Solving the homogenous Bethe-Salpeter equation with a quantum annealer

Filippo Fornetti (Università degli Studi di Perugia, INFN, Sezione di Perugia)
Alex Gnech (ECT*)
Tobias Frederico (ITA Brasil)
Francesco Pederiva (Università di Trento, INFN-TIFPA)
Matteo Rinaldi (INFN, Sezione di Perugia)
Alessandro Roggero (Università di Trento, INFN-TIFPA)
Giovanni Salmè (INFN, Sezione di Roma)
Sergio Scopetta (Università degli Studi di Perugia, INFN, Sezione di Perugia)
Michele Viviani (INFN, Sezione di Pisa)

Based on

T. Frederico, G. Salmè, and M. Viviani,
"Quantitative studies of the homogeneous Bethe-Salpeter equation in Minkowski space",
Phys. Rev. D 89, 016010 (2014)

B. Krakoff, S. M. Mniszewski and C. F. A. Negre,
"A QUBO algorithm to compute eigenvectors of symmetric matrices",
arXiv:2104.11 (2021)

S. Alliney, F. Laudiero and M. Savoia,
"A variational technique for the computation of the vibration frequencies of mechanical systems governed by nonsymmetric matrices",
Applied mathematical modelling 16, 148 (1992)

M.Ila and M.Savage
"Basic elements for simulations of standard-model physics with quantum annealers: Multigrid and clock states",
Phys.Rev.A 106 052605 (2022)

A.Gnech, F.F., T.Frederico, F.Pederiva, M.Rinaldi, A.Roggero, G.Salmè, S.Scopetta and M.Viviani
"Solving the homogeneous Bethe-Salpeter equation with a quantum annealer",
In preparation

Outline

- The Physics problem
- Symmetric Standard Eigenvalue Problem (Krakoff et al.)
- Non symmetric Generalized Eigenvalue Problem
- Quantum Annealing
- Results

Introduction to the physics problem

Our aim is to apply a variational method, tailored to a Quantum Annealer, for solving the **homogeneous Bethe-Salpeter equation (hBSE)** in Minkowski momentum-space

[E.E Salpeter and H.A. Bethe, Phys. Rev. 84, 1232 (195)]

$$\phi_b(k,p) = G_0^{(12)}(k,p) \int \frac{d^4k'}{(2\pi)^4} i\mathcal{K}(k,k',p)\phi_b(k',p)$$

The solution of the **hBSE** is the **Bethe-Saltpeter amplitude**, i.e. the residue of the relevant Green's function at the bound-state mass pole

$$\phi_b(x_1,x_2,p) = < 0 \;\; T\{\varphi(x_1)\varphi(x_2)\} \;\; p >$$

Directly related to the 3-leg vertex of the bound state

The simplest hBSE is the one describing a **bound system** composed by **two massive scalars** interacting through the **exchange of a massive boson**. This problem has been solved in **[T.Frederico, G.Salmè, and M.Vivinai, Phys. Rev. D 89, 016010 (2014)]**

Nakanishi Integral Rapresentation (NIR)

[T.Frederico, G.Salmè and M.Viviani, Phys. Rev. D 89, 016010 (2014)]

One can write the BS amplitude in terms of

(i) **Numerator:** a non singular weight function of real variables, the **Nakanishi weight function (NWF)**

(ii) Denominator: analytic structure

$$\phi_b(p,k) = i\int_{-1}^{1} dz' \int_0^{\infty} d\gamma' \frac{g_b(\gamma', z'; \kappa^2)}{[\gamma' + \kappa^2 - k^2 - p\cdot k z' - i\epsilon]^{2+n}}$$

Where $\kappa^2 = m^2 - \frac{M^2}{4} > 0$ for bound states

Projection on the Light-cone hypersurface

[T.Frederico, G.Salmè and M.Viviani, Phys. Rev. D 89, 016010 (2014)]

It is useful to introduce the **light-front variables**: $k = (k^-, \tilde{\mathbf{k}})$, with $\tilde{k} = (k^+, \mathbf{k}_\perp)$ and $k^\pm = k^0 \pm k^3$. **hBSE** transformed into an **integral equation** for the NWF

