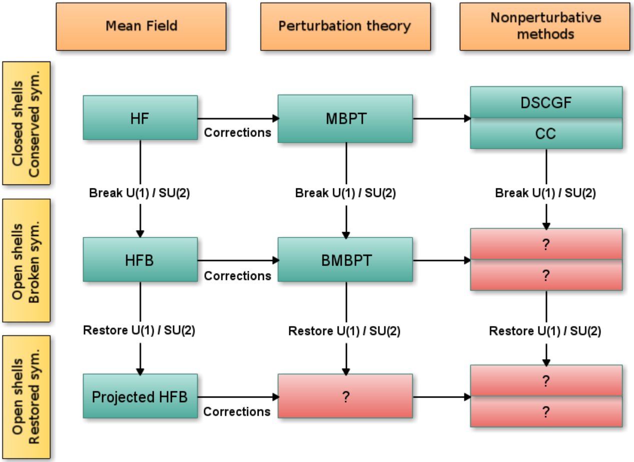
Bogoliubov Many-Body Perturbation Theory for Open-Shell Nuclei

Pierre Arthuis
IRFU, CEA, Université Paris - Saclay

with T. Duguet (CEA Saclay), J.-P. Ebran (CEA DAM), H. Hergert (MSU), R. Roth (TU Darmstadt) & A. Tichai (ESNT, CEA Saclay)

Workshop MBPT in modern quantum chemistry and nuclear physics
CEA Saclay - March 29th 2018

Quantum many-body methods

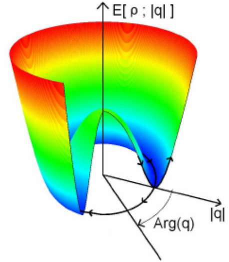


Expansion methods around unperturbed product state

On symmetry breaking

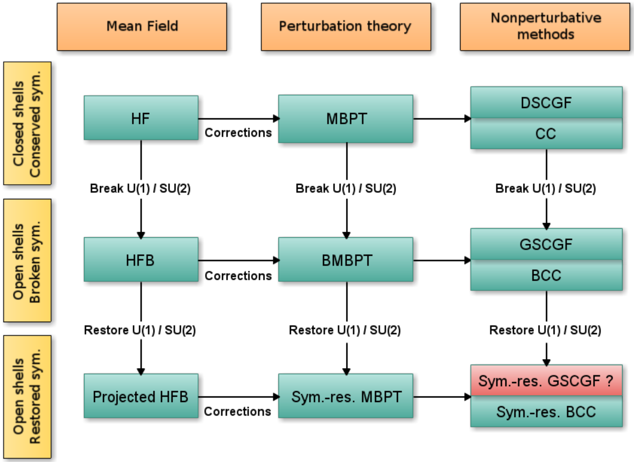
- Symmetry breaking helps incorporating non-dynamical correlations:
  - ◇ Superfluid character: $U(1)$ (particle number)
  - ◇ Deformations: $SU(2)$ (angular momentum)
- But nuclei carry good quantum numbers (e.g. number of particles)

  $\Rightarrow$ Symmetries must eventually be restored



- See Thomas' talk on Monday

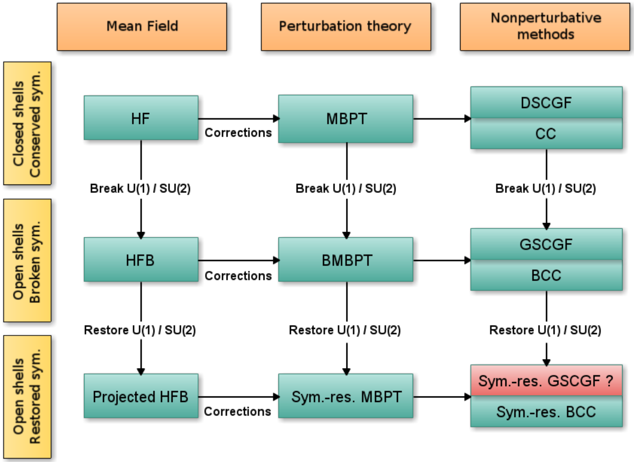
Quantum many-body methods



New methods recently proposed and implemented

- GSCGF, BCC [Somà *et al.* 2011, Signoracci *et al.* 2014]
- Sym.-res. BCC & sym.-res. BMBPT [Duguet 2015, Duguet & Signoracci 2017, Qiu *et al.* 2017]

Quantum many-body methods



MBPT reimplemented using SRG-evolved H in closed shell [Tichai *et al.* 2016] → Robert's talk

➥ MBPT competes with non-perturbative methods

Current objective: extend to (symmetry-projected) BMBPT for open shell

The BMBPT project

Particle-number-restored BMBPT formalism

Exact diagrammatic expansion with symmetry breaking *and* restoration

[Duguet and Signoracci, *J. Phys. G* **44**, 2017] $\rightarrow$ Thomas' talk on Monday

$\downarrow$

Formalism actualization

Expand off-diagonal kernels

$\langle \Psi_0^A | H | \Phi(\phi) \rangle$

$\langle \Psi_0^A | \Phi(\phi) \rangle$

Symmetry restoration

Diagonal reduction

$\langle \Psi_0^A | H | \Phi \rangle$

$\langle \Psi_0^A | \Phi \rangle$

No symmetry restoration

$\searrow$ $\swarrow$

Ab initio

Realist H

High order

The BMBPT project: Current step

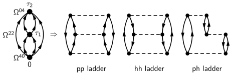
Diagonal implementation

$\langle \Psi_0^A | H | \Phi \rangle$

$\langle \Psi_0^A | \Phi \rangle$

Manual derivation of order 3

Check against MBPT limit



Automatic derivation

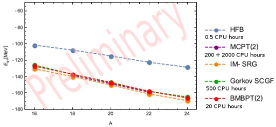
[PA, Duguet, Tichai, Lasseri, Ebran, in prep.]

Ab initio

Realist H

High order

Numerical implementation



[PA, Tichai, Hergert, Roth, Duguet, in prep.]

