

文部科学省次世代IT基盤構築のための研究開発
「イノベーション基盤シミュレーションソフトウェアの研究開発」

CISS フリーソフトウェア

量子機能解析ソルバー・ナノデバイスシミュレーターの研究開発

ASCOT Version 4.10

ユーザー・マニュアル

本ソフトウェアは文部科学省次世代IT基盤構築のための研究開発「イノベーション基盤シミュレーションソフトウェアの研究開発」プロジェクトによる成果物です。本ソフトウェアを無償でご使用になる場合「CISSフリーソフトウェア使用許諾条件」をご了承頂くことが前提となります。営利目的の場合には別途契約の締結が必要です。これらの契約で明示されていない事項に関して、或いは、これらの契約が存在しない状況においては、本ソフトウェアは著作権法など、関係法令により、保護されています。

お問い合わせ先

(契約窓口)

(財)生産技術研究奨励会

〒153-8505 東京都目黒区駒場 4-6-1

(ソフトウェア管理元) 東京大学生産技術研究所 革新的シミュレーション研究センター

〒153-8505 東京都目黒区駒場 4-6-1

FAX: 03-5452-6662

E-mail: software@ciss.iis.u-tokyo.ac.jp

COPYRIGHT of the program codes

Copyright(C) 2002-2010 Hisashi Kondo, Naoki Watanabe

It is understood by the authors that the Institute of Industrial Science (IIS), the University of Tokyo, distributes this program as "CISS Free Software" with users' agreement with the terms and conditions written in the file, LICENSE.pdf or LICENSE-J.pdf (in Japanese).

HISTORY

This set of the computer programs "ASCOT" had been developed by Hisashi Kondo since 2002 as a part of the national project "Frontier Simulation Software for Industrial Science (FSIS)" that is supported by the IT program of the Ministry of Education, Culture, Sports, Science and Technology (MEXT) of Japan. Since 2005, this program set has been developed further by Hisashi Kondo and Naoki Watanabe as a part of the national project "Revolutionary Simulation Software (RSS21)" that is supported by the next-generation IT program of MEXT of Japan. Since 2008, this program set has been developed further by Hisashi Kondo and Naoki Watanabe as a part of the national project "Research and Development of Innovative Simulation Software (RISS)" that is supported by the next-generation IT program of MEXT of Japan. The manual was written by Hisashi Kondo. The activity of "Quantum Function Analysis, Nano Device Simulator", RISS, is supervised by Takahisa Ohno.

CONTACT ADDRESS

Center for Research on Innovative Simulation Software (CISS)

Institute of Industrial Science (IIS), University of Tokyo

4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan

FAX +81-(0)3-5452-6662

E-mail software@ciss.iis.u-tokyo.ac.jp

URL <http://www.ciss.iis.u-tokyo.ac.jp>

* When distributing CISS Software duplications, the user must attach the full text in this file.

License to Use CISS Free Software for noncommercial purposes Terms and Conditions of the CISS Free Software License

The Center for Research on Innovative Simulation Software (CISS) at the Institute of Industrial Science, the University of Tokyo gives explicit permission for anyone to use any or all of the free software that is maintained and made publicly available at the CISS site free of charge, subject to the terms and conditions detailed below.

1. Definition of CISS Free Software

CISS Free Software is any software explicitly marked "CISS Free Software" in CISS project source programs, object programs, specifications, design specifications, data, implementation results, and instruction manuals.

2. Extent of Free Use

Users may use CISS Free Software free of charge to run their own data, and use any results obtained for their own personal use. Users also have the rights to copy, to modify, and to redistribute the CISS Free Software.

3. Rules for Modification and Distribution

If the user creates a modified version of CISS Free Software by modifying the software itself, by incorporating it into other software, or any other means; then copies and/or distributes the software, the user must retain the words "CISS free software" in the name of the modified version (e.g., if the CISS free software is named ProteinDF, the new software is named -----/ProteinDF.); however, this shall not apply if the user concludes separately a contract for the purpose of profit-making business. And also the user displays a copyright notice in the modified version.

The "copyright notice" in the internal code of the CISS Free Software may not be altered for any reason, except to update or add to modification records such as altering the name of the modifier or the date of modification.

4. Copyright Notice

Users must prominently and conspicuously display the copyright notice in every CISS Free Software copy at or near the beginning of the credits along with the name of the software, the version, and the copyright holder. When distributing copies of CISS Free Software, the user must attach the full text of these Terms and Conditions without any changes.

5. User Obligations

To publicly acknowledge that results have been achieved using CISS Free Software, users are obligated to clearly display the name, version, and copyright holder, and acknowledge that "these results were achieved by using Innovative Simulation Software for an Industrial Science Project."

If the user modifies the CISS Software and acknowledges that results were achieved using the software, the user must attach an explanation detailing how the software was modified.

We request that users report any bugs or problems they discover in using the CISS Software to the Center for Research on Innovative Simulation Software at the Institute of Industrial Science, the University of Tokyo. Users may not publicly announce or disclose bugs or problems they discover in CISS software without permission.

6. Commercial Use

If a user intends to use CISS Free Software for a commercial purpose such as described in examples (1)-(3) below, the user must enter into a separate commercial license agreement before using the CISS software.

- (1) A user copies and distributes CISS Free Software, then demands compensation from the recipient for the software itself as a copyrighted product or for copying and distributing the software.
- (2) A user (corporate or individual) uses CISS Free Software not for personal use but to provide services to other parties, regardless of whether the services are offered gratis or for a fee.
- (3) A user seeks to assume a right of pledge, a security interest, or some other form of commercial interest in CISS Free Software, including portions of the software that were modified by the user.

However, if a public entity seeks to provide services using CISS software for the purpose disseminating the software, we require an exchange of memorandums between the CISS and the entity (in lieu a conventional for-profit license agreement) detailing the nature of the service, regardless of whether the proposed service is offered gratis or for a fee.

The user acknowledges in advance that if he or she violates any of the provisions of this agreement, the copyright holder of any software shall prohibit the user from using the software. The user also acknowledges in advance that the copyright holder is entitled to be compensated by an amount equivalent to any profit gained by the user through the violation of the terms of this agreement.

7. No Warranty

The Institute of Industrial Science (IIS), the University of Tokyo, the Foundation for the Promotion of Industrial Science, and other concerned parties disclaim all warranties with respect to the quality, the performance, or the results of CISS Free Software, either express or implied. The user assumes sole responsibility for the use of CISS software including any damages or losses arising out of the use of the CISS software.

8. Violations of Terms and Conditions

If a user is found to be in violation of these Terms and Conditions, he or she agrees to immediately pursue any and all steps required by the Institute of Industrial Science, the University of Tokyo to get back into compliance.

- End of the terms and conditions-

CISS フリーソフトウェア使用許諾条件

東京大学生産技術研究所 革新的シミュレーション研究センター（以下 革新センター）は、次の条件や制限のもとで、革新センターで管理・公開するプロジェクト等による成果物の全てまたは一部を無償で使用することを許諾します。

1. CISS フリーソフトウェアの定義

革新センター（CISS）で管理しているソースプログラム、オブジェクトプログラム、仕様書、設計書、データ、実行結果および マニュアルなどの内、インターネット上に公開しているソフトウェアを CISS フリーソフトウェアと呼びます。

2. 無償使用の範囲

利用者が CISS フリーソフトウェアを無償で使用できる行為には、自己のために CISS フリーソフトウェアを任意のデータを用いて実行する行為、その結果を利用者の自己のために使用する行為、CISS フリーソフトウェアを複製し頒布する行為、および、CISS フリーソフトウェアを改変しそれを実行する行為等を含みます。

3. 改変・頒布での遵守事項

CISS フリーソフトウェアを変更したり、他のソフトに組み込む等の行為により、改変した CISS フリーソフトウェアを複製・頒布する場合は、そのソフトウェア名には CISS フリーソフトウェアの名称を残して（例えば、CISS フリーソフトウェアの名称を ProteinDF とした場合、 / ProteinDF のようにネーミング）下さい。ただし、別途営利目的の場合における実施許諾契約を締結している場合はこの限りではありません。また、著作権表示を行うことを義務づけます。

目的の如何を問わず、CISS フリーソフトウェア内部コードの『著作権表示』記載部分を修正する行為は、改変者氏名や改変日時などの改変記録を追加する場合を除き、禁止されています。

4. 著作権の表示

利用者は、各々の CISS フリーソフトウェアの複製物に、ソフトウェア名・バージョン・著作者氏名などの著作権表示を表示の先頭部等の箇所に適切かつ目立つように掲載するとともに、頒布する場合は、複製物に本許諾条件の全文をそのまま添付しなければなりません。

5. 利用者義務

CISS フリーソフトウェアを利用した結果を公表する場合には、関連プロジェクト等の成果を利用した（例：『革新的シミュレーションソフトウェアの研究開発プロジェクトの成果を利用した』）旨を、使用した CISS ソフトウェアの名前、バージョン、著作者氏名などの記載とともに、明示して下さい。

利用者が CISS ソフトウェアを改変し、その実行結果を公表する場合は、改変内容や改変履歴が特定できる説明を添付して公表しなければなりません。

利用者が CISS ソフトウェアのバグや不具合を発見した場合、革新センターに報告して下さい。発見したバグや不具合を許可なく公表したり、第三者に知らせることを禁止します。

6. 営利目的に使用する場合

利用者は、CISS フリーソフトウェアを下記(1)～(3)に例示するような営利目的に使用する場合には、事前に別途営利目的の場合における実施許諾契約を締結する必要があります。

- (1) 利用者が CISS フリーソフトウェアを複製・頒布する場合、著作物としての対価のみならず、複製ないし頒布に必要な経費など経済的価値を、頒布を受ける者に対して提示ないし要求すること。
- (2) 法人を含み利用者は、自己の目的に限り CISS フリーソフトウェア実行が許諾されているものであり、有償無償を問わず第三者へのサービスのために CISS ソフトウェアを実行する行為をすること。
- (3) 利用者は、自己が改変した部分も含み、CISS フリーソフトウェアを質権や担保など、いかなる商取引の対象に加えること。

ただし、公的機関が当該ソフトウェアの普及促進を目的としてそれを利用したサービスを提供する場合は、そのサービスの有償無償を問わず、別途その内容に関して革新センターとの間で覚書等をお互いに交わすことをもって営利目的の実施許諾契約締結の代用とすることができるものとします。

利用者が本項に反する行為を行った場合には、各ソフトウェア等の著作権者によりその利用を差し止められることを利用者は予め了解します。かつ、利用者は、それにより得た利益相当額の賠償をもとめられることも予め了解します。

7. 無保証

CISS フリーソフトウェアは、その品質や性能あるいは実行結果について、利用者に対してはいかなる保証もされていません。利用者は自己の責任において使用することに同意することとし、もし使用することにより損害が生じた場合には、第三者への損害や被害の修復も含み、その結果責任は全て利用者に戻ることとします。

8. 利用者が本使用許諾条件に違反した場合

利用者が本使用許諾条件に違反した場合には、利用者は、革新センターがその状態を是正するために必要と認めて行う措置に無条件に従うものとします。

目次

1	はじめに	1
2	使用条件	1
3	動作環境	2
4	非平衡グリーン関数法	2
5	計算手順	4
6	擬原子軌道および擬ポテンシャルについて	6
7	タイトバインディングパラメータについて	6
8	ディレクトリー構成	8
9	入力ファイル	9
9.1	平衡系の電子状態計算	9
9.2	密度汎関数理論に基づいた伝導特性解析	11
9.3	タイトバインディング法を用いた炭素系の伝導特性解析	13
10	計算例	24
10.1	ベンゼンジチオール分子架橋系	24
10.1.1	Procedure 1-2	24
10.1.2	Procedure 1-1	27
10.1.3	Procedure 2 & 3	28
10.2	Au(111) 表面に挟まれたベンゼンジチオール分子	30
10.2.1	Procedure 1-2	30
10.2.2	Procedure 1-1	31
10.2.3	Procedure 2 & 3	32
10.3	C ₆₀ 内包カーボンナノチューブ	36
10.3.1	Procedure 3	36

表の目次

1	擬原子軌道・擬ポテンシャルデータファイル	8
2	平衡系の電子状態計算のための入力ファイルで使用されるパラメータの説明・その1	15
3	平衡系の電子状態計算のための入力ファイルで使用されるパラメータの説明・その2	16
4	平衡系の電子状態計算のための入力ファイルで使用されるパラメータの説明・その3	17
5	平衡系の電子状態計算のための入力ファイルで使用されるパラメータの説明・その4	18
6	メイン入力ファイルで使用されるパラメータの説明・その1	19
7	メイン入力ファイルで使用されるパラメータの説明・その2	20
8	メイン入力ファイル内で指定されたファイル”file{scf_file}”で使用されるパラメータの説明・その1	21
9	メイン入力ファイル内で指定されたファイル”file{scf_file}”で使用されるパラメータの説明・その2	22
10	メイン入力ファイル内で指定されたファイル”file{tb_coordinate}”で使用されるパラメータの説明	23
11	10.1 節の計算で用いられた擬原子軌道	25

図の目次

1	対象となる架橋系の例	3
2	密度行列 $\rho_{S,ij}$ を計算する際の積分経路	4
3	パイアス下における密度行列 $\rho_{S,ij}$ を計算する際の積分経路	4
4	反復計算の概略	5
5	計算手順	6
6	Al 原子の擬原子軌道	7
7	Al 原子の擬ポテンシャル	7
8	ベンゼンジチオール分子架橋系の結晶構造	24
9	ベンゼンジチオール分子架橋系のトランスミッション	30
10	Au(111) 表面に挟まれたベンゼンジチオール分子の結晶構造	30
11	Au(111) 表面に挟まれたベンゼンジチオール分子架橋系のトランスミッション	35
12	C ₆₀ 内包カーボンナノチューブの結晶構造	36
13	C ₆₀ 内包カーボンナノチューブのバンド図	38
14	C ₆₀ 内包カーボンナノチューブのトランスミッション	39
15	C ₆₀ 内包カーボンナノチューブの状態密度	39

1 はじめに

伝導特性計算プログラム ASCOT (*Ab initio Simulation COde for quantum T*ransport) は、密度汎関数法 (DFT) および非平衡グリーン関数法 (NEGF) を用いたナノ構造の量子伝導特性の解析を効率的に精度良く行うことができるプログラムである。またタイトバインディング法およびグリーン関数法を用い、炭素系の伝導特性を求めることも可能である。

