The Coupled-Cluster Method

Thomas Papenbrock

The University of Tennessee, Knoxville, tpapenbr@utk.edu

June 25, 2018

© 2018, Thomas Papenbrock. Released under CC Attribution-NonCommercial 4.0 license

Contents

© 2018, Thomas Papenbrock. Released under CC Attribution-NonCommercial 4.0 license

Chapter 1

The Coupled-Cluster Method

Introduction

The coupled-cluster method is an efficient tool to compute atomic nuclei with an effort that grows polynomial with system size. While this might still be expensive, it is now possible to compute nuclei with mass numbers about $A \approx 100$ with this method. Recall that full configuration interaction (FCI) such as the no-core shell model exhibits an exponential cost and is therefore limited to light nuclei.

© 2018, Thomas Papenbrock. Released under CC Attribution-NonCommercial 4.0 license

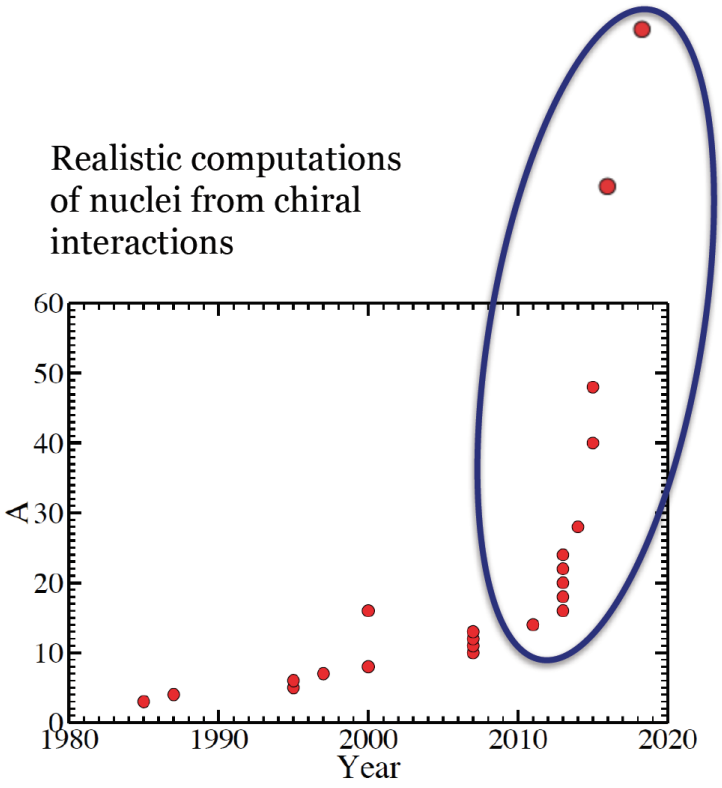


Figure 1.1: Realistic computations of atomic nuclei with interactions from chiral EFT. The slow increase prior to 2015 is based on quantum Monte Carlo and the no-core shell model. These methods are exponentially expensive (in mass number A) and meet with exponentially increasing computer power (Moore's law), thus leading to a progress that is linear in time. Methods such as coupled clusters and in-medium SRG carry a polynomial cost in mass number are transforming the field.

The normal-ordered Hamiltonian

We start from the reference state

$$|\Phi_0\rangle = \prod_{i=1}^{A} a_i^\dagger |0\rangle \tag{1.1}$$

for the description of a nucleus with mass number A. Usually, this reference is the Hartree-Fock state, but that is not necessary. In the shell-model picture, it could also be a product state where the lowest A harmonic oscillator states are occupied. Here and in what follows, the indices $i, j, k, \ldots$ run over hole states, i.e. orbitals occupied in the reference state (1.1), while $a, b, c, \ldots$ run over particle

states, i.e. unoccupied orbitals. Indices p, q, r, s can identify any orbital. Let n_u be the number of unoccupied states, and A is of course the number of occupied states. We consider the Hamiltonian

$$H = \sum_{pq} \varepsilon_q^p a_p^\dagger a_q + \frac{1}{4} \sum_{pqrs} \langle pq|V|rs\rangle a_p^\dagger a_q^\dagger a_s a_r \tag{1.2}$$

The reference state (1.1) is a non-trivial vacuum of our theory. We normal order this Hamiltonian with respect to the nontrivial vacuum state given by the Hartree-Fock reference and obtain the normal-ordered Hamiltonian

$$H_N = \sum_{pq} f_{pq} \left\{ a_p^\dagger a_q \right\} + \frac{1}{4} \sum_{pqrs} \langle pq|V|rs\rangle \left\{ a_p^\dagger a_q^\dagger a_s a_r \right\}. \tag{1.3}$$

Here,

$$f_q^p = \varepsilon_q^p + \sum_i \langle pi|V|qi\rangle \tag{1.4}$$

is the Fock matrix. We note that the Fock matrix is diagonal in the Hartree-Fock basis. The brackets $\{\cdots\}$ in Eq. (1.3) denote normal ordering, i.e. all operators that annihilate the nontrivial vaccum (1.1) are to the right of those operators that create with respect to that vaccum. Normal ordering implies that $\langle \Phi_0 | H_N | \Phi_0 \rangle = 0$.

*

Exercise 1: Practice in normal ordering
Normal order the expression $\sum_{pq} \varepsilon_q^p a_p^\dagger a_q$.

Hint.

$$\sum_{pq} \varepsilon_q^p a_p^\dagger a_q = \sum_{ab} \varepsilon_b^a a_a^\dagger a_b + \sum_{ai} \varepsilon_i^a a_a^\dagger a_i + \sum_{ai} \varepsilon_a^i a_i^\dagger a_a + \sum_{ij} \varepsilon_j^i a_i^\dagger a_j \tag{1.5}$$

Answer. We have to move all operators that annihilate the reference state to the right of those that create on the reference state. Thus,

$$\sum_{pq} \varepsilon_q^p a_p^\dagger a_q = \sum_{ab} \varepsilon_b^a a_a^\dagger a_b + \sum_{ai} \varepsilon_i^a a_a^\dagger a_i + \sum_{ai} \varepsilon_a^i a_i^\dagger a_a + \sum_{ij} \varepsilon_j^i a_i^\dagger a_j \tag{1.6}$$

$$= \sum_{ab} \varepsilon_b^a a_a^\dagger a_b + \sum_{ai} \varepsilon_i^a a_a^\dagger a_i + \sum_{ai} \varepsilon_a^i a_i^\dagger a_a + \sum_{ij} \varepsilon_j^i \left(-a_j a_i^\dagger + \delta_i^j \right) \tag{1.7}$$

$$= \sum_{ab} \varepsilon_b^a a_a^\dagger a_b + \sum_{ai} \varepsilon_i^a a_a^\dagger a_i + \sum_{ai} \varepsilon_a^i a_i^\dagger a_a - \sum_{ij} \varepsilon_j^i a_j a_i^\dagger + \sum_i \varepsilon_i^i \tag{1.8}$$

$$= \sum_{pq} \varepsilon_q^p \left\{ a_p^\dagger a_q \right\} + \sum_i \varepsilon_i^i \tag{1.9}$$

===== =====

We note that $H = E_{HF} + H_N$, where

$$E_{HF} \equiv \langle \Phi_0 | H | \Phi_0 \rangle = \sum_i \varepsilon_i^i + \frac{1}{2} \sum_{ij} \langle ij | V | ij \rangle \tag{1.10}$$

is the Hartree-Fock energy.

The coupled-cluster method is a very efficient tool to compute nuclei when a "good" reference state is available. Let us assume that the reference state results from a Hartree-Fock calculation.

*

Exercise 2: What does "good" mean?

How do you know whether a Hartree-Fock state is a "good" reference? Which results of the Hartee-Fock computation will inform you?

Answer. Once the Hartree-Fock equations are solved, the Fock matrix (1.4) becomes diagonal, and its diagonal elements can be viewed as single-particle energies. Hopefully, there is a clear gap in the single-particle spectrum at the Fermi surface, i.e. after A orbitals are filled.

===== =====

If symmetry-restricted Hartree-Fock is used, one is limited to compute nuclei with closed subshells for neutrons and for protons. On a first view, this might seem as a severe limitation. But is it?

