CCDT: Diagrams in the t_3 amplitude

This document lists all diagrams entering the t_3 equation, as well as the index realignment to perform matrix multiplications.

Notationwise, the operation

$t_{ij}^{ab} \rightarrow t_{bj}^{ai}$

indicates a index transformation where we simply align the matrix representation of this tensor in a fashion corresponding to the element order above. The purpose of this operation is to align tensors so that contractions may be performed as matrix multiplications. Upper indices is mapped to a row index, while lower indices are mapped to columns.

Technically, this means to recalculate the row and column indices of the matrix elements in the COO format (flexmat class). The corresponding code to generate the two different representations of the flexmat object t_2 above is

t_{ij}^{ab} = t2.pq_rs()

t_{bj}^{ai} = t2.pr_qs()

From a theoretical point of view, this operation may be interpreted as a generalized transpose for tensors of $rank > 2$.

The $(t_2 t_3)$ terms

Diagram Label	Factor	Permutation	Index Transform	Code translation
$(t_2 t_3)_a$	$+1$	$\hat{P}(i/jk\|a/bc)$	$\sum_{ldme} \langle lm\|\|de\rangle t_{il}^{ad} t_{mjk}^{ebc} \rightarrow \sum_{me} \sum_{ld} t_{me}^{bjck} \langle me\|\|ld\rangle t_{ai}^{ld}$	update_as_qtru_ps(t3.qtru_sp() $*$ vhhpp.qs_pr() $*$ t2.sq_pr())
$(t_2 t_3)_b$	$-\frac{1}{2}$	$\hat{P}(i/jk)$	$\sum_{ldme} \langle lm\|\|de\rangle t_{li}^{de} t_{mjk}^{abc} \rightarrow \sum_{m} \sum_{lde} t_{m}^{abjck} \langle m\|\|lde\rangle t_{i}^{lde}$	update_as_pqtru_s(t3.pqtru_s()$*$vhhpp.q_prs()$*$t2.rpq_s())
$(t_2 t_3)_c$	$-\frac{1}{2}$	$\hat{P}(a/bc)$	$\sum_{ldme} \langle lm\|\|de\rangle t_{lm}^{da} t_{ijk}^{ebc} \rightarrow \sum_{e} \sum_{lmd} t_{e}^{ibjck} \langle e\|\|lmd\rangle t_{a}^{lmd}$	update_as_sqtru_p(t3.sqtru_p()$*$vhhpp.s_pqr()$*$t2.rsp_q())
$(t_2 t_3)_d$	$-\frac{1}{2}$	$\hat{P}(k/ij\|a/bc)$	$\sum_{ldme} \langle lm\|\|de\rangle t_{ij}^{ad} t_{lmk}^{bec} \rightarrow \sum_{lme} \sum_{d} t_{lme}^{bck} \langle lme\|\|d\rangle t_{aij}^{d}$	update_as_qru_pst(t3.pru_stq()$*$vhhpp.pqs_r()$*$t2.q_prs())
$(t_2 t_3)_e$	$-\frac{1}{2}$	$\hat{P}(i/jk\|c/ab)$	$\sum_{ldme} \langle lm\|\|de\rangle t_{il}^{ab} t_{jmk}^{dec} \rightarrow \sum_{mde} \sum_{l} t_{mde}^{jck} \langle mde\|\|l\rangle t_{abi}^{l}$	update_as_tru_pqs(t3.sru_tpq()$*$vhhpp.qrs_p()$*$t2.s_pqr())
$(t_2 t_3)_f$	$+\frac{1}{4}$	$\hat{P}(k/ij)$	$\sum_{ldme} \langle lm\|\|de\rangle t_{ij}^{de} t_{lmk}^{abc} \rightarrow \sum_{lm} \sum_{de} t_{lm}^{abck} \langle lm\|\|de\rangle t_{ij}^{de}$	update_as_pqru_st(t3.pqru_st()$*$vhhpp.pq_rs()$*$t2.pq_rs())
$(t_2 t_3)_q$	$+\frac{1}{4}$	$\hat{P}(c/ab)$	$\sum_{ldme} \langle lm\|\|de\rangle t_{lm}^{ab} t_{ijk}^{dec} \rightarrow \sum_{de} \sum_{lm} t_{de}^{ijck} \langle de\|\|lm\rangle t_{ab}^{lm}$	update_as_stru_pq(t3.stru_pq()$*$vhhpp.rs_pq()$*$t2.rs_pq())

The $(t_2 t_2)$ terms

These are incorrectly generated due to unconnected lines in the interaction, so they are not yet ready for implementation.