Bogoliubov Many-Body Perturbation Theory

- Bogoliubov vacuum $|\Phi\rangle$, $\beta_k|\Phi\rangle = 0\,\forall k$ with

$$\beta_k = \sum_p U^*_{pk}\, c_p + V^*_{pk}\, c^\dagger_p$$

$$\beta^\dagger_k = \sum_p U_{pk}\, c^\dagger_p + V_{pk}\, c_p$$

- Particle number symmetry broken: $A|\Phi\rangle \neq \mathrm{A}|\Phi\rangle$

- Grand potential $\Omega \equiv H - \lambda A$ in qp basis, normal-ordered w.r.t. $|\Phi\rangle$

$$\begin{aligned}
\Omega = \Omega^{00} &+ \frac{1}{1!}\sum_{k_1 k_2} \Omega^{11}_{k_1 k_2}\beta^\dagger_{k_1}\beta_{k_2} + \frac{1}{2!}\sum_{k_1 k_2}\left\{\Omega^{20}_{k_1 k_2}\beta^\dagger_{k_1}\beta^\dagger_{k_2} + \Omega^{02}_{k_1 k_2}\beta_{k_2}\beta_{k_1}\right\} \\
&+ \frac{1}{(2!)^2}\sum_{k_1 k_2 k_3 k_4}\Omega^{22}_{k_1 k_2 k_3 k_4}\beta^\dagger_{k_1}\beta^\dagger_{k_2}\beta_{k_4}\beta_{k_3} \\
&+ \frac{1}{3!}\sum_{k_1 k_2 k_3 k_4}\left\{\Omega^{31}_{k_1 k_2 k_3 k_4}\beta^\dagger_{k_1}\beta^\dagger_{k_2}\beta^\dagger_{k_3}\beta_{k_4} + \Omega^{13}_{k_1 k_2 k_3 k_4}\beta^\dagger_{k_1}\beta_{k_4}\beta_{k_3}\beta_{k_2}\right\} \\
&+ \frac{1}{4!}\sum_{k_1 k_2 k_3 k_4}\left\{\Omega^{40}_{k_1 k_2 k_3 k_4}\beta^\dagger_{k_1}\beta^\dagger_{k_2}\beta^\dagger_{k_3}\beta^\dagger_{k_4} + \Omega^{04}_{k_1 k_2 k_3 k_4}\beta_{k_4}\beta_{k_3}\beta_{k_2}\beta_{k_1}\right\} + \dots
\end{aligned}$$

- Fully correlated state obtained via the evolution operator

$$
\begin{aligned}
|\Psi_0^A(\tau)\rangle &\equiv \mathcal{U}(\tau)|\Phi\rangle \\
&= e^{-\tau\Omega_0} \mathrm{T} e^{-\int_0^\tau d\tau \Omega_1(\tau)} |\Phi\rangle
\end{aligned}
$$

Ground state energy of an open-shell nucleus

$$
\mathrm{E}_0^A = \lim_{\tau \to \infty} \langle \Psi_0^A(\tau)|\Omega|\Phi\rangle_c
$$

Bogoliubov Many-Body Perturbation Theory

- Fully correlated state obtained via the evolution operator

$$|\Psi_0^A(\tau)\rangle \equiv \mathcal{U}(\tau)|\Phi\rangle$$
$$= e^{-\tau\Omega_0}\mathrm{T}e^{-\int_0^\tau d\tau\Omega_1(\tau)}|\Phi\rangle$$

Ground state energy of an open-shell nucleus

$$\mathrm{E}_0^A = \lim_{\tau\to\infty}\langle\Psi_0^A(\tau)|\Omega|\Phi\rangle_c$$

- Diagonal propagators (no anomalous)

$$G^{+-(0)}_{k_1k_2}(\tau_1,\tau_2) \equiv \frac{\langle\Phi|\mathrm{T}[\beta^\dagger_{k_1}(\tau_1)\beta_{k_2}(\tau_2)]|\Phi\rangle}{\langle\Phi|\Phi\rangle}$$
$$G^{-+(0)}_{k_1k_2}(\tau_1,\tau_2) \equiv \frac{\langle\Phi|\mathrm{T}[\beta_{k_1}(\tau_1)\beta^\dagger_{k_2}(\tau_2)]|\Phi\rangle}{\langle\Phi|\Phi\rangle}$$

with antisymmetry relation

$$G^{+-(0)}_{k_1k_2}(\tau_1,\tau_2) = -G^{-+(0)}_{k_2k_1}(\tau_2,\tau_1)$$

Bogoliubov Many-Body Perturbation Theory

- Perturbative expansion of ground-state energy ($\Omega = \Omega_0 + \Omega_1$)

$$
\mathrm{E}_0 = \langle \Phi | \left\{ \Omega(0) - \int_0^\infty d\tau_1 \mathrm{T}\left[\Omega_1(\tau_1)\Omega(0)\right] + \frac{1}{2!}\int_0^\infty d\tau_1 d\tau_2 \mathrm{T}\left[\Omega_1(\tau_1)\Omega_1(\tau_2)\Omega(0)\right] + \dots \right\} |\Phi\rangle_c
$$

$$
= \sum_{p=0}^{\infty} \frac{(-1)^p}{p!} \sum_{\substack{i_0+j_0=2,4 \\ \dots \\ i_p+j_p=2,4}} \int_0^\infty d\tau_1 \dots d\tau_p
$$

$$
\times \sum_{\substack{k_1 \dots k_{i_1}, k_{i_1+1} \dots k_{i_1+j_1} \\ \dots \\ l_1 \dots l_{i_p}, l_{i_p+1} \dots l_{i_p+j_p}}} \frac{\Omega^{i_1 j_1}_{k_1 \dots k_{i_1}\, k_{i_1+1} \dots k_{i_1+j_1}}}{(i_1)!(j_1)!} \dots \frac{\Omega^{i_p j_p}_{l_1 \dots l_{i_p}\, l_{i_p+1} \dots l_{i_p+j_p}}}{(i_p)!(j_p)!} \frac{\Omega^{i_0 j_0}_{m_1 \dots m_{i_0}\, m_{i_0+1} \dots m_{i_0+j_0}}}{(i_0)!(j_0)!}
$$

$$
\times \langle \Phi | \mathrm{T} \Big[\beta^\dagger_{k_1}(\tau_1) \dots \beta^\dagger_{k_{i_1}}(\tau_1)\, \beta_{k_{i_1+j_1}}(\tau_1) \dots \beta_{k_{i_1+1}}(\tau_1) \dots \beta^\dagger_{l_1}(\tau_p) \dots \beta^\dagger_{l_{i_p}}(\tau_p) \dots
$$

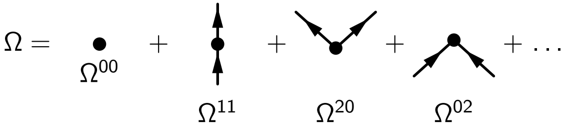
$$
\times \beta_{l_{i_p+j_p}}(\tau_p) \dots \beta_{l_{i_p+1}}(\tau_p)\, \beta^\dagger_{m_1}(0) \dots \beta^\dagger_{m_{i_0}}(0) \beta_{m_{i_0+j_0}}(0) \dots \beta_{m_{i_0+1}}(0) \Big] |\Phi\rangle_c
$$

