X-Ability
クロスアビリティ

Acceleration of computational quantum chemistry by heterogeneous computer architectures

Ryota Koga

President of X-Ability Co.,Ltd.

Collaborative researchers

Yuki Furukawa(X-Ability Co.,Ltd.), Koji Yasuda(Nagoya University)

15 Dec 2011

GTC Asia 2011, Beijing

X-Ability
クロスアビリティ

About the X-Ability Co.,Ltd.

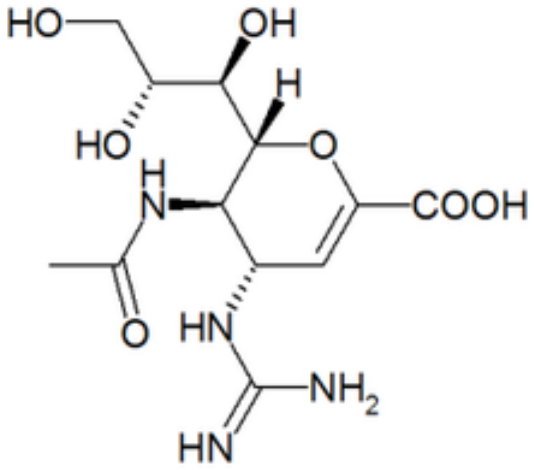
- 3 Office (foundation : 15th Jan 2008)
  Hongo, Tokyo / The K-comp, Kobe / U.S. branch

- 3 Business
  (1) <u>Scientific Computing (mainly Chemistry)</u>
  developing XA-CHEM-SUITE
  (2) Sensor Networking
  (3) ChemBioinformatics (machine-learning etc.)

- By only 4 Regular Board Members

X-Ability
クロスアビリティ

Drugs using Computational Chemistry

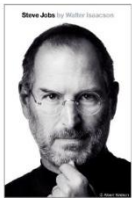


©wikipedia
Zanamivir
trade name **Relenza**

Influenza preventive medicine

©wikipedia
Erlotinib hydrochloride
(trade name **Tarceva**)

Lung cancer, pancreatic cancer



***Steve Jobs* might have used this to extend his life…**

X-Ability
クロスアビリティ

JSST 2011
International Conference on Modeling and Simulation Technology
October 22-23, 2011, Tokai University Takanawa Campus,
Tokyo, JAPAN

Acceleration of computational quantum chemistry by heterogeneous computer architectures

Yuki Furukawa [1], Ryota Koga [1], and Koji Yasuda [2]

[1] X-Ability Co., Ltd., Tokyo, Japan

[2] Ecotopia Science Institute, Nagoya University, Nagoya, Japan

Abstract—Computational quantum chemistry mehods such as the Hartree-Fock (HF), the density functional theory (DFT) or the fragment molecular orbital (FMO) require heavy computational resources. In this study they are accelerated by using graphics processing units (GPUs) and the vector instruction set (AVX) of latest CPU. PRISM algorithm to evaluate the electron repulsion integrals was vectorized to utilize AVX as much as possible. We found that this new program makes the Fock matrix formation in HF 2 to 3 times faster than ever before. The Coulomb and the exchange-correlation potentials in DFT were evalualted on GPU, result in about 4 times overall speedup. The programs developed were used to accrelerate FMO. We found that our new algorithm and GPU are very suitable for the calculation of the environmental electrostatic potential. The total computational time was reduced to about 1/3.

