Recent developments in Bogoliubov Many-Body Perturbation Theory

Pierre Arthuis

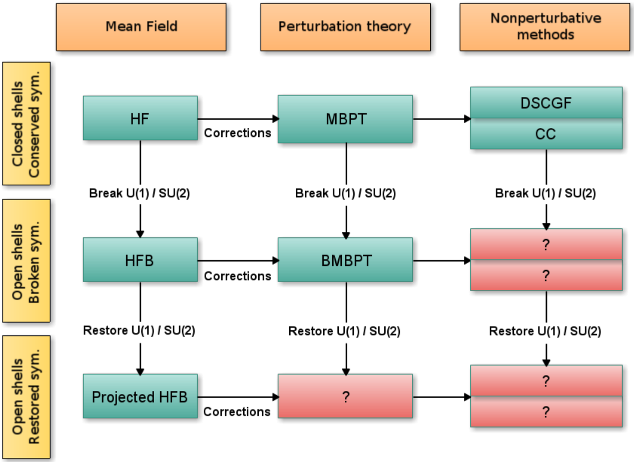
PHENIICS Fest
Université Paris-Saclay - May 30th 2017

Outline

Outline

1. Motivation
2. On Bogoliubov Many-Body Perturbation Theory
3. Recent progress

Quantum many-body methods



Expansion methods around unperturbed product state

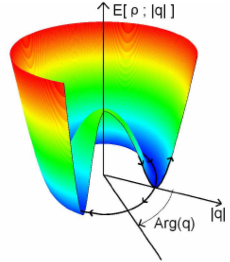
On symmetry breaking

Symmetry breaking helps incorporating non-dynamical correlations:

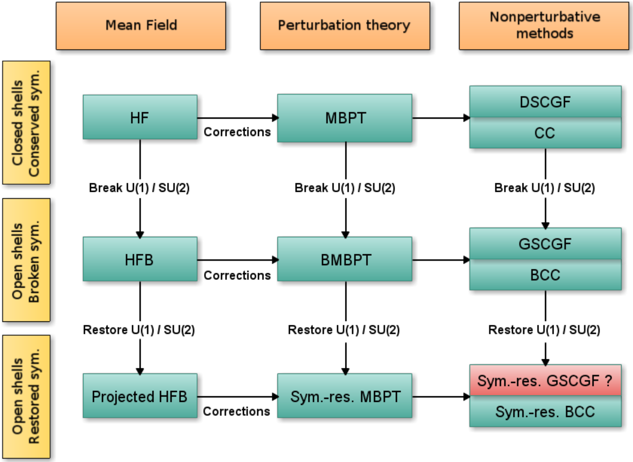
- Superfluid character: $U(1)$ (particle number)
- Deformations: $SU(2)$ (angular momentum)

But nuclei carry good quantum numbers (e.g. number of particles)

$\Rightarrow$ Symmetries must eventually be restored

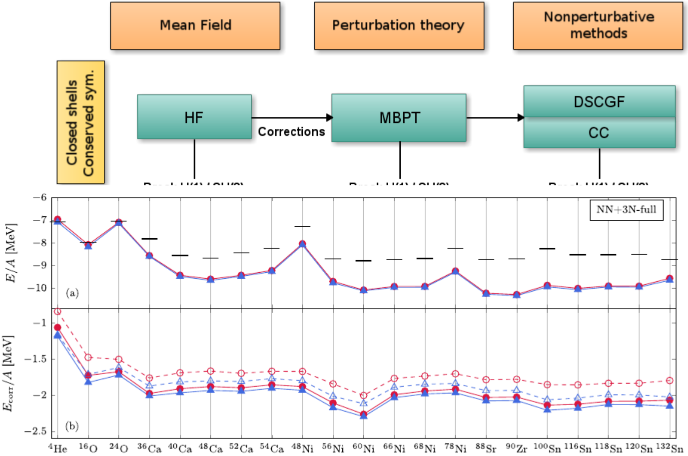


Quantum many-body methods



MBPT: Recently (re)implemented with SRG-evolved H [Tichai *et al.* 2016]
GSCGF, BCC: Recently proposed and implemented [Somà *et al.* 2011, Signoracci *et al.* 2014]
Sym.-res. BCC & sym.-res. BMBPT: Recently proposed [Duguet 2015, Duguet & Signoracci 2016]

Quantum many-body methods



MBPT: Recently (re)implemented with SRG-evolved H [Tichai *et al.* 2016]

GSCGF, BCC: Recently proposed and implemented [Somà *et al.* 2011, Signoracci *et al.* 2014]

Sym.-res. BCC & sym.-res. BMBPT: Recently proposed [Duguet 2015, Duguet & Signoracci 2016]

Outline

1. Motivation
2. On Bogoliubov Many-Body Perturbation Theory
3. Recent progress

Bogoliubov Many-Body Perturbation Theory

1. Use a Bogoliubov vacuum $|\Phi\rangle$ with $\beta_k|\Phi\rangle = 0$ for all k

2. Define grand potential operator Ω from chiral interaction

$$\Omega \equiv H - \lambda A$$

   then normal-order and split: $\Omega = \Omega_0 + \Omega_1$

3. Define evolved state in imaginary time

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

4. Expand and truncate the grand potential kernel $\Omega(\tau) \equiv \langle\Psi(\tau)|\Omega|\Phi\rangle$...
   ...and the norm kernel $N(\tau) \equiv \langle\Psi(\tau)|\Phi\rangle$

5. Extract ground state energy via

$$\mathrm{E}_0 = \lim_{\tau\to\infty} \frac{\Omega(\tau)}{N(\tau)} = \lim_{\tau\to\infty} \omega(\tau)$$