Diagonal case: $\varphi = 0$

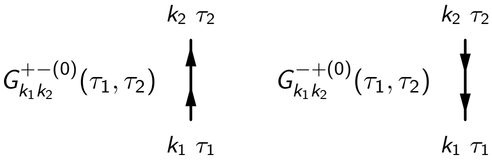
- No anomalous propagator, no self-contraction
- Standard Wick's theorem with respect to $|\Phi\rangle$

Building blocks of the diagrammatic

- Normal-ordered form of Ω with respect to $|\Phi\rangle$



- Diagonal propagators



- Main diagrammatic rules from Wick theorem
  - ◇ No external legs
  - ◇ No oriented loop between vertices
  - ◇ No self-contraction
  - ◇ Propagators go out of the Ω vertex at time 0
  - ◇ Equivalent lines
  - ◇ Discard topologically equivalent diagrams

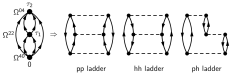
Diagonal implementation

$\langle \Psi_0^A | H | \Phi \rangle$

$\langle \Psi_0^A | \Phi \rangle$

Manual derivation of order 3

Check against MBPT limit



Automatic derivation

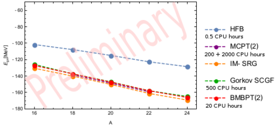
[PA, Duguet, Tichai, Lasseri, Ebran, in prep.]

Ab initio

Realist H

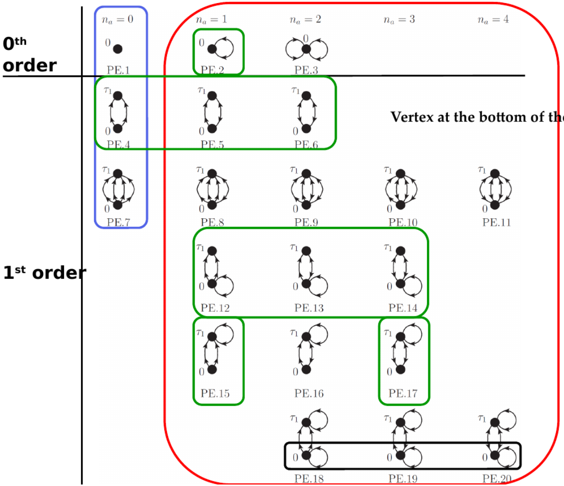
High order

Numerical implementation



[PA, Tichai, Hergert, Roth, Duguet, in prep.]

BMBPT diagrams landscape



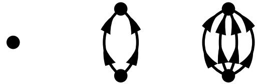
20 diagrams
3 diagrams
17 diagrams
9 diagrams

Vertex at the bottom of the diagrams at fixed time 0

Only contributions to φ = 0
Carry the φ dependence
Null for HFB reference state

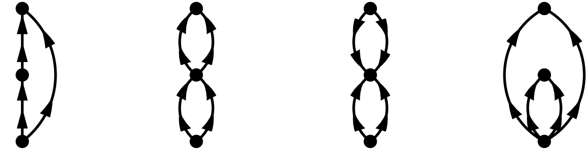
Low-order diagrams

- First- and second-order diagrams [Duguet and Signoracci, *J. Phys. G* **44**, 2017]

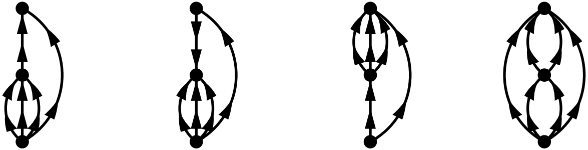


PE0.1 PE1.1 PE1.2

- Third-order diagrams



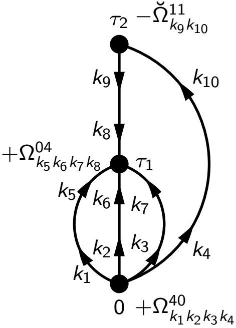
PE2.1 PE2.2 PE2.3 PE2.4



PE2.5 PE2.6 PE2.7 PE2.8

Validation of the manual derivation by checking the MBPT limit

Derivation of a third-order diagram



Feynman (time-dependent) and Goldstone (time-integrated) expressions:

$$\text{PE2.6} = -\frac{1}{3!} \sum_{k_1 k_2 k_3 k_4 k_8} \Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{04}_{k_1 k_2 k_3 k_8} \breve{\Omega}^{11}_{k_8 k_4} \int_0^\infty \mathrm{d}\tau_1 \mathrm{d}\tau_2 \theta(\tau_1 - \tau_2) e^{-\tau_1 \left(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_8}\right)} e^{\tau_2 \left(E_{k_8} - E_{k_4}\right)}$$

$$= -\frac{1}{3!} \sum_{k_1 k_2 k_3 k_4 k_8} \frac{\Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{04}_{k_1 k_2 k_3 k_8} \breve{\Omega}^{11}_{k_8 k_4}}{\left(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_4}\right) \left(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_8}\right)}$$

Status of manual derivation and implementation

- All diagrams from 2N vertices derived and implemented up to order 3

  [PA, Tichai, Ebran, Duguet]

- Want to go to higher orders
  - At least up to order 4
    - Check convergence pattern
    - Grasp effect from quadruples $\leftrightarrow$ 8 qp excitations
  - Derivation time-consuming
  - Derivation error-prone

Status of manual derivation and implementation

- All diagrams from 2N vertices derived and implemented up to order 3

  [PA, Tichai, Ebran, Duguet]

- Want to go to higher orders
  - ◇ At least up to order 4
    - ▶ Check convergence pattern
    - ▶ Grasp effect from quadruples $\leftrightarrow$ 8 qp excitations
  - ◇ Derivation time-consuming
  - ◇ Derivation error-prone

Develop automatic tool

- ◇ To generate all possible connected diagrams at order n
- ◇ To extract associated time-integrated expressions
- ◇ To be both quick and safe

The BMBPT project: Automatic derivation

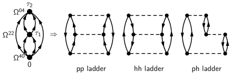
Diagonal implementation

$\langle \Psi_0^A | H | \Phi \rangle$

$\langle \Psi_0^A | \Phi \rangle$

Manual derivation of order 3

Check against MBPT limit



Automatic derivation

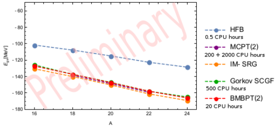
[PA, Duguet, Tichai, Lasseri, Ebran, in prep.]

Ab initio

Realist H

High order

Numerical implementation



[PA, Tichai, Hergert, Roth, Duguet, in prep.]

Why and how?