X-Ability
クロスアビリティ

Abstract

Benchmark : multicore CPUs

- Vectorized PRISM by AVX on Sandybridge CPU
  - HF (Hartree-Fock) : x 3 faster
- DFT by CUDA on NVIDIA GPU
  - J-matrix formation : x 245
  - Exchange correlation : x 4

NVIDIA GPU

- ERI J-matrix (Coulomb potential) by CUDA on NVIDIA GPU
  - FMO Environmental Electrostatic Potential : x 17

GAMESS-US
(free application)

Replace HF, DFT, FMO ESP
with CUDA/AVX scratch code

Acceleration so much

GAMESS-US
+ XA-AVX/CUDA-QM

X-Ability
クロスアビリティ

CPU[AVX]/GPU[CUDA] hybrid programming

- No GPU Program run without CPUs.
  - CPU+GPU architecture is general.
- SSE/AVX should be appropriate if the process is suitable for SIMD parallelization and not for GPU.
    - AVX computes 4 double precision arithmetics at once.
    - AVX is theoretically twice faster than SSE.
  - **AVX is available on latest CPU (SandyBridge).**
    - A normal Desktop PC is used in this research.

X-Ability
クロスアビリティ

AVX : Pros and Cons

Basically, it is like a vector processor.

Pros	Cons
- Twice faster than SSE - SIMD parallelization - utilizes main memory same as FPU computing	- Poor for algorithms that use conditional branching and/or random memory access - Automatic vectorization is efficient only for simple cases

X-Ability
クロスアビリティ

Plan for implementation

AVX	GPU
- Define overloaded operators of C++ to avoid using AVX intrinsic functions explicitly. - Use function pointers of C/C++ to eliminate branches in most inner loop.	- Use best tuned algorithm as far as GPU's resource permits. - Shared memory of Fermi core is used as L1 cache. - Use CPU threads (pthread) to harness GPU calculations for complicated thread control (synchronization, queueing…) . - Use OpenMP for CPU thread parallelization. - Single/double mixed-precision calculation.

X-Ability
クロスアビリティ

System Configuration

- Core i5 2500 @ 3.30GHz
- NVIDIA Geforce GTX580
  - 512 cuda cores
- DDR3 8GB
- CUDA 4.0
- Intel Composer XE 2011

X-Ability
クロスアビリティ

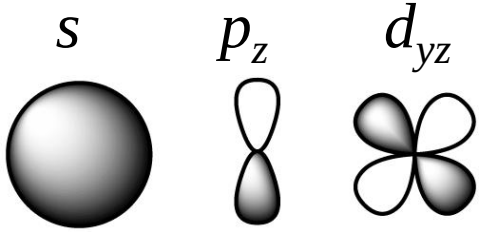
How to solve Schrödinger equation?

$$H(1\cdots N)\Psi(1\cdots N) = E\Psi(1\cdots N)$$

Expand wave function with

"contracted Gaussian basis set"

(Linear Combination of Atomic Orbital approximation)



$$\psi_i(r) = \sum_k C_k^{(i)} \chi_k(r)$$

$$s\ :\ \chi(r) = \sum_{k=1}^{K} d_k e^{-\alpha_k (r-A)^2} \qquad p_x\ :\ \chi(r) = \sum_{k=1}^{K} d_k (x - A_x) e^{-\alpha_k (r-A)^2}$$

$$F(C)C = SC\varepsilon$$

Matrix form of eigenvalue problem (Self consistent)

X-Ability
クロスアビリティ

Summary of Hartree-Fock Procedure

$$F(C)C = SC\varepsilon$$: SCF

$$(ab|cd) = \int \frac{\chi_a(r)\chi_b(r)\chi_c(r')\chi_d(r')}{|r-r'|} dr'dr$$: ERI

$$F_{\mu\nu} = H_{\mu\nu}^{core} + \sum_a^{2/N} \sum_{\lambda\sigma} C_{\lambda a} C_{\sigma a}^* \left[2(\mu\nu|\sigma\lambda) - (\mu\lambda|\sigma\nu) \right]$$

$$= H_{\mu\nu}^{core} + \sum_{\lambda\sigma} D_{\lambda\sigma} \left[(\mu\nu|\sigma\lambda) - \frac{1}{2}(\mu\lambda|\sigma\nu) \right]$$

$$= H_{\mu\nu}^{core} + G_{\mu\nu}$$

Density Matrix (D) : is updated at every SCF cycle to solve nonlinear equation after initial guess.