*

Exercise 3: How many nuclei are accessible with the coupled cluster method based on spherical mean fields?

If one limits oneself to nuclei with mass number up to mass number $A = 60$, how many nuclei can potentially be described with the coupled-cluster method? Which of these nuclei are potentially interesting? Why?

Answer. Nuclear shell closures are at $N, Z = 2, 8, 20, 28, 50, 82, 126$, and subshell closures at $N, Z = 2, 6, 8, 14, 16, 20, 28, 32, 34, 40, 50, \ldots$.

In the physics of nuclei, the evolution of nuclear structure as neutrons are added (or removed) from an isotope is a key interest. Examples are the rare isotopes of helium (^{8,10} He) oxygen (^{22,24,28} O), calcium (^{52,54,60} Ca), nickel (^{78} Ni) and tin (^{100,132} Sn). The coupled-cluster method has the potential to address questions regarding these nuclei, and in a several cases was used to make predictions before experimental data was available. In addition, the method can be used to compute neighbors of nuclei with closed subshells.

===== =====

The similarity-transformed Hamiltonian

There are several ways to view and understand the coupled-cluster method. A first simple view of coupled-cluster theory is that the method induces correlations into the reference state by expressing a correlated state as

$$|\Psi\rangle = e^T |\Phi_0\rangle, \tag{1.11}$$

Here, T is an operator that induces correlations. We can now demand that the correlate state (1.11) becomes and eigenstate of the Hamiltonian H_N, i.e. $H_N|\Psi\rangle = E|\Psi\rangle$. This view, while correct, is not the most productive one. Instead, we left-multiply the Schroedinger equation with e^{-T} and find

$$\overline{H_N}|\Phi_0\rangle = E_c|\Phi_0\rangle. \tag{1.12}$$

Here, E_c is the correlation energy, and the total energy is $E = E_c + E_{HF}$. The similarity-transformed Hamiltonian is defined as

$$\overline{H_N} \equiv e^{-T} H_N e^T. \tag{1.13}$$

A more productive view on coupled-cluster theory thus emerges: This method seeks a similarity transformation such that the uncorrelated reference state (1.1) becomes an exact eigenstate of the similarity-transformed Hamiltonian (1.13).

*

Exercise 4: What T leads to Hermitian $\overline{H_N}$?

What are the conditions on T such that $\overline{H_N}$ is Hermitian?

Answer. For a Hermitian $\overline{H_N}$, we need a unitary e^T, i.e. an anti-Hermitian T with $T = -T^\dagger$

===== =====

As we will see below, coupld-cluster theory employs a non-Hermitian Hamiltonian.

*

Exercise 5: Understanding (non-unitary) similarity transformations

Show that $\overline{H_N}$ has the same eigenvalues as H_N for arbitrary T. What is the spectral decomposition of a non-Hermitian $\overline{H_N}$?

Answer. Let $H_N|E\rangle = E|E\rangle$. Thus

$$\begin{aligned} H_N e^T e^{-T}|E\rangle &= E|E\rangle, \\ \left(e^{-T} H_N e^T\right) e^{-T}|E\rangle &= E e^{-T}|E\rangle, \\ \overline{H_N} e^{-T}|E\rangle &= E e^{-T}|E\rangle. \end{aligned}$$

Thus, if $|E\rangle$ is an eigenstate of H_N with eigenvalue E, then $e^{-T}|E\rangle$ is eigenstate of $\overline{H_N}$ with the same eigenvalue.

A non-Hermitian $\overline{H_N}$ has eigenvalues E_α corresponding to left $\langle L_\alpha|$ and right $|R_\alpha\rangle$ eigenstates. Thus

$$\overline{H_N} = \sum_\alpha |R_\alpha\rangle E_\alpha \langle L_\alpha| \tag{1.14}$$

with bi-orthonormal $\langle L_\alpha|R_\beta\rangle = \delta_\alpha^\beta$.

===== =====

To make progress, we have to specify the cluster operator T. In coupled cluster theory, this operator is

$$T \equiv \sum_{ia} t_i^a a_a^\dagger a_i + \frac{1}{4}\sum_{ijab} t_{ij}^{ab} a_a^\dagger a_b^\dagger a_j a_i + \cdots + \frac{1}{(A!)^2} \sum_{i_1 \ldots i_A a_1 \ldots a_A} t_{i_1 \ldots i_A}^{a_1 \ldots a_A} a_{a_1}^\dagger \cdots a_{a_A}^\dagger a_{i_A} \cdots a_{i_1}. \tag{1.15}$$

Thus, the operator (1.15) induces particle-hole (p-h) excitations with respect to the reference. In general, T generates up to $Ap - Ah$ excitations, and the unknown parameters are the cluster amplitides t_i^a, t_{ij}^{ab}, ..., $t_{i_1,\ldots,i_A}^{a_1,\ldots,a_A}$.

*

Exercise 6: How many unknowns?

Show that the number of unknowns is as large as the FCI dimension of the problem, using the numbers A and n_u.

Answer. We have to sum up all $np-nh$ excitations, and there are $\binom{n_u}{n}$ particle states and $\binom{A}{A-n}$ hole states for each n. Thus, we have for the total number

$$\sum_{n=0}^{A} \binom{n_u}{n}\binom{A}{A-n} = \binom{A+n_u}{A}. \tag{1.16}$$

The right hand side are obviously all ways to distribute A fermions over $n_0 + A$ orbitals.

===== =====

Thus, the coupled-cluster method with the full cluster operator (1.15) is exponentially expensive, just as FCI. To make progress, we need to make an approximation by truncating the operator. Here, we will use the CCSD (coupled clusters singles doubles) approximation, where

$$T \equiv \sum_{ia} t_i^a a_a^\dagger a_i + \frac{1}{4}\sum_{ijab} t_{ij}^{ab} a_a^\dagger a_b^\dagger a_j a_i. \tag{1.17}$$

We need to dermine the unknown cluster amplitudes that enter in CCSD. Let

$$|\Phi_i^a\rangle = a_a^\dagger a_i |\Phi_0\rangle, \tag{1.18}$$

$$|\Phi_{ij}^{ab}\rangle = a_a^\dagger a_b^\dagger a_j a_i |\Phi_0\rangle \tag{1.19}$$

be 1p-1h and 2p-2h excitations of the reference. Computing matrix elements of the Schroedinger Equation (1.12) yields

$$\langle\Phi_0|\overline{H_N}|\Phi_0\rangle = E_c, \tag{1.20}$$

$$\langle\Phi_i^a|\overline{H_N}|\Phi_0\rangle = 0, \tag{1.21}$$

$$\langle\Phi_{ij}^{ab}|\overline{H_N}|\Phi_0\rangle = 0. \tag{1.22}$$

The first equation states that the coupled-cluster correlation energy is an expectation value of the similarity-transformed Hamiltonian. The second and third equations state that the similarity-transformed Hamiltonian exhibits no 1p-1h and no 2p-2h excitations. These equations have to be solved to find the

unknown amplitudes t_i^a and t_{ij}^{ab}. Then one can use these amplitudes and compute the correlation energy from the first line of Eq. (1.20).

We note that in the CCSD approximation the reference state is not an exact eigenstates. Rather, it is decoupled from simple states but $\overline{H}$ still connects this state to 3p-3h, and 4p-4h states etc.

At this point, it is important to recall that we assumed starting from a "good" reference state. In such a case, we might reasonably expect that the inclusion of 1p-1h and 2p-2h excitations could result in an accurate approximation. Indeed, empirically one finds that CCSD accounts for about 90% of the corelation energy, i.e. of the difference between the exact energy and the Hartree-Fock energy. The inclusion of triples (3p-3h excitations) typically yields 99% of the correlation energy.

We see that the coupled-cluster method in its CCSD approximation yields a similarity-transformed Hamiltonian that is of a two-body structure with respect to a non-trivial vacuum. When viewed in this light, the coupled-cluster method "transforms" an A-body problem (in CCSD) into a two-body problem, albeit with respect to a nontrivial vacuum.

