

量子格子模型プログラムパッケージ DCore (integrated DMFT software for CORrelated Electrons)の開発

品岡寛
埼玉大学



発表概要

1. 背景：強相関電子電子系
2. 第一原理計算・動的平均場近似の概要
3. 既存のソフトウェア紹介
4. DCoreの説明

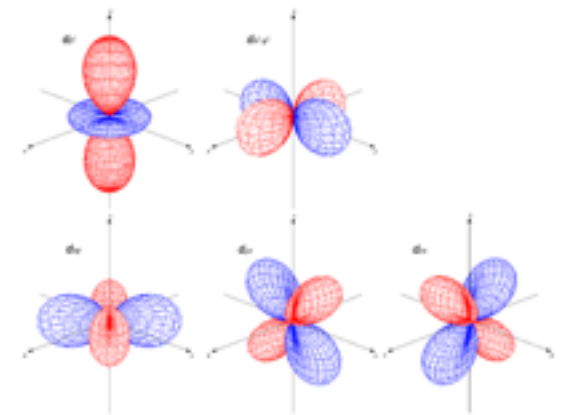
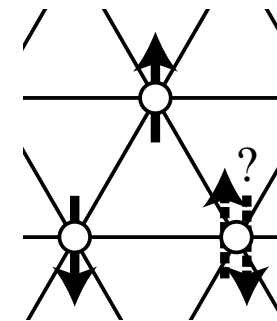
背景: 強相関電子系の広がり

超伝導

鉄系遷移金属酸化物、有機導体等

スピン、軌道、
電荷秩序

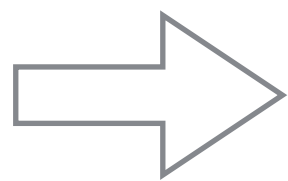
フラストレート磁性体、マルチフェロイック



トポロジカル相

5d遷移金属酸化物、界面・表面

定量的な物性予測を強く要請されている
スペクトル関数など



第一原理計算とは？

密度汎関数理論

P. Hohenberg and W. Kohn (1964)
W. Kohn and L. J. Sham (1965)

- 様々な近似: 局所密度近似(LDA)、一般化勾配近似 (GGA)
- 全電子計算 or not
- 基底関数の種類: 平面波 + α 、FP-LAPW、数値原子基底

		全電子計算	基底関数
Free	VASP	N	PAW
	Quantum ESPRESSO	N	PAW
Free	Wien2k	Y	FP-LAPW
	OpenMX	N	数値原子基底

Abinit, Elk, AkaiKKR, QMAS, *etc.*

Quantum ESPRESSO

<https://www.quantum-espresso.org>



NEWS

10.05.18

THE WALTER KOHN PRIZE

Nominations are now being accepted for the second Walter Kohn Prize for quantum-mechanical materia...

30.01.18

QE DEVELOPERS' MEETING 2018

Agenda

February 1st 2018

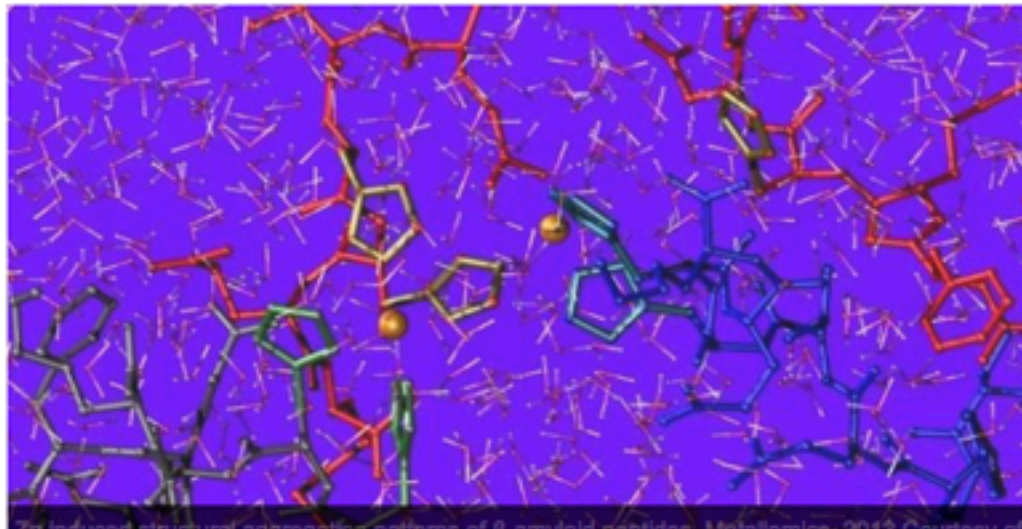
9:30 - 9:55 -- Paolo Giannozzi: *Introduction*. slides

10:00 -...

11.12.17

QUANTUM ESPRESSO V.6.2.1

Version 6.2.1 of QUANTUM ESPRESSO is available for download from GitLab and on qe-forge..



QUANTUM ESPRESSO

is an integrated suite of Open-Source computer codes for electronic-structure calculations and materials modelling at the nanoscale. It is based on density-functional theory, plane waves, and pseudopotentials.

[READ MORE >](#)

- 擬ポテンシャル法
- Wannier90との連携
(symmetry-adapted Wannier functions)

OpenMX

<http://www.openmx-square.org>

Welcome to OpenMX
open source package for Material explorer

Contents

- **What's new**

International Summer Workshop for July 2nd-12th, 2018

Patch (Ver. 3.8.5) to OpenMX Ver. 3.8 (June 12, 2018)

- **What is OpenMX?**
- **Download**
- **Manual of Ver. 3.8**
- **Manual of Ver. 3.7**
- **Technical Notes**
- **Video Lectures**
- **Publications**
- **OpenMX Forum**
- **OpenMX Viewer**
- **Workshop**
- **Database of VPS and PAO**
Ver. 2013
- **ADPACK**
- **Miscellaneous informations**
- **Contributors**
- **Acknowledgment**
- **Opening positions**
- **Links**