ERIs (ab|cd) : may be calculated only once and store them on memory in principle, but it needs large O(N^4) memory. They are recalculated at every cycle (direct SCF) to reduce expensive disk I/Os. **This step is bottleneck.**

Density Matrix×**ERI = J-matrix** : Coulomb potential matrix

Density Matrix×**ERI = K-matrix** : HF exchange matrix. GPU implementation is not easy because it needs a lot of registers.

X-Ability
クロスアビリティ

Hermite Gaussian Basis Set

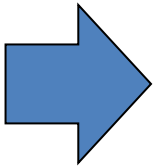
$$|p] = H_t(x - P_x)e^{-\varsigma(x-P_x)^2} \times (y\,\text{factor}) \times (z\,\text{factor})$$

t-th Hermite polynomial

A Product of two Gaussian functions |ab] is expanded exactly by Hermite Gaussians |p].

(Cartecian) Gaussian

$$J_{ab} = D_{cd}[ab|cd]$$



Hermite Gaussian

$$D_{cd}\,|cd] = \tilde{D}_q\,|q]$$

$$\tilde{J}_p = \tilde{D}_q[p|q]$$

$$J_{ab}\,|ab] = \tilde{J}_p\,|p]$$

ERI calculation in Hermite Gaussians space is faster than that in the normal Gaussians.

Because of the communication cost we can't get each ERI $[p|q]$ from GPU.

X-Ability
クロスアビリティ

Procedure of J-matrix (Coulomb)

Construct shell-pair data $|ab]$, $|p]$, D_q

expand them in terms of primitive HGs

Sort $|p]$ and $|q]$ for faster exit of loops.
$|p]$: $[p|p]^{1/2}$
$|q]$: $[q|q]^{1/2}$ D_q
descending order

p, q, D_q

$$J_p = \sum_q [p \mid q] D_q$$

1. Each thread between each pair of $[p|q]$.

2. No Integral symmetry to avoid expensive memory access.

Accumulated J_p

converted to J_{ab}

Less than 10% of total cost

X-Ability
クロスアビリティ

Density Functional Theory (DFT)

Wave function of electron (orbital)

$$[-\nabla^2/2 + v_{eff}(r)]\psi_i(r) = \varepsilon_i \psi_i(r)$$

Kinetic energy

Coulomb potential from nuclear **(Hcore)** and electron **(J)** + **exchange-correlation potential**

Hartree-Fock : $F_{\mu\nu} = H^{core}_{\mu\nu} + J_{\mu\nu} + K_{\mu\nu}$

DFT : $F_{\mu\nu} = H^{core}_{\mu\nu} + J_{\mu\nu} + v_{\mu\nu}$

Exchange correlation $v_{kl} = \int \chi_k(r)\chi_l(r) f(\rho(r), \nabla\rho(r)) dr$

Acceleration of J-matrix + Exchange Correlation is necessary.

X-Ability
クロスアビリティ

Procedure of Exchange-Correlation

$$E_{xc} = \sum_i w_i \varepsilon(\rho(r_i), \gamma(r_i)), \quad \gamma(r) = \nabla\rho(r) \cdot \nabla\rho(r)$$

$$\langle \chi_k | v_{xc} | \chi_l \rangle = \sum_i w_i \chi_k(r_i) \left[\frac{\partial \varepsilon(r_i)}{\partial \rho} \chi_l(r_i) + \frac{2\partial \varepsilon(r_i)}{\partial \gamma} \nabla\rho(r_i) \cdot \nabla\chi_l(r_i) \right]$$

Quadrature point r_i and weights w_i are generated.