Computing the similarity-transformed Hamiltonian

The solution of the CCSD equations, i.e. the second and third line of Eq. (1.20), and the computation of the correlation energy requires us to compute matrix elements of the similarity-transformed Hamiltonian (1.13). This can be done with the Baker-Campbell-Hausdorff expansion

$$\overline{H_N} = e^{-T} H_N e^T \tag{1.23}$$

$$= H_N + [H_N, T] + \frac{1}{2!}[[H_N, T], T] + \frac{1}{3!}[[[H_N, T], T], T] + \dots . \tag{1.24}$$

We now come to a key element of coupled-cluster theory: the cluster operator (1.15) consists of sums of terms that consist of particle creation and hole annihilation operators (but no particle annihilation or hole creation operators). Thus, all terms that enter T commute with each other. This means that the commutators in the Baker-Campbell-Hausdorff expansion (1.23) can only be non-zero because each T must connect to H_N (but no T with another T). Thus, the expansion is finite.

*

Exercise 7: When does CCSD truncate?

In CCSD and for two-body Hamiltonians, how many nested commutators yield nonzero results? Where does the Baker-Campbell-Hausdorff expansion terminate? What is the (many-body) rank of the resulting $\overline{H_N}$?

Answer. CCSD truncates for two-body operators at four-fold nested commutators, because each of the four annihilation and creation operators in $\overline{H_N}$ can be knocked out with one term of T.

===== =====

We see that the (disadvantage of having a) non-Hermitian Hamiltonian $\overline{H_N}$ leads to the advantage that the Baker-Campbell-Hausdorff expansion is finite, thus leading to the possibility to compute $\overline{H_N}$ exactly. In contrast, the IMSRG deals with a Hermitian Hamiltonian throughout, and the infinite Baker-Campbell-Hausdorff expansion is truncated at a high order when terms become very small.

We write the similarity-transformed Hamiltonian as

$$\overline{H_N} = \sum_{pq} \overline{H}^p_q a^\dagger_q a_p + \frac{1}{4} \sum_{pqrs} \overline{H}^{pq}_{rs} a^\dagger_p a^\dagger_q a_s a_r + \dots \tag{1.25}$$

with

$$\overline{H}^p_q \equiv \langle p|\overline{H_N}|q\rangle, \tag{1.26}$$

$$\overline{H}^{pq}_{rs} \equiv \langle pq|\overline{H_N}|rs\rangle. \tag{1.27}$$

Thus, the CCSD Eqs. (1.20) for the amplitudes can be written as $\overline{H}^a_i = 0$ and $\overline{H}^{ab}_{ij} = 0$.

*

Exercise 8: Compute the matrix element $\overline{H}^{ij}_{ab} \equiv \langle ij|\overline{H_N}|ab\rangle$

Answer. This is a simple task. This matrix element is part of the operator $\overline{H}^{ij}_{ab} a^\dagger_i a^\dagger_j a_b a_a$, i.e. particles are annihilated and holes are created. Thus, no contraction of the Hamiltonian H with any cluster operator T (remember that T annihilates holes and creates particles) can happen, and we simply have $\overline{H}^{ij}_{ab} = \langle ij|V|ab\rangle$.

===== =====

We need to work out the similarity-transformed Hamiltonian of Eq. (1.23). To do this, we write $T = T_1 + T_2$ and $H_N = F + V$, where T_1 and F are one-body operators, and T_2 and V are two-body operators.

Example: The contribution of $[F, T_2]$ to $\overline{H_N}$

The commutator $[F, T_2]$ consists of two-body and one-body terms. Let us compute first the two-body term, as it results from a single contraction (i.e. a single application of $[a_p, a_q^\dagger] = \delta_p^q$). We denote this as $[F, T_2]_{2b}$ and find

$$
\begin{aligned}
[F,T_2]_{2b} &= \frac{1}{4}\sum_{pq}\sum_{rsuv} f_p^q t_{ij}^{ab}\left[a_q^\dagger a_p, a_a^\dagger a_b^\dagger a_j a_i\right]_{2b} \\
&= \frac{1}{4}\sum_{pq}\sum_{abij} f_p^q t_{ij}^{ab}\delta_p^a a_q^\dagger a_b^\dagger a_j a_i \\
&- \frac{1}{4}\sum_{pq}\sum_{abij} f_p^q t_{ij}^{ab}\delta_p^b a_q^\dagger a_a^\dagger a_j a_i \\
&- \frac{1}{4}\sum_{pq}\sum_{abij} f_p^q t_{ij}^{ab}\delta_q^j a_a^\dagger a_b^\dagger a_p a_i \\
&+ \frac{1}{4}\sum_{pq}\sum_{abij} f_p^q t_{ij}^{ab}\delta_q^i a_a^\dagger a_b^\dagger a_p a_j \\
&= \frac{1}{4}\sum_{qbij}\left(\sum_a f_a^q t_{ij}^{ab}\right) a_q^\dagger a_b^\dagger a_j a_i \\
&- \frac{1}{4}\sum_{qaij}\left(\sum_b f_b^q t_{ij}^{ab}\right) a_q^\dagger a_a^\dagger a_j a_i \\
&- \frac{1}{4}\sum_{pabi}\left(\sum_j f_p^j t_{ij}^{ab}\right) a_a^\dagger a_b^\dagger a_p a_i \\
&+ \frac{1}{4}\sum_{pabj}\left(\sum_i f_p^i t_{ij}^{ab}\right) a_a^\dagger a_b^\dagger a_p a_j \\
&= \frac{1}{2}\sum_{qbij}\left(\sum_a f_a^q t_{ij}^{ab}\right) a_q^\dagger a_b^\dagger a_j a_i \\
&- \frac{1}{2}\sum_{pabi}\left(\sum_j f_p^j t_{ij}^{ab}\right) a_a^\dagger a_b^\dagger a_p a_i .
\end{aligned}
$$

Here we exploited the antisymmetry $t_{ij}^{ab} = -t_{ji}^{ab} = -t_{ij}^{ba} = t_{ji}^{ba}$ in the last step. Using $a_q^\dagger a_b^\dagger a_j a_i = -a_b^\dagger a_q^\dagger a_j a_i$ and $a_a^\dagger a_b^\dagger a_p a_i = a_a^\dagger a_b^\dagger a_i a_p$, we can make the

expression manifest antisymmetric, i.e.

$$[F,T_2]_{2b} = \frac{1}{4}\sum_{qbij}\left[\sum_a \left(f_a^q t_{ij}^{ab} - f_a^b t_{ij}^{qa}\right)\right] a_q^\dagger a_b^\dagger a_j a_i$$
$$- \frac{1}{4}\sum_{pabi}\left[\sum_j \left(f_p^j t_{ij}^{ab} - f_i^j t_{pj}^{ab}\right)\right] a_a^\dagger a_b^\dagger a_p a_i.$$

Thus, the contribution of $[F,T_2]_{2b}$ to the matrix element $\overline{H}_{ij}^{ab}$ is

$$\overline{H}_{ij}^{ab} \leftarrow \sum_c \left(f_c^a t_{ij}^{cb} - f_c^b t_{ij}^{ac}\right) - \sum_k \left(f_j^k t_{ik}^{ab} - f_i^k t_{jk}^{ab}\right)$$

Here we used an arrow to indicate that this is just one contribution to this matrix element. We see that the derivation straight forward, but somewhat tedious. As no one likes to commute too much (neither in this example nor when going to and from work), and so we need a better approach. This is where diagramms come in handy.

===== =====

Diagrams. The pictures in this Subsection are taken from Crawford and Schaefer.

By convention, hole lines (labels $i,j,k,\ldots$) are pointing down.

By convention, particle lines (labels $a,b,c,\ldots$) are pointing up.

Let us look at the one-body operator of the normal-ordered Hamiltonian, i.e. Fock matrix. Its diagrams are as follows.

We now turn to the two-body interaction. It is denoted as a horizontal dashed line with incoming and outgoing lines attached to it. We start by noting that the following diagrams of the inetraction are all related by permutation symmetry.

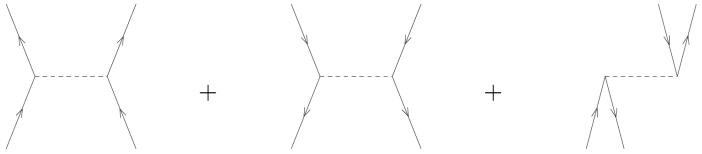
*

Exercise 9: Assign the correct matrix element $\langle pq|V|rs\rangle$ to each of the following diagrams of the interaction

Remember: $\langle \text{left}-\text{out}, \text{right}-\text{out}|\text{V}|\text{left}-\text{in}, \text{right}-\text{in}\rangle$.