The new kernel V_b^{LF} (Nakanishi kernel) is related to the BS kernel $i\mathscr{K}$. To obtain a GEVP we need to use the Ladder Approximation for the kernel:

$$i\mathscr{K}(k, k_i) = \frac{i(-ig)^2}{k - k_i^2 - \mu^2 + i\epsilon}$$

Integral equation for the NWF:

$$\int_0^\infty d\gamma' \frac{g_b(\gamma', z'; \kappa^2)}{[\gamma' + \gamma + z^2 m^2 + (1 - z^2)\kappa^2 - i\epsilon]^2} = \int_0^\infty \int_{-1}^1 dz' \, V_b^{LF}(\gamma, z; \gamma', z') \, g_b(\gamma', z'; \kappa^2)$$

hBSE as a Generalized Eigenvalue Problem (GEVP)

[T.Frederico, G.Salmè and M.Viviani, Phys. Rev. D 89, 016010 (2014)]

The NWF can be expanded in orthonormal basis. In particular one can use:

(i) Gegenbauer polynomials

(ii) Laguerre polynomials

$$g_b^{(Ld)}(\gamma, z; \kappa^2) = \sum_{l=0}^{N_z} \sum_{j=0}^{N_\gamma} c_{lj}(\kappa^2) G_l(z) \mathscr{L}_j(\gamma)$$

Finally, inserting the expansion of the NWF into the integral equation we obtain a GEVP:

$$A\,\mathbf{v}_i = \lambda_i\, B\,\mathbf{v}_i$$

The nxn ($n = N_z \times N_\gamma$) **real** matrices A (which contains the effect of the interacting kernel) and B. **A is non-symmetric and B is symmetric**

The eigenvectors $\mathbf{v}_i$ contain the coefficients $c_{lj}(\kappa^2)$

The eigenvalues are $\lambda_i = \frac{1}{\alpha_i}$, where $\alpha_i = \frac{g_i^2}{32\pi^2}$,with g_i the **coupling constant**.

Symmetric Eigenvalue Problem

We need to solve the GEVP through a **variational problem** since the Quantum Annealer (QA) can solve Quadratic Uncostrained Binary Optimization (QUBO) problems:

We started from a standard symmetric eigenvalue problem:

$$A\ \mathbf{v}_i = \lambda_i\ \mathbf{v}_i \qquad A = A^T$$

We need to minimize the **Rayleigh- Ritz quotient:**

$$R[\mathbf{v}] = \frac{\mathbf{v}^T A \mathbf{v}}{\mathbf{v}^T \mathbf{v}}$$

Algorithm I

[B. Krakoff, S. M. Mniszewski, and C. F. A. Negre, arXiv:2104.11 (2021)]

First we need to rewrite the minimization of the Rayleigh-Ritz quotient in a **quadratic form**:

$$\mathbf{v}_0 = \arg\min F[\mathbf{v}] = \arg\min\{\mathbf{v}^T[A - \lambda_g I]\mathbf{v}\} = \arg\min \mathbf{v}^T H \mathbf{v}$$

We need to add this Lagrangian multiplier to avoid the null solution

Set λ_g **near but higher** the lowest eigenvalue

Calculate the minimum eigenvalue $\lambda_0 = R[\mathbf{v_0}]$

Replace λ_g with λ_0

Formulate the problem in a QUBO form

The **QA** is designed to deal with transverse-Ising model and **QUBO** problems in an hypercube $C_{n,b} = [0,1]^{n*b}$

$$\mathbf{x}_0 = \arg \min_{\mathbf{x} \in C_{n,b}} \mathbf{x}^T Q \mathbf{x}$$

We need to approximate our **quadratic form** into a **QUBO** one, rewriting the matrices elements (that are real in our case) in a **binary basis**