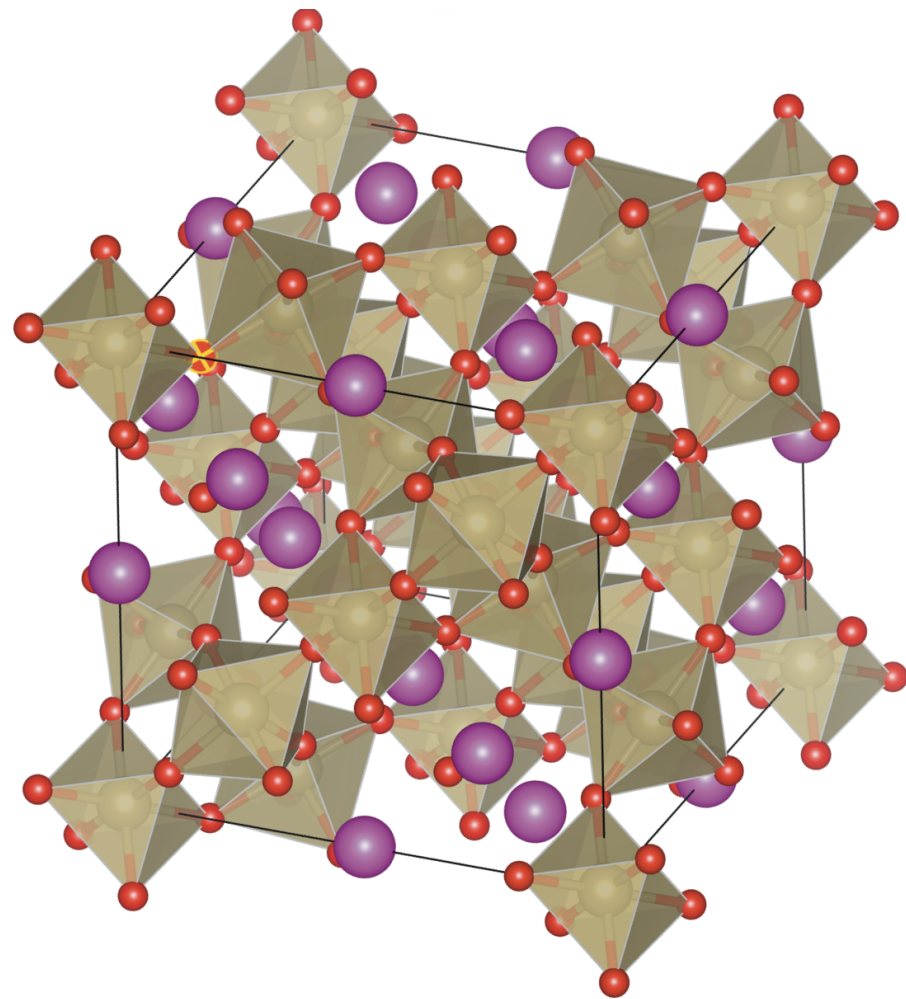


- 擬ポテンシャル＋数値原子軌道基底
- 大規模系の計算
- Wannier90との連携
- 対称操作は無し

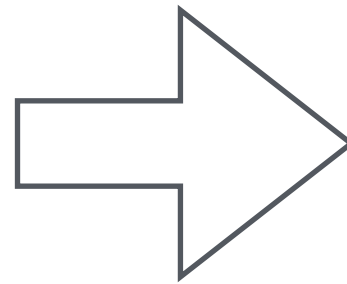
密度汎関数理論＋動的平均場近似

LDA/DFT+DMFT

G. Kotliar *et al.*, RMP **78**, 865 (2006)



相関効果が強い原子軌道への射影



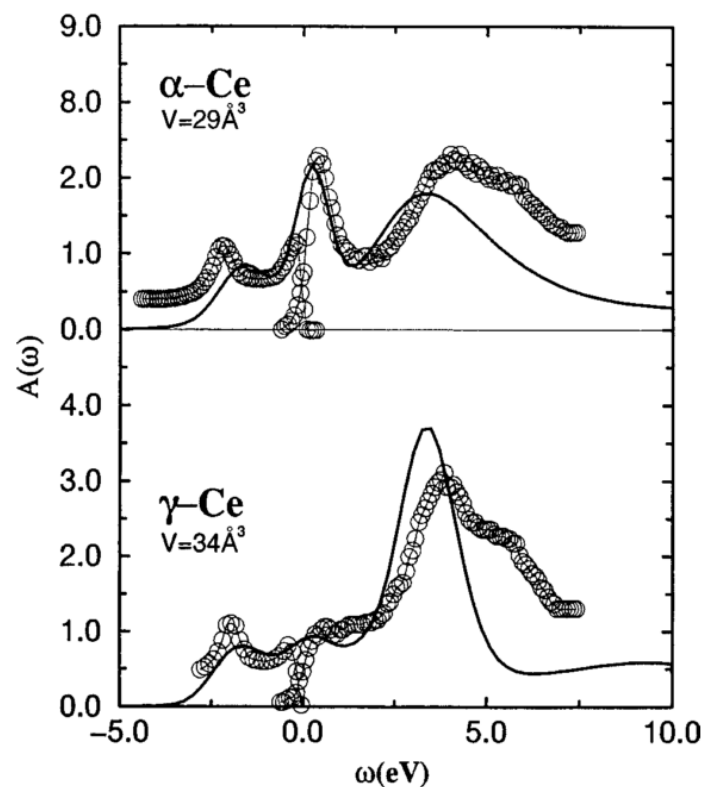
$$\Sigma(k, \omega) \rightarrow \Sigma(\omega)$$

- 格子問題を量子不純物問題へマップ
- 動的物理量の計算: スペクトル関数、動的感受率
- 摂動理論による拡張が可能 (DMFT+GWなど)

計算できる量

📍 スペクトル関数

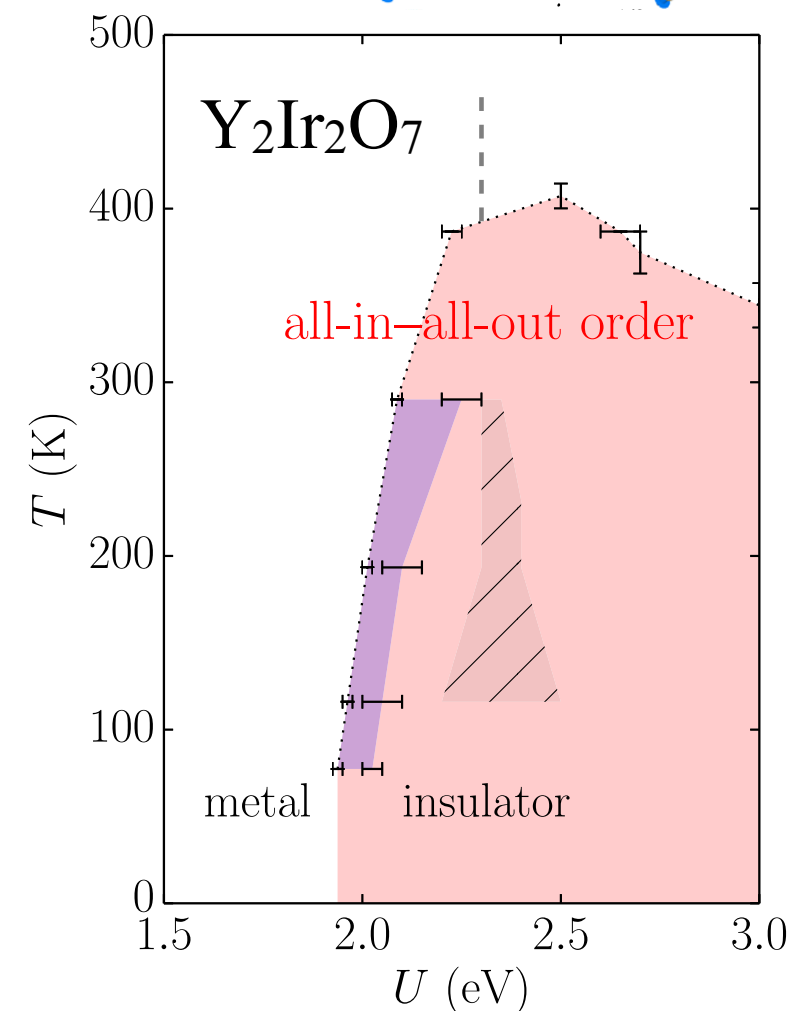
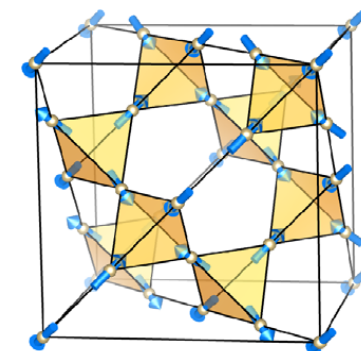
$$A(k, \omega) = -\frac{1}{\pi} \text{Im} G(k, \omega + i0^+)$$