一般的な電子状態計算では境界条件として周期系や孤立系を考え、その平衡状態での諸性質を調べる。これに対して、伝導現象はそのような平衡状態計算では扱えない。本プログラムでは、解析すべきナノ構造体の両側に半無限の電極を考え、そこから入射した電子の散乱状態および伝導特性を求めることができる。実際の計算に際しては半無限電極の効果を表面グリーン関数を用いた自己エネルギーとして記述し、ナノ構造部分のグリーン関数へ繰り込むことによって電子状態の計算を行う。このとき両電極については境界条件（開放系の境界条件）となっていて、電子状態は中央部分でのみ計算される。このため、散乱が起これらと思われる範囲までの電極部分も中央のナノ構造体に組み込んだ拡張されたナノ構造部分（拡張分子）として扱う必要がある。求めたナノ構造部分の電子状態から、電極表面や結合状態も反映した伝導特性を求めることができる。

更新情報

○ ver. 4.1 における更新点 (2010 年 6 月)

- 力の計算の基本機能、および平衡系の電子状態計算の解析機能（バンド分散、状態密度、分子軌道の計算機能）を追加・整備。

○ ver. 4.0 における更新点 (2009 年 6 月)

- 複数の波数 k を用いた伝導特性計算の機能を追加。（これに伴いデータファイルなどの仕様を変更。）
- 密度行列の更新に対して加速法を導入。
- 平衡系の電子状態計算の並列化。

○ ver. 3.0 における更新点 (2007 年 12 月)

- 密度汎関数理論および非平衡グリーン関数法を用いたプログラムの基本機能を追加。

○ ver. 1.0 のリリース (2005 年 6 月)

- タイトバインディングパラメータを用いたグリーン関数法による伝導特性の解析機能を公開。

2 使用条件

本プログラム ASCOT は文部科学省次世代 IT 基盤構築のための研究開発「イノベーション基盤シミュレーションソフトウェアの研究開発」プロジェクトによる成果物である。本ソフトウェアを無償で使用する場合「CISS フリーソフトウェア使用許諾条件」への同意が前提となる。本マニュアルまたはパッケージ内の LICENSE_J.pdf などを確認していただきたい。

文献 1 および文献 2 に ASCOT による結果を紹介している。実行結果を公表する場合は、タイトバインディング版を使用の際は文献 1、密度汎関数法版を使用の際は文献 2 を、ソフトウェア名などとともに記載していただきたい。また、ASCOT を用いた成果は文献 1 および文献 2 の他、文献 3-10 に紹介されている。

3 動作環境

ASCOT は UNIX マシン、Linux マシンにおいてコンパイルおよび実行されることを想定している。その際、mpich ライブラリー、lapack ライブラリー および fftw ライブラリー にあたるライブラリーが必要である。動作確認は、SGI Altix ICE8200EX および Altix 4700 (fortran: intel compiler ver. 11 + library: SGI MPT (intel MPI), intel MKL, fftw3) において行った。

4 非平衡グリーン関数法

ASCOT では、原子軌道による線形結合 (LCAO) 法に基づく密度汎関数理論 (DFT) および非平衡グリーン関数 (NEGF) 法 [2, 3, 11] を適用することで伝導特性を求めることができる。ここでは、NEGF 法の理論をまとめておく。ただし、簡略化して、スピン遍極の無い場合を示す。

図 1 に示される構造が対象となる架橋系である。このような系を次の 3 つの部分、すなわち (a) 左側半無限電極 (レフト領域: 図 1 の左側の濃色で示された原子) (b) 分子などの散乱体と左右電極の数層を含む散乱領域 (ナノ構造部分)、および (c) 右側半無限電極 (ライト領域: 図 1 の右側の濃色で示された原子) に分割して考える。このとき、レフト領域とライト領域の間の相互作用はないものと仮定する。するとそれぞれのスピンに対してハミルトニアン \mathcal{H} および重なり積分 S は

$$\mathcal{H} = \begin{pmatrix} H_L & H_{LS} & 0 \\ H_{SL} & H_S & H_{SR} \\ 0 & H_{RS} & H_R \end{pmatrix}, \quad (1)$$

$$S = \begin{pmatrix} S_L & S_{LS} & 0 \\ S_{SL} & S_S & S_{SR} \\ 0 & S_{RS} & S_R \end{pmatrix}, \quad (2)$$

と書くことができる。ここで、レフト領域とライト領域は結晶の電子状態と一致するものと仮定する。すなわちハミルトニアン成分 $H_{L/R}$ 、 $H_{SL/SR}$ および $H_{LS/RS}$ は周期境界条件のもとで計算される。 H_S は、下に示す電極の状態によって決められる境界条件の下で反復計算を実行した上で求まる。

散乱領域の遅延グリーン関数は

$$G_S^r(\varepsilon) = [\varepsilon^+ S_S - H_S - \Sigma_L^r(\varepsilon) - \Sigma_R^r(\varepsilon)]^{-1}, \quad (3)$$

$$\Sigma_L^r(\varepsilon) = (\varepsilon^+ S_{SL} - H_{SL}) G_L^r(\varepsilon) (\varepsilon^+ S_{LS} - H_{LS}), \quad (4)$$

$$\Sigma_R^r(\varepsilon) = (\varepsilon^+ S_{SR} - H_{SR}) G_R^r(\varepsilon) (\varepsilon^+ S_{RS} - H_{RS}), \quad (5)$$

と求められる。ただし、 $G_{L/R}(z) = (z S_{L/R} - H_{L/R})^{-1}$ はレフト・ライト領域の表面グリーン関数である [z は任意の複素変数]。 $G_S^r(\varepsilon)$ を求めるのに必要な散乱領域に隣接した原子に対する表面グリーン関数は、漸化式を解くことによって求めることができる [12, 13]。

散乱領域の密度行列 $\rho_{S,ij}$ は、 $G_{S,ij}^<(\varepsilon)$ を散乱領域のレスサージグリーン関数とすると、

$$\rho_{S,ij} = \frac{1}{2\pi i} \int_{-\infty}^{\infty} d\varepsilon G_{S,ij}^<(\varepsilon), \quad (6)$$

と求められる。平衡系では、 $\mu = \varepsilon_F$ をケミカルポテンシャル、そして $f(\varepsilon - \mu)$ をフェルミ分布関数とすると、 $G_{S,ij}^<(\varepsilon) \propto f(\varepsilon - \mu)$ であり、 $\varepsilon < \mu$ では $G_{S,ij}^<(\varepsilon)/2\pi i = -\Im G_{S,ij}^r(\varepsilon)/\pi$ であるので、密度行列 $\rho_{S,ij}$ は

$$\rho_{S,ij} = -\frac{1}{\pi} \Im \left[\int_{-\infty}^{\mu} d\varepsilon G_{S,ij}^r(\varepsilon) \right], \quad (7)$$

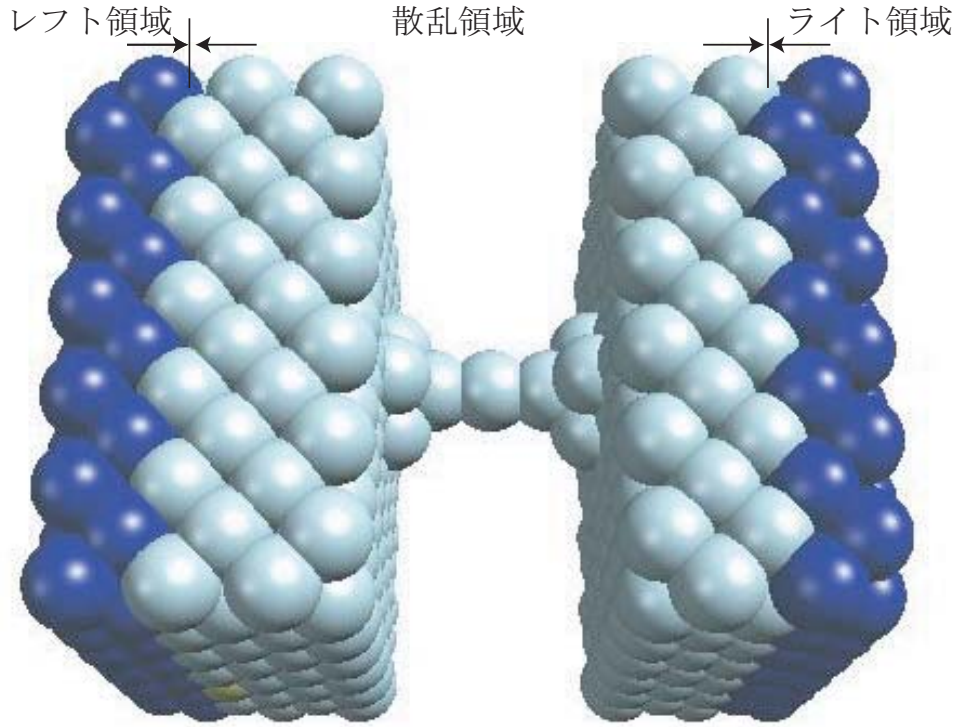


図 1: 対象となる架橋系の例 (Al 原子細線架橋系の構造)。薄色の原子は散乱領域として扱われる原子、濃色の原子がレフト領域、ライト領域として扱われる原子を表す。

によって計算することができる。式 (7) の ε に関する積分は複素平面の上半面の適切な積分経路を使って求められる。その積分経路のひとつを図 2 ($E_{\min} \rightarrow E_{\min} + i\delta \rightarrow \mu + i\delta \rightarrow \mu$) に示す。ただし、 E_{\min} は価電子の取り得る最小のエネルギーよりも小さな値であるとし、 δ は適当な正の値であるとする。非平衡系の場合、密度行列 $\rho_{S,ij}$ は

$$\rho_{S,ij} = -\Im \left[\frac{1}{\pi} \int_{-\infty}^E d\varepsilon G_{S,ij}^r(\varepsilon) + \frac{1}{2\pi i} \int_E^{\infty} d\varepsilon G_{S,ij}^<(\varepsilon) \right], \quad (8)$$

$$\begin{aligned} G_S^<(\varepsilon) &= G_S^r(\varepsilon) \Sigma_S^<(\varepsilon) G_S^a(\varepsilon) \\ &\approx -G_S^r(\varepsilon) \{ \Sigma_L^r(\varepsilon) - \Sigma_L^a(\varepsilon) \} f(\varepsilon - \mu_L) - \{ \Sigma_R^r(\varepsilon) - \Sigma_R^a(\varepsilon) \} f(\varepsilon - \mu_R) G_S^a(\varepsilon), \end{aligned} \quad (9)$$

により求まる。ただし、 $G^a(\varepsilon) = G(\varepsilon^-)$ は先進グリーン関数、 $\Sigma_{L/R}^a(\varepsilon)$ は先進自己エネルギーを表し、 $E < \min(\mu_L, \mu_R)$ ($\mu_{L/R}$ は左/右の電極のケミカルポテンシャルを表す) である。式 (8) の第一項目は式 (7) と同様に計算される。式 (8) の計算で用いられる積分経路を図 3 に示す。

電荷密度は、密度行列 $\rho_{S,ij}$ より、

$$\rho(\mathbf{r}) = 2 \sum_{ij} \phi_i(\mathbf{r}) \rho_{S,ij} \phi_j^*(\mathbf{r}), \quad (10)$$

と計算される。ここで、 $\phi_i(\mathbf{r})$ は i 番目の原子の原子軌道を表し、係数 2 はスピン自由度による。散乱領域のハートリーポテンシャル $V_H(\mathbf{r})$ はポアソン方程式: $\nabla^2 V_H(\mathbf{r}) = -4\pi\rho(\mathbf{r})$ を解くことによって与えられる。その際、レフト・ライト領域のハートリーポテンシャル $V_{H,L/R}(\mathbf{r})$ が境界条件となる。そして、運動エネルギー項、擬ポテンシャルの非局所項および $V_{\text{hxcl}}(\mathbf{r}) = V_H(\mathbf{r}) + V_{\text{xc}}(\mathbf{r}) + V_{\text{loc}}(\mathbf{r})$ ($V_{\text{xc}}(\mathbf{r})$ は交換相関ポテンシャル、 $V_{\text{loc}}(\mathbf{r})$ は擬ポテンシャルの局所項を表す) からハミルトニアン H_S が求まる。以上の手順による反復計算は密度行列 ρ_S が自己無撞着に決定されるまで繰り返す。

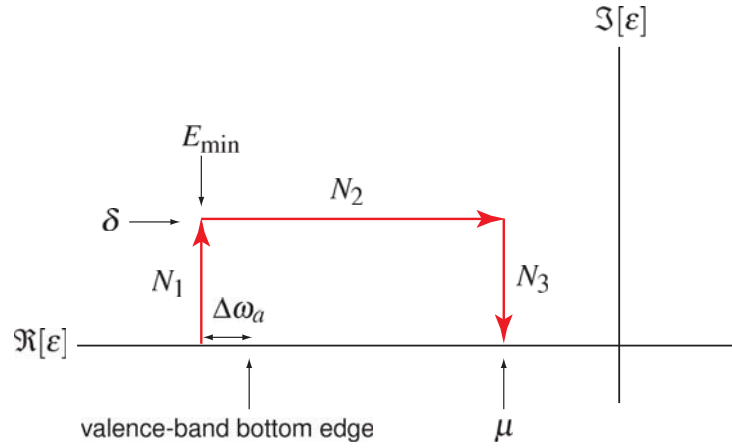


図 2: 密度行列 $\rho_{S,ij}$ を計算する際の積分経路。

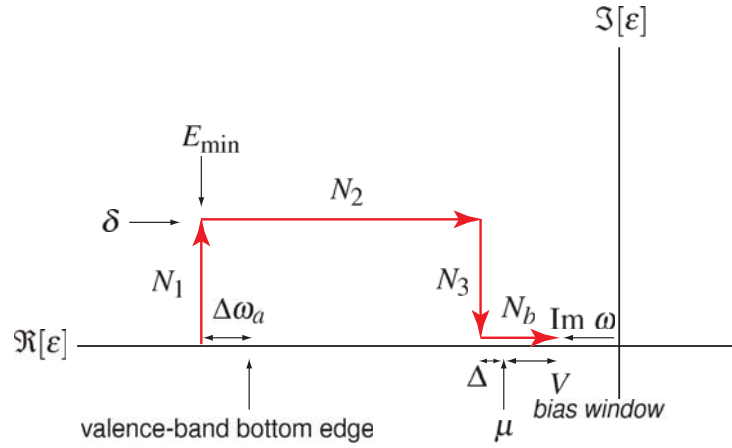


図 3: バイアス下における密度行列 $\rho_{S,ij}$ を計算する際の積分経路。

レフト領域から流れ込む電流値 I は

$$I = 2 \frac{e}{h} \int_{-\infty}^{\infty} d\varepsilon T(\varepsilon) [f(\varepsilon - \mu_R) - f(\varepsilon - \mu_L)], \quad (11)$$