Our goal

An automatic and systematic way of producing diagrams

Our tool

Adjacency matrices in graph theory

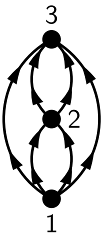
Our challenge

From BMBPT diagrammatic rules to constraints on matrices

Each Feynman diagram to be represented by an adjacency matrix

- a_{ij} indicate the number of edges going from node i to node j

$$A = \begin{pmatrix} 0 & 2 & 2 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix} \Leftrightarrow$$



  - ◇ Carry detailed information for directed graphs
  - ◇ Symmetry properties and connectivity properties directly readable

- Only two propagators, readable as one once reading direction is fixed
  - ◇ Perfectly adapted for diagonal BMBPT
  - ◇ Extension needed for off-diagonal diagrams with anomalous propagator

Constraints from the diagrammatic rules

Each vertex belongs to $\Omega^{[2]}$, $\Omega^{[4]}$ or $\Omega^{[6]}$

For each vertex i, $\sum_j (a_{ij} + a_{ji})$ is 2, 4 or 6

No self-contraction

Every diagonal element is zero

Every propagator coming out of the vertex at time 0 goes upward

First column of the matrix is zero

No loop between vertices

Can restrict to upper triangular matrices

Generate BMBPT diagrams

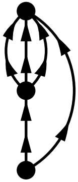
- Generate all upper triangular $n \times n$ matrices for n-th order BMBPT diagrams
  - Fill the matrices "vertex-wise" with all allowed integers
  - Check the degree of each vertex before moving on

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 & a_{12} & a_{13} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 & a_{12} & a_{13} \\ 0 & 0 & a_{23} \\ 0 & 0 & 0 \end{pmatrix}$$

- Discard matrices leading to topologically identical diagrams

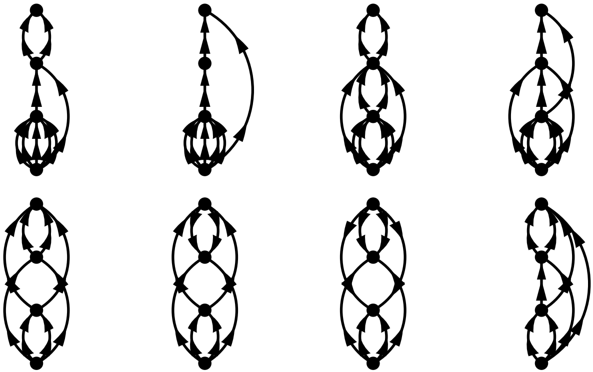
- Read the matrix and translate it into drawing instructions

```
\begin{fmfgraph*}(60,60)
\fmftop{v2}\fmfbottom{v0}
\fmf{phantom}{v0,v1}
\fmfv{d.shape=circle,d.filled=full,d.size=3thick}{v0}
\fmf{phantom}{v1,v2}
\fmfv{d.shape=circle,d.filled=full,d.size=3thick}{v1}
\fmfv{d.shape=circle,d.filled=full,d.size=3thick}{v2}
\fmffreeze
\fmf{prop_pm}{v0,v1}
\fmf{prop_pm,right=0.6}{v0,v2}
\fmf{prop_pm}{v1,v2}
\fmf{prop_pm,left=0.5}{v1,v2}
\fmf{prop_pm,right=0.5}{v1,v2}
\end{fmfgraph*}
```



Time to cook some diagrams

Run the code at order 4 with 2N and 3N interactions, obtain...



...and 388 others!

Status of the numerical derivation

- Number of diagrams with 2N interactions (using an HFB vacuum)
  - 8 (1) diagrams at order 3
  - 59 (10) diagrams at order 4
  - 568 (82) diagrams at order 5
  - 6 805 (938) diagrams at order 6

- Number of diagrams with 2N and 3N interactions (using an HFB vacuum)
  - 23 (8) diagrams at order 3
  - 396 (177) diagrams at order 4
  - 10 716 (5 055) diagrams at order 5
  - 100 000+ diagrams at order 6?

- Obtained in only a few minutes...

Automated expression derivation

All BMBPT diagrams produced automatically at a given order

➥ Need to derive automatically the diagrams' expressions

Automated expression derivation

All BMBPT diagrams produced automatically at a given order

➥ Need to derive automatically the diagrams' expressions

- Feynman diagrams recast different time-orderings
  - ✔ Less diagrams to set up
  - ✘ But time-integrated (Goldstone) expressions are to be coded

Automated expression derivation

All BMBPT diagrams produced automatically at a given order

➥ Need to derive automatically the diagrams' expressions

- Feynman diagrams recast different time-orderings
  - ✔ Less diagrams to set up
  - ✘ But time-integrated (Goldstone) expressions are to be coded
- Goldstone diagrams capture each time ordering separately
  - ✔ Time-integrated expressions obtained directly from diagrammatic rules
  - ✘ Many more diagrams to consider

Automated expression derivation

All BMBPT diagrams produced automatically at a given order

➥ Need to derive automatically the diagrams' expressions

- Feynman diagrams recast different time-orderings
  - ✔ Less diagrams to set up
  - ✘ But time-integrated (Goldstone) expressions are to be coded
- Goldstone diagrams capture each time ordering separately
  - ✔ Time-integrated expressions obtained directly from diagrammatic rules
  - ✘ Many more diagrams to consider

Challenge: Extract Goldstone expressions from Feynman diagrams

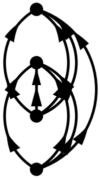
- ◇ Capture all time ordering at once
- ◇ Challenging because of structure of corresponding time integrals
- ◇ Undone task to our knowledge (even for standard diagrammatic)

Extract the Feynman expression of the diagrams

- Extract graph structure info as well
  - Associate labels with vertices, propagators, etc.
  - In- / out-degree of vertices associated with annihilators / creators
  - Run routines for symmetry factors
- Have your code write the corresponding equations in your .tex file

$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{40}_{k_5 k_6 k_7 k_8} \Omega^{04}_{k_5 k_1 k_2 k_3} \Omega^{04}_{k_6 k_7 k_8 k_4} \int_0^{\tau} \mathrm{d}\tau_1 \mathrm{d}\tau_2 \mathrm{d}\tau_3 \theta(\tau_2 - \tau_1)\theta(\tau_3 - \tau_1)$$

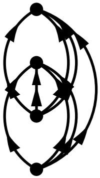
$$\times e^{-\tau_1(-E_{k_5} - E_{k_6} - E_{k_7} - E_{k_8})} e^{-\tau_2(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_5})} e^{-\tau_3(E_{k_4} + E_{k_6} + E_{k_7} + E_{k_8})}$$



Extract the Feynman expression of the diagrams

- Extract graph structure info as well
  - Associate labels with vertices, propagators, etc.
  - In- / out-degree of vertices associated with annihilators / creators
  - Run routines for symmetry factors
- Have your code write the corresponding equations in your .tex file

$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{40}_{k_5 k_6 k_7 k_8} \Omega^{04}_{k_5 k_1 k_2 k_3} \Omega^{04}_{k_6 k_7 k_8 k_4} \int_0^\tau \mathrm{d}\tau_1 \mathrm{d}\tau_2 \mathrm{d}\tau_3 \theta(\tau_2 - \tau_1)\theta(\tau_3 - \tau_1)$$
$$\times e^{-\tau_1(-E_{k_5} - E_{k_6} - E_{k_7} - E_{k_8})} e^{-\tau_2(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_5})} e^{-\tau_3(E_{k_4} + E_{k_6} + E_{k_7} + E_{k_8})}$$

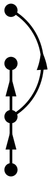


Sign, prefactor and operators left unchanged in Goldstone expression

➥ Only need to extract the denominator

Time-structure diagrams [PA, Duguet, Tichai, Lasseri, Ebran, in prep.]