r_i, w_i, D

Electron density $\rho(r_i)$,$\nabla\rho(r_i)$ on quadrature points parallelization

$$\rho(r_i) = \sum_{kl} D_{kl} \chi_k(r_i) \chi_l(r_i)$$

$\rho(r_i)$
$\nabla\rho(r_i)$

E_{xc}, potential on quadrature point f_i, $\mathbf{g}_i$

$$f_i = w_i \frac{\partial \varepsilon(r_i)}{\partial \rho}$$

$$\mathbf{g}_i = 2w_i \frac{\partial \varepsilon(r_i)}{\partial \gamma} \nabla\rho(r_i)$$

f_i, $\mathbf{g}_i$

v_{kl} matrix

V_{xc} matrix :

$$v_{kl} = \sum_i \left[f_i \chi_k(r_i) + \mathbf{g}_i \cdot \nabla\chi_k(r_i) \right] \chi_l(r_i)$$

X-Ability
クロスアビリティ

DFT acceleration [GPU]

Model, Basis set	Time [sec]	Total energy [a.u]
Paclitaxel($C_{47}H_{51}NO_{14}$), 3-21G		
GAMESS	929.407	-2912.2041896614
This work	305.709	-2912.2041830108
Paclitaxel($C_{47}H_{51}NO_{14}$), 6-31G		
GAMESS	1296.509	-2927.4589680121
This work	370.807	-2927.4589838167
Valinomycin($C_{54}H_{90}N_6O_{18}$), 3-21G		
GAMESS	2186.225	-3772.6098820622
This work	651.099	-3772.6098692643
Valinomycin($C_{54}H_{90}N_6O_{18}$), 6-31G		
GAMESS	3010.225	-3792.2248655337
this work	800.743	-3792.2248881864

X-Ability
クロスアビリティ

Timing details of DFT calculation [GPU]

Valinomycin,BLYP/3-21G unit:sec

	GAMESS	This work	speedup
HF Fock Matrix formation	248.927	76.241	x 3.27
DFT J matrix formation	619.465	2.529	x 244.94(※)
DFT exchange correlatin matrix formation	972.364	218.387	x 4.45
total	2186.225	651.099	x 3.36

 # original GAMESS J-matrix is very slow because ERIs are explicitly calculated in Cartesian Gaussian basis.

X-Ability
クロスアビリティ

PRISM implementation (1) [AVX]

- PRISM Algorithm
  - ✓One of the fastest algorithms to evaluate ERIs.
  - ✓adopted by Gaussian (famous QM software)
  - ✓Constantly fast for various contraction length / angular momentum.
  - ✓Convert Boys function $[0]^{(m)} \propto \int u^m \exp(-Tu)du$ to ERI using recurrence with a guide of “PATH”. Chooses the best path for different contraction length / angular momentum.

X-Ability
クロスアビリティ

PRISM implementation (2) [AVX]