McMahan *et al.*, 2003

→ 光学実験との直接比較

📍 複雑な磁気構造



HS, S. Hoshino, M. Troyer and P. Werner (2015)

📍 静的・動的感受率

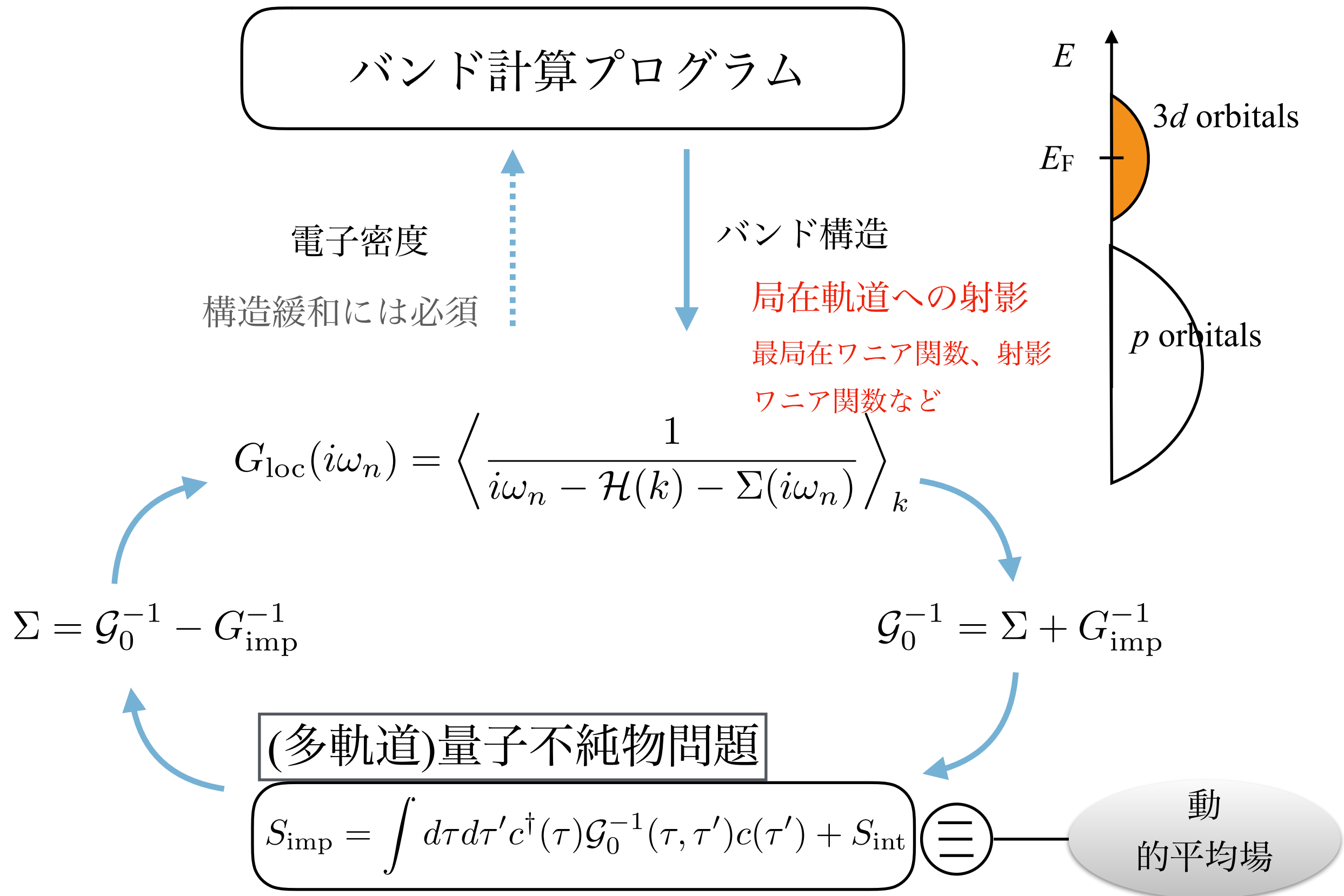
M. Jarrell (1992), H. Park *et al.* (2011), L. Boehnke *et al.* (2011), J. Kuneš *et al.* (2017)

📍 構造緩和

K. Haule and G. L. Pascut (2016)

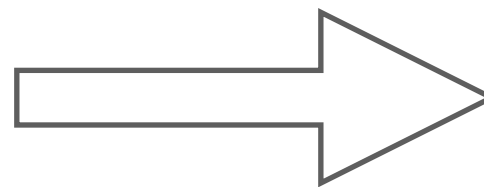
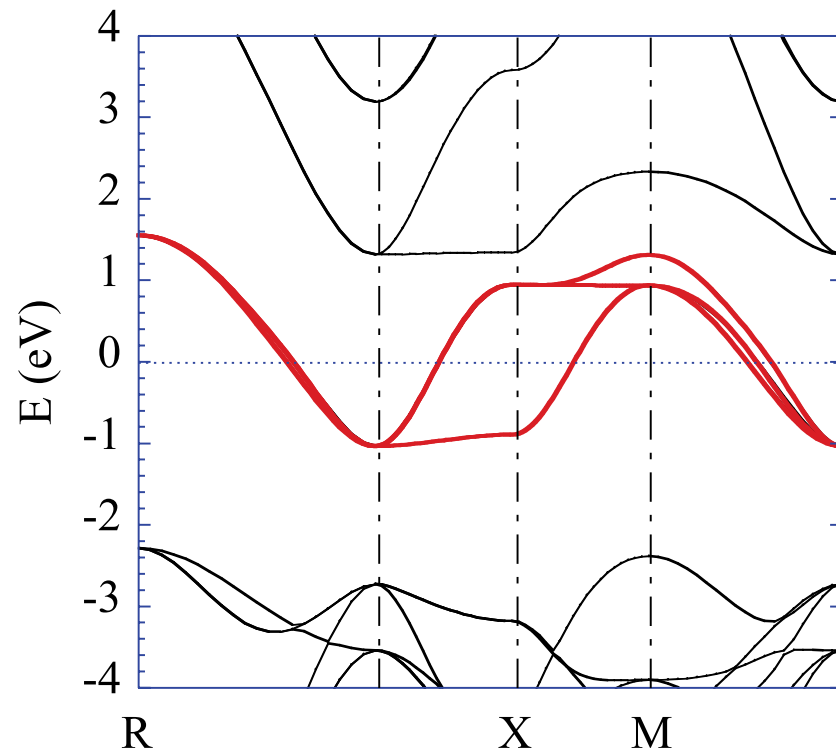
DFT+DMFTの計算スキーム

