computing

 Grid and P2P: Transparent allocation of the resources to authenticated users
 scheduling supports ressource connections / disconnections



fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud

Fault tolerance

Parallel Exact linear algebra

ninese emainder ar

Principle

verview of the omputation cheme

Further

mprovements

Ordering, sign Early termination Updated computat

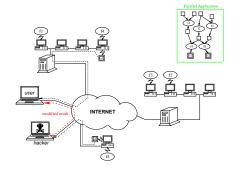
Application to computing Determinants

Softwares Live demo



Global computing and task forgery

 Grid and P2P: Transparent allocation of the resources to authenticated users scheduling supports ressource connections / disconnections



- Yet a task can be **forged** \iff $f(input) \neq \hat{f}(input)$
- ▶ forgery can affect many tasks [e.g. patched client in SETI@home]

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Fault tolerance

Parallel Exact linear algebra

hinese emainder and ror correcting

Principle Correction

Overview of the omputation cheme

Further mprovemen

Ordering, sign Early termination

Updated comput scheme

Application to computing
Determinants

oftwares

State-of-the-Art in Result Certification

- Mainly target programs P composed of independent tasks
- General approach: Replication-based
 - Voting [e.g. BOINC, SETI@home]
 - Spot-checking [Germain-Playez03, based on Wald test]
 - checkpoint and restart ⇒30' to 1h on large clusters!
 - Blacklisting, Credibility-based fault-tolerance [Sarmenta03]
 - Partial execution on reliable resources [Gao-Malewicz04]

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

computing

Fault tolerance

Parallel Exact linear algebra

mainder and or correcting

rinciple

verview of omputation

Further

mprovement

Ordering, sign Early termination Jpdated computation

Application to computing

Softwares

State-of-the-Art in Result Certification

- Mainly target programs P composed of independent tasks
- General approach: Replication-based
 - Voting [e.g. BOINC, SETI@home]
 - Spot-checking [Germain-Playez03, based on Wald test]
 - ► checkpoint and restart ⇒30' to 1h on large clusters!
 - Blacklisting, Credibility-based fault-tolerance [Sarmenta03]
 - Partial execution on reliable resources [Gao-Malewicz04]
- Specific approach: check post-condition on the results
 - ► Eg: Sorting $\mathcal{O}(n \log n)$ Simple-Checker $\mathcal{O}(n)$ [Blum97]
 - The most efficient approach when possible!

Yet, no guarantee of result correction without hypothesis on the attack

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction
Distributed and Cloud

Fault tolerance

Parallel Exact linear algebra

mainder and or correcting

rinciple correction

verview of

urther iprovemen

arly termination lpdated computa cheme

Application to computing Determinants

oftwares

Approach: Massive versus localized attacks

In practice:

- DAG of tasks, highly parallel

Yet, no blind trust:

► large scale falsifications are possible

can be detected by trustable verification of randomly selected tasks:
Extended Monte-Carlo Certification EMCT

[Krings&al 06]

• few falsifications are possible

corrected by Algorithm-based fault tolerance (ABFT)

[Beckman 93, Plank&al 97, Saha 2006]

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Fault tolerance

Parallel Exact linear algebra

hinese

ode

Principle Correction

Overview of the computation scheme

Further

Ordering sign

Ordering, sign Early termination Updated computation scheme

Application to computing

Softwares

Approach: Massive versus localized attacks

In practice:

- DAG of tasks, highly parallel

► Yet, no blind trust:

► large scale falsifications are possible

can be detected by trustable verification of randomly selected tasks:

Extended Monte-Carlo Certification **EMCT**

[Krings&al 06]

▶ few falsifications are possible can be efficiently corrected by Algorithm-based fault tolerance (ABFT) [Beckman 93, Plank&al 97, Saha 2006]