$$\mathbf{v}^T H \mathbf{v} = \sum_{\alpha,\beta=1}^{n} v^{\alpha} H_{\alpha}^{\beta} v_{\beta} \simeq \sum_{\alpha,\beta=1}^{n} \sum_{ij=1}^{b} q^{i,\alpha} Q_{i,\alpha}^{j,\beta} q_{j,\beta} = \mathbf{x}^T Q \mathbf{x}$$

$$Q_{i,\alpha}^{j,\beta} = 2^{i+j-2b} (-1)^{\delta_{ib}+\delta_{jb}} H_{\alpha}^{\beta}$$

$$\mathbf{x} = (q_{1,1}, \ldots, q_{b,1}, \ldots, q_{1,n}, \ldots, q_{b,n}) \in C_{n,b}$$

$$v^{\alpha} = -q^{b,\alpha} + \sum_{i=1}^{b-1} \frac{q^{i,\alpha}}{2^{b-1}} \qquad \mathbf{v} = (v_1, v_2, \ldots, v_n) \in [-1,1[^n$$

Algorithm II: step descendent

At the **end of the first part** of the algorithm we find a solution $\mathbf{v}_{(z=0)}$

Second part: iterative algorithm that founds a new solution $\mathbf{v}_{(z)}$ for each step z, until $2^{-z} < \epsilon$. The function can be expanded around this solution:

$$F[\mathbf{v}] = F[\mathbf{v}_{z=0}] + \nabla_{\mathbf{v}} F[\mathbf{v}]\Big|_{\mathbf{v}_{(z)}} \cdot \delta + \delta^T \frac{H}{2} \delta$$

The new function to be minimize is a quadratic form in δ: $G[\delta]$

$$Q_{i,\alpha}^{j,\beta} = 2^{i+j-2b-2z}(-1)^{\delta_{ib}+\delta_{jb}} H_{\beta}^{\alpha} + 2\delta_{\beta\alpha}\delta_{ij}(-1)^{\delta_{ib}} \sum_{\gamma} v_{(z)}^{\gamma} H_{\gamma}^{\alpha}$$

$$\delta^{\alpha} = -q^{b,\alpha} + \sum_{i=1}^{b-1} \frac{q^{i,\alpha}}{2^{b-i+z}} \qquad v_{(z+1)}^{\alpha} = v_{z}^{\alpha} + 2^{-z}\delta^{\alpha}$$

[M.Ila and M.Savage, Phys.Rev.A 106 052605 (2022)]

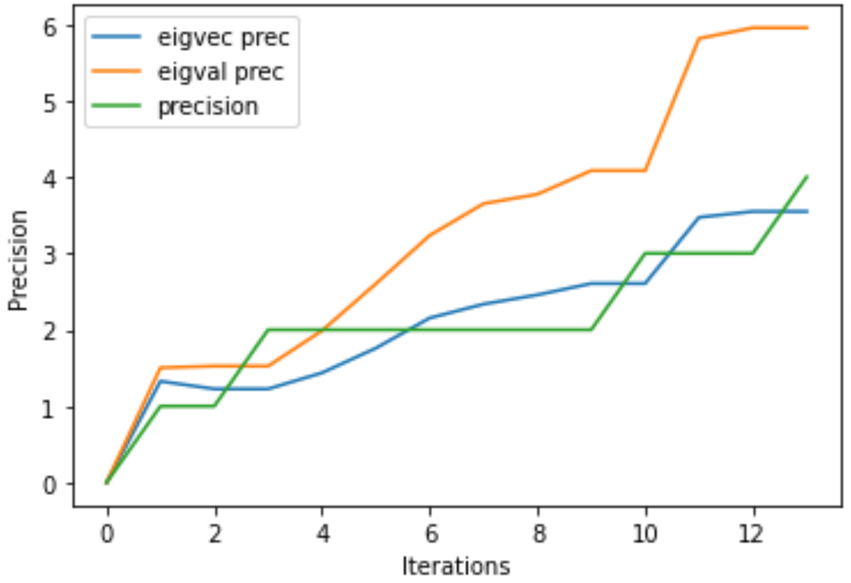
Results for symmetric standard eigenvalue problem

The results obtained in [B. Krakoff, S. M. Mniszewski and C. F. A. Negre, arXiv:2104.11 (2021)] are well reproduced for symmetric standard eigenvalue problem