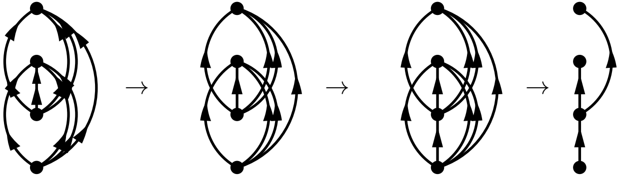
- Introduce time-structure diagrams (TSDs)
  - ◇ Links carry time-ordering relations, moving towards higher times
  - ◇ Contain only the minimal set of links to describe all the time relations



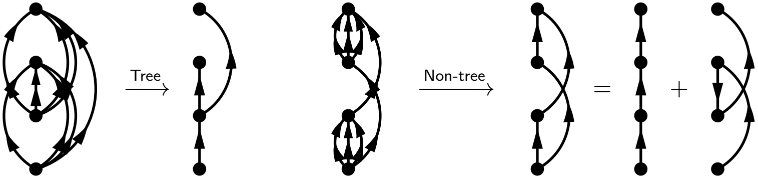
- Determine the time-structure diagram (TSD) associated to BMBPT one
  - ◇ Propagators carry time-ordering relations
  - ◇ Ω vertex at time 0 is a lower limit for time
  - ◇ One TSD recast several Feynman, even more Goldstone

Time-structure diagrams [PA, Duguet, Tichai, Lasseri, Ebran, in prep.]

- Each TSD produced from the BMBPT diagram
  - Replace propagators by links
  - Add links between vertex at time 0 and other vertices
  - Remove links carrying unnecessary information



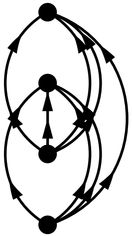
- Extraction of time-integrated expression depends on tree / non-tree



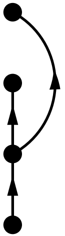
Denominator extraction algorithm for tree TSDs

For each perturbation vertex in the diagram with an associated tree TSD

1. Determine all its descendants using the TSD diagram
2. Form a subgraph using the vertex and its descendants
3. For all propagators entering the subgraph, add the associated qpe



$\leftrightarrow$

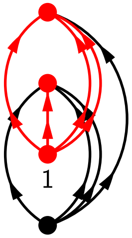


$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \frac{\Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{40}_{k_5 k_6 k_7 k_8} \Omega^{04}_{k_5 k_1 k_2 k_3} \Omega^{04}_{k_6 k_7 k_8 k_4}}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_5})(E_{k_4}+E_{k_6}+E_{k_7}+E_{k_8})}$$

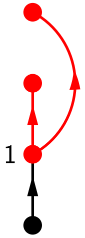
Denominator extraction algorithm for tree TSDs

For each perturbation vertex in the diagram with an associated tree TSD

1. Determine all its descendants using the TSD diagram
2. Form a subgraph using the vertex and its descendants
3. For all propagators entering the subgraph, add the associated qpe



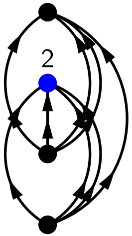
$\leftrightarrow$



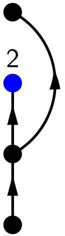
$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \frac{\Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{40}_{k_5 k_6 k_7 k_8} \Omega^{04}_{k_5 k_1 k_2 k_3} \Omega^{04}_{k_6 k_7 k_8 k_4}}{(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_4})(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_5})(E_{k_4} + E_{k_6} + E_{k_7} + E_{k_8})}$$

For each perturbation vertex in the diagram with an associated tree TSD

1. Determine all its descendants using the TSD diagram
2. Form a subgraph using the vertex and its descendants
3. For all propagators entering the subgraph, add the associated qpe



$\leftrightarrow$

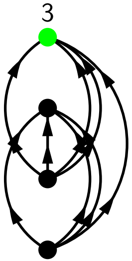


$$\frac{-(-1)^3}{(3!)^2}\sum_{k_i}\frac{\Omega^{40}_{k_1k_2k_3k_4}\Omega^{40}_{k_5k_6k_7k_8}\Omega^{04}_{k_5k_1k_2k_3}\Omega^{04}_{k_6k_7k_8k_4}}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_5})(E_{k_4}+E_{k_6}+E_{k_7}+E_{k_8})}$$

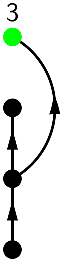
Denominator extraction algorithm for tree TSDs

For each perturbation vertex in the diagram with an associated tree TSD

1. Determine all its descendants using the TSD diagram
2. Form a subgraph using the vertex and its descendants
3. For all propagators entering the subgraph, add the associated qpe



$\leftrightarrow$

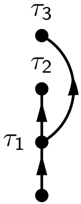


$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \frac{\Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{40}_{k_5 k_6 k_7 k_8} \Omega^{04}_{k_5 k_1 k_2 k_3} \Omega^{04}_{k_6 k_7 k_8 k_4}}{(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_4})(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_5})(E_{k_4} + E_{k_6} + E_{k_7} + E_{k_8})}$$

Denominator extraction and integral structure

Why a so simple denominator algorithm for all trees?