と求まる。ただし、

$$T(\varepsilon) = \text{Tr}[\Gamma_L(\varepsilon) G_S^r(\varepsilon) \Gamma_R(\varepsilon) G_S^a(\varepsilon)], \quad (12)$$

$$\Gamma_L(\varepsilon) = i[\Sigma_L^r(\varepsilon) - \Sigma_L^a(\varepsilon)], \quad (13)$$

$$\Gamma_R(\varepsilon) = i[\Sigma_R^r(\varepsilon) - \Sigma_R^a(\varepsilon)], \quad (14)$$

である。 $T(\varepsilon)$ が対象とした架橋系のトランスミッションである。

上記の非平衡グリーン関数を用いた反復計算の概略を、一般の周期系に対する計算と対比させて図 4 に示す。

5 計算手順

ここでは ASCOT の計算手順を簡単に紹介する。図 5 にその計算手順を示す。第 4 章に NEGF 法の理論を紹介したが、散乱領域の電子状態および系の伝導特性を求めるためにはレフト・ライト領域の電子状態を求める必要

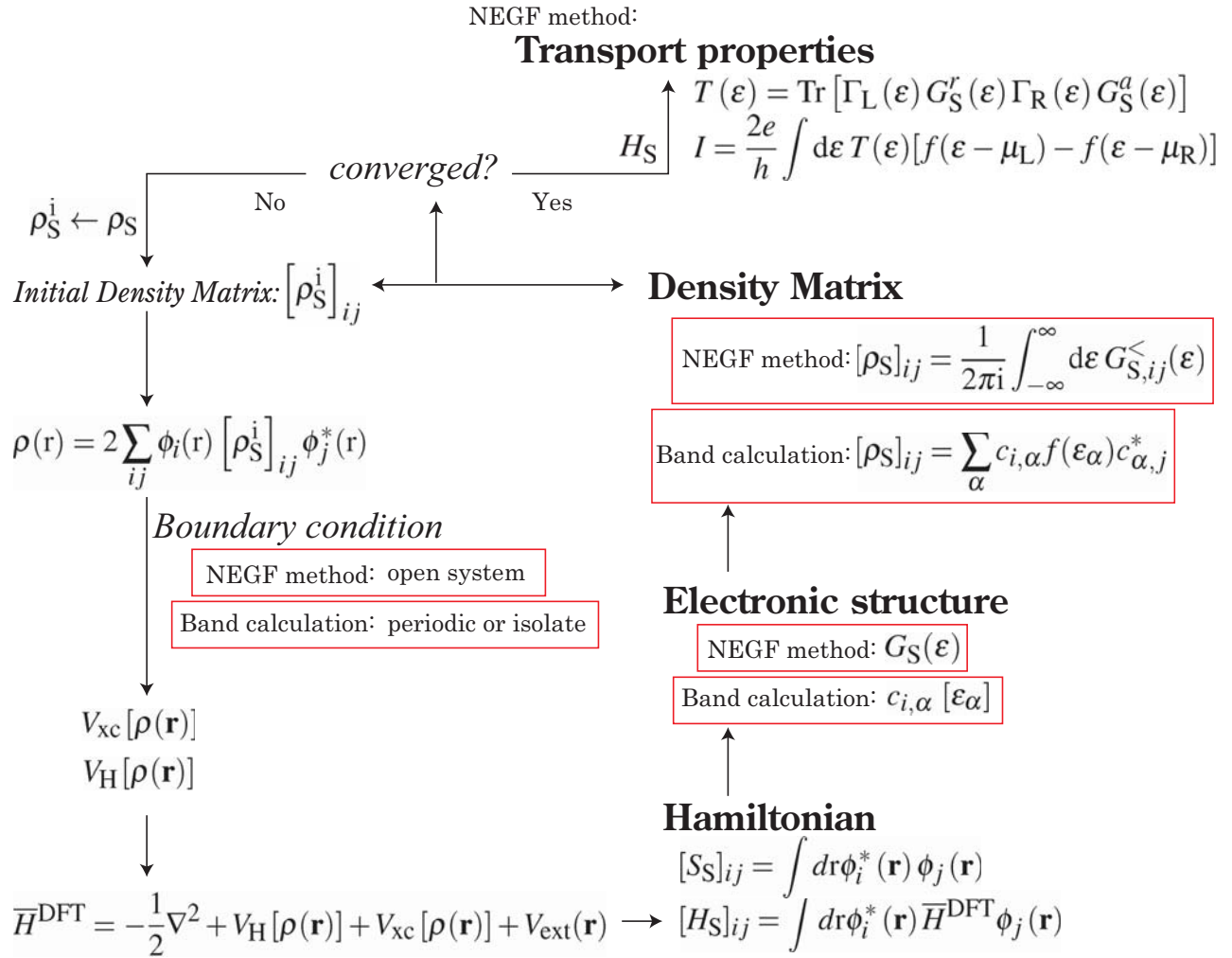


図 4: 反復計算の概略。

がある (図 5 の Procedure 1-1)。つまり、まず $H_{L/R}$ 、 $H_{SL/SR}$ 、 $H_{LS/RS}$ およびポアソン方程式を解く際の境界条件となるレフト・ライト領域のハートリーポテンシャル $V_{H,L/R}(\mathbf{r})$ を決定する。同時に、散乱領域を周期的に取り扱い、その電子状態を求める (図 5 の Procedure 1-2)。ここで求めた密度行列または電荷密度を反復計算の初期値 $\rho_S^i[\rho(\mathbf{r})]$ とする。これらの計算では一般の周期系に対する電子状態計算 (通常のバンド計算) が行われる。以上の結果を基に散乱領域の電子状態を求める (図 5 の Procedure 2)。ここでは開放系の境界条件のもとで反復計算が実行され、対象となっている半無限の電極を持つ系の散乱領域の電子状態 [H_S および ρ_S など] が求まる。以上の手順はタイトバインディング法を用いた炭素系の伝導特性解析を行う際は省略できる。求められた H_S などから、式 (12) に従ってトランスミッションが、そしてバイアス下では式 (11) に従って電流値が求められる。

密度汎関数理論に基づいた計算の実行には、計算実行体と入力ファイルの他、各元素の擬原子軌道 (PAO) 基底関数データファイルと擬ポテンシャルデータファイルが必要となる。PAO 基底関数データファイルと擬ポテンシャルデータファイルに関しては第 6 章を参照のこと。入力ファイルの作成法は第 9 章および第 10 章を、実行方法は第 10 章を参照のこと。

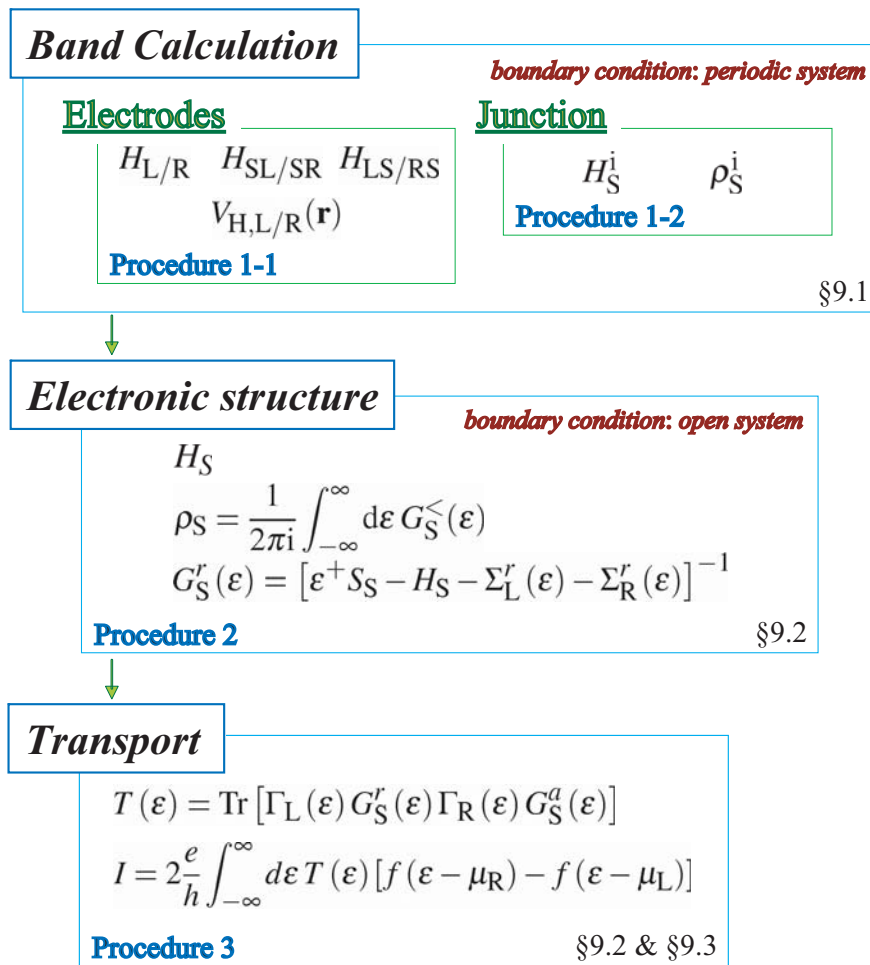


図 5: 計算手順。

6 擬原子軌道および擬ポテンシャルについて

ASCOT の計算には、CIAO [15] を用いて計算された擬原子軌道 $\phi_i(\mathbf{r})$ が必要である (CIAO Ver. 2.16 では非対応)。この擬原子軌道 $\phi_i(\mathbf{r})$ は文献 14 において提案された方法に基づいて計算される。Al 原子の例を図 6(a) に示す。図 6(b) に示すように指定されたカットオフ半径 r_c のところに非常に高いポテンシャルを設定し (このため孤立原子の電子状態があるカットオフ半径 r_c の外にはしみ出さない)、電子の状態を解いて、その固有状態に対する波動関数を擬原子軌道としている [14]。

また、擬ポテンシャル [16, 17] も CIAO [15] を用いて求められる (図 7)。

7 タイトバインディングパラメータについて

ASCOT は、炭素系のパラメータとして文献 18, 19 で提案されたタイトバインディングパラメータと文献 20 で提案されたものにも対応している。

文献 18 で提案されたパラメータは DFT に基づいて求められたダイヤモンドおよびグラファイトのバンド構造を再現するように決定されている。さらに、文献 19 においてフラレン内包カーボンナノチューブのバンド構造を再現するよう拡張されている。フラレン内包カーボンナノチューブを計算する際、便宜的に、ナノチューブを形成する炭素原子を C と表記し、内包された炭素原子を C1 と表記することにする (10.3 節を参照)。

詳細は文献 18-20 を参照していただきたい。

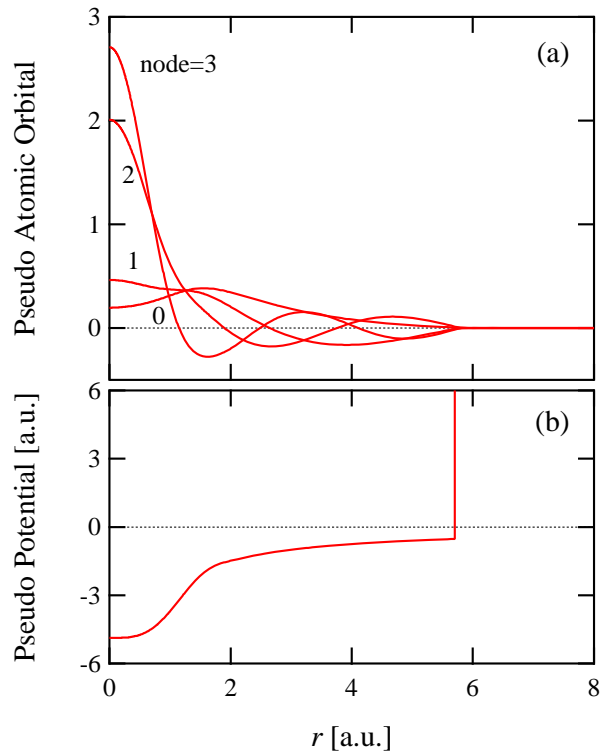


図 6: (a) カットオフ半径 $r_c = 6.0$ a.u. の Al 原子の擬原子軌道 $\phi_i(\mathbf{r})$ (s 軌道)。 (b) 擬原子軌道 $\phi_i(\mathbf{r})$ を求める際に用いた擬ポテンシャルの局所成分。

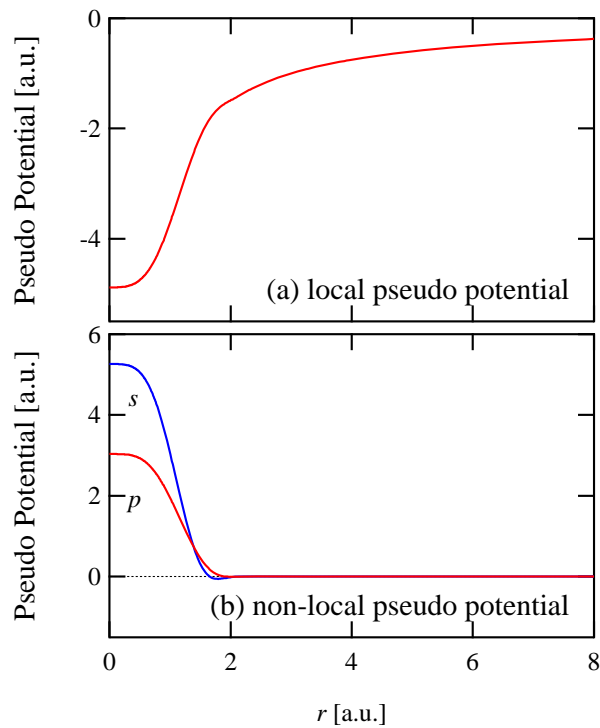


図 7: Al 原子の擬ポテンシャルの (a) 局所成分と (b) 非局所成分の動径成分。

表 1: 擬原子軌道・擬ポテンシャルデータファイル名およびカットオフ半径 r_c とファイルに格納されている s 、 p 、 d 軌道の軌道の数 m_x ($x = s, p, d$)