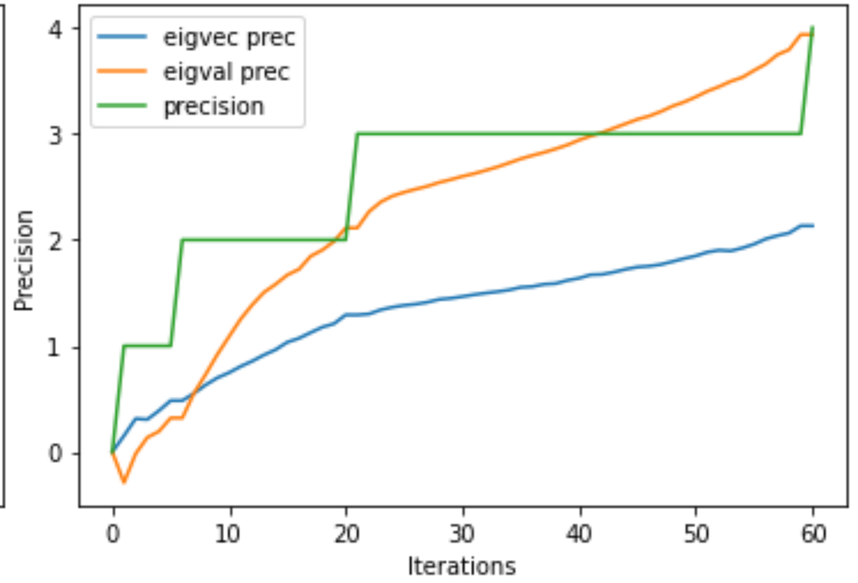
No dependance on the dimension of the matrix

Dependance on the gap among first and second eigenvalues: required more iterations

Y axis: precisions in logarithmic scale $\log(10^{-n})$
X axis: Iterations of the gradient descent phase (z)



n=8, $\lambda = -11.95621(-11.95623), \epsilon = 0.001$



n=32, $\lambda = -19.8999(-19.900), \epsilon = 0.001$

Non symmetric Eigenvalue Problem

[S. Alliney, F. Laudiero and M. Savoia, Applied mathematical modelling 16, 148 (1992)]

Complex eigenvalues ⇒we can't minimize the Rayleigh-Ritz quotient. We need to define a **new objective function**

$$J[\mathbf{v}, \lambda_g] = \mathbf{v}^T[A - \lambda_g I]^T[A - \lambda_g I]\mathbf{v} = \mathbf{v}^T[A^T A - 2\lambda_g M + \lambda_g^2 I^T I]\mathbf{v} \geq 0 \text{ with } M = \frac{I^T A + A^T I}{2}$$

The minimum is for $J[\mathbf{v}_0, \lambda_0] = 0$, so $\lambda_0 = \dfrac{\mathbf{v_0}^T M \mathbf{v}_0 \pm i\sqrt{R[\mathbf{v}_0]}}{\mathbf{v}_0^T \mathbf{v}_0}$

But one can notice that $R[\mathbf{v}, \lambda]$ is **proportional** to $J[\mathbf{v}, \lambda]$, so when we find the minimum of $J[\mathbf{v}, \lambda]$ we automatically set $R[\mathbf{v}, \lambda] = 0$.

So we find the lowest real eigenvalue as $\lambda_0 = \dfrac{\mathbf{v}_0^T M \mathbf{v}_0}{\mathbf{v}_0^T \mathbf{v}_0}$

GEVP

GEVP $\Rightarrow$ Replace I with B, but we didn't get results as good as the standard one

Hybrid algorithm:

- **Classical part:** decompose the B matrix to transform the GEVP into a standard eigenvalue problem
- **Quantum part:** finds the lower eigenvalue with QA

We use the **LDL decomposition**, where a real symmetric matrix B is factorized as $B = LDL^T$, with L a lower triangular matrix and D a diagonal matrix