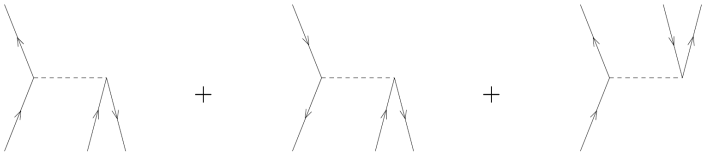
aragraph!paragraph>paragraph>-0.5em

a)



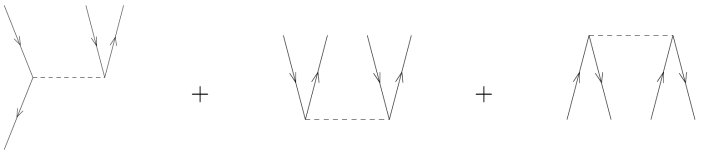
Answer. $\langle ab|V|cd\rangle + \langle ij|V|kl\rangle + \langle ia|V|bj\rangle$
aragraph!paragraph>paragraph>-0.5em

b)



Answer. $\langle ai|V!bc\rangle + \langle ij|V|ka\rangle + \langle ab|V|ci\rangle$
aragraph!paragraph>paragraph>-0.5em

c)



Answer. $\langle ia|V|jk\rangle + \langle ab|V|ij\rangle + \langle ij|V|ab\rangle$
===== =====

Finally, we have the following diagrams for the T_1 and T_2 amplitudes.

We are now in the position to construct the diagrams of the similarity-transformed Hamiltonian, keeping in mind that these diagrams correspond to matrix elements of $\overline{H_N}$. The rules are as follows.

1. Write down all *topologically different* diagrams corresponding to the desired matrix element. Topologically different diagrams differ in the number and type of lines (particle or hole) that connect the Fock matrix F or the interaction V to the cluster amplitudes T, but not whether these

connections are left or right (as those are related by antisymmetry). As an example, all diagrams in Fig. 1.8 are topologically identical, because they consist of incoming particle and hole lines and of outgoing particle and hole lines.

2. Write down the matrix elements that enter the diagram, and sum over all internal lines.

3. The overall sign is (-1) to the power of [(number of hole lines) – (number of loops)].

4. Symmetry factor: For each pair of equivalent lines (i.e. lines that connect the same two operators) multiply with a factor 1/2. For n identical vertices, multiply the algebraic expression by the symmery factor $1/n!$ to account properly for the number of ways the diagram can be constructed.

5. Antisymmetrize the outgoing and incoming lines as necessary.

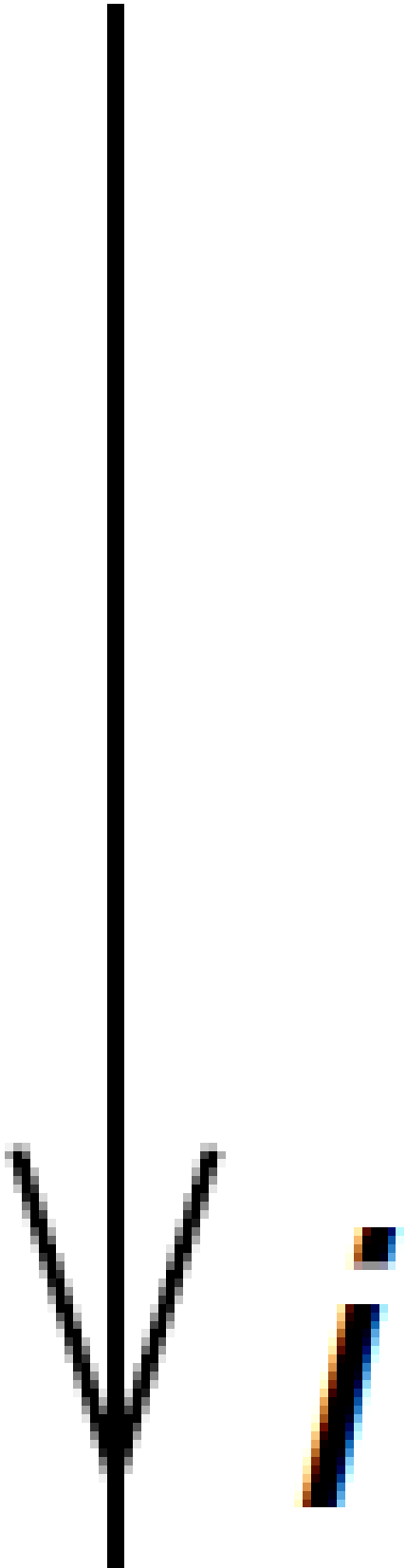
Please note that this really works. You could derive these rules for yourself from the commutations and factors that enter the Baker-Campbell-Hausdorff expansion. The sign comes obviously from the arrangement of creation and annilhilation operators, while the symmetry factor stems from all the different ways, one can contract the cluster operator with the normal-ordered Hamiltonian.

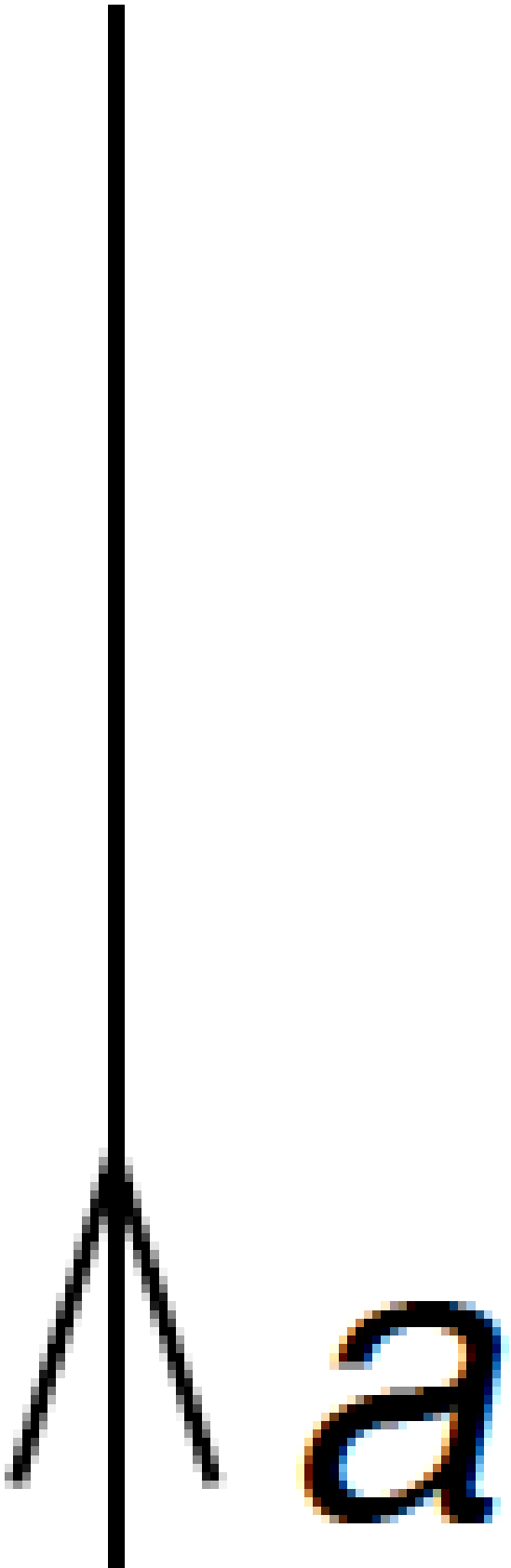
Example: CCSD correlation energy

The CCSD correlation energy, $E_c = \langle\Phi_0|\overline{H_N}|\Phi_0\rangle$, is the first of the CCSD equations (1.20). It is a vacuum expectation value and thus consists of all diagrams with no external legs. There are three such diagrams:

The correponding algebraic expression is $E_c = \sum_{ia} f_a^i t_i^a + \frac{1}{4}\sum_{ijab}\langle ij|V|ab\rangle t_{ij}^{ab} + \frac{1}{2}\sum_{ijab}\langle ij|V|ab\rangle t_i^a t_j^b$.