	擬原子軌道					擬ポテンシャル ファイル名
	ファイル名	r_c [a.u.]	m_s	m_p	m_d	
H	h40_ldapw91.pao2	4.0	5	5	5	h_ldapw91.gncpp2
C	c40_ldapw91.pao2	4.0	5	5	5	c_ldapw91.gncpp2
N	n40_ldapw91.pao2	4.0	5	5	5	n_ldapw91.gncpp2
O	o40_ldapw91.pao2	4.0	5	5	5	o_ldapw91.gncpp2
	o55_ldapw91.pao2	5.5	5	5	5	—
Al	al60_ldapw91.pao2	6.0	5	5	5	al_ldapw91.gncpp2
Si	si60_ldapw91.pao2	6.0	5	5	5	si_ldapw91.gncpp2
S	s55_ldapw91.pao2	5.5	5	5	5	s_ldapw91.gncpp2
Cu	cu60_ldapw91.pao2	6.0	5	5	5	cu_ldapw91.gncpp2
Se	se60_ldapw91.pao2	6.0	3	3	3	se_ldapw91.gncpp2
Ag	ag60_ldapw91.pao2	6.0	5	5	5	ag_ldapw91.gncpp2
Te	te65_ldapw91.pao2	6.5	3	3	3	te_ldapw91.gncpp2
Au	au60_ldapw91.pao2	6.0	5	5	5	au_ldapw91.gncpp2
Pt	pt60_ldapw91.pao2	6.0	3	3	3	pt_ldapw91.gncpp2

8 ディレクトリー構成

ASCOT のディレクトリー構成を以下に示す。

```
ascot/doc ..... ユーザーマニュアル格納ディレクトリー
/src ..... ソースファイル格納ディレクトリー
/bin ..... 計算実行体格納ディレクトリー
/data ..... データファイル格納ディレクトリー
/samples ..... サンプルデータ格納ディレクトリー
/work ..... 作業用ディレクトリー
```

ソースファイル格納ディレクトリー”src”にはソースファイルが格納されている。このディレクトリーで、”makefile”を使用環境にあわせて書き換えた上で、

```
% make
```

とすると、”makefile”内の計算実行体名の変更を行わない限り、計算実行体”ascot_v410”が作成される。さらに、

```
% make install
```

とすると計算実行体”ascot_v410”を計算実行体格納ディレクトリー”bin”にコピーする。

データファイル格納ディレクトリー”data”には計算実行時に必要な擬原子軌道・擬ポテンシャルおよびタイトバインディングパラメータのデータファイルがある。擬原子軌道・擬ポテンシャルデータファイルのリストを表 1 に示す。これらは CIAO [15] により作成された。その際交換相関汎関数として LDA-PW91 [21] を用いた。擬原子軌道の条件 [カットオフ半径およびファイルに格納されている s 、 p 、 d 軌道の軌道の数 m_x ($x = s, p, d$)] も表 1 も示されている。計算に使用する際、擬原子軌道の数をおこの m_x を超えて指定することはできない (詳細は 9.1 節および 10.1 節参照)。また、文献 18, 19 に対するタイトバインディングパラメータのデータファイルは”rtbh_c.parameter.in”であり、文献 20 に対するそれは”gsp_c.parameter.in”である。

サンプルデータ格納ディレクトリー”samples”には例題として入力ファイルとその結果(トランスミッションのデータなど)が収められている。入力ファイルの書き方などは第9章を参照のこと。

9 入力ファイル

計算を行う際に必要な入力ファイルについて説明する。

各パラメータは役割ごとに分類され、それらは”*keyword*{...}*keyword*”によってまとめられて指定される。例えば、

```
general{
    dft__negf    = negf          #(dft | negf)
    ...
}general
```

のように項目”*keyword* = general”内でいくつかのパラメータが設定される。各パラメータは上記の例 (dft__negf = negf) のように、”変数名”、”等号記号”、”値”の3つの要素を並べて指定される。このとき、各要素は空白で区切られている必要がある。”#”より右側は注釈となる。

以下でASCOTの入力ファイルの例とパラメータについての説明(表2-10)を示す。ここでは便宜的に複数のファイルに分けているが、すべてを一つのファイルに指定することも可能である。

9.1 平衡系の電子状態計算

平衡系の電子状態計算(周期系および孤立系の境界条件の下での計算)を行う際の入力ファイルの例を以下に示す。パラメータの説明は表2-5に示す。

```
general{
    dft__negf    = dft          #{dft | negf}
    system_name = junction    # necessary
}general

accuracy{
    energy_cutoff      = 100.0 Ry          # default=150Ry
    temperature        = 500.0 K          # default=300K
    scf_convergence    = 1.e-6 hartree    # default=0.000001hartree
    scf_criterion_type = density          # {fermi|orbital|total|density}, default=fermi
    num_max_iteration  = 100              # default=100

    ksampling{
        method = mesh                # {gamma|mesh}, default=gamma
        mesh{
            1 1 1 # nx ny nz
        }mesh
    }ksampling

    xctype          = ldapw91          # {ldapz81|ldapw91}, default=ldapw91
    element_data_path_pao = "../../data/" # default="."
    element_data_path_vps = "../../data/" # default="."

    num_element      = 4
    element_list{
        Au "au60_ldapw91.pao2"          "au_ldapw91.gncpp2"          s2p1d1
        S  "s55_ldapw91.pao2"          "s_ldapw91.gncpp2"          s2p2d2
        C  "c40_ldapw91.pao2"          "c_ldapw91.gncpp2"          s2p2
        H  "h40_ldapw91.pao2"          "h_ldapw91.gncpp2"          s2p2
    }element_list

    calc_force = off
}accuracy
```



```

structure{
  spin_polarization = off # {on|off}, default=off
  unit_cell{
    unit = ang # {au|ang}
    vector{
      0.00000000000000 5.86991000000000 10.16699000000000
      0.00000000000000 11.73981000000000 0.00000000000000
      26.91954293108431 0.00000000000000 0.00000000000000
    }vector
  }unit_cell
  atom_list{
    num_atom = 40
    num_left_atom = 11
    num_right_atom = 11
    unit = ang # {au|ang}
    atoms{
      1 Au 0.00000000000000 1.46748000000000 2.54175000000000 5.50000 5.50000
      2 Au 0.00000000000000 2.93496000000000 5.08349000000000 5.50000 5.50000
      3 Au 0.00000000000000 4.40243000000000 2.54175000000000 5.50000 5.50000
      4 Au 2.39638000000000 1.46748000000000 0.84725000000000 5.50000 5.50000
      5 Au 2.39638000000000 2.93496000000000 3.38899000000000 5.50000 5.50000
      6 Au 2.39638000000000 4.40243000000000 0.84725000000000 5.50000 5.50000
      7 Au 2.39638000000000 4.40244000000000 5.93073000000000 5.50000 5.50000
      8 Au 2.39638000000000 1.46749000000000 5.93073000000000 5.50000 5.50000
      9 Au 4.79276000000000 1.46748000000000 4.23624000000000 5.50000 5.50000
      10 Au 4.79276000000000 2.93496000000000 1.69450000000000 5.50000 5.50000
      11 Au 4.79276000000000 4.40243000000000 4.23624000000000 5.50000 5.50000
      1 Au 7.18914000000000 1.46748000000000 2.54175000000000 5.50000 5.50000
      2 Au 7.18914000000000 2.93496000000000 5.08349000000000 5.50000 5.50000
      3 Au 7.18914000000000 4.40243000000000 2.54175000000000 5.50000 5.50000
      4 S 9.18299548024847 2.93495988846062 3.38975654380107 3.00000 3.00000
      5 C 10.95963958507695 2.93495988845724 3.39368730502919 2.00000 2.00000
      6 C 11.64960600108224 4.14118064210738 3.39146737287916 2.00000 2.00000
      7 C 11.64960600108237 1.72873913480768 3.39146737288114 2.00000 2.00000
      8 C 13.02092241017248 4.14118064209453 3.38651247803053 2.00000 2.00000
      9 C 13.02092241017267 1.72873913482101 3.38651247803271 2.00000 2.00000
      10 H 11.10289648540533 0.77965812370857 3.39340713314200 0.50000 0.50000
      11 H 11.10289648540441 5.09026165320659 3.39340713314238 0.50000 0.50000
      12 H 13.56763192583338 0.77965812372957 3.38457271768971 0.50000 0.50000
      13 H 13.56763192583269 5.09026165318615 3.38457271769030 0.50000 0.50000
      14 C 13.71088882614472 2.93495988845818 3.38429254610896 2.00000 2.00000
      15 S 15.48753293108431 2.93495988845838 3.38822330844022 3.00000 3.00000
      16 Au 17.33402293108431 1.46748000000000 4.23624000000000 5.50000 5.50000
      17 Au 17.33402293108431 2.93496000000000 1.69450000000000 5.50000 5.50000
      18 Au 17.33402293108431 4.40243000000000 4.23624000000000 5.50000 5.50000
      1 Au 19.73040293108431 1.46748000000000 2.54175000000000 5.50000 5.50000
      2 Au 19.73040293108431 2.93496000000000 5.08349000000000 5.50000 5.50000
      3 Au 19.73040293108431 4.40243000000000 2.54175000000000 5.50000 5.50000
      4 Au 22.12678293108431 1.46748000000000 0.84725000000000 5.50000 5.50000
      5 Au 22.12678293108431 2.93496000000000 3.38899000000000 5.50000 5.50000
      6 Au 22.12678293108431 4.40243000000000 0.84725000000000 5.50000 5.50000
      7 Au 22.12678293108431 4.40244000000000 5.93073000000000 5.50000 5.50000
      8 Au 22.12678293108431 1.46749000000000 5.93073000000000 5.50000 5.50000
      9 Au 24.52316293108431 1.46748000000000 4.23624000000000 5.50000 5.50000
      10 Au 24.52316293108431 2.93496000000000 1.69450000000000 5.50000 5.50000
      11 Au 24.52316293108431 4.40243000000000 4.23624000000000 5.50000 5.50000
    }atoms
  }atom_list
}structure

charge_mixing{
  mixing_methods = pulay # {simple|pulay|anderson}, default=simple
  mixing_weight = 0.10 # default=0.30
  mixing_weight_simple = 0.01 # default=0.30
  pulay_start = 10 # default=4
  pulay_history = 7 # default=4
}charge_mixing

atom_int{
  num_k = 900 # default=900
  num_r = 900 # default=900
  energy_cutoff = 2500.0 Ry # default=2500.0 Ry
}

```

```

}atom_int

file_in{
  matrices      = ""          # "filename", default=""
}file_in

file_out{
  save_at       = finalstep  # {finalstep|everystep|never}, default=never
  field_format  = cube       # {cube|dx}, default=cube
  density_scf   = off        # {on|off}, default=off
  matrices      = on         # {on|off}, default=off
  f_negf        = on         # {on|off}, default=off
}file_out

postprocessing{
  band{
    sw_band = off
    unit_cell{
      unit = ang
      vector{
        0.000000000000000  5.869910000000000  10.166990000000000
        0.000000000000000  11.739810000000000  0.000000000000000
        26.91954293108431  0.000000000000000  0.000000000000000
      }vector
    }unit_cell
    num_band_path = 1
    band_path{
      0.000 0.000 0.000  0.500 0.000 0.000  32
    }band_path
  }band
  dos{
    sw_dos = off
    method = gaussian
    gaussian_broadening = 0.1 eV
    e_range{
      501 -20.0 +20.0 eV
    }e_range
    ksampling{
      1 1 1
    }ksampling
  }dos
  mo{
    sw_mo = on
    m_bandnum = ho_lumo    # {band_num|ho_lumo}, default=ho_lumo
    band_range{
      -5 +5
    }band_range
    num_k_points = 1
    k_points{
      0.0 0.0 0.0
    }k_points
  }mo
}postprocessing

```

9.2 密度汎関数理論に基づいた伝導特性解析

伝導計算のメイン入力ファイルの例を以下に示す。パラメータの説明は表 6 および表 7 に示す。

```

general{
  dft_negf      = negf          #{dft | negf}
  model_type    = scf_calc      #{rtb_h | gsp | input | scf_calc}
  calc_type     = trans         #{trans | iv | scf_only | off}
  output_G      = diagonal     #{diagonal | all}
  output_T      = diagonal     #{diagonal | tr | all}
}general

```

```

file{
#  tb_cordinate      = cordinate.in
#  tb_parameter      = "../data/rtbh_c_parameter.in"
#  input_junc        = "../junction/junction.matrices.dat"
#  input_left        = "../electrode/electrode.matrices.dat"
#  input_right       = "../electrode/electrode.matrices.dat"
  scf_file           = condition_scf.in
  self_e_file        = none                #{'file name' | none}
}file

gra_energy{
  mesh_number        = 128
  start_energy       = -10.0000           # eV
  end_energy         =  0.0000           # eV
  im_energy          =  0.000001         # eV
}gra_energy

#iv_energy{
#  mesh_number       = 500
#  im_energy         = 0.1                 # eV
#}iv_energy

surface_green_func{
  method            = 1                   #{1|2}
  max_number        = 40
  criterion          = 0.00001
}surface_green_func

# band & DOS
electrode_left{
  s_band            = on                  #{on | off}
  s_electrod_DOS    = on                  #{on | off}
  k_point_mesh      = 32
}electrode_left

# band & DOS
electrode_right{
  s_band            = off                 #{on | off}
  s_electrod_DOS    = off                 #{on | off}
  k_point_mesh      = 32
}electrode_right