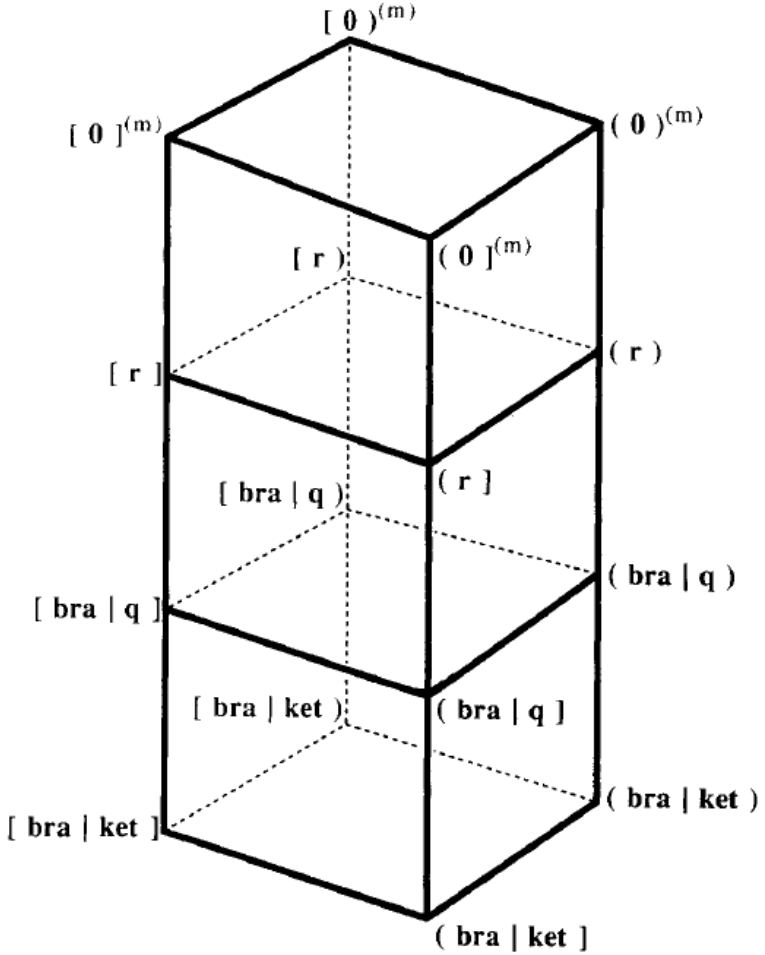


Figure 2. The general PRISM from $[\mathbf{0}]^{(m)}$ to (**bra**|**ket**).

Gill and Pople(1991)

All recurrence formulae are

(Integral A) = a * (Integral B) + b * (Integral C)

The same recurrence formula is applied to the intermediate integrals of the same shell type (angular momentum, bra/ket contraction length).
It would be suitable for SIMD parallelization.

X-Ability
クロスアビリティ

PRISM implementation (3) [AVX]

```
class double4                                   # operator
overload
{
public:
   double4& operator=(double d){
      m_d = _mm256_set1_pd(d);
      return *this;
   }
   double4 operator+(const double4& dd){
      return double4(_mm256_add_pd( m_d, dd.m_d) );
   }
   // … define other operators
   __m256d m_d;
}
```

X-Ability
クロスアビリティ

Difference between PRISM of ours and Gaussian‘s

	Gaussian	X-Ability
What is it	classifies shell quartets $(ab\|cd)$ by shell type and number of bra/ket contraction so that “Driver” is determined on each shell quartet on the fly.	generates source code of all PRISM paths and sets appropriate **function pointer** for each calculation target dynamically.
Pros	A recurrence formula is applied to long-enough batch of shell quartets so that it is suitable for traditional vector processors.	Only 4 shell quartets are processed at once, **resulting to avoid cache misses.**
Cons	Not suitable for scalar processors because **it might results in cash misses. A compiler could not optimize the program efficiently.**	The code has tens of thousands lines and 300-400MB volume.

X-Ability
クロスアビリティ

Calculation Result of Hartree-Fock [AVX]

	Time [sec]	Total energy [a.u]
Paclitaxel, 3-21G		
GAMESS	184.048	-2895.7814570171
This work	78.271	-2895.7814570169
Paclitaxel, 6-31G		
GAMESS	324.386	-2910.6633340322
This work	153.632	-2910.6633340179
Valinomycin, 3-21G		
GAMESS	476.829	-3750.9205018138
This work	155.481	-3750.9205017267
Valinomycin, 6-31G		
GAMESS	752.839	-3770.0595984968
this work	323.098	-3770.0595984236

10^{-7}hartree is enough accuracy. **We replaced GAMESS ERI with XA PRISM.**

X-Ability
クロスアビリティ

What's FMO(Fragment MO)

Ab initio for large Insulators (protein etc.)

Divide a molecule into N fragments

Generate initial density matrices for all fragments (monomers)

Prepare environmental electrostatic potentials (ESP) using previous density matrices

Bottleneck

Solve Fock equations for all monomers $F^I C^I = S^I C^I e^I$, for I=1 to N

NO

Are all monomer densities converged?