The first algebraic expression is clear. We have one hole line and one loop, giving it a positive sign. There are no equivalent lines or vertices, giving it no symmetry factor. The second diagram has two loops and two hole lines, again leading to a positive sign. We have a pair of equivalent hole lines and a pair of equivalent particle lines, each giving a symmetry factor of 1/2. The third diagram has two loops and two hole lines, again leading to a positive sign. We have two indentical vertices (each connecting to a T_1 in the same way) and thus a symmetry factor 1/2.





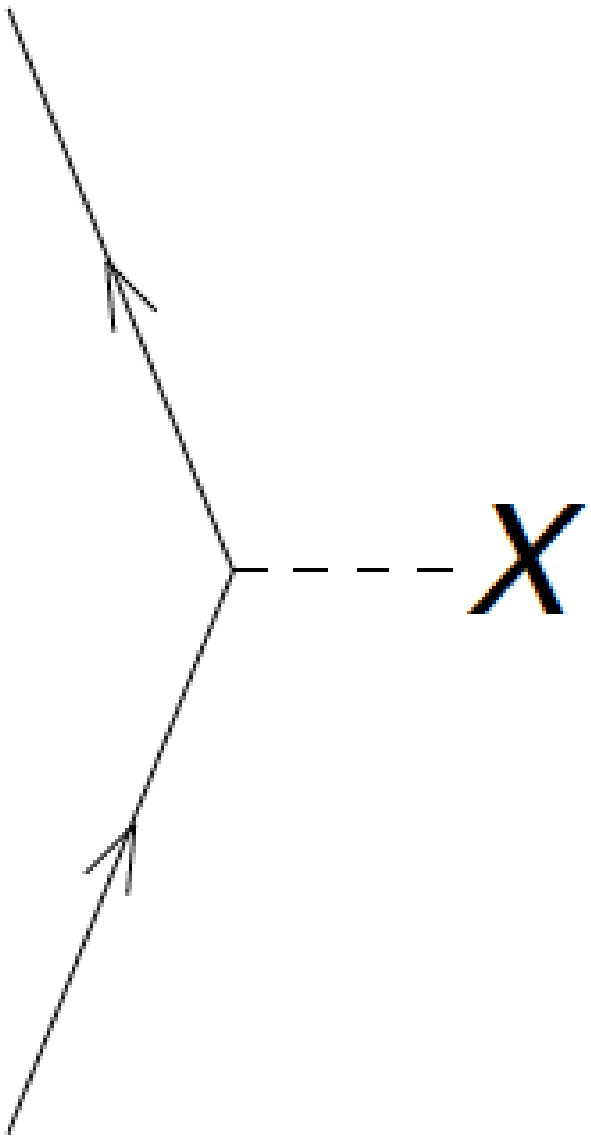


Figure 1.4: The diagrams corresponding to f_a^b. The dashed line with the 'X' denotes the interaction F between the incoming and outgoing lines. The labels a and b are not denoted, but you should label the outgoing and incoming lines accordingly.

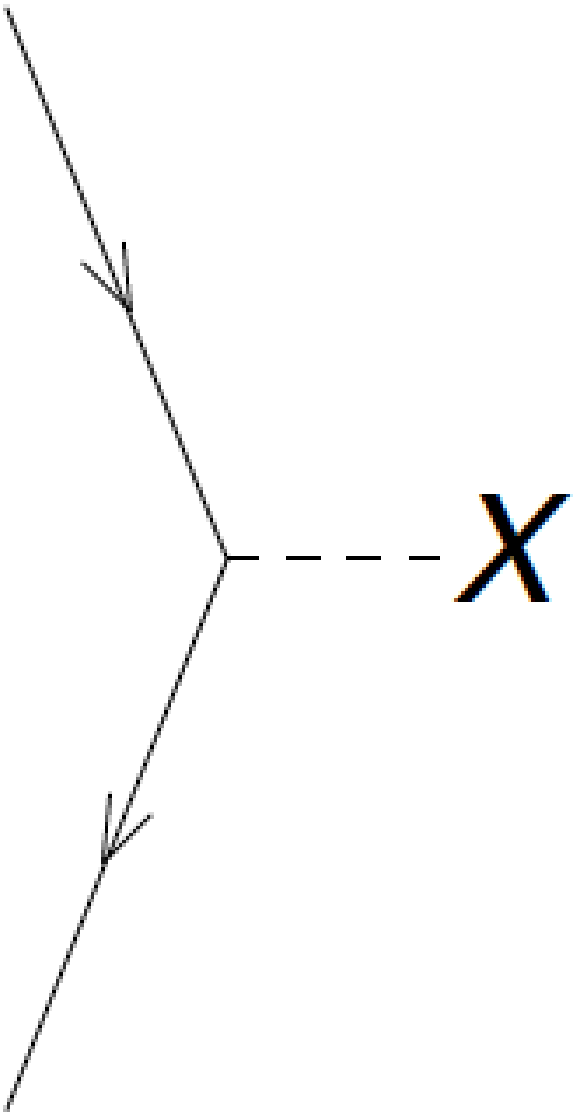


Figure 1.5: The diagrams corresponding to f_i^j. The dashed line with the 'X' denotes the interaction F between the incoming and outgoing lines.

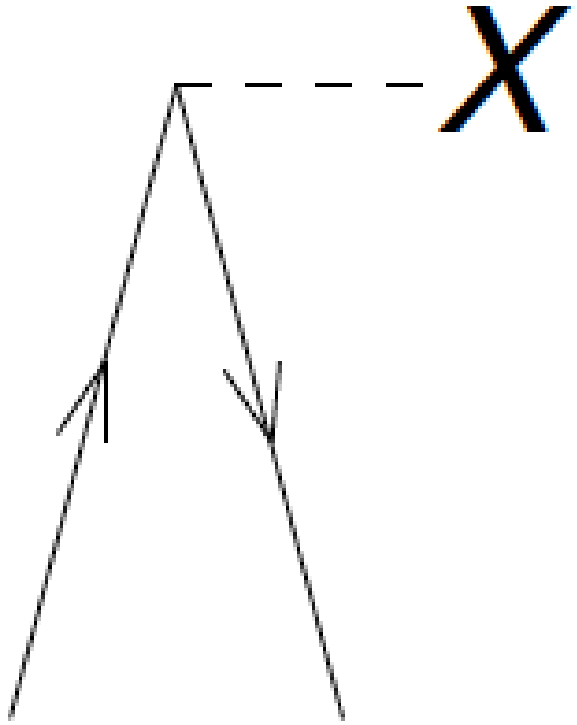


Figure 1.6: The diagrams corresponding to f_a^i. The dashed line with the 'X' denotes the interaction F between the incoming and outgoing lines.

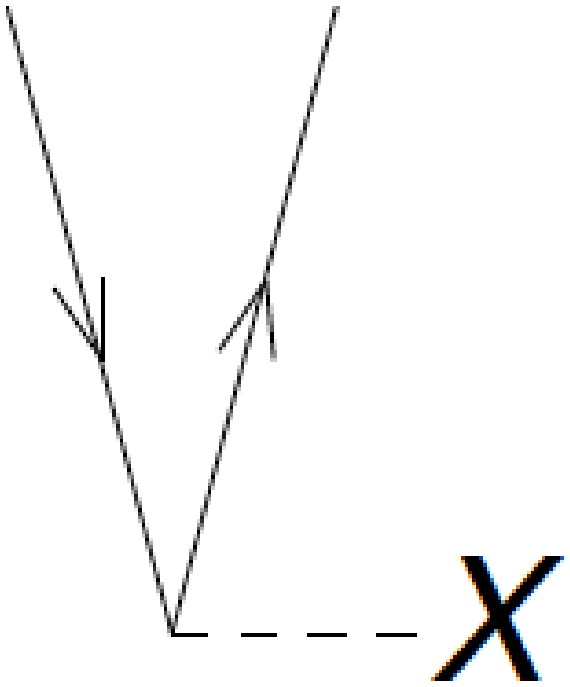


Figure 1.7: The diagrams corresponding to f_i^a. The dashed line with the 'X' denotes the interaction F between the incoming and outgoing lines.