Link between tree TSD structure and time integrals structure

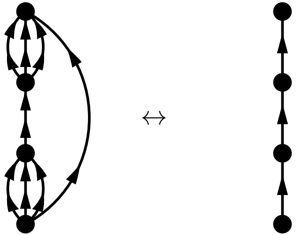


$$
\begin{aligned}
D &= \lim_{\tau\to\infty} \int_0^{\tau} d\tau_1 d\tau_2 d\tau_3 \theta(\tau_3-\tau_1)\theta(\tau_2-\tau_1) e^{a\tau_1} e^{b\tau_2} e^{c\tau_3} \\
&= \lim_{\tau\to\infty} \int_0^{\tau} d\tau_1 e^{a\tau_1} \int_0^{\tau_1} d\tau_2 e^{b\tau_2} \int_0^{\tau_1} d\tau_3 e^{c\tau_3} \\
&= \lim_{\tau\to\infty} \frac{1}{bc} \int_0^{\tau} d\tau_1 e^{a\tau_1} \left(e^{b\tau} - e^{b\tau_1}\right)\left(e^{c\tau} - e^{c\tau_1}\right) \\
&= \frac{1}{bc(a+b+c)}
\end{aligned}
$$

- Integrate from the leaves first
- Go down each branch
- Each vertex depends on the vertices above it

Same algorithm applied on linear tree

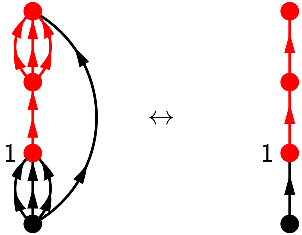
Classic Goldstone rule recovered on a Feynman graph



$$\frac{(-1)^3}{(3!)^2}\sum_{k_i}\frac{\Omega^{40}_{k_1k_2k_3k_4}\Omega^{13}_{k_5k_1k_2k_3}\Omega^{31}_{k_6k_7k_8k_5}\Omega^{04}_{k_6k_7k_8k_4}}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_5}+E_{k_4})(E_{k_4}+E_{k_6}+E_{k_7}+E_{k_8})}$$

Same algorithm applied on linear tree

Classic Goldstone rule recovered on a Feynman graph

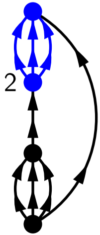


$$\frac{(-1)^3}{(3!)^2} \sum_{k_i} \frac{\Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{13}_{k_5 k_1 k_2 k_3} \Omega^{31}_{k_6 k_7 k_8 k_5} \Omega^{04}_{k_6 k_7 k_8 k_4}}{(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_4})(E_{k_5} + E_{k_4})(E_{k_4} + E_{k_6} + E_{k_7} + E_{k_8})}$$

Algorithm for denominator extraction: Special case

Same algorithm applied on linear tree

Classic Goldstone rule recovered on a Feynman graph



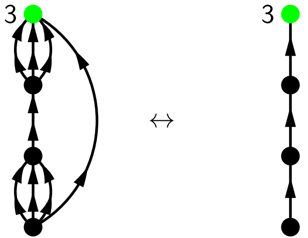
$\leftrightarrow$

$$\frac{(-1)^3}{(3!)^2} \sum_{k_i} \frac{\Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{13}_{k_5 k_1 k_2 k_3} \Omega^{31}_{k_6 k_7 k_8 k_5} \Omega^{04}_{k_6 k_7 k_8 k_4}}{(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_4})(E_{k_5} + E_{k_4})(E_{k_4} + E_{k_6} + E_{k_7} + E_{k_8})}$$

Algorithm for denominator extraction: Special case

Same algorithm applied on linear tree

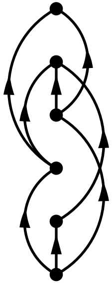
Classic Goldstone rule recovered on a Feynman graph



$$\frac{(-1)^3}{(3!)^2} \sum_{k_i} \frac{\Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{13}_{k_5 k_1 k_2 k_3} \Omega^{31}_{k_6 k_7 k_8 k_5} \Omega^{04}_{k_6 k_7 k_8 k_4}}{(E_{k_1} + E_{k_2} + E_{k_3} + E_{k_4})(E_{k_5} + E_{k_4})(E_{k_4} + E_{k_6} + E_{k_7} + E_{k_8})}$$

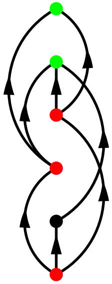
Algorithm for cycle finding

- For each node_a with out_degree $\geq$ 2:
  - For each node_b different from node_a:
    - List all paths going from node_a to node_b
    - If in_degree(node_b) $\geq$ 2 and nb_paths $\geq$ 2:
      node_a and node_b are end nodes of the cycle
    - Check that the two paths share only their end ones



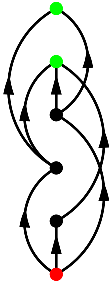
Algorithm for cycle finding

- For each node_a with out_degree ≥ 2:
  - For each node_b different from node_a:
    - List all paths going from node_a to node_b
    - If in_degree(node_b) ≥ 2 and nb_paths ≥ 2:
      node_a and node_b are end nodes of the cycle
    - Check that the two paths share only their end ones



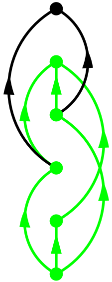
Algorithm for cycle finding

- For each node_a with out_degree $\geq$ 2:
  - For each node_b different from node_a:
    - List all paths going from node_a to node_b
    - If in_degree(node_b) $\geq$ 2 and nb_paths $\geq$ 2:
      node_a and node_b are end nodes of the cycle
    - Check that the two paths share only their end ones



Algorithm for cycle finding

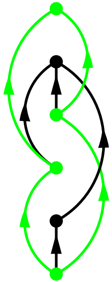
- For each node_a with out_degree ≥ 2:
  - For each node_b different from node_a:
    - List all paths going from node_a to node_b
    - If in_degree(node_b) ≥ 2 and nb_paths ≥ 2:
      node_a and node_b are end nodes of the cycle
    - Check that the two paths share only their end ones



➥ Two pairs of end nodes producing cycles to be addressed

Algorithm for cycle finding

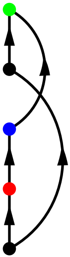
- For each node_a with out_degree ≥ 2:
  - For each node_b different from node_a:
    - List all paths going from node_a to node_b
    - If in_degree(node_b) ≥ 2 and nb_paths ≥ 2:
      node_a and node_b are end nodes of the cycle
    - Check that the two paths share only their end ones



➥ Two pairs of end nodes producing cycles to be addressed

Algorithm for cycle disentangling

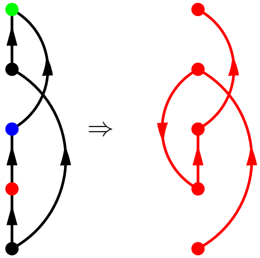
- Set node_to_insert as the first node of path_1 after start node
- For each daughter_node in path_2 but the starting node:
  - Make a copy of the graph
  - Add an edge from node_to_insert to daughter_node
  - Set mother_node as the node preceding daughter_node in path_2
  - Add an edge from mother_node to daughter_node
  - Remove the edges carrying unnecessary information



$\Rightarrow$

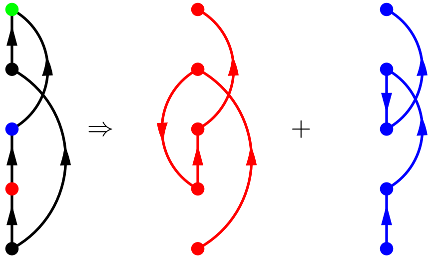
Algorithm for cycle disentangling

- Set node_to_insert as the first node of path_1 after start node
- For each daughter_node in path_2 but the starting node:
  - ◇ Make a copy of the graph
  - ◇ Add an edge from node_to_insert to daughter_node
  - ◇ Set mother_node as the node preceding daughter_node in path_2
  - ◇ Add an edge from mother_node to daughter_node
  - ◇ Remove the edges carrying unnecessary information