YES

Prepare environmental electrostatic potentials for all fragment pairs (dimers) using monomer densities

Solve Fock equations for all monomers $F^I C^I = S^I C^I e^I$, for I=2 to N, J=1 to I-1

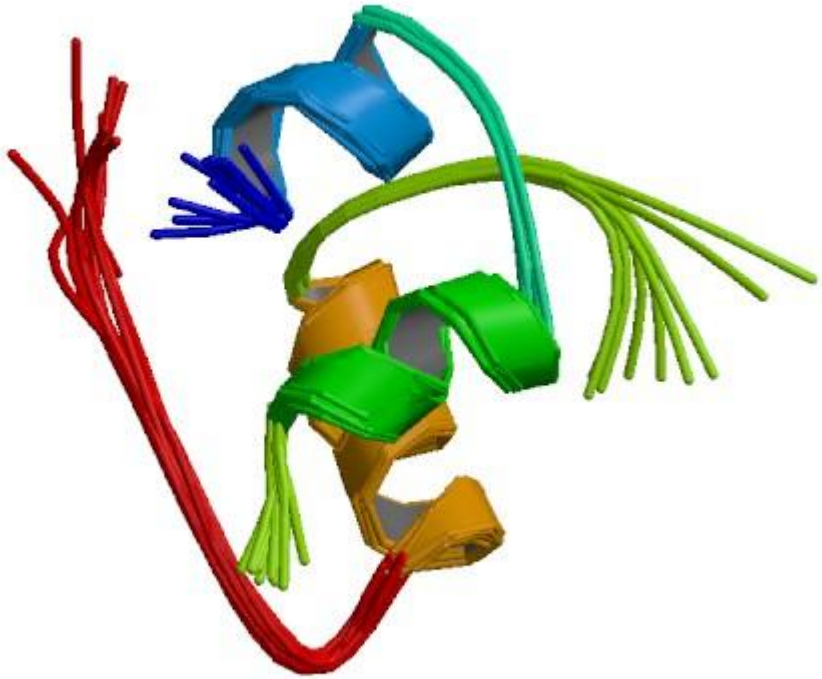
Calculate total energy and properties of molecule

X-Ability
クロスアビリティ

Test Model

Insulin (PDBID:2HIU)

44 amino acids



X-Ability
クロスアビリティ

Acceleration of Environmental Electrostatic Potential (ESP)

Fragment A Fragment B

$$J_{ab} = \sum_{cd} (ab|cd) D_{cd}$$

- Utilize ERI J-matrix

  cd: basis on neighboring fragments

- Decompose protein into many small fragments such as amino acid molecules.

  Conventional SCF (ERIs saved on disk) is effective for usual amino acids (# of basis < 180).

  ERI is not bottleneck in FMO.

  Environmental electrostatic potential dominates.

- Utilize J-matrix acceleration program by

X-Ability
クロスアビリティ

Total Energy of Insulin

	Time [sec]	Total Energy [a.u.]
Original GAMESS	14158.0	-21635.4488653044
Our work	3807.5	-21635.4488649218

- Error of total energy of 44 fragments is small enough.
- Energy of each amino acid is highly consistent.

X-Ability
クロスアビリティ

FMO ESP[GPU] and SCF[AVX] calculation time of Insulin

	GAMESS	This work	speedup
ESP part(GPU)	10696.2	640.0	x 16.71
SCF part(AVX)	2475.1	2136.8	x 1.16
Total	14158.0	3807.5	x 3.72

ESP part is much better than total acceleration ratio.

X-Ability
クロスアビリティ

Summary of performance results & discussion

	AVX	GPU
Summary of performance results	Hartree-Fock x 3 by Vectorized PRISM	DFT J-matrix formation: x 245 Exchange correlations: x 4 FMO ESP : x 17
Discussion	Unpredictable implementation cost	Difficulty of simple parallelization by register shortage. Few cost of the transfer speed between host and device using contracted Gaussian basis set.