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Fault tolerance

Parallel Exact linear algebra

hinese

error cor

Principle Correction

Overview of the computation scheme

Further

mprovement

Ordering, sign
Early termination
Updated computation

Application to computing

Softwares



Idea: incorporate the redundancy in the algorithm

⇒use some specific properties of the
problem

fault tolerant linear algebra

Majid Khonji and Clément Pernet

nroduction

Distributed and Cloud computing

Fault tolerance

Parallel Exact linear algebra

Chinese Remainde

Error corr code

> Principle Correction

rerview of the mputation heme

Further

nprovements

ordering, sign

Jpdated computa

Application to computing Determinants

Softwares

Idea: incorporate the redundancy in the algorithm

⇒use some specific properties of the
problem

Example: Matrix vector product [Dongarra 2006]



u

uA

fault tolerant linear algebra

Majid Khonji and Clément Pernet

nroduction

Distributed and Clo

Fault tolerance

Parallel Exact linear algebra

ninese emainde

rror corr ode

Principle Correction

Overview of the omputation cheme

Further

mprovemen

Ordering, sign Early termination

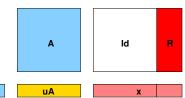
Application to computing

Softwares

vo domo

Idea: incorporate the redundancy in the algorithm ⇒use some specific properties of the problem

Example: Matrix vector product [Dongarra 2006]



fault tolerant linear algebra

Majid Khonji and Clément Pernet

nroduction

Distributed and

Fault tolerance

Parallel Exact linear algebra

inese mainder a

Error corr code

Principle Correction

overview of the omputation

Further

improvemer

Ordering, sign Early termination

Updated comput

Application to computing Determinants

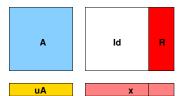
Softwares

live demo



Idea: incorporate the redundancy in the algorithm ⇒use some specific properties of the problem

Example: Matrix vector product [Dongarra 2006]



- precompute the product $B = A \times I$
- ightharpoonup compute x = uB in parallel
- decode/correct x

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Fault tolerance



Parallel Exact linear algebra

Mathematics is the art of reducing any problem to linear algbra

W. Stein

⇒the core to be optimized for a lot of computations

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Distributed and Cloud computing

Parallel Exact linear algebra

Chinese Remainder and Error correcting code

Principle Correction

Overview of the computation scheme

urther

Ordering, sign
Early termination
Updated computatio

Application to computing Determinants

Softwares Live demo



Parallel Exact linear algebra

Mathematics is the art of reducing any problem to linear algbra

W. Stein

- ⇒the core to be optimized for a lot of computations
- Matrix multiplication over GF(p)
- Eliminations: Gauss, Gram-Schmidt (LLL), ...
- Krylov iteration
- Chinese Remainder Algorithm

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Distributed and Cloud computing

Parallel Exact linear algebra

hinese emainder and rror correcting

Principle Correction

Overview of

urther

mprovement

Early termination

Jpdated compu

Application to computing Determinants

Softwares Live demo



Parallel Exact linear algebra

Mathematics is the art of reducing any problem to linear algbra

W. Stein

- ⇒the core to be optimized for a lot of computations
- Matrix multiplication over GF(p)
- Eliminations: Gauss, Gram-Schmidt (LLL), ...
- Krylov iteration
- ► Chinese Remainder Algorithm

fault tolerant linear algebra

Majid Khonji and Clément Pernet

nroduction
Distributed and Cloud
Computing

Parallel Exact linear algebra

ninese emainder and fror correcting

Principle Correction

Overview of the omputation

urther

Ordering, sign
Early termination

Updated compu scheme

Application to computing

Softwares

Outline

Inroduction

Distributed and Cloud computing

Fault tolerance

Parallel Exact linear algebra

Chinese Remainder and Error correcting code

Principle

Correction

Overview of the computation scheme

Further improvements

Ordering, sign

Early termination

Updated computation scheme

Application to computing Determinants

Softwares

Live demo

Perspectives

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Fault tolerance

Chinese Remainder and Error correcting

> rinciple Correction

Overview of the omputation cheme

urther

Ordering, sign

Updated comput scheme

Application to computing Determinants

Softwares Live demo

Poreposti

Perspecti

Chinese Remainder Theorem



where $p_1 \times \cdots \times p_k > x$ and $x_i = x \mod p_i \forall i$

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Principle

Chinese Remainder Theorem



where $p_1 \times \cdots \times p_n > x$ and $x_i = x \mod p_i \forall i$

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Principle

 X_k

 $X_{k+1} \dots X_n$

Chinese Remainder Theorem

$$x \in \mathbb{Z}$$
 \longrightarrow $x_1 \mid x_2$

where $p_1 \times \cdots \times p_n > x$ and $x_i = x \mod p_i \, \forall i$

Definition

(n, k)-code:

 $C = \{(x_1, \dots, x_n) \in \mathbb{Z}_{p_1} \times \dots \times \mathbb{Z}_{p_n} \text{ s.t. there exists } x, 0 \leq x < p_1 \dots p_k \text{ and } x_i = x \mod p_i \, \forall i \}.$

fault tolerant linear algebra

Majid Khonji and Clément Pernet

nroduction

Distributed and Cloud computing Fault tolerance

hinese emainder an

Principle

Principie Correction

verview of the omputation cheme

Further

Ordering, sign
Early termination
Updated computation

Application to computing Determinants

Softwares ive demo

Perspectives

4日ト 4周ト 4 三ト 4 三 り 9 0 0

 $X_{k+1} \dots X_n$

Property

$$X \in C$$
 iff $X < M_k$.

$$M_n = p_1 \times \cdots \times p_n$$

$$p_1 \mid p_2 \mid \cdots \mid p_k \mid p_{k+1} \mid p_n$$

$$M_k = p_1 \times \cdots \times p_k$$

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing
Fault tolerance

Chinaga

emainder and ror correcting

Principle

Correction

Overview of the computation

Further

improvements

Ordering, sign

Early termination

Updated computation

Application to computing Determinants

Softwares

vrenoctivos

Transmission channel



Computation

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

Chinese

emainder and ror correcting

Principle

orrection

verview of the mputation

Further

nprovements

Ordering, sign Early termination Updated computation

Application to computing Determinants

Softwares Live demo

Parenactiva

◆□▶◆□▶◆団▶ ■ めの○

Noisy Transmission channel

Unsafe Computation

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

Chinese

lemainder and rror correcting

Principle

orrection

verview of the mputation

Further

mprovements

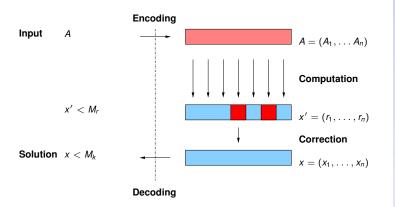
Ordering, sign Early termination Updated computation

Application to computing Determinants

Softwares

. .

Noisy Transmission channel ≡ Unsafe Computation



fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing
Fault tolerance

Parallel Exact linear algebra

ror coi ide

Principle

Overview of the omputation cheme

Further

Ordering, sign

Early termination Updated computation

Application to computing Determinants

Softwares

Property of the code

Redundancy : r = n - k

$$M_n = p_1 \times \cdots \times p_n$$

$$p_1 p_2 \dots p_k p_{k+1} \dots p_n$$

$$M_k = p_1 \times \cdots \times p_k$$

Detects up to *r* errors:

For
$$X' = X + E$$
 with $X \in C$, $||E|| \le r$,

$$X' > M_k$$

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

Chinaga

emainder ar

Principle

Correction

verview of the omputation cheme

Further

improvement

Ordering, sign
Early termination
Updated computation

Application to computing

Softwares

ve demo

Property of the code

Redundancy :
$$r = n - k$$

$$\underbrace{\begin{array}{c|cccc}
M_n = p_1 \times \cdots \times p_n \\
\hline
p_1 & p_2 & \cdots & p_k & p_{k+1} & p_n \\
M_k = p_1 \times \cdots \times p_k
\end{array}}_{M_k = p_1 \times \cdots \times p_k}$$

Detects up to *r* errors:

For
$$X' = X + E$$
 with $X \in C$, $||E|| \le r$,

$$X' > M_k$$
.

- ▶ Distance: *r* + 1
- ▶ Corrects up to $\left|\frac{r}{2}\right|$ errors: the closest code word

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing
Fault tolerance

Uninese Remainder

Principle

Correction

verview of the omputation cheme

Further

mprovement

Ordering, sign
Early termination
Updated computa

Application to computing

oftwares

е аетто

Property of the code

Redundancy :
$$r = n - k$$

$$M_n = p_1 \times \cdots \times p_n$$

$$p_1 \mid p_2 \mid \dots \mid p_k \mid p_{k+1} \mid \dots \mid p_n$$

$$M_k = p_1 \times \cdots \times p_k$$

Detects up to *r* errors:

For
$$X' = X + E$$
 with $X \in C$, $||E|| \le r$,

$$X' > M_k$$
.