```

メイン入力ファイル内の”file{scf_file}”で指定されたファイルの例を以下に示す。パラメータの説明は表 8 および 9 に示す。

```

negf_scf_general{
#  calc_start        = f_DM              #{f_DM | f_density}
  read_self_file     = none              #{exist | none}

  calc_force = off
}negf_scf_general

negf_scf_file{
  input_hs_junc      = "../junction/junction.matrices.dat"
  input_pr_junc      = "../junction/junction.vd.data"
  input_junc_cond    = "../junction/junction.in.data"
#
  input_hs_left      = "../electrode/electrode.matrices.dat"
  input_pr_left      = "../electrode/electrode.vd.data"
#
  input_hs_right     = "../electrode/electrode.matrices.dat"
  input_pr_right     = "../electrode/electrode.vd.data"
#
  outputt_hs_junc    = "matrix/abred_matrices.dat"
  output_pr_junc     = scf.vd.data

```

```

#
  self_e_file      = "sel/sel_scf.in"
}negf_scf_file

negf_scf_accuracy{
  scf_convergence  = 0.01
  num_max_iteration = 10000
#   ksampling = gamma
}negf_scf_accuracy

negf_scf_mixing{
  mixing_methods   = pulay_dm          #{simple | pulay_dm | anderson_dm}
  mixing_dm_weight = 0.01
  mixing_dm_s_weight = 0.01
  pulay_mix_history = 7
  pulay_mix_start  = 10
}negf_scf_mixing

negf_scf_energy_m1{
  min_energy_pos    = 5.                # eV
  m_v_min_energy_pos = relative         #{real | relative}
  block_cond0{
    im_energy       = 0.63              # eV
    mesh_num_min    = 5
    mesh_num_ima    = 40
    mesh_num_ef     = 15
    mesh_method_ef  = linear            #{linear | log}
  }block_cond0
}negf_scf_energy_m1

#negf_scf_bias{
#  switch           = off                #{on | off}
#  v_left           = 0.d0               # [V]
#  v_right          = 0.d0               # [V]
#  cdm_mesh_num     = 5
#  cdm_im_energy    = 0.0001            # [eV]
#}negf_scf_bias

#negf_spin_rot{
#  switch_left      = off
#  switch_right     = off
#  boundary_g_num   = 0
#  boundary_g_den   = 1
#}negf_spin_rot

```

9.3 タイトバインディング法を用いた炭素系の伝導特性解析

メイン入力ファイル内の”file{tb.coordinate}”で指定されたファイルの例を以下に示す。パラメータの説明は表 10 に示す。

```

tb_buffer_size{
  left_region      = 1
  junc_region      = 1
  right_region     = 1
}tb_buffer_size

tb_distance{
  between_l_c      = 0.00000            # Ang
  between_c_r      = 0.00000            # Ang
}tb_distance

tb_junc_region{
  unit_vector{
    13.58818        0.00000        0.00000
    0.00000         13.58818        0.00000
    0.00000         0.00000         4.26000
  }
}tb_junc_region

```

```
    }unit_vector
    coordinate_system = cartesian
    atom_number       = 3
    atom_list{
      1  C    0.000000   0.000000   0.000000
      2  C    0.000000   0.000000   1.420000
      3  C    0.000000   0.000000   2.840000
    }atom_list
  }tb_junc_region

tb_left_region{
  unit_vector{
    13.58818   0.00000   0.00000
    0.00000   13.58818   0.00000
    0.00000   0.00000   4.26000
  }unit_vector
  coordinate_system = cartesian
  atom_number       = 3
  atom_list{
    1  C    0.000000   0.000000   0.000000
    2  C    0.000000   0.000000   1.420000
    3  C    0.000000   0.000000   2.840000
  }atom_list
}tb_left_region

tb_right_region{
  unit_vector{
    13.58818   0.00000   0.00000
    0.00000   13.58818   0.00000
    0.00000   0.00000   4.26000
  }unit_vector
  coordinate_system = cartesian
  atom_number       = 3
  atom_list{
    1  C    0.000000   0.000000   0.000000
    2  C    0.000000   0.000000   1.420000
    3  C    0.000000   0.000000   2.840000
  }atom_list
}tb_right_region
```

表 2: 平衡系の電子状態計算のための入力ファイルで使用されるパラメータの説明・その 1

識別子	説明
general{	計算の制御
dft_negf	電極などの電子状態の計算を行う場合は”dft”
system_name	名前
}general	
accuracy{	計算の精度に関するパラメータ
energy_cutoff	エネルギーカットオフ ”値 (単位)”と指定。 ”単位” = a.u., eV, hartree, rydberg (def: rydberg)。
temperature	電子温度 ”値 (単位)”と指定。 ”単位” = a.u., eV, K (def: K)。
scf_convergence	収束判定の基準値 ”値 (単位)”と指定。 ”単位” = a.u., eV, hartree, rydberg (def: hartree)。
scf_criterion_type	収束判定の方法 density: 一電子あたりの電子密度の変動 total: 一原子あたりの全エネルギー
num_max_iteration	最大反復回数。 (def: 100)
ksampling{ method mesh{ }mesh }ksampling	波数 k の指定 指定方法 ”gamma”(Γ 点) ”mesh”(後述の”mesh{...}mesh”でメッシュ数を指定) メッシュ数を $(n_a \ n_b \ n_c)$ と指定。 def: (1,1,1)
xctype	交換相関汎関数の種類。 (def: ldapw91) ldapw91: LDA の PW91 型の交換相関汎関数 [21] ldapz81: LDA の PZ81 型の交換相関汎関数 [22]
element_data_path_pao	擬原子軌道データファイルが格納されているディレクトリーへのパス
element_data_path_vps	擬ポテンシャルデータファイルが格納されているディレクトリーへのパス
num_element	element_list で指定される元素の種類の数
element_list{ }element_list	元素のデータファイルなどの指定 一行で元素ごとに”元素名 PAO_file VPS_file PAO_mode”と指定 PAO_file: 擬原子軌道データファイル名 VPS_file: 擬ポテンシャルデータファイル名 PAO_mode: ”s2p2d1”などと指定 [14] この数は表 1 に示した m_x の数を超えてはならない
calc_force	力の計算の実行。 off/on/(opt) (opt: 構造最適化計算の実行)
opt_force_convergence	構造最適化計算 (calc_force = opt) における収束判定の基準値 ”値 (単位)”と指定。 ”単位” = a.u., hartree/bohr (def: 1.0e-3 hartree/bohr)。
opt_num_max_iteration	構造最適化計算 (calc_force = opt) における最大反復回数。 (def: 1)
}accuracy	

表 3: 平衡系の電子状態計算のための入力ファイルで使用されるパラメータの説明・その 2

識別子	説明
structure{	系の構造に関するパラメータ
spin_polarization	スピン分極の有無 off/on
unit_cell{ unit vector{ }vector }unit_cell	単位胞の指定 後述の単位胞の基本ベクトルの単位。"ang" (def) or "au" 単位胞の基本ベクトル (上から順に、a、b、c)
atom_list{ num_atom num_left_atom num_right_atom unit atoms{ }atoms }atom_list	結晶構造の指定 系に含まれる原子の総数 左電極とみなす原子の数 右電極とみなす原子の数 後述の原子の座標の単位。"ang" (def) or "au" 原子ごとに一行でデータを指定 "番号 原子種 x 座標 y 座標 z 座標 Q_{up} Q_{dw} (fixed/free)"と指定 $Q_{up} : Q_{dw}$ は初期値電子分布の上下スピンの比率 fixed/free は構造最適化計算において原子座標の固定・非固定の指定
}structure	
charge_mixing{	密度行列の更新法に関するパラメータ
mixing_methods	密度行列の更新法 simple: 計算された密度行列と現ステップの初期値との線形和 pulay: pulay 法 [23] anderson: anderson 法 [24]
mixing_weight	密度行列の更新の割合。(def: 0.3)
mixing_weight_simple	simple 法で用いる密度行列の更新の割合。(def: 0.3)
pulay_start	pulay で使用するステップ数。(def: 4)
pulay_history	pulay を開始するステップ数。(def: 4)
}charge_mixing	
atom_int{	重なり積分などの計算精度に関するパラメータ
num_k	1次元積分の逆空間のメッシュ数
num_r	1次元積分の実空間のメッシュ数
energy_cutoff	1次元積分のカットオフ "値 (単位)"と指定。"単位" = a.u., eV, hartree, rydberg (def: rydberg)。
}atom_int	

表 4: 平衡系の電子状態計算のための入力ファイルで使用されるパラメータの説明・その3

識別子	説明
file_in{	継続計算のための入力ファイルに関するパラメータ
matrices	継続計算用の密度行列などの行列要素ファイル 原子の番号”_*”は指定しない。
}file_in	
file_out{	計算結果の出力ファイルに関するパラメータ
save_at	計算結果の保存のタイミング finalstep: SCF ループ終了後 everystep: SCF ループの各ループ never: 出力なし
matrices	ハミルトニアン, 密度行列などの行列要素の保存 (継続計算および NEGF 計算で使用) off/on、原子の数だけファイルが作成される。
f_negf	伝導計算のためのファイルの作成 (NEGF 計算で使用) off/on
density_scf	電荷密度の保存 off/on
density_val	孤立原子の電荷密度の保存 off/on
density_pcc	部分内殻補正の保存 off/on
potential_ext	外場ポテンシャルの保存 off/on
potential_har	ハートリーポテンシャルの保存 off/on
potential_exc	交換相関ポテンシャルの保存 off/on
potential_tot	全ポテンシャルの保存 off/on
field_format	保存ファイル形式 cube: Gaussian CUBE 形式
}file_out	

表 5: 平衡系の電子状態計算のための入力ファイルで使用されるパラメータの説明・その 4

識別子	説明
postprocessing{	バンド分散、状態密度、分子軌道の計算に関するパラメータ
band{	バンド分散計算に関する指定
sw_band	バンド分散の計算を行うかのスイッチ
unit_cell{	バンド分散の計算における単位胞の指定
unit	後述の単位胞の基本ベクトルの単位。"ang" (def) or "au"
vector{	単位胞の基本ベクトル (上から順に、a、b、c)
}vector	
}unit_cell	
num_band_path	band_path{...}band_path で指定のバンドを計算する k 点の経路の数
band_path{	バンド分散を計算する k 点の経路と k 点数
	"始点の逆格子ベクトルに対する内部座標
	終点の逆格子ベクトルに対する内部座標 k 点数"と指定
}band_path	
}band	
dos{	状態密度計算に関する指定
sw_dos	状態密度の計算を行うかのスイッチ
method	状態密度の計算手法 (gaussian/tetrahedron) の指定
gaussian_broadening	gaussian 法で用いる broadening 幅。"値 (単位)"と指定 (def: 0.1 eV)。 "単位" = a.u., eV, hartree, rydberg
e_range{	状態密度の計算をするエネルギーの数と範囲を指定
	"エネルギーの数 エネルギーの最小値 最大値 (単位)"と指定 (def: 101 -10 10 eV)。"単位" = a.u., eV, hartree, rydberg
}e_range	
ksampling{	波数 k の指定。メッシュ数を (n_a n_b n_c) と指定。
	無指定の場合は accuracy{ksampling{...}ksampling}accuracy の値
}ksampling	
}dos	
mo{	分子軌道計算に関する指定
sw_mo	分子軌道の計算を行うかのスイッチ
m_bandnum	保存する分子軌道のレベル番号の指定方法
	band_num: レベル番号をレベルの底から数える
	ho_lumo: レベル番号をフェルミ準位を基準にして数える (def)
band_range{	保存する分子軌道のレベル番号の範囲を " N_{\min} N_{\max} " のように指定
}band_range	
unit	分子軌道の計算で用いる後述の単位胞の基本ベクトルの単位。 "ang" (def) or "au"
vector{	分子軌道の計算で用いる単位胞の基本ベクトル (上から順に、a、b、c)
}vector	
num_k_points	計算する分子軌道のサンプリング k 点の数
k_points{	計算する分子軌道の各サンプリング k 点の逆格子ベクトル
	k_a k_b k_c のように内部座標として指定。
}k_points	
}mo	
}postprocessing	

表 6: メイン入力ファイルで 사용되는パラメータの説明・その 1

識別子	説明
general{	計算の制御
dft_negf	伝導特性の解析を行う場合は"negf"
model_type	ハミルトニアンとの与え方 scf_calc: SCF 計算 input: ASCOT による周期境界条件下での計算などで決定したハミルトニアン rtb_h: 現実的タイトバインディングモデル [18,19] gsp: GSP 型タイトバインディングモデル [20]
calc_type	求める量の設定 trans: "gra_energy" で指定されたエネルギーでのトランスミッションを計算 iv: 電流値を計算 (パイアスの値は"file{scf_file}" で指定したファイル内で設定) scf_only: 開放系における電子状態のみを計算 off: 電極の電子状態計算のみを実行
output_G	グリーン関数の出力レベル diagonal: 対角成分のみ all: 全成分
output_T	トランスミッション行列の出力レベル diagonal: 対角成分のみ tr: トランスミッション行列の全成分 all: トランスミッション行列の全成分、 Γ および遅延グリーン関数
band_calc_type	固有値の計算方法 ("electrode_left/electrode_right{s.band}=on" の時に有効) general: 対称行列を用いない計算 symmetry: 対称行列を用いた計算
}general	
file{	計算に用いるファイル名を指定
scf_file	SCF 計算に必要な条件が書かれたファイル名 (このファイルを指定してもよい) "general{model_type}=scf_accel/scf_abred" の時に有効
input_junc	外部で決定した中央部のハミルトニアンを収めたファイル名 "general{model_type}=input" の時に有効。原子の番号"_"**" は指定しない。
input_left	外部で決定した左電極のハミルトニアンを収めたファイル名 "general{model_type}=input" の時に有効。原子の番号"_"**" は指定しない。
input_right	外部で決定した右電極のハミルトニアンを収めたファイル名 "general{model_type}=input" の時に有効。原子の番号"_"**" は指定しない。
tb_cordinate	原子座標を設定しているファイル名 (このファイルを指定してもよい) "general{model_type}=rtb_h/gsp" の時に有効
tb_parameter	タイトバインディング法のパラメータファイル "general{model_type}=rtb_h/gsp" の時に有効 def: "../data/rtbh-c-parameter.in" or "../data/gsp-c-parameter.in"
}file	