G. Kotliar *et al.*, RMP **78**, 865 (2006)

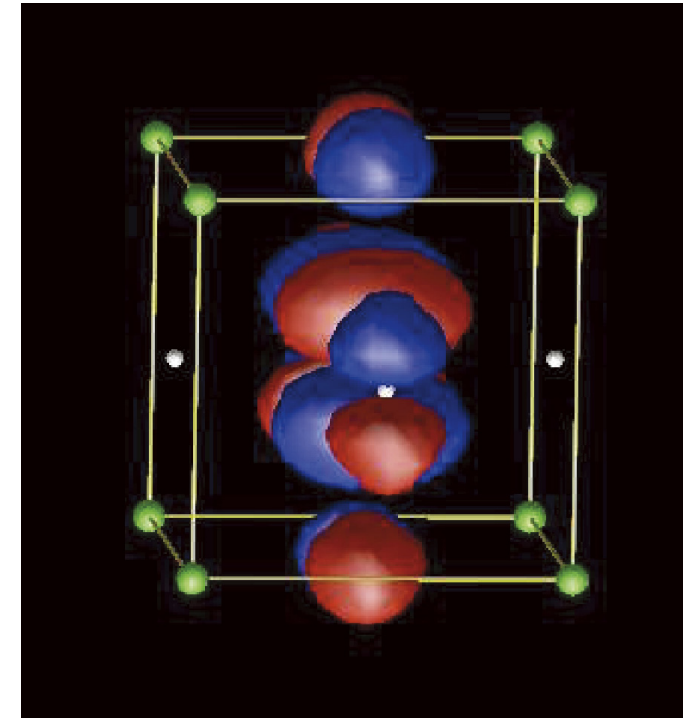


最局在ワニア関数

N. Marzari and D. Vanderbilt (1997), I. Souza *et al.* (2001)



波数、軌道に依存
したゲージ変換



SrVO₃: M. Imada and T. Miyake (2010)

- DFTコードが使う基底に依存しない
- ワニア関数の最適化が必要

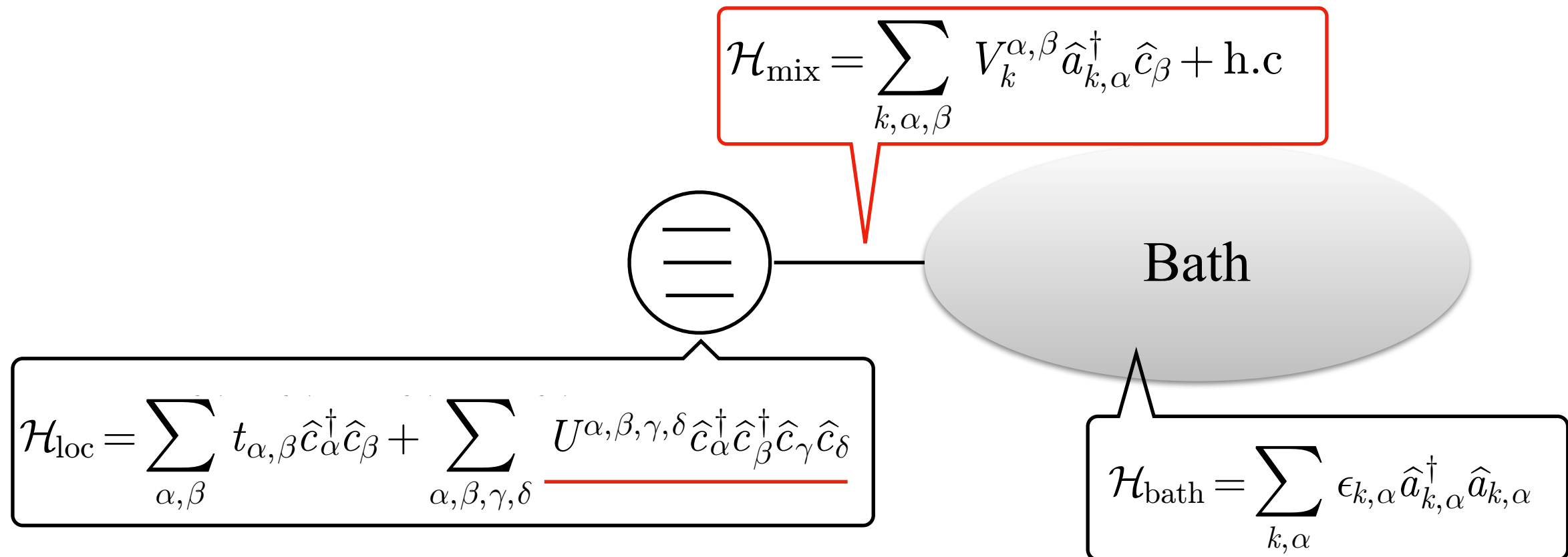
Projectors

<http://hauleweb.rutgers.edu/tutorials/Overview.html>

M. Aichorn *et al.*, PRB **80**, 085101 (2009)

https://triqs.github.io/dft_tools/master/_downloads/TutorialDmftproj.pdf

Quantum (Anderson) impurity problem



NRG, Exact diagonalization, Hubbard-I, **CT-QMC** *etc.*

Continuous-time Monte Carlo method

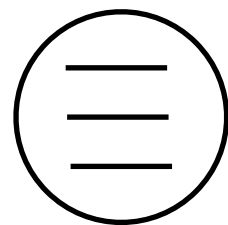
Review: E. Gull *et al.*, RMP **83**, 349 (2011)

$$\mathcal{H}_{\text{mix}} = \sum_{k,\alpha,\beta} V_k^{\alpha,\beta} \hat{a}_{k,\alpha}^\dagger \hat{c}_\beta + \text{h.c}$$

Hybridization
expansion (CT-HYB)

P. Werner *et al.*, PRL **97** 076405 (2006)

P. Werner *et al.*, PRB **74** 155107 (2006)



Bath

$$\mathcal{H}_{\text{loc}} = \sum_{\alpha,\beta} t_{\alpha,\beta} \hat{c}_\alpha^\dagger \hat{c}_\beta + \sum_{\alpha,\beta,\gamma,\delta} U^{\alpha,\beta,\gamma,\delta} \hat{c}_\alpha^\dagger \hat{c}_\beta^\dagger \hat{c}_\gamma \hat{c}_\delta$$

Interaction expansion (CT-INT)

A. N. Rubtsov *et al.*, PRB
72, 035122 (2005)

$$\mathcal{H}_{\text{bath}} = \sum_{k,\alpha} \epsilon_{k,\alpha} \hat{a}_{k,\alpha}^\dagger \hat{a}_{k,\alpha}$$

CT-AUX, CT-J, CT-X, *etc.*

- A serious expansion of partition function
- Sign problem in solving multi-orbital model

CT-HYB

P. Werner *et al.*, PRL **97** 076405 (2006)

P. Werner *et al.*, PRB **74** 155107 (2006)

$$\begin{aligned} Z = \text{Tr}[e^{-\beta\mathcal{H}}] &= \text{Tr}\left[e^{-\beta\mathcal{H}_1} T e^{-\int_0^\beta d\tau \mathcal{H}_2(\tau)}\right] \\ &= \sum_{n=0}^{\infty} \int_0^\beta d\tau_1 \cdots \int_{\tau_{n-1}}^\beta d\tau_n (-1)^n \text{Tr}\left[e^{-(\beta-\tau_n)\mathcal{H}_1} \mathcal{H}_2 e^{-(\tau_n-\tau_{n-1})\mathcal{H}_1} \cdots \mathcal{H}_2 e^{-\tau_1\mathcal{H}_1}\right] \end{aligned}$$

$$\mathcal{H}_1 = \mathcal{H}_{\text{loc}} + \mathcal{H}_{\text{bath}} \quad \text{and} \quad \mathcal{H}_2 = \mathcal{H}_{\text{mix}}$$