- ▶ Distance: r + 1
- ► Corrects up to $\left|\frac{r}{2}\right|$ errors: the closest code word

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Principle

Correction

[Mandelbaum 78]

CRT

- $\qquad \qquad = p_1 \times \cdots \times p_n,$
- $\blacktriangleright \pi_i = \pi/p_i$

$$x = \sum x_i \pi_i y_i$$

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Parallal Event line

- . . .

Chinese

emainder and rror correcting

Principle

Correction

Overview of the omputation cheme

urther

nprovements Ordering, sign

Early termination Updated computa scheme

Application to computing Determinants

Softwares Live demo

Correction

[Mandelbaum 78]

CRT

$$\qquad \qquad = p_1 \times \cdots \times p_n,$$

$$\pi_i = \pi/p_i$$

$$y_i = \pi_i^{-1} \mod p_i.$$

$$x=\sum x_i\pi_iy_i$$

- ▶ Error *e* of weight $t \leq \lfloor \frac{r}{2} \rfloor$
- \triangleright C: the product of the p_i of the errors

$$x = x' - e$$

$$= x' - \sum_{\text{supp}(e)} x_i \pi_i y_i$$

$$= x' - \frac{M_n}{C} B'$$

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

Parallel Exact linear algebra

nese mainder ar or correctir

Principle Correction

Correction

Overview of the computation scheme

Further mprovemen

Ordering, sign Early termination Updated computat

Application to computing

Softwares Live demo

Correction

Theorem

If
$$e \leq (n-k) \frac{\log p_{\min} - \log 2}{\log p_{\max} + \log p_{\min}}$$

$$\left|\frac{x'}{M_n} - \frac{B'}{C}\right| \le \frac{1}{C^2}$$

 \Rightarrow use continued fractions to approximate $\frac{\chi'}{M_0}$

 \Rightarrow recover $B'/C \Rightarrow e$

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Correction

Continued fractions

Continued fractions: Every rational can be written as

$$\frac{p}{q} = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \dots}} = [a_0, \dots, a_n]$$

Convergents: a truncation $\frac{p_i}{q_i} = [a_0, \dots, a_i]$ for $i \le n$

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

computing
Fault tolerance

Chinese

emainder and rror correctin

Principle

Correction

Overview of the omputation cheme

Further

mprovements

Ordering, sign Early termination Updated computation scheme

Application to computing Determinants

Softwares Live demo

Continued fractions

Continued fractions: Every rational can be written as

$$\frac{p}{q} = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \dots}} = [a_0, \dots, a_n]$$

Convergents: a truncation $\frac{p_i}{q_i} = [a_0, \dots, a_i]$ for $i \leq n$

Property

$$\left| \frac{p}{q} - \frac{p_i}{q_i} \right| \leq \frac{1}{q_i^2}$$
 and is decreasing.

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

arallel Exact linear alg

iinese mainde

ode

Tinciple

Correction

Overview of the computation scheme

Further

mprovements

Ordering, sign
Early termination
Updated computation

Application to computing Determinants

Softwares Live demo

Continued fractions

Continued fractions: Every rational can be written as

$$\frac{p}{q} = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \dots}} = [a_0, \dots, a_n]$$

Convergents: a truncation $\frac{p_i}{q_i} = [a_0, \dots, a_i]$ for $i \leq n$

Property

$$\left| \frac{p}{q} - \frac{p_i}{q_i} \right| \leq \frac{1}{q_i^2}$$
 and is decreasing.

 \Rightarrow

- ▶ compute iteratively the convergents of $\frac{\chi'}{M_n}$ until $\left|\frac{\chi'}{M_n} \frac{p_i}{q_i}\right| \leq \frac{1}{C^2}$ is satisfied.
- ▶ If $M_n \frac{p_i}{q_i} \in \mathbb{Z}$ and $x = x' e \le M_k$ ⇒Corrected

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

Chinese

Remainder ar Error correctir

rinciple

Correction

Overview of the computation scheme

Further

Ordering, sign
Early termination

Updated computa scheme

Application to computing Determinants

oftwares

Outline

Overview of the computation scheme

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Overview of the computation scheme



Computation resources

3 types of resources:

- ▶ Trusted
 - ▶ 100% secure
- Untrusted
 - no security measures (public machines)
- Semi-Trusted
 - accessible by untrusted machines but with security measures
 - firewall, chroot, input checking...etc