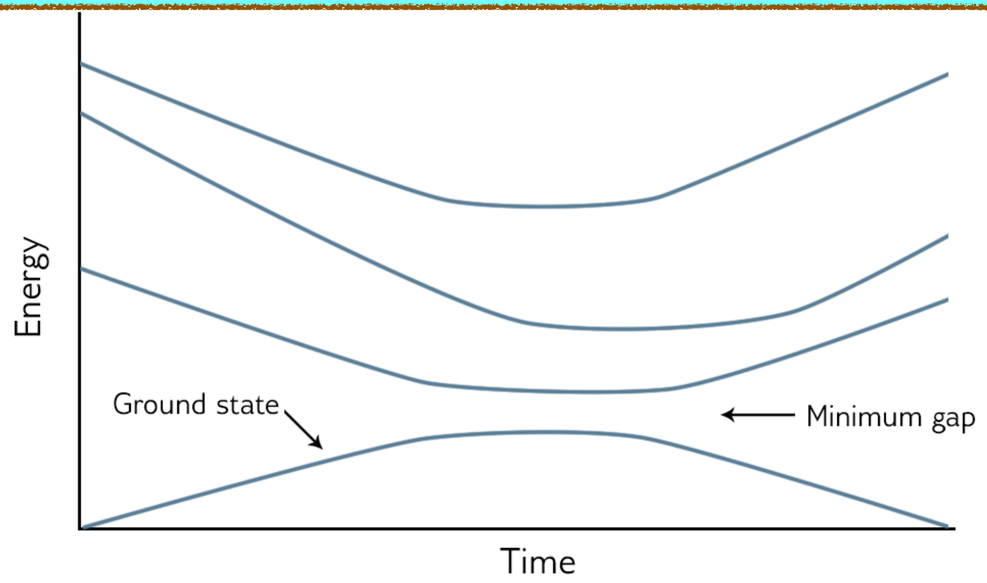
$$A\mathbf{v}_i = \lambda_i B\mathbf{v}_i \Rightarrow [(LD)^{-1}A(L^T)^{-1}][L^T\mathbf{v}_i] = \lambda_i[L^T\mathbf{v}_i] \Rightarrow \tilde{A}\mathbf{w}_i = \lambda_i\mathbf{w}_i$$

Quantum Annealing

We are using both **Quantum Annealing (QA)** and **Simulated Annealing (SA)**

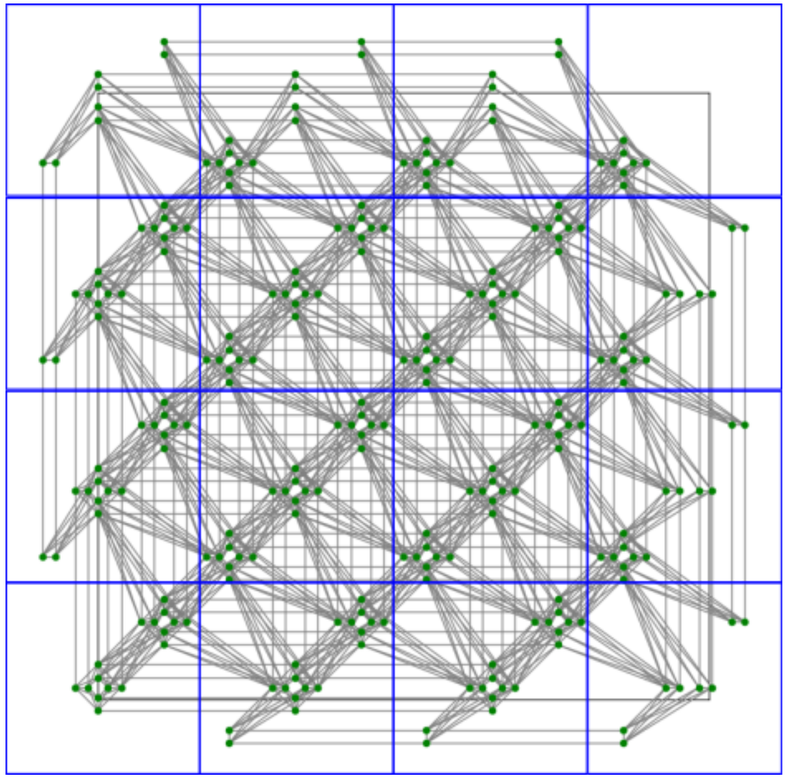
Quantum Annealing ⇒ Adiabatic transformation from an initial Hamiltonian to a final Hamiltonian (where the QUBO matrix of the problem is encoded to a transverse-Ising Hamiltonian).