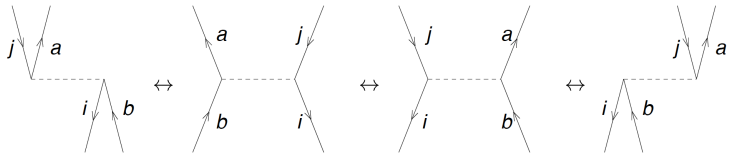


Figure 1.8: The diagrams corresponding to $\langle ai|V|jb\rangle = -\langle ai|V|bj\rangle = -\langle ia|V|jb\rangle = \langle ia|V|bj\rangle$.

$$\hat{T}_1 = \sum_{ia} t_i^a \{a_a^\dagger a_i\} =$$

$$\hat{T}_2 = \frac{1}{4}\sum_{ijab} t_{ij}^{ab} \{a_a^\dagger a_b^\dagger a_j a_i\} =$$

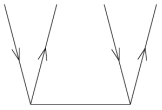


Figure 1.9: The horizontal full line is the cluster amplitude with incoming hole lines and outgoing particle lines as indicated.

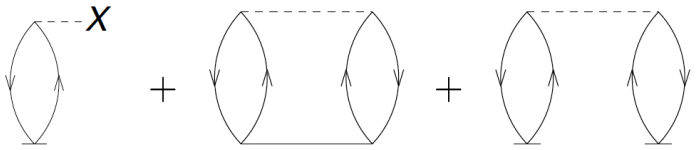


Figure 1.10: Three diagrams enter for the CCSD correlation energy, i.e. all diagrams that leave no external legs.

CCD Approximation

In what follows, we will consider the coupled cluster doubles (CCD) approximation. This approximation is valid in cases where the system cannot exhibit any particle-hole excitations (such as nuclear matter when formulated on a momentum-space grid) or for the pairing model (as the pairing interactions only excites pairs of particles). In this case $t_i^a = 0$ for all i, a, and $\overline{H}_i^a = 0$. The CCD approximation is also of some sort of leading order approximation in the Hartree-Fock basis (as the Hartree-Fock Hamiltonian exhibits no particle-hole excitations).

*

Exercise 10: Derive the CCD equations!

Let us consider the matrix element $\overline{H}_{ij}^{ab}$. Clearly, it consists of all diagrams (i.e. all combinations of T_2, and a single F or V that have two incoming hole lines and two outgoing particle lines. Write down all these diagrams.

Hint. Start systematically! Consider all combinations of F and V diagrams with 0, 1, and 2 cluster amplitudes T_2.

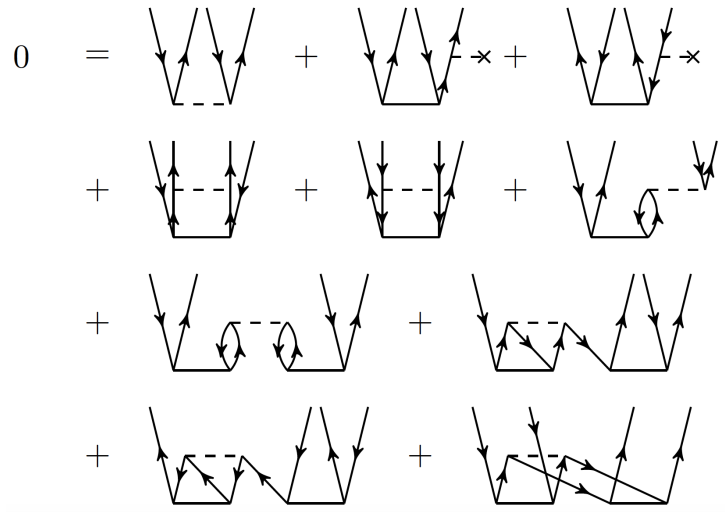


Figure 1.11: The diagrams for the T_1 equation, i.e. the matrix elements of $\overline{H}_i^a$. Taken from Baardsen et al (2013).

Answer. The corresponding algebraic expression is

$$\begin{aligned}\overline{H}_{ij}^{ab} &= \langle ab|V|ij\rangle + P(ab)\sum_c f_c^b t_{ij}^{ac} - P(ij)\sum_k f_j^k t_{ik}^{ab} \\ &+ \frac{1}{2}\sum_{cd}\langle ab|V|cd\rangle t_{ij}^{cd} + \frac{1}{2}\sum_{kl}\langle kl|V|ij\rangle t_{kl}^{ab} + P(ab)P(ij)\sum_{kc}\langle kb|V|cj\rangle t_{ik}^{ac} \\ &+ \frac{1}{2}P(ij)P(ab)\sum_{kcld}\langle kl|V|cd\rangle t_{ik}^{ac} t_{lj}^{db} + \frac{1}{2}P(ij)\sum_{kcld}\langle kl|V|cd\rangle t_{ik}^{cd} t_{lj}^{ab} \\ &+ \frac{1}{2}P(ab)\sum_{kcld}\langle kl|V|cd\rangle t_{kl}^{ac} t_{ij}^{db} + \frac{1}{4}\sum_{kcld}\langle kl|V|cd\rangle t_{ij}^{cd} t_{kl}^{ab}.\end{aligned}$$

===== =====

Let us now turn to the computational cost of a CCD computation.

*

Exercise 11: Computational scaling of CCD

For each of the diagrams in Fig. 1.11 write down the computational cost in terms of the number of occupied A and the number of unoccupied n_u orbitals.

Answer. The cost is $A^2n_u^2$, $A^2n_u^3$, $A^3n_u^2$, $A^2n_u^4$, $A^4n_u^2$, $A^3n_u^3$, $A^4n_u^4$, $A^4n_u^4$, $A^4n_u^4$, and $A^4n_u^4$ for the respective diagrams.

===== =====

Note that $n_u \gg A$ in general. In textbooks, one reads that CCD (and CCSD) cost only $A^2n_u^4$. Our most expensive diagrams, however are $A^4n_u^4$. What is going on?

To understand this puzzle, let us consider the last diagram of Fig. 1.11. We break up the computation into two steps, computing first the intermediate

$$\chi_{ij}^{kl} \equiv \frac{1}{2}\sum_{cd}\langle kl|V|cd\rangle t_{ij}^{cd} \tag{1.28}$$

at a cost of $A^4n_u^2$, and then

$$\frac{1}{2}\sum_{kl}\chi_{ij}^{kl} t_{kl}^{ab} \tag{1.29}$$

at a cost of $A^4n_u^2$. *This is affordable. The price to pay is the storage of the intermediate* χ_{ij}^{kl}, i.e. we traded memory for computational cycles. This trick is known as "factorization."

*

Exercise 12: Factorize the remaining diagrams of the CCD equation
Diagrams 7, 8, and 9 of Fig. 1.11 also need to be factorized.

Answer. For diagram number 7, we compute

$$\chi_{id}^{al} \equiv \sum_{kc} \langle kl|V|cd\rangle t_{ik}^{ac} \tag{1.30}$$

at a cost of $A^3 n_u^3$ and then compute

$$\frac{1}{2} P(ij)P(ab) \sum_{ld} \chi_{id}^{al} t_{lj}^{db} \tag{1.31}$$

at the cost of $A^3 n_u^3$.

For diagram number 8, we compute

$$\chi_i^l \equiv -\frac{1}{2} \sum_{kcd} \langle kl|V|cd\rangle t_{ik}^{cd} \tag{1.32}$$

at a cost of $A^3 n_u^2$, and then compute

$$-P(ij) \sum_{l} \chi_i^l t_{lj}^{ab} \tag{1.33}$$

at the cost of $A^3 n_u^2$.

For diagram number 9, we compute

$$\chi_d^a \equiv \frac{1}{2} \sum_{kcl} \langle kl|V|cd\rangle t_{kl}^{ac} \tag{1.34}$$

at a cost of $A^2 n_u^3$ and then compute

$$P(ab) \sum_{d} \chi_d^a t_{ij}^{db} \tag{1.35}$$

at the cost of $A^3 n_u^3$.

===== =====

We are now ready, to derive the full CCSD equations, i.e. the matrix elements of $\overline{H}_i^a$ and $\overline{H}_{ij}^{ab}$.