Special attention will need to be given to the antisymmetric elements in the multiplication.

Diagram Label	Factor	Permutation	Index Transform	Code translation
$(t_2 t_2)_a$	-1	$\hat{P}(k/ij\|a/bc)$	$\sum_{ld} \langle l\|f\|d\rangle t_{ij}^{ad} t_{lk}^{bc} \rightarrow \sum_{l} \sum_{d} t_{l}^{bck} \langle l\|f\|d\rangle t_{aij}^{d} = 0$	(canonical HF basis) $\rightarrow$ no contribution
$(t_2 t_2)_b$	$+1$	$\hat{P}(i/jk\|abc)$	$\sum_{lde} \langle lb\|\|de\rangle t_{il}^{ad} t_{jk}^{ec} \rightarrow \sum_{ld} \sum_{e} (t_{ld}^{ai} \langle ld\|\|be\rangle)_{e}^{aib} t_{cjk}^{e}$	update_as_psq_rtu(t2.pr_sq()$*$vhppp.pr_qs()$*$t2.p_qrs())

$(t_2t_2)_c$	$-\frac{1}{2}$	$\hat{P}(i/jk\vert c/ab)$	$\sum_{ldce}\langle lc\Vert de\rangle t_{il}^{ab}t_{jk}^{de} \rightarrow \sum_{de}\sum_{l}(t_{de}^{jk}\langle de\Vert lc\rangle)_{l}^{jkc}t_{abi}^{l}$	update_as_tur_pqs(t2.rs_pq()∗vhppp.rs_pq()∗t2.s_pqr())
$(t_2t_2)_d$	$+\frac{1}{2}$	$\hat{P}(k/ij\vert a/bc)$	$\sum_{ldmk}\langle lm\Vert dk\rangle t_{ij}^{ad}t_{lm}^{bc} \rightarrow \sum_{lm}\sum_{d}(t_{lm}^{bc}\langle lm\Vert dk\rangle)_{d}^{bck}t_{aij}^{d}$	update_as_qru_pst(t2.pq_rs()∗vhhph.pq_rs()∗t2.q_prs())

The problem of unconnected lines leaving the interaction may be solved by performing the multiplication and alignment in three steps:

1. Align and multiply inside paranthesis.
2. Align the resulting product to the final amplitude and multiply.
3. Align the resulting product to the amplitudes.

The linear t_3 terms

The following terms are linear in the t_3 amplitude.

Diagram Label	Factor	Permutation	Index Transform	Code translation
$(t_3)_a$	$+\frac{1}{2}$	$\hat{P}(c/ab)$	$\sum_{de}\langle ab\Vert de\rangle t_{ijk}^{dec} \rightarrow \sum_{de}\langle ab\Vert de\rangle t_{cijk}^{de}$	update_as_pq_rstu(vpppp.pq_rs() ∗ t3.pq_rstu())
$(t_3)_b$	$+\frac{1}{2}$	$\hat{P}(k/ij)$	$\sum_{lm}\langle lm\Vert ij\rangle t_{lmk}^{abc} \rightarrow \sum_{lm} t_{lm}^{abck}\langle lm\Vert ij\rangle$	update_as_pqru_st(t3.pqrs_tu()∗ vphhp.pq_rs())
$(t_3)_c$	$+1$	$\hat{P}(i/jk\vert a/bc)$	$\sum_{ld}\langle al\Vert id\rangle t_{ljk}^{dbc} \rightarrow \sum_{ld}\langle ai\Vert ld\rangle t_{bcjk}^{ld}$	update_as_ps_qrtu(vphhp.pr_qs() ∗ t3.sp_qrtu())

The diagram $(t_3)_a$ includes the ladder operator from $\hat{V}$, so it will have to be calulcated using some block scheme.