表 7: メイン入力ファイルで使用されるパラメータの説明・その 2

識別子	説明
gra_energy{	トランスミッションを計算するエネルギー範囲を指定
mesh_number	分割数
start_energy	始点のエネルギー値 (単位 eV)
end_energy	終点のエネルギー値 (単位 eV)
im_energy	虚部の値 (単位 eV)
}gra_energy	
iv_energy{	file{scf_file} で指定されたバイアスの下での電流値を計算する際の条件
mesh_number	積分のメッシュ数
im_energy	虚部の値 (単位 eV)
}iv_energy	
gra_ksampling{	トランスミッションを計算する波数 k の指定
point	メッシュ数を (n_a, n_b) と指定。def: (1,1)。"gamma" のときは Γ 点で計算
}gra_ksampling	
surface_green_func{	表面グリーン関数の計算条件
method	計算方法の指定 1: 文献 12 の方法 2: 文献 13 の方法
max_number	反復回数の最大値
criterion	収束条件 (単位 1/eV)
}surface_green_func	
electrode_left{	左電極の電子状態計算
s_band	伝導方向のバンド計算の実行 off/on
s_electrod_DOS	状態密度計算の実行 off/on
k_point_mesh	電子状態計算で用いる伝導方向の k_z -点のメッシュ数
}electrode_left	
electrode_right{	右電極の電子状態計算
s_band	伝導方向のバンド計算の実行 off/on
s_electrod_DOS	状態密度計算の実行 off/on
k_point_mesh	電子状態計算で用いる伝導方向の k_z -点のメッシュ数
}electrode_right	

表 8: メイン入力ファイル内で指定されたファイル”file{scf_file}”で使用されるパラメータの説明・その 1

識別子	説明
negf_scf_general{	SCF 計算の制御
calc_start	計算の出発点となるデータ f_DM: ”negf_scf_file{input_hs_junc}”に収められた密度行列 f_density: ”negf_scf_file{input_pr_junc}”に収められた電荷密度
read_self_file	電荷密度計算時の自己エネルギーが収められたファイルの有無 exist: ”negf_scf_file{self_e_file}”で指定されたファイルが存在する場合 none: ”negf_scf_file{self_e_file}”で指定されたファイルが作成される
fft_type	フーリエ変換の計算方法 dft: 離散フーリエ変換 f_2_ft: 2成分高速フーリエ変換 fft: 高速フーリエ変換 ($n=2, 3, 5, 7$)
method_charge_sum	電荷密度の計算の積分経路の指定 square: 図 2 参照。
poisson_method	ハートリーポテンシャルの計算方法 ft: フーリエ変換を用いた方法 gr: グリーン関数を用いた方法 [25]
poisson_K_value	K の値 (”poisson_method=gr”の時に有効。文献 25 参照)
calc_force	力の計算の実行 off/(on)
}negf_scf_general	
negf_scf_file{	SCF 計算に用いるファイル名を指定
input_hs_junc	中央部の密度行列およびハミルトニアンを収めたファイル名 原子の番号”_***”は指定しない。
input_pr_junc	ポテンシャル、電荷密度の初期値を収めたファイル名
input_junc_cond	散乱領域の計算時に用いた計算条件を収めたファイル名 散乱領域の計算時に作成される
input_hs_left	左電極のハミルトニアンを収めたファイル名 原子の番号”_***”は指定しない。
input_pr_left	左電極のポテンシャル(境界条件)、電荷密度を収めたファイル名
input_hs_right	右電極のハミルトニアンを収めたファイル名 原子の番号”_***”は指定しない。
input_pr_right	右電極のポテンシャル(境界条件)、電荷密度を収めたファイル名
output_hs_junc	計算された密度行列およびハミルトニアンを収めるファイル名 原子の番号”_***”が付加されたファイルが原子の数だけ作成される。
output_pr_junc	計算されたポテンシャル、電荷密度を収めるファイル名
self_e_file	電荷密度計算時の積分系路上の自己エネルギーを収めた/収めるファイル名
}negf_scf_file	
negf_scf_accuracy{	SCF 計算の制御
scf_convergence	収束判定の基準値(密度行列の残差により判定)
num_max_iteration	反復回数の最大値
ksampling	k 点メッシュ数を (n_a n_b) と指定。”gamma”のときは Γ 点で計算 指定がない場合は電極の電子状態を計算した際のメッシュ数を用いる
}negf_scf_accuracy	

表 9: メイン入力ファイル内で指定されたファイル”file{scf_file}”で使用されるパラメータの説明・その 2

識別子	説明
negf_scf_mixing{	反復計算時の密度行列の更新法の指定
mixing_methods	密度行列の更新法 simple_dm: 計算された密度行列と現ステップの初期値との線形和 pulay_dm: pulay 法 [23] anderson_dm: anderson 法 [24]
mixing_dm_weight	密度行列の更新の割合。(def: 0.1)
mixing_dm_s_weight	simple 法で用いる密度行列の更新の割合。(def: 0.01)
pulay_start	pulay で使用するステップ数。(def: 4)
pulay_history	pulay を開始するステップ数。(def: 10)
}negf_scf_mixing	
negf_scf_energy_m1{	”negf_scf_general{method_charge_sum}=square”の時の積分経路の指定、図 2。
min_energy_pos	図 2 の E_{\min} の設定 ”m_v_min_energy_pos=relative”の時: バンド端からの相対値 $\Delta\omega_a$ を指定 ”m_v_min_energy_pos=real”の時: E_{\min} の値を指定
m_v_min_energy_pos	”min_energy_pos”の値の意味。 ”relative”: バンド端からの相対値 ”real”: 絶対値
block_cond0{	図 2 の経路の設定
im_energy	積分経路の虚部。図 2 の δ
mesh_num_min	メッシュ数。図 2 の N_1
mesh_num_ima	メッシュ数。図 2 の N_2
mesh_num_ef	メッシュ数。図 2 の N_3
mesh_method_ef	μ 上の経路の分割方法 linear: 線形メッシュで分割 log: 対数メッシュで分割
}block_cond0	
}negf_scf_energy_m1	
negf_scf_bias{	バイアス指定 (0 V の計算の際は省略可)
switch	on: $V(=“v_left”-“v_right”)$ における計算 off: 0 V の計算
v_left	左電極にかけるバイアス値 (単位 V)
v_right	右電極にかけるバイアス値 (単位 V)
cdm_mesh_num	バイアスウィンドウ内の積分経路のエネルギーのメッシュ数。図 3 の N_b
cdm_im_energy	バイアスウィンドウ内の積分経路のエネルギーの虚部 (単位 eV)。図 3 の $\text{Im } \omega$
cdm_om_del_energy	実軸直上の積分経路の増加分 (単位 eV)。図 3 の Δ
}negf_scf_bias	
negf_spin_rot{	スピンの向きの初期条件 (省略可)
switch_left	左電極側のスピンのマジョリティとマイノリティを逆転
switch_right	右電極側のスピンのマジョリティとマイノリティを逆転
boundary_g_num	反転の境界の割合を左から $\frac{B}{A}$ と与えた時の B (整数)
boundary_g_den	反転の境界の割合を左から $\frac{B}{A}$ と与えた時の A (整数)
}negf_spin_rot	

表 10: メイン入力ファイル内で指定されたファイル”file{tb.coordinate}”で使用されるパラメータの説明

識別子	説明
tb_buffer_size{	各領域の単位胞を並べる個数
left_region	レフト領域の個数
junc_region	散乱領域の個数
right_region	ライト領域の個数
}tb_buffer_size	
tb_distance{	各領域間の距離の指定 (単位 Å)
between_l.c	レフト領域と散乱領域との間の距離
between_c.r	ライト領域と散乱領域との間の距離
}tb_distance	
tb_junc_region{	散乱領域の構造などの設定
unit_vector	単位胞の基本ベクトルの設定 (単位 Å) 上から順に、a、b、c 左から順に、x 座標、y 座標、z 座標
coordinate_system	座標系の指定 cartesian: ”atom_list”で指定される座標を実座標値で指定 pucv: 内部座標値で指定
atom_number	セル内の原子数
atom_list	原子座標の設定 ”coordinate_system=cartesian”: 座標を実座標値で指定 (単位 Å) ”coordinate_system=pucv”: 内部座標値 (0 から 1 までの数値) で指定 左から順に、番号、原子種、x 座標、y 座標、z 座標
}tb_junc_region	
tb_left_region{/tb_right_region{	レフト/ライト領域の構造などの設定
unit_vector	単位胞の基本ベクトルの設定 (単位 Å) 上から順に、a、b、c 左から順に、x 座標、y 座標、z 座標
coordinate_system	座標系の指定 cartesian: ”atom_list”で指定される座標を実座標値で指定 pucv: 内部座標値で指定
atom_number	セル内の原子数
atom_list	原子座標の設定 ”coordinate_system=cartesian”: 座標を実座標値で指定 (単位 Å) ”coordinate_system=pucv”: 内部座標値 (0 から 1 までの数値) で指定 左から順に、番号、原子種、x 座標、y 座標、z 座標
}tb_left_region/}tb_right_region	



図 8: ベンゼンジチオール分子架橋系の結晶構造。濃色の原子がレフト領域、ライト領域として扱われる Au 原子を表す。

10 計算例

ここでいくつかの計算例を示す。計算実行体格納ディレクトリー”bin”に第 8 章に記した計算実行体”ascot_v410”が、データファイル格納ディレクトリー”data”に擬原子軌道などのデータファイルがあるものとする。また、サンプルデータ格納ディレクトリー”samples”にあるディレクトリーをそのまま作業用ディレクトリー”work”にコピーして使用するものとする。

10.1 ベンゼンジチオール分子架橋系

まず、図 8 に示す 1 次元 Au 細線の電極間にベンゼンジチオール分子 ($C_6H_4S_2$) が架橋された系の伝導特性を求める。これはディレクトリー”samples/BDT_1d”にある例題である。このディレクトリーには 3 つのディレクトリー(”trans”、”junction”、”electrode”)がある。ディレクトリー”junction”では図 5 の Procedure 1-2 の計算を、ディレクトリー”electrode”では図 5 の Procedure 1-1 の計算を行う。これらの計算の後に、ディレクトリー”trans”において図 5 の Procedure 2 および 3 の計算を行う。入力ファイル内のデータの説明は第 9 章にあるので、ここでは作成時の注意点と結果について述べることにする。

10.1.1 Procedure 1-2

最初にディレクトリー”junction”における、散乱領域の周期境界条件下での電子状態計算 (通常のバンド計算) について述べる。Procedure 2 の計算の初期値を求めるために実行される。

入力ファイル

計算条件は”condition_jun.in”で設定される。

周期系に対する電子状態計算を行うので

```
general{
    dft__negf = dft
    ...
}general
```

と設定する。

”accuracy{...}accuracy”内の element_data_path_pao および element_data_path_vps でディレクトリー”data”の相対位置などを設定する。このディレクトリー”junction”の階層を変えて実行する場合は書き換える必要がある。

この計算で用いられる擬原子軌道の数などを表 11 に示す。これは次のような意味を持つ。 r_c はカットオフ半径を表し、孤立 Au 原子の場合電子の波動関数は 6.0 a.u. よりも外にしみ出していない。 n_s は s 軌道に対する軌道

表 11: 計算に用いられた擬原子軌道のカットオフ半径 r_c と s 、 p および d 軌道の軌道の数 (n_s 、 n_p および n_d)。

	r_c [a.u.]	n_s	n_p	n_d
Au	6.0	2	1	1
S	5.5	2	2	2
C	4.0	2	2	—
H	4.0	2	2	—

として節を持たないものと節を一つもつものの2つを用いることを意味する (図 6 参照)。 n_p と n_d も同様である。これを入力ファイルの "accuracy{element_list{...}element_list}accuracy" 内の "PAO_mode" に "s(n_s)p(n_p)d(n_d)" と指定する。例えば

```
accuracy{
  ...
  num_element      = 4
  element_list{
    Au "au60_ldapw91.pao2"      "au_ldapw91.gncpp2"      s2p1d1
    S  "s55_ldapw91.pao2"      "s_ldapw91.gncpp2"      s2p2d2
    C  "c40_ldapw91.pao2"      "c_ldapw91.gncpp2"      s2p2
    H  "h40_ldapw91.pao2"      "h_ldapw91.gncpp2"      s2p2
  }element_list
}accuracy
```

である。これらの値 (n_s 、 n_p 、 n_d) を大きくとることによって精度よい計算を行うことができる。ただし、表 1 に示した m_x の値を超えて指定することはできない。

"structure{atom_list{...}atom_list}structure" 内に散乱領域の原子構造を設定する。この例では散乱領域に 40 原子あり、その座標などを

```
structure{
  ...
  unit_cell{
    unit      = ang    # {au|ang}
    vector{
      0.00000000000000  5.86991000000000  10.16699000000000
      0.00000000000000  11.73981000000000  0.00000000000000
      26.91954293108431  0.00000000000000  0.00000000000000
    }vector
  }unit_cell
  atom_list{
    num_atom      = 40
    num_left_atom = 11
    num_right_atom = 11
    unit          = ang    # {au|ang}
    atoms{
      1 Au  0.00000000000000  1.46748000000000  2.54175000000000  5.50000  5.50000
      2 Au  0.00000000000000  2.93496000000000  5.08349000000000  5.50000  5.50000
      3 Au  0.00000000000000  4.40243000000000  2.54175000000000  5.50000  5.50000
```



```

...
  9 Au 24.52316293108431 1.46748000000000 4.23624000000000 5.50000 5.50000
 10 Au 24.52316293108431 2.93496000000000 1.69450000000000 5.50000 5.50000
 11 Au 24.52316293108431 4.40243000000000 4.23624000000000 5.50000 5.50000
}atoms
}atom_list
}structure

```

と指定する。伝導方向は c 軸方向に設定される。これらの原子は c 軸方向に順番に並んでいる (この例では x 座標が小さい順番に並んでいる) 必要がある。このうち左右の数個の原子は電極の構造を含む。この例では電極の単位胞の数 (11 個) の原子がそれにあたる。また、番号に意味は無い。

最後に、後の伝導計算 (Procedure 2 & 3) のために

```

file_out{
...
  matrices      = on          # {on|off}, default=off
  f_negf        = on          # {on|off}, default=off
}file_out

```

としておく。

計算の実行

この計算は、"% 計算実行体 入力ファイル名" と指定することで実行される。例えば

```
% mpirun -np 1 ../../bin/ascot_v410 condition_jun.in
```

とする。従来版 [ver. 3] ではこの部分の計算は並列化されていなかったが、ver. 4 では並列計算が可能となった。ただし、原子数 (この例では 40) を超えて並列化しても意味はないので注意のこと。