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Parallol Evact lin

rarallel Exact linear algebra

inese mainda

rror con

rinciple

orrection

Overview of the computation scheme

Further

nprovements Ordering, sign

Ordering, sign Early termination

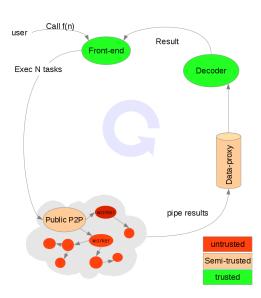
Updated comp

Application to computing

Softwares

ive demo





fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Fault tolerance

Chinese

rror correcting

Principle Correction

Overview of the computation scheme

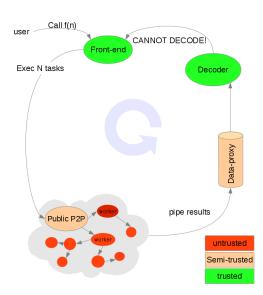
Further

improvement

Early termination
Updated comput

Application to computing Determinants

Softwares



fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and

Fault tolerance

Parallal Event linear o

Chinese

Remaind Error co

Principle

Correction

Overview of the computation scheme

Further

improvemen

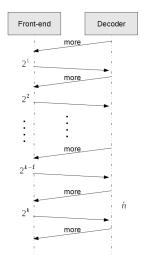
Ordering, sign Early termination Updated comp

Application to computing Determinants

Softwares



- we don't know the number of tasks!
- assuming the error rate is not fixed
- Trade-off: iterations vs #forks



fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

Chinese

Error corr

Principle Correction

Overview of the computation scheme

Furthe

improvements

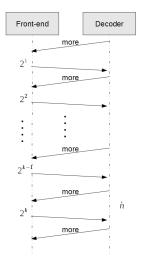
Early termination
Updated computati

Application to computing Determinants

Softwares



- we don't know the number of tasks!
- assuming the error rate is not fixed
- Trade-off: iterations vs #forks



increasing iteratively the number of tasks:

- quadratic growth: 2ⁱ
- ρ-amortized: ρ^{f(i)} with f sub-linear
 - $f(i) = i^{\alpha}$ with $0 < \alpha < 1$

$$f(i) = \frac{i}{\log i}, \dots$$

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

ninese emainde

rinciple

Overview of the computation scheme

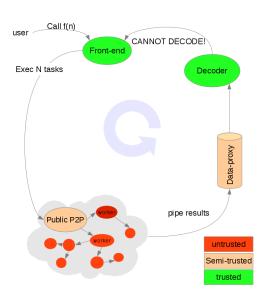
Further improvement

ordering, sign arly termination lpdated computat

Application to computing Determinants

oftwares

Online Post certification scheme



fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Chinese

Remainder and Error correcting

Principle Correction

Overview of the computation scheme

-urthe

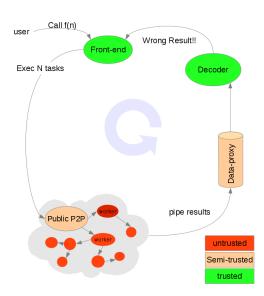
improvement

Ordering, sign
Early termination
Updated computs

Application to computing Determinants

Softwares

Online Post certification scheme



fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Parallel Exact linear algeb

Chinese

Remainder and Error correcting

Principle Correction

Overview of the computation scheme

-urthe

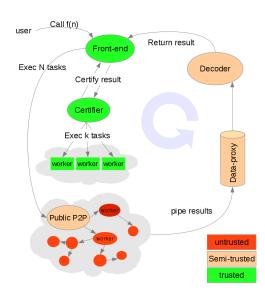
improvements

Early termination Updated comput

Application to computing Determinants

Softwares

Online Post certification scheme



fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and (

Fault tolerance

Parallel Exact linear algebr

Chinese

Remain Error co

Principle

Overview of the computation

scheme Further

improveme

Early termination
Updated comput

Application to computing Determinants

Softwares

Certification process

Read the decoder result R'pick up a random prime p_i compute r = f over \mathbb{Z}_{p_i} reduce $r' = R' \mod p_i$ if r! = r' then Return ERROR.