Expansion of the grand potential kernel

Inserting the operator Ω at time 0 and expanding

$$
\begin{aligned}
\mathrm{E}_0 &= \lim_{\tau\to\infty} \frac{\langle \Psi(\tau)|\Omega|\Phi\rangle}{\langle \Psi(\tau)|\Phi\rangle} \\
&= \langle \Phi| \Big\{ \Omega(0) - \int_0^{\infty} d\tau_1 \mathrm{T}\left[\Omega_1(\tau_1)\Omega(0)\right] \\
&\qquad + \frac{1}{2!}\int_0^{\infty} d\tau_1 d\tau_2 \mathrm{T}\left[\Omega_1(\tau_1)\,\Omega_1(\tau_2)\Omega(0)\right] + \ldots \Big\} |\Phi\rangle_c
\end{aligned}
$$

Then expressing the grand potential in the qp basis

$$
\Omega = \Omega^{00} + \frac{1}{1!}\sum_{k_1 k_2} \Omega^{11}_{k_1 k_2} \beta^\dagger_{k_1}\beta_{k_2} + \frac{1}{2!}\sum_{k_1 k_2} \left\{ \Omega^{20}_{k_1 k_2}\beta^\dagger_{k_1}\beta^\dagger_{k_2} + \Omega^{02}_{k_1 k_2}\beta_{k_2}\beta_{k_1} \right\} + \ldots
$$

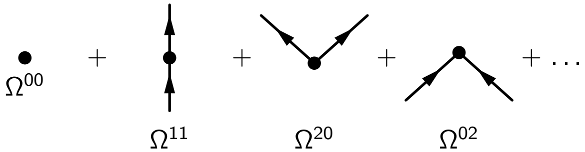
Expansion of the grand potential kernel

$$
\begin{aligned}
\mathrm{E}_0 = \sum_{p=0}^{\infty} \frac{(-1)^p}{p!} \sum_{\substack{i_0+j_0=2,4 \\ \vdots \\ i_p+j_p=2,4}} \int_0^{\infty} d\tau_1 \dots d\tau_p \\
\times \sum_{\substack{k_1 \dots k_{i_1} \\ k_{i_1+1} \dots k_{i_1+j_1} \\ \vdots \\ l_1 \dots l_{i_p} \\ l_{i_p+1} \dots l_{i_p+j_p}}} \frac{\Omega^{i_1 j_1}_{k_1 \dots k_{i_1} k_{i_1+1} \dots k_{i_1+j_1}}}{(i_1)!(j_1)!} \dots \frac{\Omega^{i_p j_p}_{l_1 \dots l_{i_p} l_{i_p+1} \dots l_{i_p+j_p}}}{(i_p)!(j_p)!} \frac{\Omega^{i_0 j_0}_{m_1 \dots m_{i_0} m_{i_0+1} \dots m_{i_0+j_0}}}{(i_0)!(j_0)!} \\
\times \langle \Phi | \mathrm{T} \Big[\beta^{\dagger}_{k_1}(\tau_1) \dots \beta^{\dagger}_{k_{i_1}}(\tau_1) \beta_{k_{i_1+j_1}}(\tau_1) \dots \beta_{k_{i_1+1}}(\tau_1) \dots \\
\dots \beta^{\dagger}_{l_1}(\tau_p) \dots \beta^{\dagger}_{l_{i_p}}(\tau_p) \beta_{l_{i_p+j_p}}(\tau_p) \dots \beta_{l_{i_p+1}}(\tau_p) \\
\times \beta^{\dagger}_{m_1}(0) \dots \beta^{\dagger}_{m_{i_0}}(0) \beta_{m_{i_0+j_0}}(0) \dots \beta_{m_{i_0+1}}(0) \Big] | \Phi \rangle_c
\end{aligned}
$$

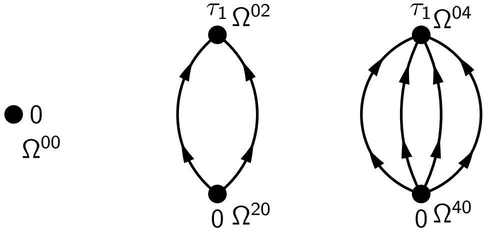
All contributions computable algebraically and diagramatically

First- and second-order diagrams

Diagrammatic representation of the grand potential Ω

$\Omega =$ 

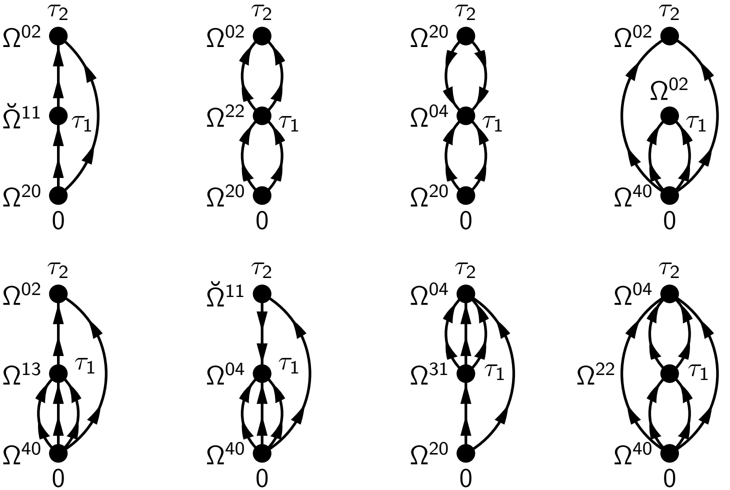
Extracting and applying diagrammatic rules

$\mathrm{E}_0^{(1+2)} =$ 

Outline

1. Motivation
2. On Bogoliubov Many-Body Perturbation Theory
3. Recent progress

Third-order diagrams



Derivation of all diagrams up to third order

Validation of formal derivation

BMBPT must match standard MBPT in Slater determinant limit

$\rightarrow$ Matching must be true at each order

$\rightarrow$ Proof of consistent formalism for BMBPT

Validation of formal derivation

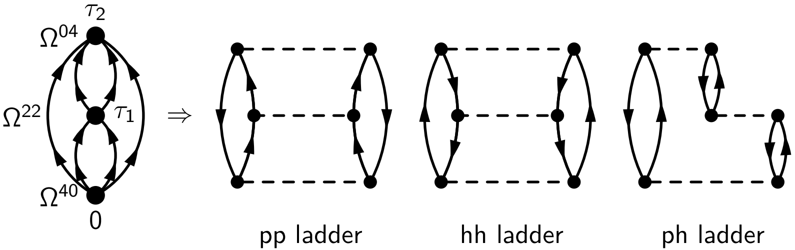
BMBPT must match standard MBPT in Slater determinant limit