The qubits are initially in the ground state of the initial hamiltonian, after the annealing the qubits are in the ground state of the final Hamiltonian



Topologies

In a real quantum computer we can't link each qubit to the others. How the qubits are linked through the couplers define the **topology of the Quantum Annealer**. The QUBO matrix so is encoded to an Hamiltonian that can be represented by the topology of the QA.



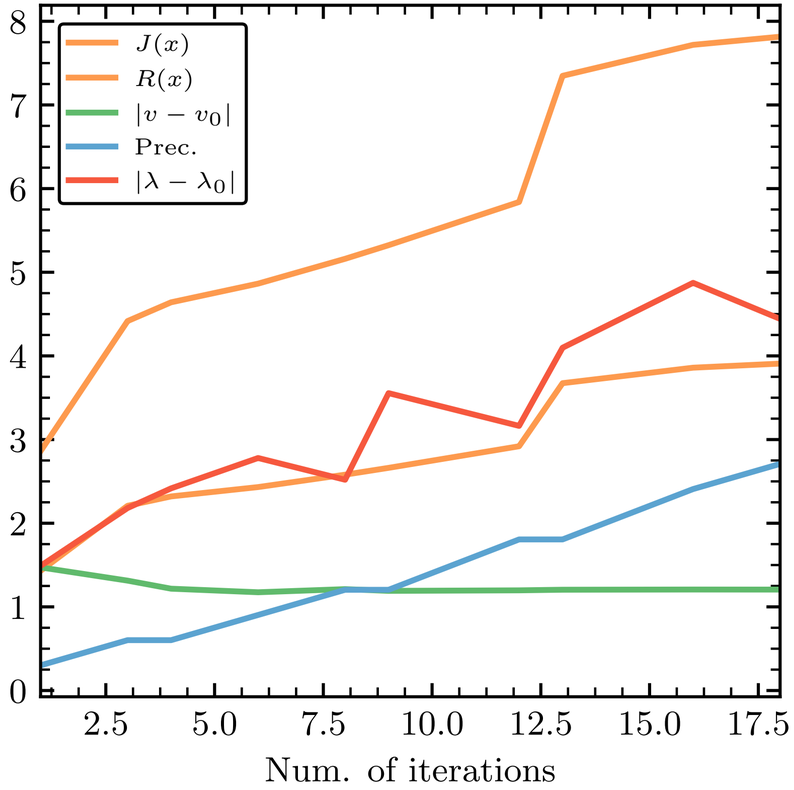
Unit cell of the pegasus topology. Green dots: qubits; Gray lines: couplers