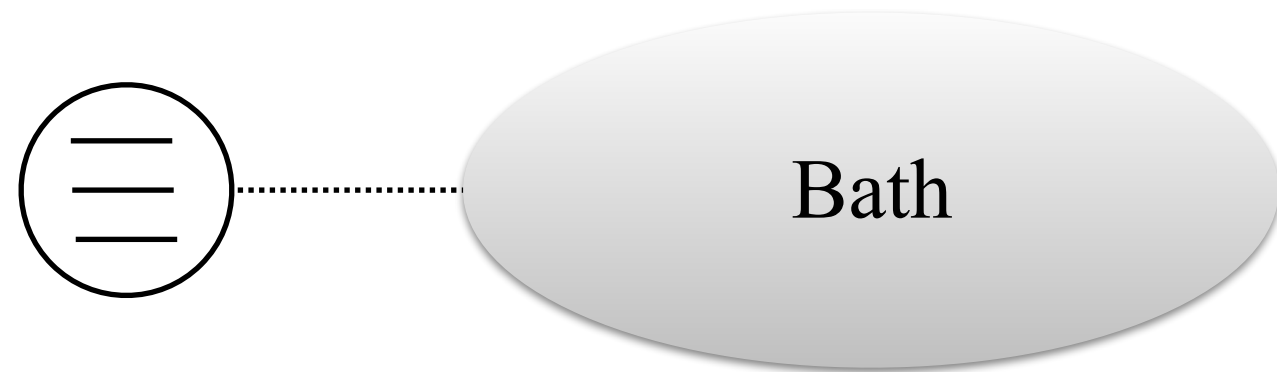


- Multi-orbital models with general instantaneous interactions
- Low expansion order in correlated regime
- Measurement of single-/two-particle Green's functions

Hubbard I approximation

J. Hubbard, Proc. Roy. Soc. London A**276**, 238 (1963)

- Insulating solution at integer fillings
- Multi-orbital systems
- Real-frequency data



発表概要

1. 背景：強相関電子電子系
2. 第一原理計算・動的平均場近似の概要
3. 既存のソフトウェア紹介
4. DCoreの説明

既存のソフトウェア

👤 TRIQS: A Toolbox for Research on Interacting Quantum Systems

<https://triqs.ipht.cnrs.fr/>

- ▶ Green's function libraries
- ▶ Quantum impurity solvers
- ▶ Interface with *ab-initio* codes

👤 ALPSCore project

<https://alpscore.org>

- ▶ Continuous-time quantum impurity solvers
- ▶ Hybridization-expansion algorithm
HS, E. Gull, P. Werner (2017)
- ▶ Interaction-expansion algorithm
HS, Y. Nomura, E. Gull, in preparation

Pythonでライブラリを組み合わせて、物質、模型ごとにプログラムを作る必要あり

👤 DFT + Embedded DMFT Functional

<http://hauleweb.rutgers.edu/tutorials/>

Rutger's university

- ▶ Quantum impurity solvers
- ▶ Charge self consistency with Wien2k
- ▶ *License issues*

👤 iQIST (Interacting Quantum Impurity Solver Toolkit)

<https://github.com/iqist/iqist>

- ▶ Quantum impurity solvers

👤 w2dynamics

M. Wallerberger *et al.*, arXiv:1801.10209v1

- ▶ Quantum impurity solver

誰もが簡単に使えるようなソフトウェアが必要
(理論家にも実験家にも) → **DCore**

TRIQS: A Toolbox for Research on Interacting Quantum Systems

<https://triqs.ipht.cnrs.fr/>

<https://triqs.github.io/triqs/master/tour/ctqmc.html>

▶ C++, Python ライブラリ

主要な機能をPythonから利用可能

▶ 開発者は欧米中心

▶ 最新のC++言語環境が必要

TRIQS 1.4 (C++14), TRIQS 2.0 (C++17)

物性研システム
B OK

DCoreはまだ
未対応

```
from pytriqs.gf import *
from pytriqs.operators import *
from pytriqs.applications.impurity_solvers.cthyb import Solver

# Parameters
D, t, U = 1.0, 0.2, 4.0
e_f, beta = -U/2.0, 50.0

# Construct the impurity solver with the inverse temperature
# and the structure of the Green's functions
S = Solver(beta = beta, gf_struct = {'up':[0], 'down':[0]})

# Initialize the non-interacting Green's function S.G0_iw
for spin, g0 in S.G0_iw:
    g0 << inverse( iOmega_n - e_f - t**2 * Flat(D) )

# Set the solver parameters.
params = {}
params['n_cycles'] = 1000000 # Number of QMC cycles
params['length_cycle'] = 200 # Length of one cycle
params['n_warmup_cycles'] = 10000 # Warmup cycles
params['measure_g_l'] = True # Measure G in Legendre

# Run the solver. The result will be stored in S.G_tau.
S.solve(h_int = U * n('up',0) * n('down',0), **params)

# Save the results in an hdf5 file (only on the master node).
from pytriqs.archive import HDFArchive
import pytriqs.utility.mpi as mpi

if mpi.is_master_node():
    Results = HDFArchive("solution.h5", 'w')
    Results["G_tau"] = S.G_tau
    Results["G_l"] = S.G_l
```

P. Seth *et al.* (2016)

CT-HYB

オープンソース実装

	Correct measurement of off-diagonal components of Green's function	Two-particle Green's functions (<i>not supported DCore vet</i>)
ALPSCore/CT-HYB <small>https://github.com/ALPSCore/CT-HYB</small>	Y	Y
TRIQS/cthyb <small>https://github.com/ALPSCore/CT-HYB</small>	N	N
iQIST	N	N
w2dynamics	Y	Y

ALPS/CT-HYB

HS, E. Gull, P. Werner, Computer Physics Communications (2017)

<https://github.com/ALPSCore/CT-HYB>

- ▶ C++プログラム (かなり古い環境でも動く
C++03)
- ▶ テキストファイル+HDF5ファイルベース
のインターフェース
- ▶ スピン軌道相互作用
- ▶ グリーン関数の非対角成分の計測

```
seed=20
timelimit=600

model.sites=3
model.spins=2
model.coulomb_tensor_input_file="Uijkl.txt"
model.hopping_matrix_input_file="hopping.txt"
model.beta=20.0
model.delta_input_file="delta.txt"
model.n_tau_hyb=1000
```

TRIQSからシームレス呼び出す
ことも可能 (Python, オプション)

DCore : 開発背景 既存のソフトウェア

🧑 TRIQS: A Toolbox for Research on Interacting Quantum Systems

<https://triqs.ipht.cnrs.fr/1.x/index.html>

- ▶ Green's function libraries
- ▶ Quantum impurity solvers
- ▶ Interface to Wannier 90, Wien2k

🧑 ALPSCore project

<https://alpscore.org>

- ▶ Monte Carlo libraries
- ▶ Quantum impurity solvers

Only single-orbital code in *original* ALPS

Pythonでライブラリを組み合わせて、物質、模型ごとにプログラムを作る必要あり

🧑 DFT + Embedded DMFT Functional

<http://hauleweb.rutgers.edu/tutorials/>

Rutger's university

- ▶ Quantum impurity solvers
- ▶ Charge self consistency with Wien2k
- ▶ *License issues*

🧑 iQIST (Interacting Quantum Impurity Solver Toolkit)

<https://github.com/iqist/iqist>

- ▶ Quantum impurity solvers

🧑 w2dynamics

M. Wallerberger *et al.*, arXiv:1801.10209v1

- ▶ Quantum impurity solver



→ 既存のライブラリを組み合わせたソフトウェア開発
誰もが簡単に使えるようにする！

発表概要

1. 背景：強相関電子電子系
2. 第一原理計算・動的平均場近似の概要
3. 既存のソフトウェア紹介
4. **DCore**の説明

DCore : 開発背景 ソフトウェア開発・高度化プロジェクト

学術的・社会的ニーズが高い、物質科学計算用ソフトウェアの開発・高度化
簡便なシミュレーション環境を物性研スパコン上に構築
全国共同利用スパコンユーザー層の底上げをする。