$\rightarrow$ Matching must be true at each order

$\rightarrow$ Proof of consistent formalism for BMBPT

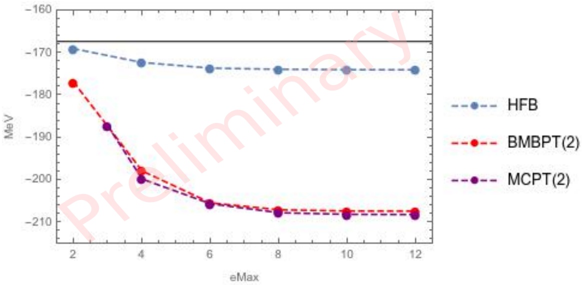
BMBPT(3) diagrams match MBPT(3) ones exactly

Canonical HF-MBPT diagrams were recovered from only one BMBPT



Proof of principle calculations

First BMBPT(2) proof of principle calculation of $^{20}\mathrm{O}$:



[Arthuis, Tichai, Hergert, Roth and Duguet, in prep.]
[Tichai, Gebrerufael and Roth, arXiv:1703.05664 (2017)]

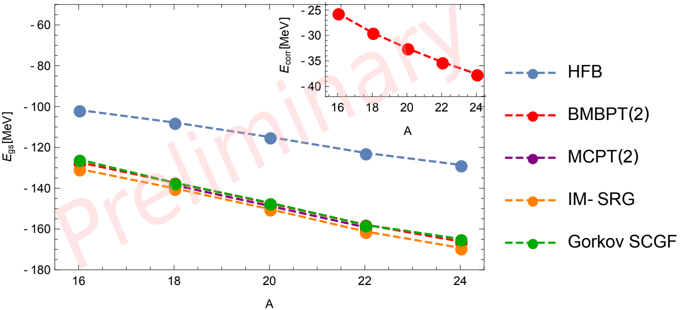
using NN SRG-evolved chiral interaction

On MCPT:

- Multi-configurational MBPT
- Alternative method for open-shell nuclei

Isotopic chains calculations at second order

First BMBPT(2) calculations on O, Ca and Sn isotopic chains



[Arthuis, Tichai, Hergert, Roth and Duguet, in prep.]
[Tichai, Gebrerufael and Roth, arXiv:1703.05664 (2017)]
[Hergert, Phys. Scripta 92 (2017)]
[Cipollone, Barbieri and Navràtil, Phys. Rev. C (2015)]

using NN and 3N SRG-evolved chiral interaction

Same chains under investigation at third order at the moment

Numerical derivations of higher orders

Use of diagrammatic rules and graph theory

$\Rightarrow$ Produce graphs and their expressions numerically:

- Produce higher orders diagrams
  - 59 diagrams at order 4
  - 568 diagrams at order 5

- Extend to three-body diagrams
  - 15 diagrams at order 3
  - 337 diagrams at order 4
  - 10 148 diagrams at order 5

Prospects

- Go up to fourth order
  - → Even better than other *ab initio* methods?
  - → Test for computational cost
- Push BMBPT to heavier nuclei
  - → Can go further than other *ab initio* methods
  - → Good test for the computational cost
- Implement particle-number restored BMBPT for the first time
  - → Required for precise study of open-shell nuclei
  - → Proof of concept of symmetry-restored BMBPT / BCC
- *Ab initio* driven EDF method [T. Duguet et al. (2015)]
  - → Safe/correlated/improvable off-diagonal EDF kernels
  - → Based on PNR-BMBPT

Conclusion

- *Ab initio* expansion methods are a powerful framework
  - ✔ Rigorous approach to the many-body problem
  - ✘ Computationally intensive (polynomial scaling)
  - ✘ Cannot describe the whole nuclear chart
- Many-Body Perturbation Theory and its daughters are one of them
  - ✔ Computationally friendly
  - ✔ Potentially as precise as others when using SRG-evolved H
- BMBPT has been formulated and is being implemented
  - ✔ First derivation and calculations up to third order
  - ✔ Appropriate framework to tackle open-shell nuclei
  - ✔ Systematic studies at third and fourth order to come
- Symmetry-restored BMBPT is the next step

Our collaborators

BMBPT Project

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

A. Tichai
R. Roth

H. Hergert

On broader aspects

M. Drissi
J. Ripoche

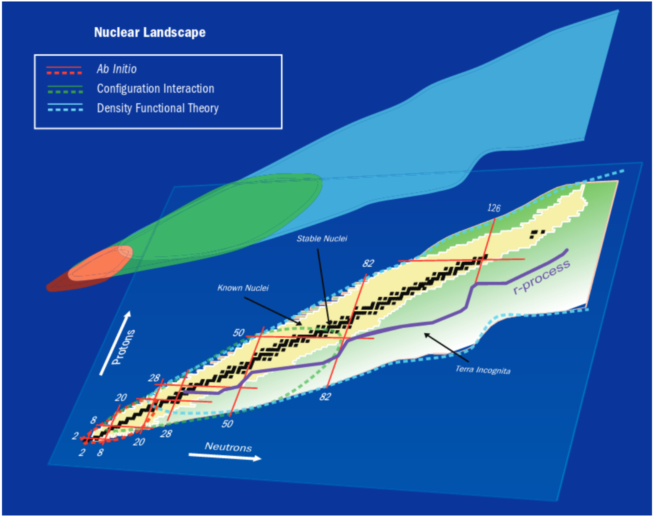
R. Lasseri

In case of emergency

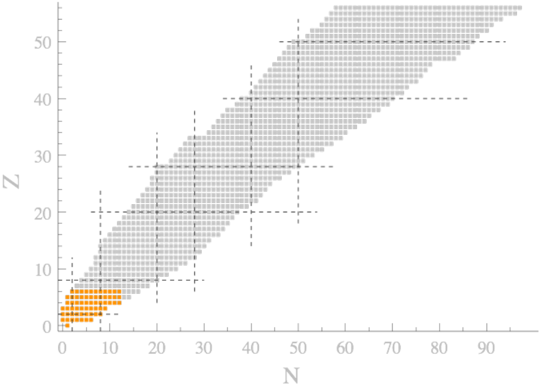
Backup slides!