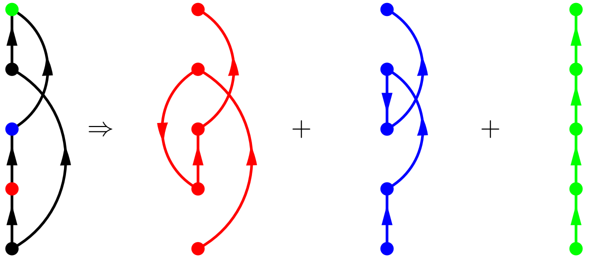
Algorithm for cycle disentangling

- Set node_to_insert as the first node of path_1 after start node
- For each daughter_node in path_2 but the starting node:
  - ◇ Make a copy of the graph
  - ◇ Add an edge from node_to_insert to daughter_node
  - ◇ Set mother_node as the node preceding daughter_node in path_2
  - ◇ Add an edge from mother_node to daughter_node
  - ◇ Remove the edges carrying unnecessary information



Algorithm for cycle disentangling

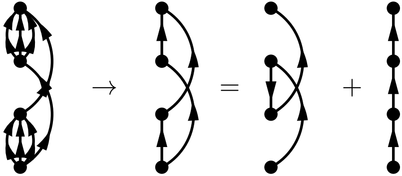
- Set node_to_insert as the first node of path_1 after start node
- For each daughter_node in path_2 but the starting node:
  - Make a copy of the graph
  - Add an edge from node_to_insert to daughter_node
  - Set mother_node as the node preceding daughter_node in path_2
  - Add an edge from mother_node to daughter_node
  - Remove the edges carrying unnecessary information



Denominator extraction: Non-tree case

If the associated time-structure diagram is not a tree:

- Separate the TSD in a sum of tree TSDs
- Apply the tree denominator algorithm, sum the results



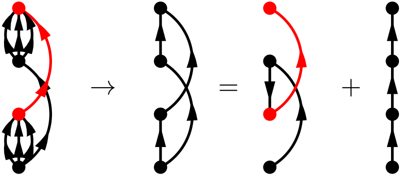
$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{13}_{k_5 k_1 k_2 k_3} \Omega^{31}_{k_6 k_7 k_8 k_4} \Omega^{04}_{k_6 k_7 k_8 k_5}$$

$$\times \left[\frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_6}+E_{k_7}+E_{k_8})(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} + \frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_4}+E_{k_5})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} \right]$$

Denominator extraction: Non-tree case

If the associated time-structure diagram is not a tree:

- Separate the TSD in a sum of tree TSDs
- Apply the tree denominator algorithm, sum the results



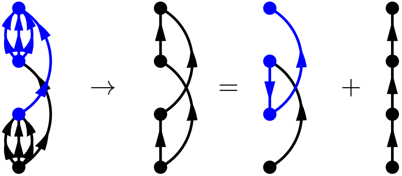
$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{13}_{k_5 k_1 k_2 k_3} \Omega^{31}_{k_6 k_7 k_8 k_4} \Omega^{04}_{k_6 k_7 k_8 k_5}$$

$$\times \left[\frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_6}+E_{k_7}+E_{k_8})(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} + \frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_4}+E_{k_5})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} \right]$$

Denominator extraction: Non-tree case

If the associated time-structure diagram is not a tree:

- Separate the TSD in a sum of tree TSDs
- Apply the tree denominator algorithm, sum the results

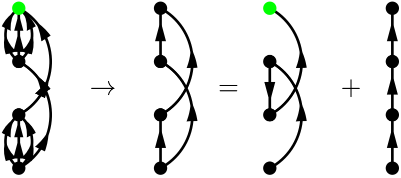


$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{13}_{k_5 k_1 k_2 k_3} \Omega^{31}_{k_6 k_7 k_8 k_4} \Omega^{04}_{k_6 k_7 k_8 k_5}$$

$$\times \left[\frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_6}+E_{k_7}+E_{k_8})(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} + \frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_4}+E_{k_5})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} \right]$$

If the associated time-structure diagram is not a tree:

- Separate the TSD in a sum of tree TSDs
- Apply the tree denominator algorithm, sum the results



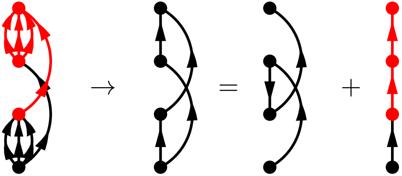
$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{13}_{k_5 k_1 k_2 k_3} \Omega^{31}_{k_6 k_7 k_8 k_4} \Omega^{04}_{k_6 k_7 k_8 k_5}$$

$$\times \left[\frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_6}+E_{k_7}+E_{k_8})(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} + \frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_4}+E_{k_5})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} \right]$$

Denominator extraction: Non-tree case

If the associated time-structure diagram is not a tree:

- Separate the TSD in a sum of tree TSDs
- Apply the tree denominator algorithm, sum the results



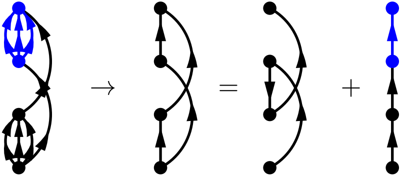
$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{13}_{k_5 k_1 k_2 k_3} \Omega^{31}_{k_6 k_7 k_8 k_4} \Omega^{04}_{k_6 k_7 k_8 k_5}$$

$$\times \left[\frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_6}+E_{k_7}+E_{k_8})(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} + \frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_4}+E_{k_5})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} \right]$$

Denominator extraction: Non-tree case

If the associated time-structure diagram is not a tree:

- Separate the TSD in a sum of tree TSDs
- Apply the tree denominator algorithm, sum the results



$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{13}_{k_5 k_1 k_2 k_3} \Omega^{31}_{k_6 k_7 k_8 k_4} \Omega^{04}_{k_6 k_7 k_8 k_5}$$

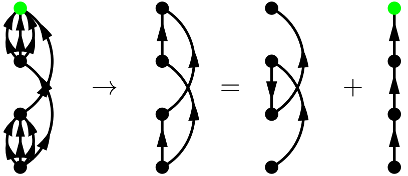
$$\times \left[\frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_6}+E_{k_7}+E_{k_8})(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} \right.$$

$$\left. + \frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_4}+E_{k_5})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} \right]$$

Denominator extraction: Non-tree case

If the associated time-structure diagram is not a tree:

- Separate the TSD in a sum of tree TSDs
- Apply the tree denominator algorithm, sum the results