ユーザビリティ向上



プリインストール

物質科学計算用ソフトウェア
のユーザビリティ向上
(機能整備、新規機能追加 etc.)
1~2ソフトウェア/年
オープンソースソフトウェアで公開

2015-2016年で高度化されたソフトウェア一覧

K ω (2016)



数値ライブラリ
新規開発



H Φ (2015) mVMC (2016)



強相関係向け量子格子模型ソルバー
既存コード整備・機能追加

OpenMX (2015)

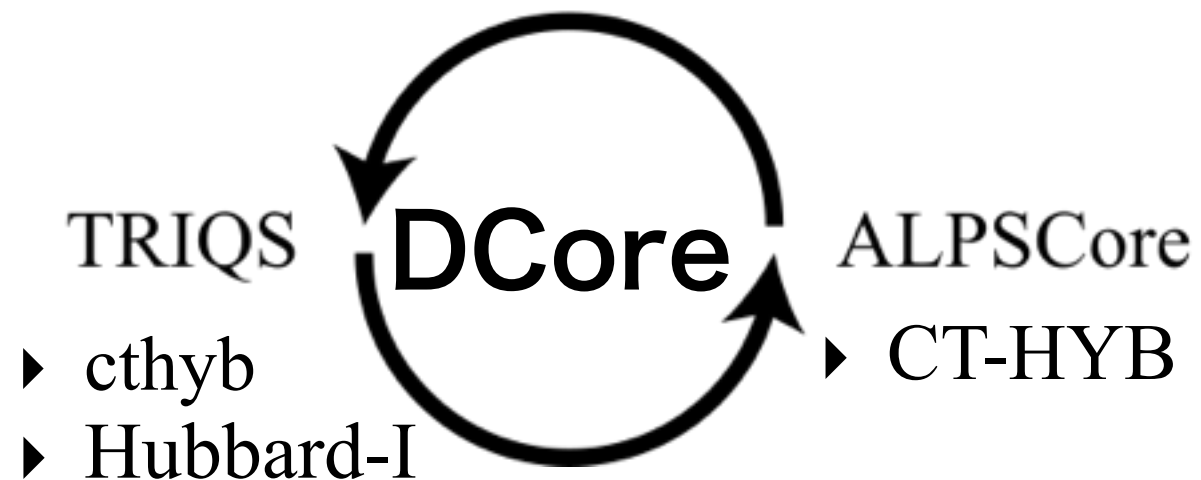


第一原理計算ソフトウェア
機能新規追加

DCore : 開発背景

2017年度ソフトウェア開発・高度化プロジェクトで開発！

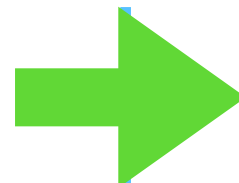
Connecting power of elaborate TRIQS and ALPS libraries



テキストベースの
インターフェース
▶ 自己無撞着計算
▶ 物理量計算
▶ モデル定義

Output: 自己エネルギー、グリーン関数、スペクトル関数

第一原理計算を用いたモデル生成
バンド構造 (Wannier90形式)



相互作用 (Wannier90形式)



DCore : 基本情報

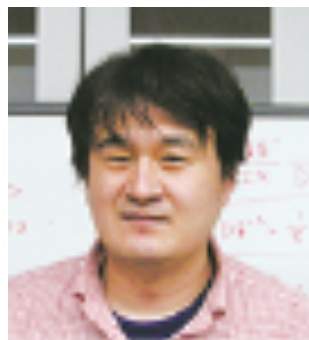
▶ DCore 開発者 (自分を除く)



吉見一慶



河村光晶



加藤岳生

ソフトウェア高度化チーム (東京大学 物性研)



品岡寛
(埼玉大学 理学部)

提案者



大槻純也
(東北大学 理学部)

連携協力者

▶ 言語

- Python2.7
- C++14 (for building TRIQS)

▶ ライセンス



オープンソースソフトウェアなので、誰でも利用可能！

▶ 動作環境

- Linux , OS Xでの動作確認済

DCore : 基本情報 機能一覧

▶ モデル

- 格子形状
 - 標準的な格子 : Bethe, chain, square, cubic lattice
 - Wannier90形式 : DFT calculations with/without spin-orbit coupling
 - 相互作用
 - Slater-Kanamori interaction,
 - 制限RPA法を用いて導出した有効相互作用 (Wannier90形式)
- RESPACK: <http://ma.cms-initiative.jp/ja/listapps/respack/respack>

▶ 自己無撞着計算

- 非磁性計算、コリニア磁気構造

▶ 物理量

- 自己エネルギー $\Sigma(i\omega_n)$
- スペクトル関数 $A(\omega), A(k, \omega)$

物理量は今後追加予定
開発者としての参加を歓迎！

▶ MPI並列化

- 自己無撞着計算 (implemented in TRIQS/DFTTools)
- 量子不純物ソルバー (ALPSCore/CT-HYB, TRIQS/cthyb)

DCoreで扱える相互作用
入力・出力ファイルに関する説明 etc.

DCore: 基本情報(6) 計算の流れ

入力ファイルの作成
(テキストインターフェイス)

第一原理計算から導出したパラメータを使用する場合、
Wannier90形式のファイルも用意
(hopping, Interaction)



dcore実行用 : dcore_pre
入力ファイル(hdf5形式)の作成

選択されたsolverに応じた
hdf5形式の入力ファイルを作成



DMFT計算 : dcore
(出力結果 : hdf5形式)

選択されたsolverを用いたDMFT計算
(TRIQSで用いられているフォーマットで結果出力)
Green関数、自己エネルギーが出力される。



計算後処理 : dcore_post
(出力形式 : テキスト)

出力された結果の加工(主に解析接続)やグラフ表示。
テキスト形式での出力もする。

DCore: 基本情報(7) 入力ファイル

全5つのblockから構成される

[model] : 模型に関する設定
格子
軌道の数・種類
電子数
相互作用の種類
相互作用の大きさ

[system] : 系に関する設定
虚時間の分点の数
松原振動数の分点の数
逆温度
化学ポテンシャル
etc...