*

Project 13: (Optional) Derive the CCSD equations!

aragraph!paragraph>paragraph>-0.5em

a) Let us consider the matrix element $\overline{H}_i^a$ first. Clearly, it consists of all diagrams (i.e. all combinations of T_1, T_2, and a single F or V that have an incoming hole line and an outgoing particle line. Write down all these diagrams.

$$
\begin{aligned}
0 \;=\; & f_{ai} + \sum_c f_{ac} t_i^c - \sum_k f_{ki} t_k^a + \sum_{kc} \langle ka||ci\rangle t_k^c + \sum_{kc} f_{kc} t_{ik}^{ac} + \frac{1}{2}\sum_{kcd} \langle ka||cd\rangle t_{ki}^{cd} - \\
& \frac{1}{2}\sum_{klc} \langle kl||ci\rangle t_{kl}^{ca} - \sum_{kc} f_{kc} t_i^c t_k^a - \sum_{klc} \langle kl||ci\rangle t_k^c t_l^a + \sum_{kcd} \langle ka||cd\rangle t_k^c t_i^d - \\
& \sum_{klcd} \langle kl||cd\rangle t_k^c t_i^d t_l^a + \sum_{klcd} \langle kl||cd\rangle t_k^c t_{li}^{da} - \frac{1}{2}\sum_{klcd} \langle kl||cd\rangle t_{ki}^{cd} t_l^a - \frac{1}{2}\sum_{klcd} \langle kl||cd\rangle t_{kl}^{ca} t_i^d
\end{aligned}
$$

Figure 1.12: The diagrams for the T_1 equation, i.e. the matrix elements of $\overline{H}_i^a$. Taken from Crawford and Schaefer. Here $\langle pq||rs\rangle \equiv \langle pq|V|rs\rangle$ and $f_{pq} \equiv f_q^p$.

Answer.

aragraph!paragraph>paragraph>-0.5em

b) Let us now consider the matrix element $\overline{H}_{ij}^{ab}$. Clearly, it consists of all diagrams (i.e. all combinations of T_1, T_2, and a single F or V that have two incoming hole lines and two outgoing particle lines. Write down all these diagrams and corresponding algebraic expressions.

Answer. ===== =====

Solving the CCD equations

The CCD equations, depicted in Fig. 1.11, are nonlinear in the cluster amplitudes. How do we solve $\overline{H}_{ij}^{ab} = 0$? We add $(f_a^a + f_b^b - f_i^i - f_j^j)t_{ij}^{ab}$ to both sides of $\overline{H}_{ij}^{ab} = 0$ and find

$$(f_a^a + f_b^b - f_i^i - f_j^j)t_{ij}^{ab} = (f_a^a + f_b^b - f_i^i - f_j^j)t_{ij}^{ab} + \overline{H}_{ij}^{ab}$$

Dividing by $(f_a^a + f_b^b - f_i^i - f_j^j)$ yields

$$t_{ij}^{ab} = t_{ij}^{ab} + \frac{\overline{H}_{ij}^{ab}}{f_a^a + f_b^b - f_i^i - f_j^j} \tag{1.36}$$

This equation is of the type $t = f(t)$, and we solve it by iteration, i.e. we start with a guess t_0 and iterate $t_{n+1} = f(t_n)$, and hope that this will converge to a solution. We take the perturbative result

$$\left(t_{ij}^{ab}\right)_0 = \frac{\langle ab|V|ij\rangle}{f_a^a + f_b^b - f_i^i - f_j^j} \tag{1.37}$$

$$
\begin{aligned}
0 \;=\;& \langle ab||ij\rangle + \sum_c \left(f_{bc}t_{ij}^{ac} - f_{ac}t_{ij}^{bc}\right) - \sum_k \left(f_{kj}t_{ik}^{ab} - f_{ki}t_{jk}^{ab}\right) + \\
& \frac{1}{2}\sum_{kl}\langle kl||ij\rangle t_{kl}^{ab} + \frac{1}{2}\sum_{cd}\langle ab||cd\rangle t_{ij}^{cd} + P(ij)P(ab)\sum_{kc}\langle kb||cj\rangle t_{ik}^{ac} + \\
& P(ij)\sum_c \langle ab||cj\rangle t_i^c - P(ab)\sum_k \langle kb||ij\rangle t_k^a + \\
& \frac{1}{2}P(ij)P(ab)\sum_{klcd}\langle kl||cd\rangle t_{ik}^{ac}t_{lj}^{db} + \frac{1}{4}\sum_{klcd}\langle kl||cd\rangle t_{ij}^{cd}t_{kl}^{ab} - \\
& P(ab)\frac{1}{2}\sum_{klcd}\langle kl||cd\rangle t_{ij}^{ac}t_{kl}^{bd} - P(ij)\frac{1}{2}\sum_{klcd}\langle kl||cd\rangle t_{ik}^{ab}t_{jl}^{cd} + \\
& P(ab)\frac{1}{2}\sum_{kl}\langle kl||ij\rangle t_k^a t_l^b + P(ij)\frac{1}{2}\sum_{cd}\langle ab||cd\rangle t_i^c t_j^d - P(ij)P(ab)\sum_{kc}\langle kb||ic\rangle t_k^a t_j^c + \\
& P(ab)\sum_{kc} f_{kc}t_k^a t_{ij}^{bc} + P(ij)\sum_{kc} f_{kc}t_i^c t_{jk}^{ab} - \\
& P(ij)\sum_{klc}\langle kl||ci\rangle t_k^c t_{lj}^{ab} + P(ab)\sum_{kcd}\langle ka||cd\rangle t_k^c t_{ij}^{db} + \\
& P(ij)P(ab)\sum_{kcd}\langle ak||dc\rangle t_i^d t_{jk}^{bc} + P(ij)P(ab)\sum_{klc}\langle kl||ic\rangle t_l^a t_{jk}^{bc} + \\
& P(ij)\frac{1}{2}\sum_{klc}\langle kl||cj\rangle t_i^c t_{kl}^{ab} - P(ab)\frac{1}{2}\sum_{kcd}\langle kb||cd\rangle t_k^a t_{ij}^{cd} -
\end{aligned}
$$

Figure 1.13: The diagrams for the T_2 equation, i.e. the matrix elements of $\overline{H}_{ij}^{ab}$. Taken from Crawford and Schaefer. Here $\langle pq||rs\rangle \equiv \langle pq|V|rs\rangle$, $f_{pq} \equiv f_q^p$, and $P(ab) = 1 - (a \leftrightarrow b)$ antisymmetrizes.

as a starting point, compute $\overline{H}_{ij}^{ab}$, and find a new t_{ij}^{ab} from the right-hand side of Eq. (1.36). We repeat this process until the amplitudes converge.

CCD for the pairing Hamiltonian

You learned about the pairing Hamiltonian earlier in this school. Convince yourself that this Hamiltonian does not induce any 1p-1h excitations. Let us solve the CCD equations for this problem. This consists of the following steps

1. Write a function that compute the potential. We need $\langle ab|V|cd\rangle$, $\langle ij|V|kl\rangle$, and $\langle ab|V|ij\rangle$. Why is there no $\langle ab|V|id\rangle$ or $\langle ai|V|jb\rangle$?
2. Write a function that computes the Fock matrix. We only need f_a^b and f_i^j. Why?
3. Initialize the cluster amplitudes according to Eq. (1.37), and solve Eq. (1.36) by iteration.

*

Project 14: Solve the CCD equations for the pairing problem

Check your results and reproduce Fig 8.5 and Table 8.12 from Lecture Notes in Physics 936.