On Nuclear Structure Theory

Different methods to treat the whole nuclear chart:



Evolution of the ab initio reach

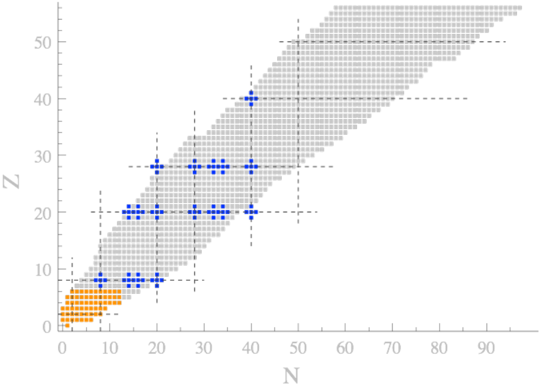


Courtesy of V. Soma, T. Duguet

"Exact" *ab initio* methods

- Since the 80's
- GFMC, NCSM, FY

Evolution of the ab initio reach

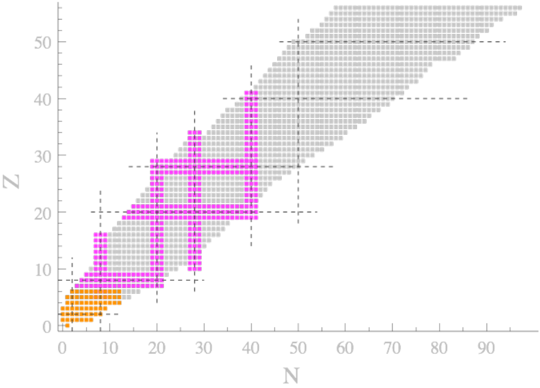


Courtesy of V. Soma, T. Duguet

Ab initio approaches for closed-shell nuclei

- Since the 2000's
- DSCGF, CC, IMSRG

Evolution of the ab initio reach

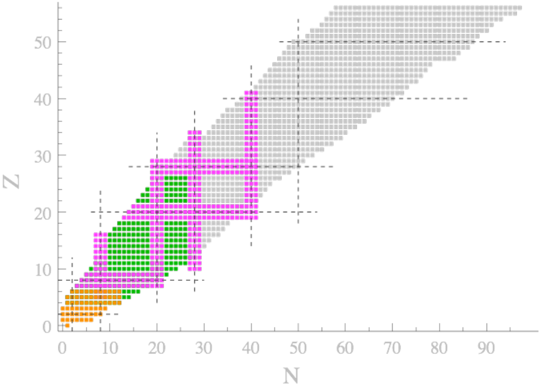


Courtesy of V. Soma, T. Duguet

Non-perturbative *ab initio* approaches for open-shell nuclei

- Since the 2010's
- GSCGF, BCC, MR-IMSRG

Evolution of the ab initio reach



Courtesy of V. Soma, T. Duguet

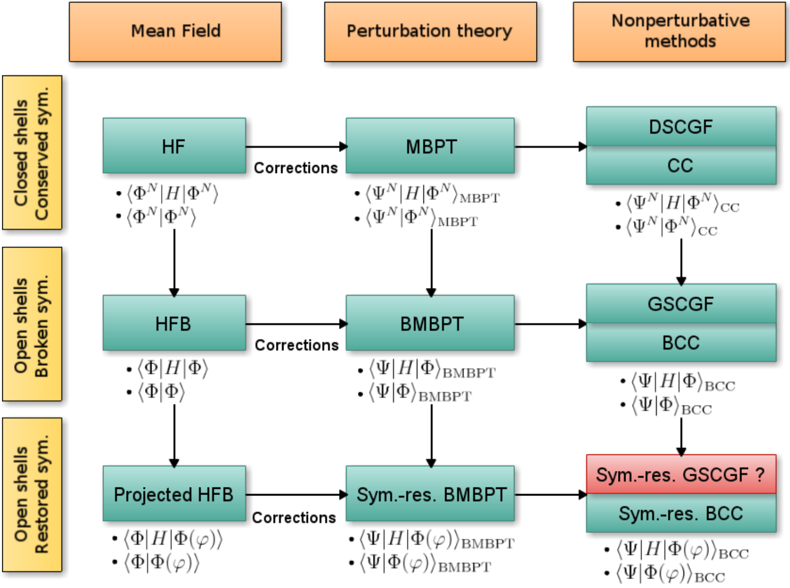
Ab initio shell model

- Since 2014
- Effective interaction via CC/IMSRG

What makes a method *ab initio*

1. Consider point-like nucleons as appropriate degrees of freedom
2. Use interactions rooted in underlying theory (i.e. QCD)
3. Expand the many-body Schrödinger equation systematically
4. Truncate at a given order and solve using computational methods
5. Estimate systematic error

Nuclear many-body methods

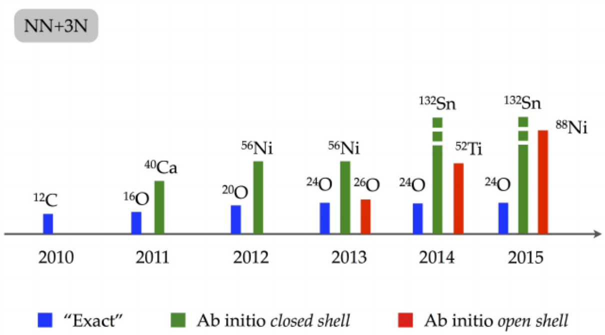


Status and perspectives of *ab initio* methods

Recent developments: symmetry breaking and restoration
$\rightarrow$ Access to open-shell nuclei