計算の入力条件・途中経過・出力ファイルの一覧は "g.output_0001" に出力される。計算の収束状況を見るためには、例えば

```
% grep "#d" g.output_0001
```

と入力した結果

```

...
#d####      33      0.000004219407      0.00000046859 #####d#
#d####      34      0.000003295381      0.00000024383 #####d#
#d####      35      0.000001760556      0.00000035380 #####d#
#d####      36      0.000001072984      0.00000001309 #####d#
#d####      37      0.000000657707      0.00000002224 #####d#

```

を見ればよい。最初の整数は反復回数、その後の数値が密度行列およびトータルエネルギーの残差値を表す。密度行列の残差値が、入力ファイルに指定した収束判定の基準値を満たすと反復計算のループの外に出る。

出力ファイル

出力ファイルの一覧は "g.output_0001" に書かれている。入力ファイルに指定したものは "g.output_0001" の最初に、それ以外のは最後に書かれている。

作成された "junction.in.data"、"junction.matrices.dat.*"、"junction.vd.data" がこの後の伝導特性の計算に必要なデータファイルである。"junction.in.data" には原子座標など伝導計算に引き渡す入力データが格納さ

れている。そして、”junction.matrices.dat_***”（”***”は原子の番号）には、重なり積分、ハミルトニアン¹の運動エネルギーからの寄与、そして、計算されたハミルトニアンおよび密度行列が、”junction.vd.data”には散乱領域の計算されたポテンシャルおよび電荷密度が格納されている。

10.1.2 Procedure 1-1

次にディレクトリー”electrode”において、レフト・ライト領域の電子状態の計算を行う。この例では左右の電極が同じなので、以下の作業で同時に求めることができる。

入力ファイル

計算条件は”condition_ele.in”で設定される。伝導計算のために Procedure 1-2 の入力条件と整合性をとらなければならない箇所がある。

”structure{atom_list{...}atom_list}structure”内にレフト・ライト領域の原子構造を3倍の周期分を、例のように設定する。この例では原子数が33原子あり、その座標などを

```
structure{
    ...
    atom_list{
        num_atom      = 33
        num_left_atom = 11
        num_right_atom = 11
        unit           = ang      # {au|ang}
        atoms{
            1 Au  0.00000  1.46748  2.54175  5.50000  5.50000
            2 Au  0.00000  2.93496  5.08349  5.50000  5.50000
            3 Au  0.00000  4.40243  2.54175  5.50000  5.50000
            ...
            9 Au 19.17104  1.46748  4.23624  5.50000  5.50000
            10 Au 19.17104  2.93496  1.69450  5.50000  5.50000
            11 Au 19.17104  4.40243  4.23624  5.50000  5.50000
        }atoms
    }atom_list
}structure
```

と指定する。この際、左側の原子11個はディレクトリー”junction”における計算で用いられた原子位置と一致している必要がある。同様に右側の原子11個も単位胞の端からの相対位置がディレクトリー”junction”における計算で用いられたものと一致していなければならない。

それから、エネルギーカットオフの値が一致していること、つまり

```
accuracy{
    energy_cutoff      = 100.0 Ry
    ...
}accuracy
```

という指定が一致していることが望ましい。この値によってポテンシャルなどを与える実空間上のメッシュ点が決定され、この点がそろっていることが望ましいためである。(メッシュ点の数が異なっている場合は、Procedure 2の計算の際、補間したハートリーポテンシャルのデータを用いてポアソン方程式を解くことになる。)

計算の実行および出力ファイル

例えば、

```
% mpirun -np 1 ../../bin/ascot_v410 condition_ele.in
```

によって計算が実行される。計算の過程に関することは Procedure 1-2 と同じである。

作成された”electrode.matrices.dat_***”、”electrode.vd.data”がこの後の計算に必要となるデータファイルである。”electrode.matrices.dat_***”（”***”は原子の番号）に、電極の重なり積分およびハミルトニアンなどが、”electrode.vd.data”には散乱領域のハートリーポテンシャルを求める際の境界条件が格納されている。また、電極のフェルミエネルギーは”energy.dat”に

```
Fermi energy
Ef =      -0.1749835994674862 [au];      -4.7615478897698109 [eV]
      ...
```

のように与えられている。

10.1.3 Procedure 2 & 3

ディレクトリー”trans”おける、半無限電極につながった散乱領域の電子状態と対象としている系の伝導特性の求め方を述べる。

入力ファイル

計算の基本的な条件は”condition.in”で設定される。

開放系における電子状態の計算と伝導特性を求めるので

```
general{
    dft__negf    = negf
    ...
}general
```

と設定する。

散乱領域の密度行列の初期値やレフト・ライト領域のハミルトニアンなどが収められているファイルの指定は、”condition.in”内の

```
file{
    scf_file      = condition_scf.in
    ...
}file
```

で指定されるファイル (condition_scf.in) 内で、

```
negf_scf_file{
    input_hs_junc = "../junction/junction.matrices.dat"
    ...
    input_hs_left  = "../electrode/electrode.matrices.dat"
    ...
    input_hs_right = "../electrode/electrode.matrices.dat"
    ...
}negf_scf_file
```

と行う。Procedure 1 で作成されたファイルには原子の番号”_**”が付加されているが、それを書く必要はない。

またこのディレクトリ”trans”はディレクトリ”junction”と並列におかれていなければならない。これは擬原子軌道などのデータファイルを Procedure 1-2 で指定された通りのものを用いるためである。もし階層を変えるのであれば、

```
negf_scf_file{
  ...
  input_junc_cond = "../junction/junction.in.data"
  ...
}negf_scf_file
```

で指定される”../junction/junction.in.data”内の擬原子軌道などのデータファイルを指定している箇所を書き変える必要がある。

計算の実行

この計算は、これまでと同様に”% 計算実行体 入力ファイル名”と指定することで実行される。例えば

```
% mpirun -np 1 ../../bin/ascot_v410 condition.in
```

とする。

計算の入力条件・途中経過・出力ファイルの一覧は”g.output_0001”に出力される。計算の収束状況を見るためには、例えば

```
% grep "#d" g.output_0001
```

と入力した結果

```
...
#dm###      27      0.0001804107235049      0.612713   ###dm#
#dm###      28      0.0001689786921160      0.612670   ###dm#
#dm###      29      0.0001303360474633      0.674306   ###dm#
#dm###      30      0.0001115951872376      0.674338   ###dm#
#dm###      31      0.0000647522892521      0.637914   ###dm#
```

を見ればよい。これらの数値は反復回数、密度行列の収束残差値、最大残差値を与える成分の値を表す。密度行列の残差値が、入力ファイルに指定した収束判定の基準値を満たすと反復計算のループの外に出る。

出力ファイル

出力ファイルの一覧は”g.output_0001”に書かれている。入力ファイルに指定したものは”g.output_0001”の最初に、それ以外のものは最後に書かれている。

トランスミッションの結果は”trans.dat”に

```
+ total
+++++
-0.996093750000000D+01    0.747525613761253D+00
-0.988281250000000D+01    0.847597747561413D+00
-0.980468750000000D+01    0.353095545122651D+00
-0.972656250000000D+01    0.258960514318159D-01
-0.964843750000000D+01    0.145848780283926D+00
...

```

というように出力されている。第一列は電子の入射エネルギー (単位: eV)、第二列にはそのエネルギーに対するトランスミッションの値である。この結果は図 9 の通りである。

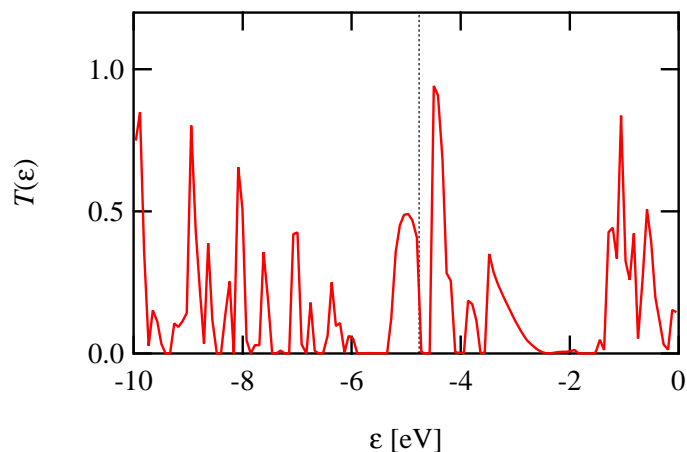


図 9: ベンゼンジチオール分子架橋系のトランスミッション。点線は電極のフェルミエネルギーを表す。

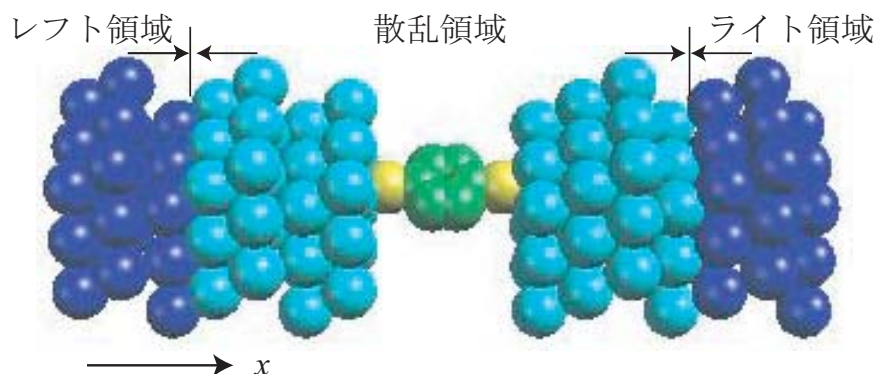


図 10: Au(111) 表面に挟まれたベンゼンジチオール分子架橋系の結晶構造。濃色の原子がレフト領域、ライト領域として扱われる Au 原子を表す。

10.2 Au(111) 表面に挟まれたベンゼンジチオール分子

ここでは、図 10 に示す Au(111) 表面間にベンゼンジチオール分子 ($C_6H_4S_2$) が架橋された系の伝導特性を求める。ここでは、複数波数 k を用いた、ディレクトリ―”samples/BDT_k”にある例題について述べる。このディレクトリ―には 3 つのディレクトリ― (“trans”、“junction”、“electrode”) がある。前節の例題と同様、ディレクトリ―”junction”では図 5 の Procedure 1-2 の計算を、ディレクトリ―”electrode”では図 5 の Procedure 1-1 の計算を行う。これらの計算の後に、ディレクトリ―”trans”において図 5 の Procedure 2 および 3 の計算を行う。

10.2.1 Procedure 1-2

最初にディレクトリ―”junction”における、散乱領域の周期境界条件下での電子状態計算 (通常のバンド計算) について述べる。

入力ファイル

計算の条件は”condition_jun.in”で設定される。

前節と異なる点は、波数 k のメッシュ数を与える

```

accuracy{
  ...
  ksampling{
    method = mesh # {gamma|mesh}, default=gamma
    mesh{
      3 3 1      # na nb nc
    }mesh
  }ksampling
  ...
}accuracy

```

と設定されていることである。伝導方向は c 軸方向に設定されているので、その方向は”1”としている。

計算の実行

この計算は、前節同様、”% 計算実行体 入力ファイル名”と指定することで実行される。例えば

```
% mpirun -np 1 ../../bin/ascot_v410 condition_jun.in
```

とする。

計算の収束状況を見るためには、例えば

```
% grep "#d" g.output_0001
```

と入力した結果

```

...
#d####      34      0.000010538882      0.00000696180 ####d#
#d####      35      0.000002668768      0.00000142573 ####d#
#d####      36      0.000002184714      0.00000053698 ####d#
#d####      37      0.000001806330      0.00000081582 ####d#
#d####      38      0.000000409442      0.00000056937 ####d#

```

を見ればよい。最初の整数は反復回数、その後の数値が密度行列およびトータルエネルギーの残差値を表す。密度行列の残差値が、入力ファイルに指定した収束判定の基準値を満たすと反復計算のループの外に出る。

出力ファイル

出力ファイルは前節に示した Γ 点計算のものと同じである。入力ファイルに指定したものは”g.output_0001”の最初に、それ以外のは最後に書かれている。

作成された”junction.in.data”、”junction.matrices.dat_***”、”junction.vd.data”がこの後の伝導特性の計算に必要なデータファイルである。”junction.in.data”には原子座標など伝導計算に引き渡す入力データが格納されている。そして、”junction.matrices.dat_***”（”***”は原子の番号）には、重なり積分、ハミルトニアン¹の運動エネルギーからの寄与、そして、計算されたハミルトニアンおよび密度行列が、”junction.vd.data”には散乱領域の計算されたポテンシャルおよび電荷密度が格納されている。

10.2.2 Procedure 1-1

次にディレクトリー”electrode”において、レフト・ライト領域の電子状態の計算について述べる。

入力ファイル

計算の条件は”condition_ele.in”で設定される。

Procedure 1-2 と同様、波数 k のメッシュ数は

```
accuracy{
  ...
  ksampling{
    method = mesh # {gamma|mesh}, default=gamma
    mesh{
      3 3 1      # na nb nc
    }mesh
  }ksampling
  ...
}accuracy
```

と設定する。伝導方向は c 軸方向に設定されているので、その方向は”1”としている。この波数 k のメッシュ数が次の伝導計算 Procedure 2 における k 点のデフォルトになる。

計算の実行および出力ファイル

例えば、

```
% mpirun -np 1 ../../bin/ascot_v410 condition_ele.in
```

によって計算が実行される。作成された”electrode.matrices.dat_***”、”electrode.vd.data”がこの後の計算に必要なデータファイルである。”electrode.matrices.dat_***”（”***”は原子の番号）には、電極の重なり積分およびハミルトニアンなどが、”electrode.vd.data”には散乱領域のハートリーポテンシャルを求める際の境界条件が格納されている。電極のフェルミエネルギーは”energy.dat”に

```
Fermi energy
Ef =      -0.1430799945057132 [au];          -3.8934062848189632 [eV]
  ...
```

のように与えられている。

10.2.3 Procedure 2 & 3

ディレクトリー”trans”における、半無限電極につながった散乱領域の電子状態と対象としている系の伝導特性の求め方を述べる。この際、電子状態計算と伝導特性（トランスミッションや I - V 曲線）を求める計算とで、波数 k のメッシュ数を指定する箇所が異なるので注意していただきたい。