GPGPU is better regarding price performance ratio.

X-Ability
クロスアビリティ

Further acceleration and improvement

- HF [AVX]

  Improvement of evaluation part of Boys function $[0]^{(m)}$

- DFT [GPU]

  Grid generation of DFT

- FMO [AVX+GPU]

  HF acceleration by AVX brings FMO acceleration, and investigate extra overhead time.

- Others, technically speaking

  OpenCL, cluster tuning by hardware, etc.

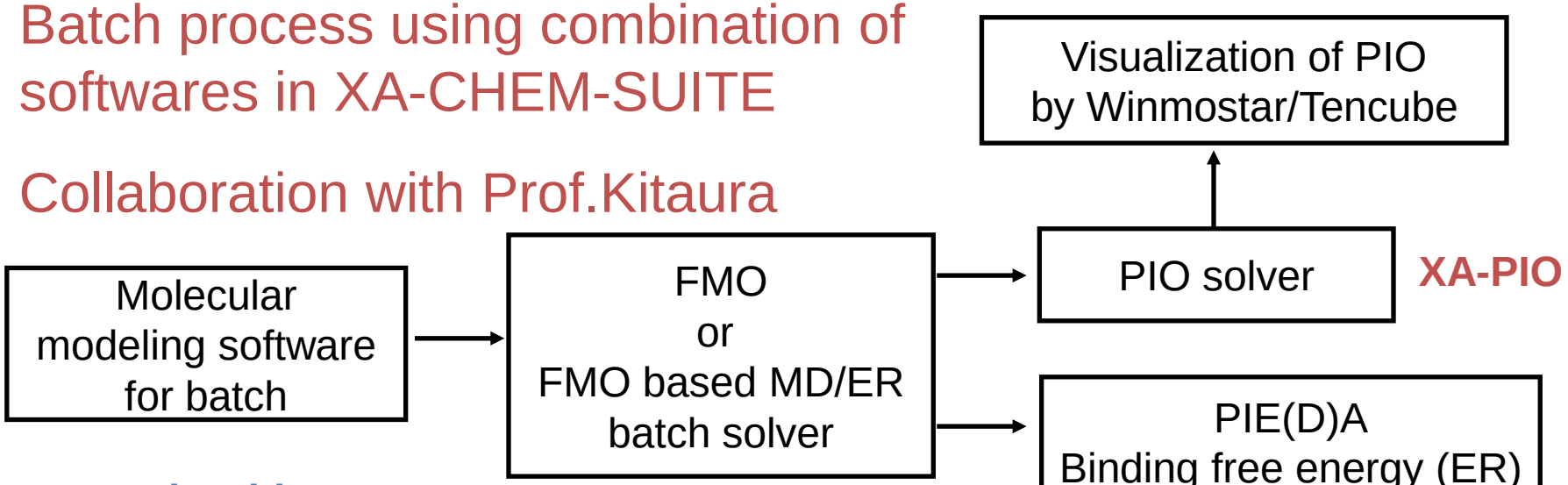
X-Ability
クロスアビリティ

Quntum Pharmaceutical System (funded by JST A-STEP)

JST is the biggest Japanese Government Funding Agency.

Batch process using combination of softwares in XA-CHEM-SUITE

Collaboration with Prof.Kitaura



Fastest algorithm on fastest architecture from electronic state calculation to affinity evaluation

Charge assignment

by GAMESS-US

+ XA-SSE/AVX/CUDA-QM

+ XA-ER

Molecular Dynamics

(Amber / Gromacs)

Start from 1st Oct 2011

X-Ability
クロスアビリティ

Thank you for your attention.

- Ryota Koga

  President

- Yuki Furukawa

  Scientific Computing

?

- Tetsu Ito

  Sensor Networking

- Naoki Nariai

  ChemBioInformatics