Reach some isotopic chains for medium-mass nuclei

Systematic and predictive methods: estimates of theoretical error



Reminder on second quantization

Introduce creation a^+_μ and annihilation a_μ operators acting on $\mathcal{F}$ such that

- a_μ and a^+_μ are hermitian conjugate: $a^+_\mu = (a_\mu)^\dagger$

- a_μ annihilates a particle in state $|\mu\rangle$:

$$
\begin{aligned}
a_\mu \; : \; \mathcal{H}_N \; &\rightarrow \; \mathcal{H}_{N-1} \\
|\alpha\,\beta\ldots\rangle \; &\rightarrow \; a_\mu\,|\alpha\,\beta\ldots\rangle \equiv 0 \qquad \text{if } \mu \text{ is not initially occupied} \\
|\mu\,\beta\ldots\rangle \; &\rightarrow \; a_\mu\,|\mu\,\beta\ldots\rangle \equiv |\beta\ldots\rangle \; \text{if } \mu \text{ is initially occupied}
\end{aligned}
$$

- a^+_μ creates a particle in the state $|\mu\rangle$:

$$
\begin{aligned}
a^+_\mu \; : \; \mathcal{H}_N \; &\rightarrow \; \mathcal{H}_{N+1} \\
|\alpha\,\beta\ldots\rangle \; &\rightarrow \; a^+_\mu\,|\alpha\,\beta\ldots\rangle \equiv |\mu\,\alpha\,\beta\ldots\rangle \; \text{if } \mu \text{ is not initially occupied} \\
|\alpha\ldots\mu\,\ldots\rangle \; &\rightarrow \; a^+_\mu\,|\alpha\ldots\mu\,\ldots\rangle \equiv 0 \qquad \text{if } \mu \text{ is initially occupied}
\end{aligned}
$$

- a_μ and a^+_μ obey anticommutation relationships

$$
\{a^+_\mu, a^+_\nu\} = 0 \quad , \quad \{a_\mu, a_\nu\} = 0 \quad , \quad \{a_\mu, a^+_\nu\} = \delta_{\mu\nu} \; .
$$

Use of the creation and annihilation operators

- Create Slater determinants:

$$|\alpha\,\beta\ldots\rangle = a^+_\alpha\, a^+_\beta \ldots\, |0\rangle$$

- Write operators:
  One-body:

$$F = \sum_{\alpha\beta} f_{\alpha\beta}\, a^+_\alpha\, a_\beta$$

  using matrix elements of the operator:

$$f_{\alpha\beta} \equiv \langle i:\alpha|f(i)|i:\beta\rangle$$

  And two-body:

$$G = \frac{1}{2}\sum_{\alpha\beta\gamma\delta} g_{\alpha\beta\gamma\delta}\; a^+_\alpha\, a^+_\beta\, a_\delta\, a_\gamma$$

  with matrix elements

$$g_{\alpha\beta\gamma\delta} \equiv \langle i:\alpha\,;j:\beta|g(i,j)|i:\gamma\,;j:\delta\rangle$$

Bogoliubov transformation

Bogoliubov transformation connects qp operators $\{\beta_k; \beta_k^\dagger\}$ to particle ones:

$$\beta_k = \sum_p U_{pk}^* \, a_p + V_{pk}^* \, a_p^\dagger$$

$$\beta_k^\dagger = \sum_p U_{pk} \, a_p^\dagger + V_{pk} \, a_p$$

They obey anticommutation rules

$$\{\beta_{k_1}, \beta_{k_2}\} = 0 \quad , \quad \{\beta_{k_1}^\dagger, \beta_{k_2}^\dagger\} = 0 \quad , \quad \{\beta_{k_1}, \beta_{k_2}^\dagger\} = \delta_{k_1 k_2}$$

Bogoliubov transformation can be written in matrix form

$$\begin{pmatrix} \beta \\ \beta^\dagger \end{pmatrix} = W^\dagger \begin{pmatrix} a \\ a^\dagger \end{pmatrix}$$

where

$$W \equiv \begin{pmatrix} U & V^* \\ V & U^* \end{pmatrix}$$

On contractions

Contractions and normal product:

$$\overbracket{AB} \equiv AB - :AB: \quad \text{with} \quad \overbracket{AB} = \frac{\langle \Phi | AB | \Phi \rangle}{\langle \Phi | \Phi \rangle}$$

Elementary contractions:

$$\overbracket{\beta^+_\alpha \beta^+_\beta} = \frac{\langle \Phi | \beta^+_\alpha \beta^+_\beta | \Phi \rangle}{\langle \Phi | \Phi \rangle} = 0 \qquad \overbracket{\beta^+_\alpha \beta_\beta} = \frac{\langle \Phi | \beta^+_\alpha \beta_\beta | \Phi \rangle}{\langle \Phi | \Phi \rangle} = 0$$

$$\overbracket{\beta_\alpha \beta^+_\beta} = \frac{\langle \Phi | \beta_\alpha \beta^+_\beta | \Phi \rangle}{\langle \Phi | \Phi \rangle} = \delta_{\alpha\beta} \qquad \overbracket{\beta_\alpha \beta_\beta} = \frac{\langle \Phi | \beta_\alpha \beta_\beta | \Phi \rangle}{\langle \Phi | \Phi \rangle} = 0$$