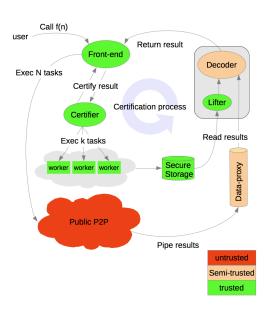
else

Return SUCCESS with a given probability end if

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Overview of the computation scheme



fault tolerant linear algebra

Majid Khonji and Clément Pernet

nroduction

Distributed and Cloud computing

Parallel Exact linear a

Chinese

Remainder Error correc

Principle

Overview of the computation scheme

Further

improveme

Early termination
Updated compu

Application to computing

Softwares

ive demo

Outline

Further improvements

Ordering, sign

Early termination

Updated computation scheme

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Further improvements



Removing the ordering of the moduli

worst case scenario:

every errors having a bigger contribution at the end,

Mandelbaum algorithm requires to sort the moduli p_i :

- ▶ to ensures accurately the t = (n k)/2 correction rate
- ▶ to compute the optimal bound $C_{max} = \prod_{i=n-k+1}^{n} p_i$ for the search of convergents.

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

Parallel Exact linear algebr

hinese lemaind

code

Principle Correction

Overview of the computation scheme

Further mprovement

Ordering, sign

arly termination
pdated computatio

Application to computing

Softwares

Live demo

Removing the ordering of the moduli

worst case scenario:

every errors having a bigger contribution at the end,

Mandelbaum algorithm requires to sort the moduli p_i :

- ▶ to ensures accurately the t = (n k)/2 correction rate
- ▶ to compute the optimal bound $C_{max} = \prod_{i=n-k+1}^{n} p_i$ for the search of convergents.

Relaxation:

- ▶ no sort, only find p_{min} and p_{max},
- more suited for a future on-the-fly model
- only an approximation of the distance (off by 1 in the worst cases)
- ▶ less accurate C_{max} ,

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

Parallel Exact linear algebr

ninese lemaind

ode

Principle Correction

Overview of the omputation cheme

mprovement

Ordering, sign

Early termination Updated computa

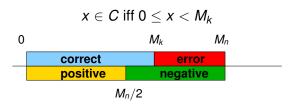
Updated compu scheme

Application to computing Determinants

oftwares ive demo



Dealing with negative numbers



⇒Mandelbaum algorithm only works with positive numbers.

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Parallel Exact linear algebra

Chinese

Error corr

Principle Correction

verview of the omputation cheme

-urther mprovemen

Ordering, sign

Early termination

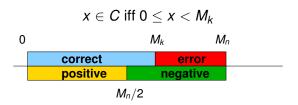
Jpdated computation

Application to computing

Softwares

ive demo

Dealing with negative numbers



⇒Mandelbaum algorithm only works with positive numbers.

(1 mod 5, 1 mod 7, 10 mod 11) could be

- ► -34 with 0 errors
- ▶ 1 with 1 error

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Parallel Exact linear

inese mainder

code

Correction

Overview of the omputation cheme

Further

Ordering, sign

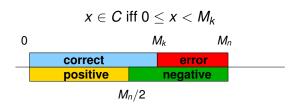
Early termination Updated comput

Application to

computing Determinan

Softwares Live demo

Dealing with negative numbers



⇒Mandelbaum algorithm only works with positive numbers.

(1 mod 5, 1 mod 7, 10 mod 11) could be

- ► -34 with 0 errors
- 1 with 1 error

Work-around solution:

reconstruct both

$$x = CRT(x_1,...,x_n)$$

 $y = CRT(-x_1,...,-x_n)$

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing
Fault tolerance

Chinese

Error corre

Principle Correction

Overview of the computation scheme

urther

mprovement Ordering, sign

rdering, sign arly termination

Updated compositions

Application t

oftwares ive demo

Chinese Remainder Algorithm requires a bound. For Determinant ⇒ Hadamard:

$$|\det(A)| \leq n^n(\max|a_{i,j}|)^{n/2} \leq \prod_i p_i$$

fault tolerant linear algebra

Majid Khonji and Clément Pernet



Chinese Remainder Algorithm requires a bound. For Determinant ⇒ Hadamard:

$$|\det(A)| \le n^n(\max|a_{i,j}|)^{n/2} \le \prod_i p_i$$

Could be **really** pessimistic with

- sparse/structured matrices,
- matrices with special properties,...