$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \Omega^{40}_{k_1 k_2 k_3 k_4} \Omega^{13}_{k_5 k_1 k_2 k_3} \Omega^{31}_{k_6 k_7 k_8 k_4} \Omega^{04}_{k_6 k_7 k_8 k_5}$$

$$\times \left[\frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_6}+E_{k_7}+E_{k_8})(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} + \frac{1}{(E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(E_{k_4}+E_{k_5})(E_{k_5}+E_{k_6}+E_{k_7}+E_{k_8})} \right]$$

All BMBPT expressions produced automatically at a given order

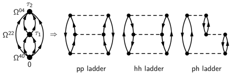
➥ Need to implement them numerically

The BMBPT project: Current step

Diagonal implementation
$\langle \Psi_0^A | H | \Phi \rangle$
$\langle \Psi_0^A | \Phi \rangle$

Manual derivation of order 3

Check against MBPT limit

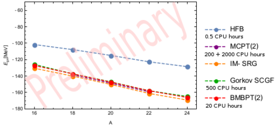


Automatic derivation
[PA, Duguet, Tichai, Lasseri, Ebran, in prep.]

Ab initio

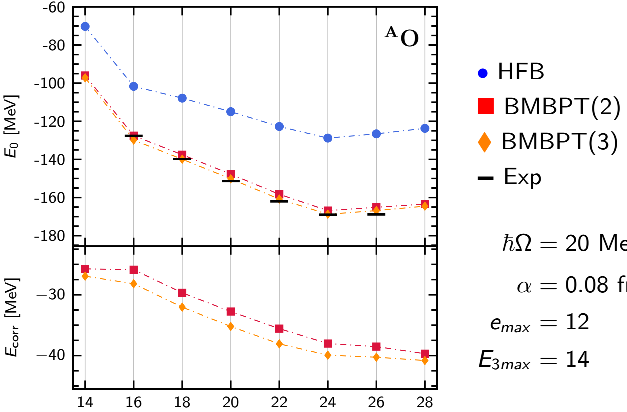
Realist H
High order

Numerical implementation



[PA, Tichai, Hergert, Roth, Duguet, in prep.]

Oxygen chain calculations - Ground state energies



$\hbar\Omega = 20$ MeV

$\alpha = 0.08$ fm^4

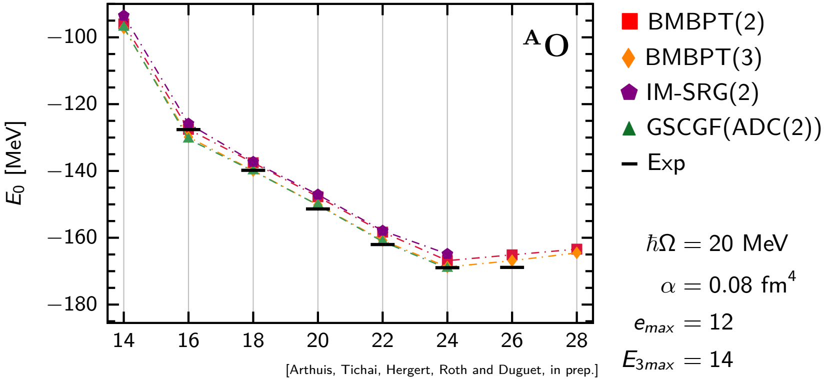
$e_{max} = 12$

$E_{3max} = 14$

[Arthuis, Tichai, Hergert, Roth and Duguet, in prep.]

- $E^{(3)}$ **one order of magnitude smaller than** $E^{(2)}$
- Computer resources independent of system size: **10-20 CPU hours**
- Error estimate on 3rd order correction: $\Delta E = \Delta A^{(3)} \cdot 8\text{MeV}/\text{A} \approx 5\text{MeV}$
- **Calculations of Ca, Ni and Sn chains coming soon**

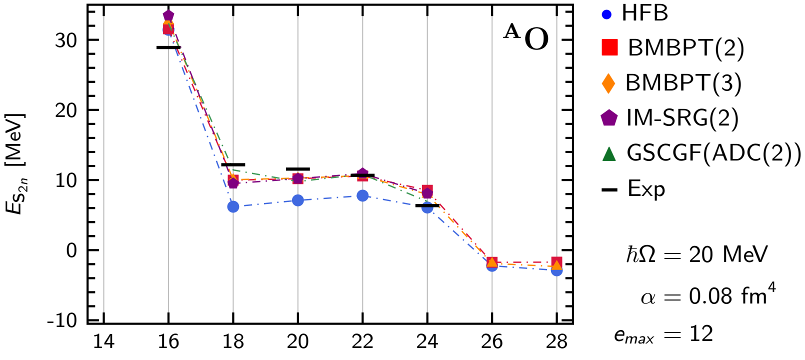
Oxygen chain calculations - Comparisons



[Arthuis, Tichai, Hergert, Roth and Duguet, in prep.]

- Consistent with different non-perturbative methods
- **Comparable accuracy within 1-5 % of computing time**
- Computational scaling independent of system size

Oxygen chain calculations - S2n



[Arthuis, Tichai, Hergert, Roth and Duguet, in prep.]

$\hbar\Omega = 20$ MeV

$\alpha = 0.08$ fm^4

$e_{max} = 12$

$E_{3max} = 14$

- **Very good agreement with state-of-the-art approaches**
- Reproduction of experimentally observed shell gaps
- Little overall effect of particle-number breaking (similar to GGF)
- **Particle-number restoration could impact near magic numbers**

Conclusion

- BMBPT diagrams now generated automatically
  - ✔ Fast and error-safe
  - ✔ No intrinsic upper limit on the order
- BMBPT analytical expressions automatically derived to all order as well
  - ✔ Feynman and Goldstone expressions for all diagrams
  - ✔ Order 4 to be implemented in BMBPT code in near future
- Project still moving on
  - ◇ Code to be published
  - ◇ Open to collaborations regarding other diagrammatic methods
- Numerical implementation of BMBPT(2) and BMBPT(3)
  - ✔ Very low-cost correlated method
  - ✔ Competes with state-of-the-art *ab initio* methods

Perspectives

- Extend the scope of ADG
  - Gorkov SCGF
  - Off-diagonal BMBPT
- Extend the scope of diagonal BMBPT
  - Excited states and new observables
  - Developments used in parallel in future BCC implementation
- Move towards symmetry-restored BMBPT
  - Extensive work on the theory
  - Automated diagram generation and derivation
  - Implementation in the BMBPT numerical code

Our collaborators

BMBPT Project

P. Arthuis
T. Duguet
J.-P. Ebran
A. Tichai

R. Roth

H. Hergert

On broader aspects

M. Drissi
J. Ripoche

R.-D. Lasseri