Wick's theorem

$$
\begin{aligned}
ABCD\ldots YZ = {} & \overbracket{AB}\,\overbracket{CD}\ldots\overbracket{YZ} - \overbracket{AC}\,\overbracket{BD}\ldots\overbracket{YZ} + \overbracket{AD}\,\overbracket{BC}\ldots\overbracket{YZ} + \ldots \\
& + \overbracket{AB}\,\overbracket{CD}\ldots : YZ : - \overbracket{AC}\,\overbracket{BD}\ldots : YZ : + \overbracket{AD}\,\overbracket{BC}\ldots : YZ : + \ldots \\
& \vdots \\
& + \overbracket{AB} : CD \ldots YZ : - \overbracket{AC} : BD \ldots YZ : + \overbracket{AD} : BC \ldots YZ : + \ldots \\
& + : ABCD \ldots YZ :
\end{aligned}
$$

$$
\begin{aligned}
\beta^+_\alpha \beta^+_\beta \beta_\delta \beta_\gamma = {} & \overbracket{\beta^+_\alpha \beta^+_\beta}\,\overbracket{\beta_\delta \beta_\gamma} - \overbracket{\beta^+_\alpha \beta_\delta}\,\overbracket{\beta^+_\beta \beta_\gamma} + \overbracket{\beta^+_\alpha \beta_\gamma}\,\overbracket{\beta^+_\beta \beta_\delta} \\
& + \overbracket{\beta^+_\alpha \beta^+_\beta} : \beta_\delta \beta_\gamma : - \overbracket{\beta^+_\alpha \beta_\delta} : \beta^+_\beta \beta_\gamma : + \overbracket{\beta^+_\alpha \beta_\gamma} : \beta^+_\beta \beta_\delta : \\
& + \overbracket{\beta^+_\beta \beta_\delta} : \beta^+_\alpha \beta_\gamma : - \overbracket{\beta^+_\beta \beta_\gamma} : \beta^+_\alpha \beta_\delta : + \overbracket{\beta_\delta \beta_\gamma} : \beta^+_\alpha \beta^+_\beta : \\
& + : \beta^+_\alpha \beta^+_\beta \beta_\delta \beta_\gamma :
\end{aligned}
$$

Time-dependent kernels

Introduce time-dependent kernel for generic operator O

$$O(\tau) \equiv \langle \Psi(\tau) | O | \Phi \rangle$$

or other operator of interest

$$\begin{aligned} N(\tau) &\equiv \langle \Psi(\tau) | \mathbb{1} | \Phi \rangle \\ H(\tau) &\equiv \langle \Psi(\tau) | H | \Phi \rangle \\ A(\tau) &\equiv \langle \Psi(\tau) | A | \Phi \rangle \\ \Omega(\tau) &\equiv \langle \Psi(\tau) | \Omega | \Phi \rangle \end{aligned}$$

and reduced kernel

$$\mathcal{O}(\tau) \equiv \frac{O(\tau)}{N(\tau)}$$

Decomposition of the time-dependent kernels

Kernels can be decomposed as

$$N(\tau) = \sum_{\mathrm{A}\in\mathbb{N}} \sum_{\mu} e^{-\tau\Omega_\mu^\mathrm{A}} |\langle\Phi|\Psi_\mu^\mathrm{A}\rangle|^2$$

$$H(\tau) = \sum_{\mathrm{A}\in\mathbb{N}} \sum_{\mu} \mathrm{E}_\mu^\mathrm{A} \, e^{-\tau\Omega_\mu^\mathrm{A}} |\langle\Phi|\Psi_\mu^\mathrm{A}\rangle|^2$$

$$A(\tau) = \sum_{\mathrm{A}\in\mathbb{N}} \sum_{\mu} \mathrm{A} \, e^{-\tau\Omega_\mu^\mathrm{A}} |\langle\Phi|\Psi_\mu^\mathrm{A}\rangle|^2$$

$$\Omega(\tau) = \sum_{\mathrm{A}\in\mathbb{N}} \sum_{\mu} \Omega_\mu^\mathrm{A} \, e^{-\tau\Omega_\mu^\mathrm{A}} |\langle\Phi|\Psi_\mu^\mathrm{A}\rangle|^2$$

Large time limit of the kernels

Defining the large τ limit of a kernel via

$$O(\infty) \equiv \lim_{\tau \to \infty} O(\tau)$$

gives

$$\begin{aligned} N(\infty) &= e^{-\tau \Omega_0^{A_0}} |\langle \Phi | \Psi_0^{A_0} \rangle|^2 \\ H(\infty) &= E_0^{A_0} \, e^{-\tau \Omega_0^{A_0}} |\langle \Phi | \Psi_0^{A_0} \rangle|^2 \\ A(\infty) &= A_0 \, e^{-\tau \Omega_0^{A_0}} |\langle \Phi | \Psi_0^{A_0} \rangle|^2 \\ \Omega(\infty) &= \Omega_0^{A_0} \, e^{-\tau \Omega_0^{A_0}} |\langle \Phi | \Psi_0^{A_0} \rangle|^2 \end{aligned}$$

Extracting observables

From the previous relations,

$$H(\infty) = \mathrm{E}_0^{\mathrm{A}_0} \, N(\infty)$$
$$A(\infty) = \mathrm{A}_0 \, N(\infty)$$
$$\Omega(\infty) = \Omega_0^{\mathrm{A}_0} \, N(\infty)$$

Or directly from the reduced kernels

$$\mathcal{H}(\infty) = \mathrm{E}_0^{\mathrm{A}_0}$$
$$\mathcal{A}(\infty) = \mathrm{A}_0$$
$$\Omega_r(\infty) = \Omega_0^{\mathrm{A}_0}$$

Expansion of the norm kernel

Expanding norm kernel through the evolution operator:

$$
\begin{aligned}
N(\tau) &= \langle \Phi | \mathcal{U}(\tau) | \Phi \rangle \\
&= \langle \Phi | e^{-\tau \Omega_0} \mathcal{U}_1(t) | \Phi \rangle \\
&= e^{-\tau \Omega_{00}} \langle \Phi | \mathrm{T} e^{-\int_0^\tau dt \Omega_1(t)} | \Phi \rangle \\
&= e^{-\tau \Omega^{00}} \langle \Phi | \Big\{ 1 - \int_0^\tau d\tau_1 \Omega_1 \left(\tau_1 \right) \\
&\qquad + \frac{1}{2!} \int_0^\tau d\tau_1 d\tau_2 \mathrm{T} \left[\Omega_1 \left(\tau_1 \right) \Omega_1 \left(\tau_2 \right) \right] + \ldots \Big\} | \Phi \rangle
\end{aligned}
$$