fault tolerant linear algebra

Majid Khonji and Clément Pernet



Chinese Remainder Algorithm requires a bound. For Determinant ⇒ Hadamard:

$$|\det(A)| \le n^n(\max|a_{i,j}|)^{n/2} \le \prod_i p_i$$

Could be **really** pessimistic with

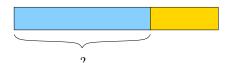
- sparse/structured matrices,
- matrices with special properties,...
- ⇒discover the bound during the computation
 - for each new modular result, reconstruct the result
- \triangleright if it stabilizes k times \Rightarrow correct result with probability $1 - 1/p^k$

fault tolerant linear algebra

Majid Khonji and Clément Pernet



Problem: how to detect errors without a bound?



fault tolerant linear algebra

Majid Khonji and Clément Pernet

Problem: how to detect errors without a bound?



Multiple solution candidates:

Solution x₁ with 1 error corrected

fault tolerant linear algebra

Majid Khonji and Clément Pernet



Problem: how to detect errors without a bound?



Multiple solution candidates:

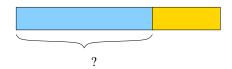
- Solution x₁ with 1 error corrected
- Solution x₂ with 3 error corrected

fault tolerant linear algebra

Majid Khonji and Clément Pernet



Problem: how to detect errors without a bound?



Multiple solution candidates:

- Solution x₁ with 1 error corrected
- Solution x_2 with 3 error corrected
- Solution x₃, with 2 errors corrected

fault tolerant linear algebra

Majid Khonji and Clément Pernet



Problem: how to detect errors without a bound?



Multiple solution candidates:

- Solution x₁ with 1 error corrected
- Solution x₂ with 3 error corrected
- Solution x₃, with 2 errors corrected

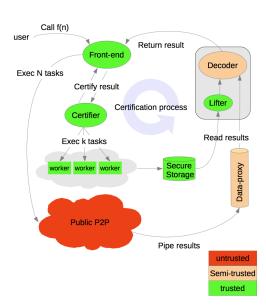
The certifier discriminates the candidates

fault tolerant linear algebra

Majid Khonji and Clément Pernet



Computation scheme



fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and

Fault tolerance

Parallel Exact linear a

Chinese

Remaind Error cor

Principle

verview of the

Further

nprovements

Ordering, sign

Updated computation scheme

Application to computing Determinants

Softwares

ive demo

Multiple decoding

sign: 2 candidates

early term.: multiple candidates, for each possible bound

```
Decoder returns a I of candidates while |I| > 1 do pick a random prime p_i Certifier compute f_i = f \mod p_i sieve un-compatible elements in I if |I| = 1 then return l.front() end if end while
```

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

Chinese

emainder a rror correcti

rinciple

Correction

omputation cheme

Further

mprovem

ordering, sign arly termination

Updated computation scheme

Application to computing

Softwares

Outline

Inroduction

Distributed and Cloud computing

Fault tolerance

Parallel Exact linear algebra

Chinese Remainder and Error correcting code

Principle

Correction

Overview of the computation scheme

Further improvements

Ordering, sign

Early termination

Updated computation scheme

Application to computing Determinants

Softwares

Live demo

Perspectives

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing
Fault tolerance

Parallal Evact li

Chinese

Chinese

emainder ror correc

rinciple

orrection

verview of the imputation heme

-urther

mprovemen

Early termination
Updated complete

Jpdated computa scheme

Application to computing Determinants

Softwares Live demo

_



fault tolerant linear algebra

Majid Khonji and Clément Pernet

nroduction

Distributed and Cloud computing

Fault tolerance

Chinese Remainder and

code Principle

Principle Correction

verview of thomputation cheme

urther

mprovement Ordering, sign

Early terminati Updated comp

cheme

Application to computing Determinants

Softwares

Live demo

Perspectives

Kernel for Adaptive Asynchronous Parallel Interface

Virtualization of the parallel platform

- SMP. distributed architectures
- virtualizes the processors
 - ⇒programmer only focus on a high level description of the parallelism
- Scheduling:
 - static optimization from the data flow graph
 - dynamic using work stealing at fine grain granularity