Answer.

```
## Coupled clusters in CCD approximation
## Implemented for the pairing model of Lecture Notes in Physics 936, Chapter 8.
## Thomas Papenbrock, June 2018

import numpy as np


def init_pairing_v(g,pnum,hnum):
    """
    returns potential matrices of the pairing model in three relevant channels

    param g: strength of the pairing interaction, as in Eq. (8.42)
    param pnum: number of particle states
    param hnum: number of hole states

    return v_pppp, v_pphh, v_hhhh: np.array(pnum,pnum,pnum,pnum),
                                   np.array(pnum,pnum,hnum,hnum),
                                   np.array(hnum,hnum,hnum,hnum),
                                   The interaction as a 4-indexed tensor in three channels.
    """
    v_pppp=np.zeros((pnum,pnum,pnum,pnum))
    v_pphh=np.zeros((pnum,pnum,hnum,hnum))
    v_hhhh=np.zeros((hnum,hnum,hnum,hnum))

    gval=-0.5*g
    for a in range(0,pnum,2):
        for b in range(0,pnum,2):
            v_pppp[a,a+1,b,b+1]=gval
            v_pppp[a+1,a,b,b+1]=-gval
            v_pppp[a,a+1,b+1,b]=-gval
```

```
                v_pppp[a+1,a,b+1,b]=gval

        for a in range(0,pnum,2):
            for i in range(0,hnum,2):
                v_pphh[a,a+1,i,i+1]=gval
                v_pphh[a+1,a,i,i+1]=-gval
                v_pphh[a,a+1,i+1,i]=-gval
                v_pphh[a+1,a,i+1,i]=gval

        for j in range(0,hnum,2):
            for i in range(0,hnum,2):
                v_hhhh[j,j+1,i,i+1]=gval
                v_hhhh[j+1,j,i,i+1]=-gval
                v_hhhh[j,j+1,i+1,i]=-gval
                v_hhhh[j+1,j,i+1,i]=gval

        return v_pppp, v_pphh, v_hhhh


    def init_pairing_fock(delta,g,pnum,hnum):
        """
        initializes the Fock matrix of the pairing model

        param delta: Single-particle spacing, as in Eq. (8.41)
        param g: pairing strength, as in eq. (8.42)
        param pnum: number of particle states
        param hnum: number of hole states

        return f_pp, f_hh: The Fock matrix in two channels as numpy arrays np.array(pnum,pnum), np.ar
        """
    # the Fock matrix for the pairing model. No f_ph needed, because we are in Hartree-Fock basis
        deltaval=0.5*delta
        gval=-0.5*g
        f_pp = np.zeros((pnum,pnum))
        f_hh = np.zeros((hnum,hnum))

        for i in range(0,hnum,2):
            f_hh[i  ,i  ] = deltaval*i+gval
            f_hh[i+1,i+1] = deltaval*i+gval

        for a in range(0,pnum,2):
            f_pp[a  ,a  ] = deltaval*(hnum+a)
            f_pp[a+1,a+1] = deltaval*(hnum+a)

        return f_pp, f_hh


    def init_t2(v_pphh,f_pp,f_hh):
        """
        Initializes t2 amlitudes as in MBPT2, see first equation on page 345

        param v_pphh: pairing tensor in pphh channel
        param f_pp:   Fock matrix in pp channel
        param f_hh:   Fock matrix in hh channel

        return t2: numpy array in pphh format, 4-indices tensor
        """
        pnum = len(f_pp)
        hnum = len(f_hh)
        t2_new = np.zeros((pnum,pnum,hnum,hnum))
        for i in range(hnum):
```

```
        for j in range(hnum):
            for a in range(pnum):
                for b in range(pnum):
                    t2_new[a,b,i,j] = v_pphh[a,b,i,j] / (f_hh[i,i]+f_hh[j,j]-f_pp[a,a]-f_pp[b,b])
    return t2_new


# CCD equations. Note that the "->abij" assignment is redundant, because indices are ordered alph
# Nevertheless, we retain it for transparency.
def ccd_iter(v_pppp,v_pphh,v_hhhh,f_pp,f_hh,t2):
    """
    Performs one iteration of the CCD equations (8.34), using also intermediates for the nonlinia

    param v_pppp: pppp-channel pairing tensor, numpy array
    param v_pphh: pphh-channel pairing tensor, numpy array
    param v_hhhh: hhhh-channel pairing tensor, numpy array
    param f_pp: Fock matrix in pp channel
    param f_hh: Fock matrix in hh channel
    param t2: Initial t2 amplitude, tensor in form of pphh channel

    return t2_new: new t2 amplitude, tensor in form of pphh channel
    """
    pnum = len(f_pp)
    hnum = len(f_hh)
    Hbar_pphh = (  v_pphh
                 + np.einsum('bc,acij->abij',f_pp,t2)
                 - np.einsum('ac,bcij->abij',f_pp,t2)
                 - np.einsum('abik,kj->abij',t2,f_hh)
                 + np.einsum('abjk,ki->abij',t2,f_hh)
                 + 0.5*np.einsum('abcd,cdij->abij',v_pppp,t2)
                 + 0.5*np.einsum('abkl,klij->abij',t2,v_hhhh)
                )

    # hh intermediate, see (8.47)
    chi_hh = 0.5* np.einsum('cdkl,cdjl->kj',v_pphh,t2)

    Hbar_pphh = Hbar_pphh - (  np.einsum('abik,kj->abij',t2,chi_hh)
                             - np.einsum('abik,kj->abji',t2,chi_hh) )

    # pp intermediate, see (8.46)
    chi_pp = -0.5* np.einsum('cdkl,bdkl->cb',v_pphh,t2)

    Hbar_pphh = Hbar_pphh + (  np.einsum('acij,cb->abij',t2,chi_pp)
                             - np.einsum('acij,cb->baij',t2,chi_pp) )

    # hhhh intermediate, see (8.48)
    chi_hhhh = 0.5 * np.einsum('cdkl,cdij->klij',v_pphh,t2)

    Hbar_pphh = Hbar_pphh + 0.5 * np.einsum('abkl,klij->abij',t2,chi_hhhh)

    # phph intermediate, see (8.49)
    chi_phph= + 0.5 * np.einsum('cdkl,dblj->bkcj',v_pphh,t2)


    Hbar_pphh = Hbar_pphh + (  np.einsum('bkcj,acik->abij',chi_phph,t2)
                             - np.einsum('bkcj,acik->baij',chi_phph,t2)
                             - np.einsum('bkcj,acik->abji',chi_phph,t2)
                             + np.einsum('bkcj,acik->baji',chi_phph,t2) )

    t2_new=np.zeros((pnum,pnum,hnum,hnum))
    for i in range(hnum):
```

```
        for j in range(hnum):
            for a in range(pnum):
                for b in range(pnum):
                    t2_new[a,b,i,j] = (  t2[a,b,i,j]
                                       + Hbar_pphh[a,b,i,j] / (f_hh[i,i]+f_hh[j,j]-f_pp[a,a]-f_pp

    return t2_new


def ccd_energy(v_pphh,t2):
    """
    Computes CCD energy. Call as
    energy = ccd_energy(v_pphh,t2)

    param v_pphh: pphh-channel pairing tensor, numpy array
    param t2: t2 amplitude, tensor in form of pphh channel

    return energy: CCD correlation energy
    """
    erg = 0.25*np.einsum('abij,abij',v_pphh,t2)
    return erg

################################
######## Main Program

# set parameters as for model
pnum = 20 # number of particle states
hnum = 10 # number of hole states
delta = 1.0

g = 0.5

print("parameters")
print("delta =", delta, ", g =", g)


# Initialize pairing matrix elements and Fock matrix
v_pppp, v_pphh, v_hhhh = init_pairing_v(g,pnum,hnum)
f_pp, f_hh = init_pairing_fock(delta,g,pnum,hnum)

# Initialize T2 amplitudes from MBPT2
t2 = init_t2(v_pphh,f_pp,f_hh)
erg = ccd_energy(v_pphh,t2)

# Exact MBPT2 for comparison, see last equation on page 365
exact_mbpt2 = -0.25*g**2*(1.0/(2.0+g) + 2.0/(4.0+g) + 1.0/(6.0+g))
print("MBPT2 energy =", erg, ", compared to exact:", exact_mbpt2)


# iterate CCD equations niter times
niter=200
mix=0.3
erg_old=0.0
eps=1.e-8
for iter in range(niter):
    t2_new = ccd_iter(v_pppp,v_pphh,v_hhhh,f_pp,f_hh,t2)
    erg = ccd_energy(v_pphh,t2_new)
    myeps = abs(erg-erg_old)/abs(erg)
    if myeps < eps: break
    erg_old=erg
    print("iter=", iter, "erg=", erg, "myeps=", myeps)
```

```
    t2 = mix*t2_new + (1.0-mix)*t2
print("Energy = ", erg)
```