[impurity_solver] : ソルバーの設定
TRIQS/hubbard-I, TRIQS/cthyb,
ALPS/cthyb

[control] : ソルバーの計算条件の設定
次のステップに進む際のmixingパラメータ
DMFT-loopの最大ループ数
再計算フラグ

[tool] : ポスト処理時の設定
最大・最小実振動数
k点の始点・終点と分点数
振動数の虚部のシフト量
etc.

ref.) <https://issp-center-dev.github.io/DCore/reference/input.html>

計算例: single-band 2D Hubbard model (1)

<https://issp-center-dev.github.io/DCore/tutorial/square/square.html>

dmft_square.ini

```
[model]
seedname = square
lattice = square
norb = 1
nelec = 1.0
t = -1.0
kanamori = [(2.0, 0.0, 0.0)]

[system]
beta = 40.0
nk = 8
n_iw = 1000
prec_mu = 0.001

[impurity_solver]
name = TRIQS/hubbard-I

[control]
max_step = 7

[tool]
broadening = 0.4
nnode = 4
knode = [(G,0.0,0.0,0.0),(X,0.5,0.0,0.0),(M,0.5,0.5,0.0),(G,0.0,0.0,0.0)]
nk_line = 100
omega_max = 6.0
omega_min = -5.0
Nomega = 400
```

格子

軌道数

電子数

ホッピング

相互作用

逆温度

不純物ソルバー

自己無撞着計算の回数

$A(k,\omega)$ を計算するパス

原子内相互作用

$$\hat{H}_{\text{int}} = \frac{1}{2} \sum_{i, \alpha\beta\gamma\delta, \sigma\sigma'} U_{\alpha\beta\gamma\delta}^i c_{i\alpha\sigma}^\dagger c_{i\beta\sigma'}^\dagger c_{i\delta\sigma'} c_{i\gamma\sigma}$$

The diagram illustrates the components of the interaction term $U_{\alpha\beta\gamma\delta}^i$ in the Hamiltonian. The summation index $i, \alpha\beta\gamma\delta, \sigma\sigma'$ is shown with color-coded labels: i is green (原子), $\alpha, \beta, \gamma, \delta$ are red (軌道), and σ, σ' are blue (スピン). The term $U_{\alpha\beta\gamma\delta}^i$ is enclosed in a dotted circle, with a line pointing to the text "原子に依存した4階テンソル" (4th-order tensor dependent on the atom).

DCoreでは、以下の形式をサポート

- 少数のパラメータに依存した標準的な形式の選択 (kanamori相互作用など)
- RESPACK形式での入力

Kanamori相互作用

$$\begin{aligned}U_{\alpha\alpha\alpha\alpha} &= U, \\U_{\alpha\beta\alpha\beta} &= U' \quad (\alpha \neq \beta), \\U_{\alpha\beta\beta\alpha} &= J \quad (\alpha \neq \beta), \\U_{\alpha\alpha\beta\beta} &= J \quad (\alpha \neq \beta),\end{aligned}$$

where U, U', J at each correlated shell are specified by the parameter `kanamori` as

```
interaction = kanamori  
kanamori = [(U_1, U'_1, J_1), (U_2, U'_2, J_2), ... ]
```

Slater型など、他のパラメータ化もサポート

計算例: single-band 2D Hubbard model (2)

<https://issp-center-dev.github.io/DCore/tutorial/square/square.html>

1. 入力ファイルからHDF5ファイルを作成

```
> dcore_pre dmft_square.ini
```

dmft_square.ini → square.h5

2. 自己無撞着計算

```
> dcore dmft_square.ini
```

▶ dmft_square.ini
▶ square.h5 → square.out.h5

3. 自己無撞着計算の収束を確認

```
> dcore_check dmft_square.ini
```

4. 物理量解析

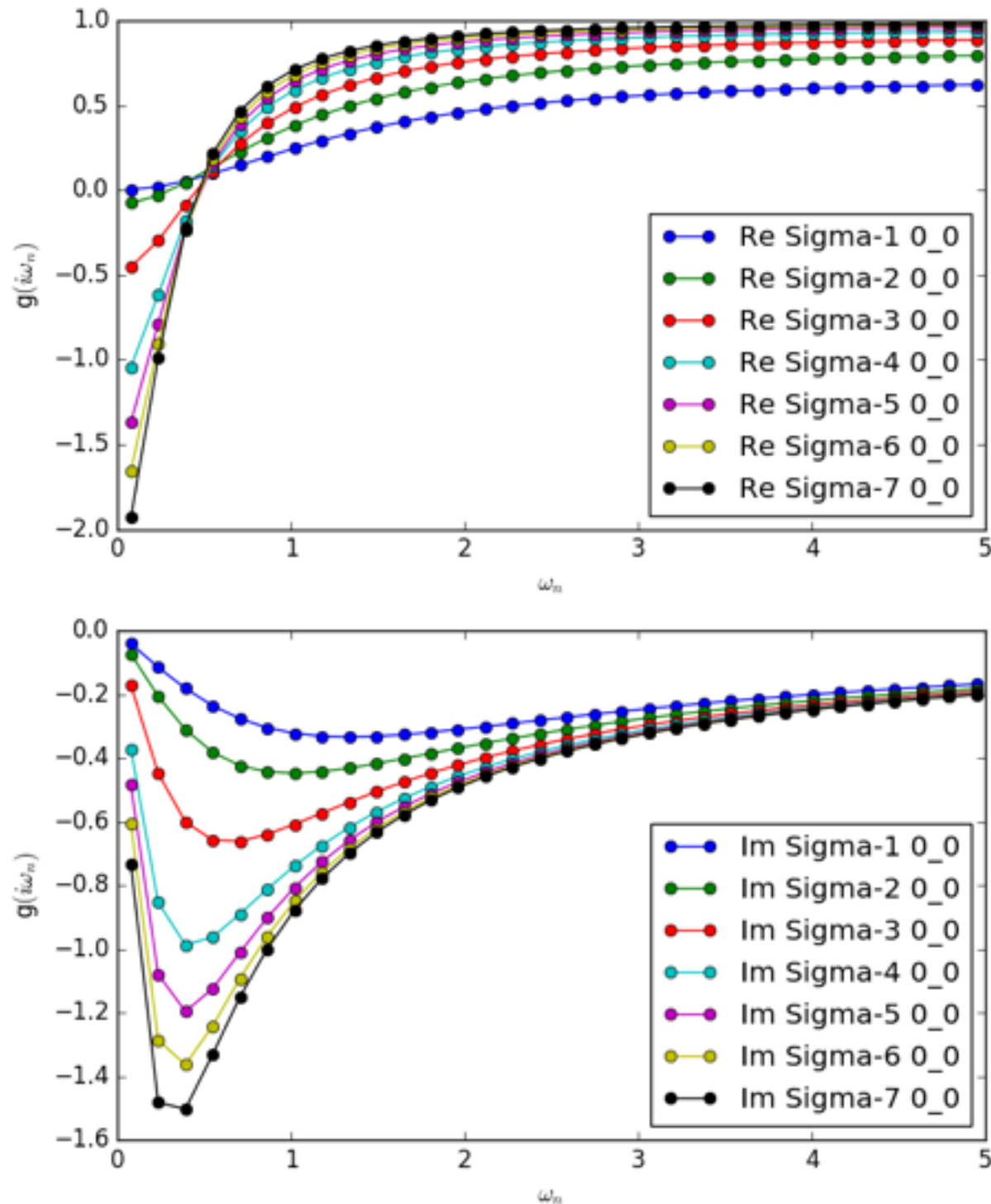
```
> dcore_post dmft_square.ini
```

計算例: single-band 2D Hubbard model (3)