Parallelism: Kaapi

Example of code

```
struct Fibonacci {
 void operator()( int n,
                                int& result )
       if (n < 2) result =
       else {
                     int subresult1:
                     int subresult2;
                   Fibonacci ()(n-1, subresult1);
                   Fibonacci ()(n-2, subresult2);
                   Sum ()(result, subresult1, subresult2);
};
struct Sum {
 void operator()(
                              int& result,
                           int srl.
                           int sr2 )
   { result =
                             + sr2
                                           ; }
                            31
```

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing
Fault tolerance

01.

lemainder and rror correcting

Principle

Overview of the computation

urther

improvements

Early termination Updated computa

Application computing

computing Determinant

Softwares

Live demo

Parallelism: Kaapi

Example of code

```
struct Fibonacci {
  void operator()( int n, al::Shared w<int> result )
       if (n < 2) result.write( n ):
       else {
           al::Shared<int> subresult1:
           al::Shared<int> subresult2;
           al::Fork<Fibonacci>()(n-1, subresult1);
           al::Fork<Fibonacci>()(n-2, subresult2);
           al::Fork<Sum>()(result, subresult1, subresult2);
};
struct Sum {
  void operator()( al::Shared w<int> result,
               al::Shared w<int> srl.
               al::Shared w<int> sr2 )
   { result.write( srl.read() + sr2.read() ); }
```

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Softwares

Computations

Linear Algebra tasks: LinBox

- Fast MatMul (BLAS, delayed modular reductions, Strassen,...)
- ► Efficient, in-place, block Gaussian elimination

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing
Fault tolerance

Parallel Exact lin

nese nainder and

Principle

verview of th

urther

nprovement

Ordering, sign
Early termination
Updated computation

Application to computing

Softwares

Live demo



Computations

Linear Algebra tasks: LinBox

- Fast MatMul (BLAS, delayed modular reductions, Strassen,...)
- ► Efficient, in-place, block Gaussian elimination

Decoder: Sage

- ease of use for experimenting different prototypes
- good efficiency
- ▶ but BIG executable ⇒hard to distribute

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction
Distributed and Cloud computing
Fault tolerance

hinese emainder and ror correcting

Principle Correction

Overview of the omputation cheme

urther

Ordering, sign

Application to

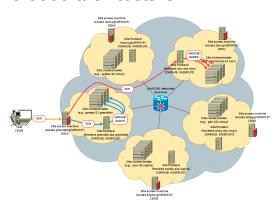
Application to computing

Softwares

Live demo



The Grid 5000 architecture



- Country-wide grid of 9 sites (Nancy, Lyon, Grenoble, Toulouse, Lille, Bordeaux, Orsay, Rennes, Sophia)
- ► Each site has 2-3 clusters of 60-600 computers
- ▶ heterogeneous architecture ⇒3202 processors, 5714 cores

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

I didiloi Exac

lemainder ar rror correctir

Principle Correction

Overview of the computation

Further

Ordering, sign

Early termination

Updated computal scheme

Application to computing Determinants

Softwares

Live demo



Live demo

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing
Fault tolerance

Chinese

Remainder and Error correcting code

Principle

Overview of the computation scheme

urther

Ordering, sign
Early termination

Application to computing

Softwares

Live demo

Outline

Perspectives

fault tolerant linear algebra

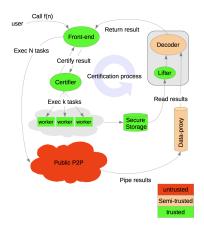
Majid Khonji and Clément Pernet

Perspectives

4 D F 4 P F A B F A B F B

Perspectives

A fully on-the-fly computation model:



- Need a fast CRT working on the fly
- ▶ Refine the amortized technique

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing

Fault tolerance
Parallel Exact linear alge

Chinese

Error co

Principle

Correction

omputation scheme

Further

Ordering sign

Early termination Updated comput scheme

Application to computing Determinant

Softwares

Live demo



Sage?

Revive the Dsage efforts?

distributed computations: Kaapi optional package?

- resource oblivious
- task parallelism, work-stealing, ABFT
- perspectives for GPU/multi-CPU support

P2P, volunteer computing

Spare your cycles and contribute solving BSD conjecture!

fault tolerant linear algebra

Majid Khonji and Clément Pernet

Inroduction

Distributed and Cloud computing Fault tolerance

Parallel Exact linear algebra

nese nainder a er correcti

Principle Correction

Overview of the computation

urther

ordering, sign

Early termination

Jpdated computation

Application to computing Determinants

Softwares ive demo

_