Expansion of the norm kernel

Eventually, one gets

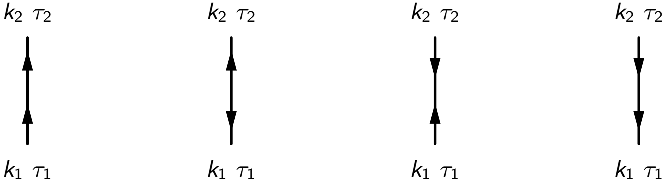
$$N(\tau) = e^{-\tau\Omega^{00}} \Big\{ \sum_{p=0}^{\infty} \frac{(-1)^p}{p!} \sum_{\substack{i_1+j_1=2,4 \\ \vdots \\ i_p+j_p=2,4}} \int_0^\tau d\tau_1 \ldots d\tau_p$$

$$\times \sum_{\substack{k_1\ldots k_{i_1} \\ k_{i_1+1}\ldots k_{i_1+j_1} \\ \vdots \\ l_1\ldots l_{i_p} \\ l_{i_p+1}\ldots l_{i_p+j_p}}} \frac{\Omega^{i_1 j_1}_{k_1\ldots k_{i_1}\, k_{i_1+1}\ldots k_{i_1+j_1}}}{(i_1)!(j_1)!} \ldots \frac{\Omega^{i_p j_p}_{l_1\ldots l_{i_p}\, l_{i_p+1}\ldots l_{i_p+j_p}}}{(i_p)!(j_p)!}$$

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

$$\ldots \beta^\dagger_{l_1}(\tau_p) \ldots \beta^\dagger_{l_{i_p}}(\tau_p)\, \beta_{l_{i_p+j_p}}(\tau_p) \ldots \beta_{l_{i_p+1}}(\tau_p) \Big] |\Phi\rangle \Big\}$$

BMBPT propagators



$G^{+-(0)}_{k_1 k_2}(\tau_1, \tau_2)$ $\quad$ $G^{--(0)}_{k_1 k_2}(\tau_1, \tau_2)$ $\quad$ $G^{++(0)}_{k_1 k_2}(\tau_1, \tau_2)$ $\quad$ $G^{-+(0)}_{k_1 k_2}(\tau_1, \tau_2)$

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

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

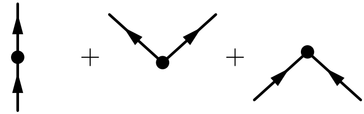
$$G^{++(0)}_{k_1 k_2}(\tau_1, \tau_2) \equiv \frac{\langle \Phi | \mathrm{T}[\beta^\dagger_{k_1}(\tau_1)\beta^\dagger_{k_2}(\tau_2)] | \Phi \rangle}{\langle \Phi | \Phi \rangle}$$

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

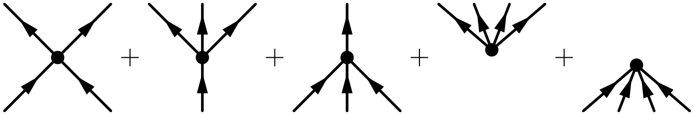
And grand potential vertices:

$\Omega^{[0]} =$

$\Omega^{[2]} =$



$\Omega^{[4]} =$



BMBPT diagrammatic rules

Then there are some rules to follow:

- Order $p \to p$ vertices $\Omega^{ij}(\tau_k)$, 1 $O^{ij}(0)$ connected via propagators
- A good labelling convention
- Make all possible contractions
- Keep only topologically distinct diagrams
- Sum all time labels from 0 to τ
- Sign factor $(-1)^{p+n_c}$
- Symmetry factors: equivalent lines and exchangeable vertices
- Sign factor linked to reading direction

BMBPT diagrammatic rules

Plus some selection rules:

- No anomalous propagators
  $\rightarrow$ Same number of creators and operators
- Propagators linking two same vertices
  $\rightarrow$ Same direction
- No contraction of a vertex on itself
- Propagators starting from vertex at time 0
  $\rightarrow$ Moving upward

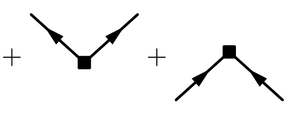
Generic operator vertices

Let us draw some operator vertices:

$O^{[0]} =$

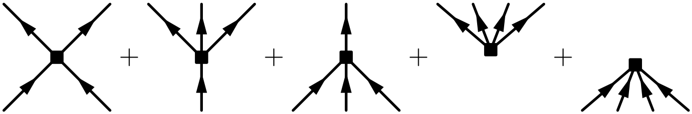
O^{00}

$O^{[2]} =$



O^{11} + O^{20} + O^{02}

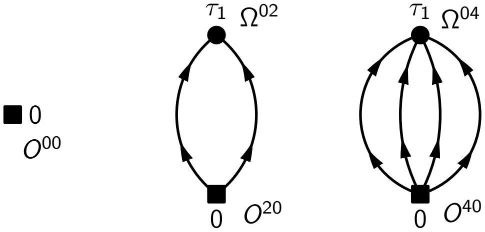
$O^{[4]} =$



O^{22} + O^{31} + O^{13} + O^{40} + O^{04}

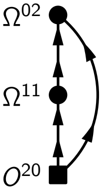
Zero- and first-order diagrams

Applying all the previous rules:

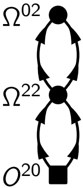


PO0.1 PO1.1 PO1.2

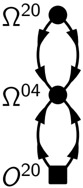
Second-order diagrams



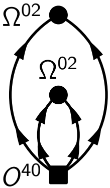
PO2.1



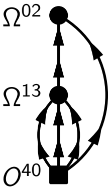
PO2.2



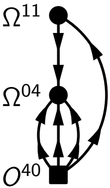
PO2.3



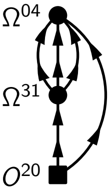
PO2.4



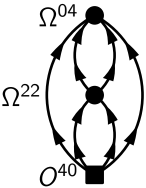
PO2.5



PO2.6



PO2.7



PO2.8

Slater determinant limit

Bogoliubov transformation simplifies to

$$
\begin{aligned}
a > \mathrm{A}_0 : \qquad & c_a = \beta_a, \quad c_a^\dagger = \beta_a^\dagger, \\
i \leq \mathrm{A}_0 : \qquad & c_i = \beta_i^\dagger, \quad c_i^\dagger = \beta_i.
\end{aligned}
$$