<https://issp-center-dev.github.io/DCore/tutorial/square/square.html>

3. 自己無撞着計算の収束を確認

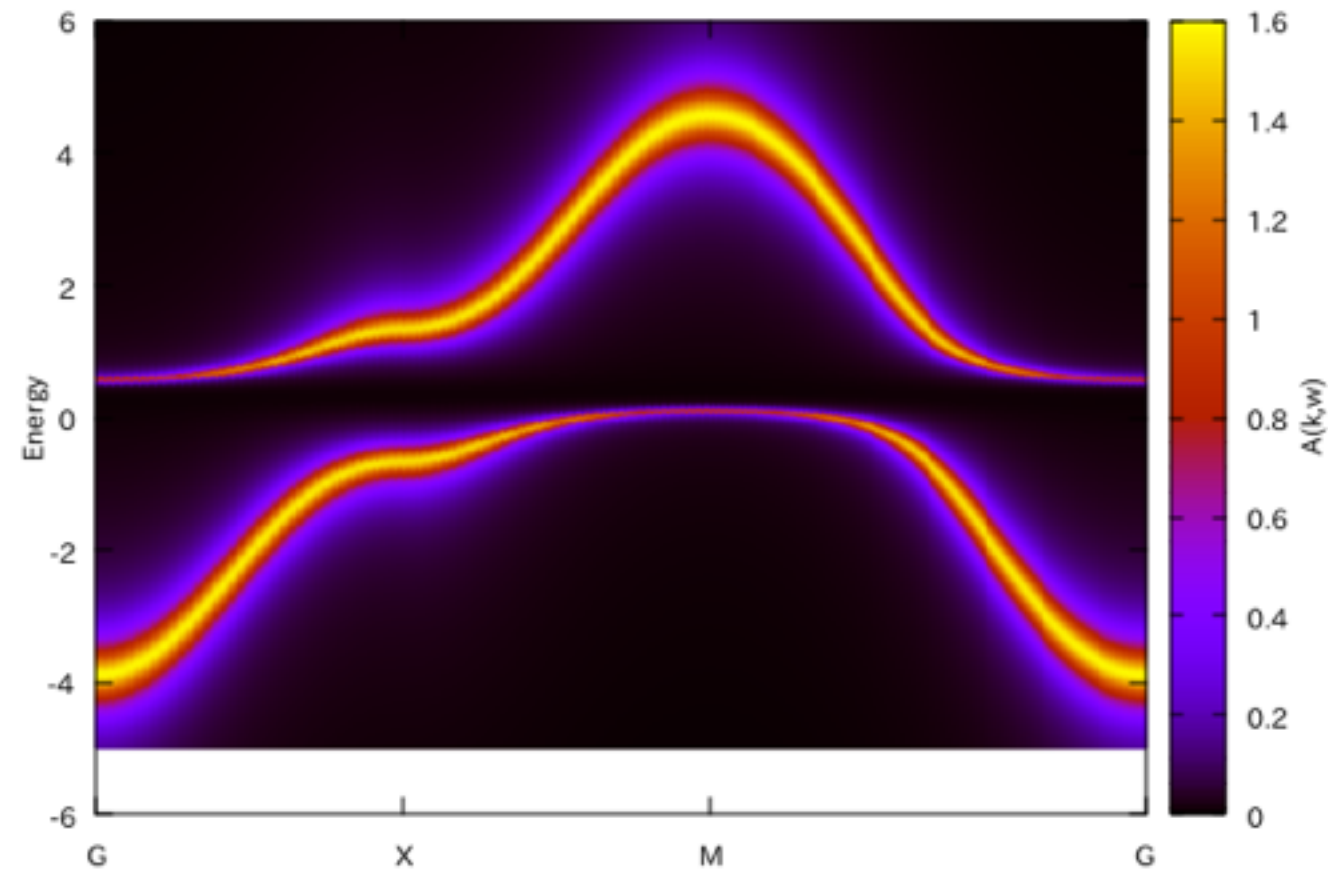
> `dcore_check dmft_square.ini`



4. 物理量解析

> `dcore_post dmft_square.ini`

> `gnuplot square_akw.gp`



計算例: SrVO₃ with Quantum ESPRESSO (1)

https://issp-center-dev.github.io/DCore/tutorial/srvo3_qe/srvo3.html

0. Quantum ESPRESSOによる最局在ワニア関数作成

1. 入力ファイルからHDF5ファイルを作成

> `dcore_pre` srvo3.ini

srvo3.ini

```
[model]
lattice = wannier90
seedname = srvo3
nelec = 1.0
ncor = 1
norb = 3
kanamori = [(3.419, 2.315, 0.530)]
bvec=[(1.627091,0.0,0.0),(0.0,1.627091,0.0),(0.0,0.0,1.627091)]
```

2. 自己無撞着計算

> `dcore` srvo3.ini

3. 自己無撞着計算の収束を確認

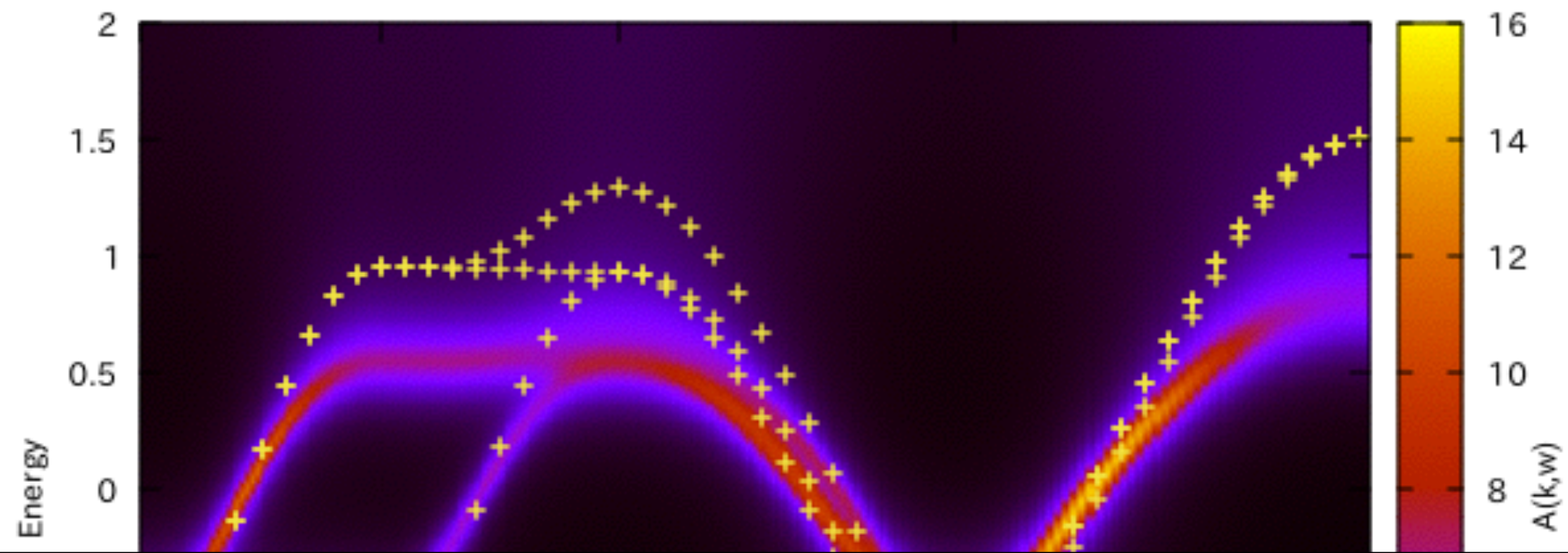
> `dcore_check` srvo3.ini

4. 物理量解析 $i\omega_n \rightarrow \omega$

> `dcore_post` srvo3.ini

計算例: SrVO₃ with Quantum ESPRESSO (2)

https://issp-center-dev.github.io/DCore/tutorial/srvo3_qe/srvo3.html



```
[model]
lattice = wannier90
seedname = srvo3
nelec = 1.0
ncor = 1
norb = 3
interaction = respack
bvec=[(1.627091,0.0,0.0),(0.0,1.627091,0.0),(0.0,0.0,1.627091)]
```

<https://issp-center-dev.github.io/DCore/documentation.html>