入力ファイル

計算の基本的な条件は”condition.in”で設定される。

開放系における電子状態の計算で用いる波数 k のメッシュ数は、”condition.in”内の

```
file{
  scf_file          = condition_scf.in
  ...
}file
```

で指定されるファイル (condition_scf.in) 内で、

```

negf_scf_accuracy{
    ...
    ksampling = 3 3
}negf_scf_accuracy

```

のように設定する。この値が設定されていない場合は、電極の電子状態を計算した際 (Procedure 1-1) に用いたメッシュ数が用いられる。サンプルファイルでは

```

negf_scf_accuracy{
    ...
    # ksampling = 3 3
}negf_scf_accuracy

```

のようにコメントされているので、Procedure 1-1 で用いた 3×3 のメッシュ点が用いられる。

伝導特性 (トランスミッションや I - V 曲線) を求める計算で用いる波数 k のメッシュ数は、"condition.in" において

```

gra_ksampling{
    point      = 3 3
}gra_ksampling

```

と設定する。この値が設定されていない場合は、 Γ 点のみの計算が実行される。

計算の実行

この計算は、これまでと同様に"% 計算実行体 入力ファイル名"と指定することで実行される。例えば

```
% mpirun -np 1 ../../bin/ascot_v410 condition.in
```

とする。

計算の入力条件・途中経過・出力ファイルの一覧は"output_0001"に出力される。計算の収束状況を見るためには、例えば

```
% grep "#d" g.output_0001
```

と入力した結果

```

...
#dm###      23      0.0002312543720825      0.643358   ###dm#
#dm###      24      0.0002140328657161      0.643385   ###dm#
#dm###      25      0.0001111535784278      0.643385   ###dm#
#dm###      26      0.0001040903881458      0.643378   ###dm#
#dm###      27      0.0000341970029053      -0.025856   ###dm#

```

を見ればよい。密度行列の残差値が入力ファイルに指定した収束判定の基準値を満たすと反復計算のループの外に出る。

出力ファイル

出力ファイルの一覧は"output_0001"に書かれている。入力ファイルに指定したものは"output_0001"の最初に、それ以外のものは最後に書かれている。

トランスミッションの結果は"trans.dat"に

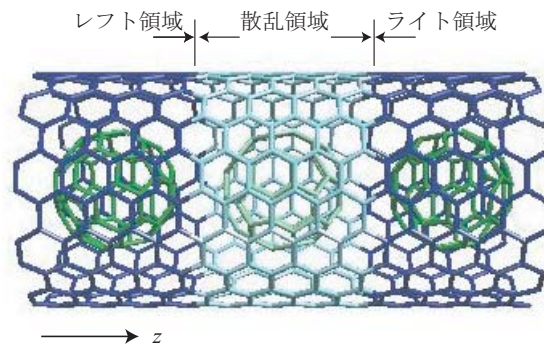


図 12: C_{60} 内包カーボンナノチューブの結晶構造。濃色の原子がレフト領域、ライト領域として扱われる C 原子を表す。

10.3 C_{60} 内包カーボンナノチューブ

図 12 に示す C_{60} 内包カーボンナノチューブの伝導特性を求める。これはディレクトリー”samples/C_NT”にある例題である。文献 18,19 で提案されたタイトバインディングパラメータを用いて計算する。タイトバインディングパラメータを用いた計算では Procedure 1 および Procedure 2 は省略される。

この系に対する結果の詳細は文献 1 にある。

10.3.1 Procedure 3

入力ファイル

基本的な条件は”condition.in”で設定される。

文献 18,19 で提案されたタイトバインディングパラメータを用いて伝導特性を求めるので

```
general{
    dft__negf    = negf
    model_type  = rtb_h
    ...
}general
```

と設定する。この計算では必ずパラメータのデータファイル”rtbh_c_parameter.in”を用いる。このデータファイルは

```
file{
    ...
    tb_parameter    = "../../data/rtbh_c_parameter.in"
    ...
}file
```

と指定される。ファイルの指定が行わなければデフォルトとして”../../data/rtbh_c_parameter.in”が設定されるので、この例の通りのディレクトリーの階層で計算を実行する時は省略が可能である。

結晶構造は

```
file{
    tb_cordinate    = cordinate.in
    ...
}file
```

で指定されるファイル (coordinate.in) 内で、図 12 に示す領域に分けて設定される。例えば

```

tb_junc_region{
    ...
    atom_list{
        ...
        158  C    3.397610    5.883531    7.383733
        159  C    2.099490    6.461566    7.383733
        160  C    1.413208    6.645490    8.614355
        161  C1   1.174050    0.381472    7.628316
        162  C1   0.725603   -0.998707    7.628316
        163  C1  -0.725603   -0.998707    7.628316
        ...
    }atom_list
}tb_junc_region
...
tb_left_region{
    ...
}tb_left_region
...
tb_right_region{
    ...
}tb_right_region

```

である。原子の記号として”C”と表記されたものはナノチューブの C 原子を、”C1”と表記されたものは内包された C₆₀ の C 原子を表す。伝導方向は c 軸方向に設定されている。

計算の実行

この計算は、これまでと同様に”% 計算実行体 入力ファイル名”と指定することで実行される。例えば

```
% mpirun -np 1 ../../bin/ascot_v410 condition.in
```

とする。計算の入力条件・途中経過・出力ファイルの一覧は”g.output_0001”に出力される。

出力ファイル

出力ファイルの一覧は”g.output_0001”に書かれている。入力ファイルに指定したものは”g.output_0001”の最初に、それ以外のものは最後に書かれている。

ライト領域のバンド構造の結果は”ev_r.dat”に

```

* cp_r= 0.427803882215966
*****
+ n_k= 1
+++++
0 -20.2256546776677
0 -19.7252029661440
0 -19.7252026564593
0 -19.7130313711377
0 -19.4511039643909

```

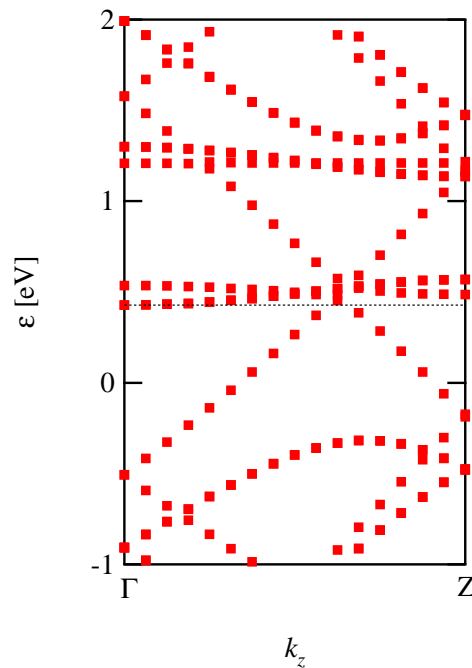


図 13: C₆₀ 内包カーボンナノチューブのバンド図。点線は電極のフェルミエネルギーを表す。

```
0 -19.3951534174472
0 -19.3951534171021
```

...

というように出力されている。最初の”cp_r”はライト領域のフェルミエネルギー (単位: eV) が出力されている。そして第一列は k_z を表す番号、第二列にはそれに対するエネルギーレベル (単位: eV) である。その結果は図 13 の通りである。レフト領域のバンド構造は”ev_l.dat”に同様に格納される。

トランスミッションの結果は”trans.dat”に

```
+ total
+++++
0.5000000000000000D-02 0.199997818598244D+01
0.1500000000000000D-01 0.199997819758579D+01
0.2500000000000000D-01 0.199997820836010D+01
0.3500000000000000D-01 0.199997821829501D+01
0.4500000000000000D-01 0.199997822737115D+01
0.5500000000000000D-01 0.199997823556062D+01
```

...

というように出力されている。第一列は電子の入射エネルギー (単位: eV)、第二列にはそのエネルギーに対するトランスミッションの値である。その結果は図 14 の通りである。

”coordinate.in”に指定した全炭素原子に対する状態密度は”em_gf.dat”に

```
+ total
+++++
0.5000000000000000D-02 0.131510676799347D-02
0.1500000000000000D-01 0.131440719788078D-02
```

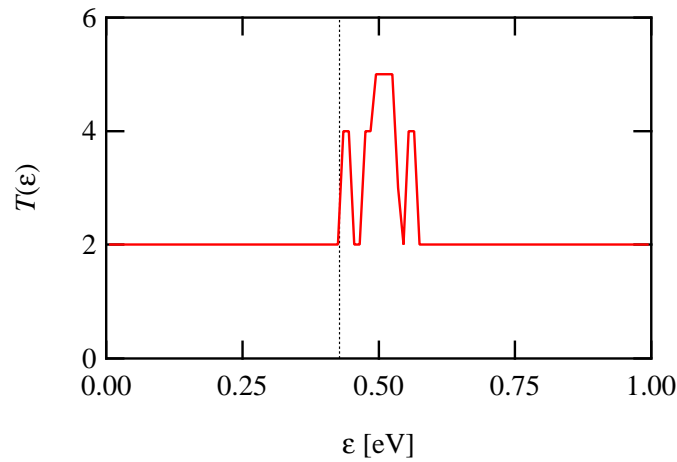


図 14: C_{60} 内包カーボンナノチューブのトランスミッション。点線は電極のフェルミエネルギーを表す。

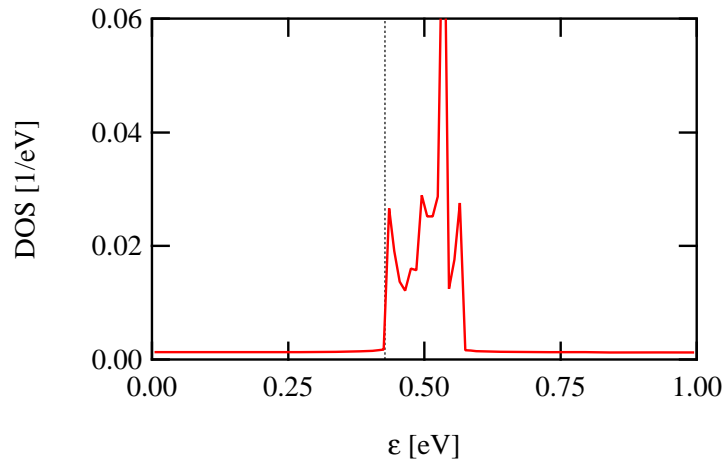


図 15: C_{60} 内包カーボンナノチューブの状態密度。点線は電極のフェルミエネルギーを表す。

0.2500000000000000D-01	0.131375763484579D-02
0.3500000000000000D-01	0.131315870490883D-02
0.4500000000000000D-01	0.131261158030394D-02
0.5500000000000000D-01	0.131211793801151D-02

...

というように出力されている。第一列は電子の入射エネルギー (単位: eV)、第二列にはそのエネルギーに対する状態密度 (単位: 1/eV) の値である。その結果は図 15 の通りである。

参考文献

- [1] H. Kondo, H. Kino and T. Ohno: Phys. Rev. B **71** (2005) 115413.
- [2] H. Kondo, J. Nara, H. Kino and T. Ohno: Jpn. J. Appl. Phys. **47** (2008) 4792.
- [3] H. Kondo, H. Kino, J. Nara, T. Ozaki and T. Ohno: Phys. Rev. B **73** (2006) 235323.

- [4] H. Kondo, H. Kino and T. Ohno: *Thin Solid Films* **464-465** (2004) 342.
- [5] W.T. Geng, H. Kondo, J. Nara, and T. Ohno: *Phys. Rev. B* **72** (2005) 125421.
- [6] H. Kondo, J. Nara, H. Kino and T. Ohno: *J. Chem. Phys.* **128** (2008) 064701.
- [7] W.T. Geng, M. Oda, J. Nara, H. Kondo, and T. Ohno: *J. Phys. Chem. B* **112**, 2795 (2008).
- [8] H. Kondo, H. Kino, J. Nara and T. Ohno: *Appl. Surf. Sci.* **254** (2008) 7985.
- [9] H. Kondo, J. Nara, H. Kino and T. Ohno: *J. Phys.: Condens. Matter* **21**, 064220 (2009).
- [10] H. Kondo, J. Nara and T. Ohno: *Phys. Rev. B* **81** (2010) 085318.
- [11] S. Datta, *Electronic Transport in Mesoscopic Systems*, Cambridge Univ. Press, 1995.
- [12] M. P. López Sancho, J. M. López Sancho, and J. Rubio: *J. Phys. F* **14** (1984) 1205.
- [13] M. P. López Sancho, J. M. López Sancho, and J. Rubio: *J. Phys. F* **15** (1985) 951.
- [14] T. Ozaki: *Phys. Rev. B* **67** (2003) 155108.
- [15] The pseudo atomic orbitals are generated using the CIAO code, which is developed within the RSS21 project supported by MEXT of the Japanese government. See the website <http://www.rss21.iis.u-tokyo.ac.jp/en/index.html>.
- [16] N. Troullier and J. L. Martins: *Phys. Rev. B* **43** (1991) 1993.
- [17] L. Kleinman and D. M. Bylander: *Phys. Rev. Lett.* **48** (1982) 1425.
- [18] S. Okada and S. Saito: *J. Phys. Soc. Jpn.* **64** (1995) 2100.
- [19] N. Hamada, M. Yamaji, S. Okada, S. Saito: *Proc. of International Symposium on Nanonetwork Materials: Fullerenes, Nanotubes, and Related Systems*, eds., S. Saito, T. Ando, Y. Iwasa, K. Kikuchi, M. Kobayashi, and Y. Saito, pp. 201, January 2001, Kamakura (American Institute of Physics, New York, 2001).
- [20] C. H. Xu, C. Z. Wang, C. T. Chan, and K. M. Ho: *J. Phys.: Condens. Matter.* **4** (1989) 6047.
- [21] J. P. Perdew and Y. Wang: *Phys. Rev. B* **45** (1992) 13244.
- [22] J. P. Perdew and A. Zunger: *Phys. Rev. B* **23** (1981) 5048.
- [23] G. Kresse and J. Furthmüller: *Phys. Rev. B* **54** (1996) 11169.
- [24] D. G. Anderson, *J. Assoc. Computing Machinery*, **12** (1965) 547.
- [25] K. Hirose and M. Tsukada: *Phys. Rev. B* **51** (1995) 5278.