The linear t_2 terms

Diagram Label	Factor	Permutation	Index Transform	Code translation
$(t_2)_a$	$+1$	$\hat{P}(k/ij\vert a/bc)$	$\sum_{d}\langle bc\Vert dk\rangle t_{ij}^{ad} \rightarrow \sum_{d}\langle bck\Vert d\rangle t_{aij}^{d}$	update_as_qru_pst(vppph.pqs_r() ∗ t2.q_prs())
$(t_2)_b$	-1	$\hat{P}(i/jk\vert c/ab)$	$\sum_{l}\langle lc\Vert jk\rangle t_{il}^{ab} \rightarrow \sum_{l} t_{l}^{abi}\langle l\Vert cjk\rangle$	update_as_pqs_rtu(t2.pqr_s() ∗ vphhh.p_qrs())

Implementation

Finally we state the basic implementation of the t3 amplitude equation. Tables is separated by horizontal lines.

Note that permutations are not yet included.

---

t2t3a.update_as_qtru_ps(t3.qtru_sp()∗vhhpp.qs_pr()∗t2.sq_pr())

t2t3b.update_as_pqtru_s(t3.pqtru_s()∗vhhpp.q_prs()∗t2.rpq_s())

t2t3c.update_as_sqtru_p(t3.sqtru_p()∗vhhpp.s_pqr()∗t2.rsp_q())

t2t3d.update_as_qru_pst(t3.pru_stq()∗vhhpp.pqs_r()∗t2.q_prs())

t2t3e.update_as_tru_pqs(t3.sru_tpq()∗vhhpp.qrs_p()∗t2.s_pqr())

t2t3f.update_as_pqru_st(t3.pqru_st()∗vhhpp.pq_rs()∗t2.pq_rs())

t2t3g.update_as_stru_pq(t3.stru_pq()∗vhhpp.rs_pq()∗t2.rs_pq())

---

t2t2b.update_as_psq_rtu(t2.pr_sq()∗vhppp.pr_qs()∗t2.p_qrs())

t2t2c.update_as_tur_pqs(t2.rs_pq()∗vhppp.rs_pq()∗t2.s_pqr())

t2t2d.update_as_qru_pst(t2.pq_rs()∗vhhph.pq_rs()∗t2.q_prs())

---

t3a.update_as_pq_rstu(vpppp.pq_rs() ∗ t3.pq_rstu()) //Note that this will probably be replaced by a block implementation.

t3b.update_as_pqru_st(t3.pqrs_tu()∗ vphhp.pq_rs())

t3c.update_as_ps_qrtu(vphhp.pr_qs() ∗ t3.sp_qrtu())

---

t2a.update_as_qru_pst(vppph.pqs_r() ∗ t2.q_prs())

t2b.update_as_pqs_rtu(t2.pqr_s() ∗ vhphh.p_qrs())

CCDT: Diagrams in the t_2 amplitude

Actually, the inclusion of tripes will only result in three extra terms in the doubles equation, and we may even remove the first due to the fact that we use a canonical HF basis where f is diagonal.

$$D_{CCD} + \sum_{me} f_e^m t_{ijm}^{abe} + \frac{1}{2}\hat{P}(ab)\sum_{mef}\langle bm||ef\rangle t_{ijm}^{aef} - \frac{1}{2}\hat{P}(ij)\sum_{mne}\langle mn||je\rangle t_{imn}^{abe} = 0$$

The terms we need to include in the t_2 equation is then (labelling as in Shavitt and Bartlett):

Diagram Label	Factor	Permutation	Index Transform	Code translation
D_{10b}	$+\frac{1}{2}$	$\hat{P}(ab)$	$\sum_{mef}\langle bm\|\|ef\rangle t_{ijm}^{aef} \to (\sum_{mef}\langle b\|\|mef\rangle t_{ija}^{mef})_{ij}^{ab}$	update_as_q_rsp(vphpp.p_qrs() ∗ t3.uqr_stp())
D_{10c}	$-\frac{1}{2}$	$\hat{P}(ij)$	$\sum_{mne}\langle mn\|\|je\rangle t_{imn}^{abe} \to (\sum_{mne} t_{mne}^{abi}\langle mne\|\|j\rangle)_{ij}^{ab}$	update_as_pqr_s(t3.pqs_tur() ∗ vhhhp.pqs_r())

This means we have to add in the following when computing the doubles contribution

D10b.update_as_q_rsp(vphpp.p_qrs() ∗ t3.uqr_stp())

D10c.update_as_pqr_s(t3.pqs_tur() ∗ vhhhp.pqs_r())

```
In [ ]:
```