U and V matrix elements are

$$
\begin{aligned}
a > \mathrm{A}_0 : \qquad & V_{ak} = 0, \qquad U_{ak} = \delta_{ak}, \\
i \leq \mathrm{A}_0 : \qquad & V_{ik} = \delta_{ik}, \quad U_{ik} = 0.
\end{aligned}
$$

And density matrices are

$$
\begin{aligned}
a > \mathrm{A}_0 : \qquad & \rho_{ap} = 0, \qquad \kappa_{ap} = 0, \\
i \leq \mathrm{A}_0 : \qquad & \rho_{ip} = \delta_{ip}, \quad \kappa_{ip} = 0,
\end{aligned}
$$

What is graph theory?

Obviously: The study of graphs and their properties

Graphs are made of:

- Nodes ($\leftrightarrow$ our vertices)
- Edges connecting the nodes ($\leftrightarrow$ our propagators)

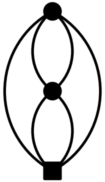
Applied to various domains and situations:

- Search engines
- Task attribution
- Energy grid
- Quantum mechanics
- And many more...

Graphs and adjacency matrix

a_{ij} indicate the number of edges connecting two nodes

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



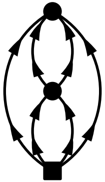
$\Rightarrow$ Exhibit symmetry properties

$\Rightarrow$ Do not distinguish between directed and undirected diagrams

Graphs and oriented adjacency matrix

$\tilde{a}_{ij}$ indicate the number of edges going from node i to node j

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



$\Rightarrow$ No such symmetry properties

$\Rightarrow$ Carry more detailed information for directed graphs

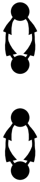
Symmetry under the exchange of two vertices

$\Rightarrow$ Symmetry under simultaneous exchange associated rows and columns

Disconnected diagram

$\Rightarrow$ Matrix recastable as block-matrix

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



Graphs and Python

NetworkX: A Python package for graph theory

- Create all kinds of graphs
- Extract adjacency matrices and all kinds of information
- Perform all sorts of operations on the graphs
- Adress some specific problems solved with graphs

Constraints from the diagrammatic rules

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

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

No self-contraction

$\Rightarrow$ Every diagonal element is zero

No loop between two vertices

$\Rightarrow$ Either a_{ij} or a_{ji} is zero

Every propagator coming out of the operator vertex goes upward

$\Rightarrow$ First column of the matrix is zero

Generate all adjacency matrices: Going brute force

- First test all possible values for first element

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 & 2 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \rightarrow \dots$$

- Then take output matrices and do the same for second element

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 & 1 & 2 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \rightarrow \dots$$

$\Rightarrow$ Produces all possible matrices

Test afterwards to exclude "unphysical" matrices

$\Rightarrow$ Time and computer memory wasted

Generate all adjacency matrices: A smarter way

To avoid generating too many matrices:

- Fill the matrices "vertex-wise"
- Leave first column blank
- Iterate on a_{ij} only if a_{ji} is zero
- Check the degree of each vertex before moving on

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 & a_{12} & a_{13} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 & a_{12} & a_{13} \\ 0 & 0 & a_{23} \\ 0 & a_{32} & 0 \end{pmatrix}$$

Some additional tests

Now, use Python / NetworkX to avoid:

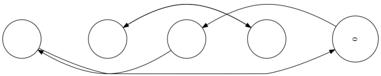
- Matrices appearing twice
- Matrices associated to vanishing graphs (e.g. loop between a set of vertices)
- Matrices associated to topologicaly identical diagrams

$\Rightarrow$ You're good to go!

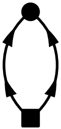
How to draw these nice graphs

Different options:

- Directly from NetworkX using dot



- Using the feynmp LaTeX instructions:
    - Use NetworkX functions to get useful graph structure info
    - Have your code write the feynmp instructions in your .tex file

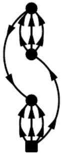


How to obtain their expression without any work

- Extract graph structure info as well
  $\rightarrow$ Possible to associate labels with vertices, propagators, etc.
  $\rightarrow$ Can use tests on subgraphs, in- and out-degree, topological sorts...

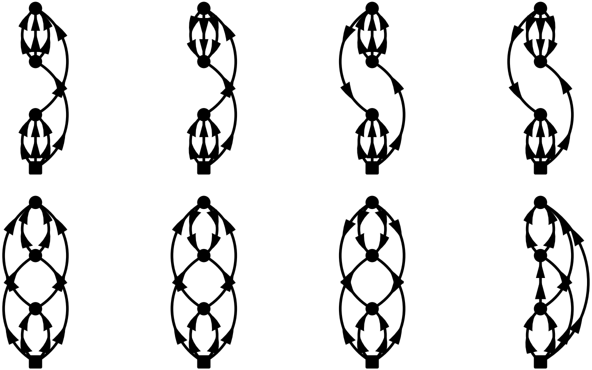
- Have your code write the corresponding equations in your .tex file

$$\frac{-(-1)^3}{(3!)^2} \sum_{k_i} \frac{O^{40}_{k_1k_2k_3k_4}\Omega^{04}_{k_1k_2k_3k_8}\Omega^{31}_{k_5k_6k_7k_4}\Omega^{13}_{k_8k_5k_6k_7}}{(+E_{k_1}+E_{k_2}+E_{k_3}+E_{k_4})(+E_{k_1}+E_{k_2}+E_{k_3}+E_{k_5}+E_{k_6}+E_{k_7})(+E_{k_1}+E_{k_2}+E_{k_3}+E_{k_8})}$$



Time to cook some diagrams

After having the code to run at order 4, obtain...



...and 388 others!

More than 10 000 at order 5