Results

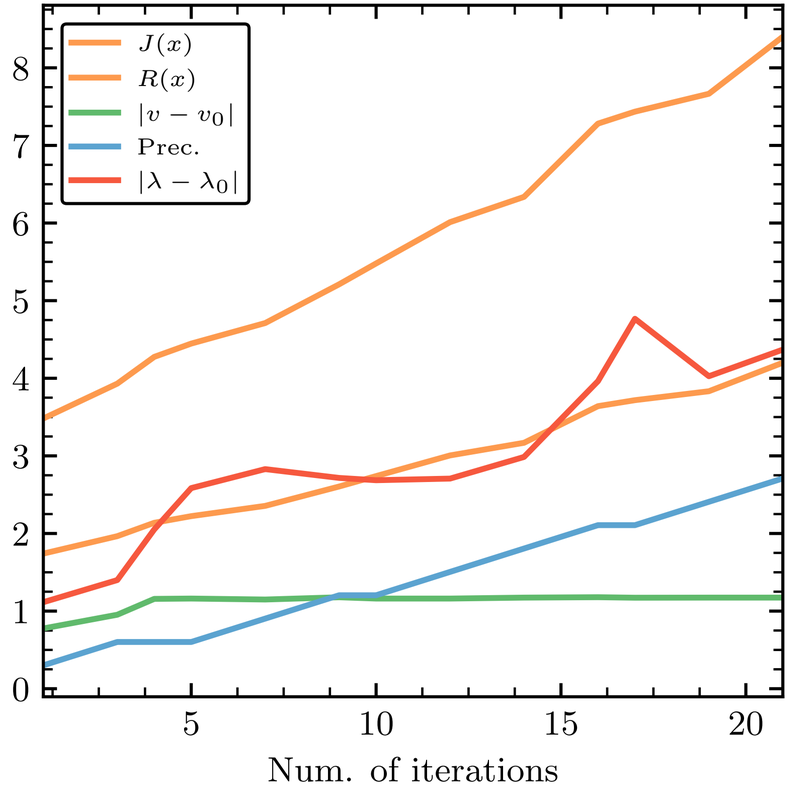
We are studying the statistical distribution of the results obtained via QA. Since for larger dimensions this require too much time we are using Simulated Annealing to simulate the algorithm

- **QA with pegasus topology**
- **Simulated Annealing without topology**: The algorithm can run multiple times and we can see the behavior of the algorithm with the dimension of the matrices
- **Simulated Annealing with pegasus topology:** The algorithm can run multiple times only for small matrices (4x4 or 6x6). Good agreement with the results obtained with the QA

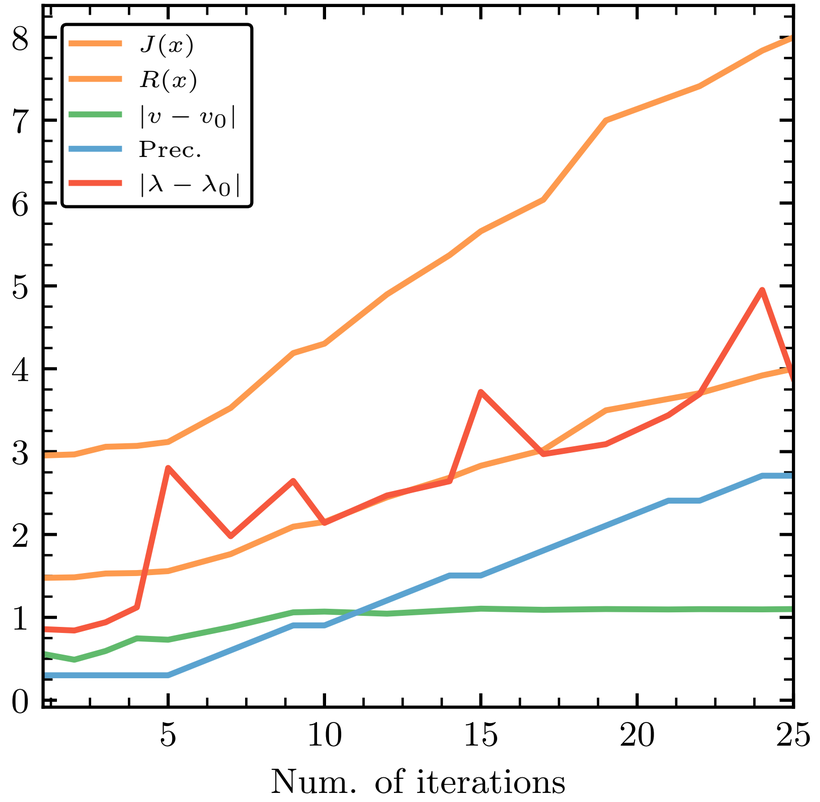
Result of the gradient-descent phase via QA



$n = 4$
$\epsilon = 0.001$
Annealing time = 0.48s
$\lambda = 0.475311(0.475328)$
$\frac{\mathbf{v} - \mathbf{v}_c}{\mathbf{v}_c} = 0.062$

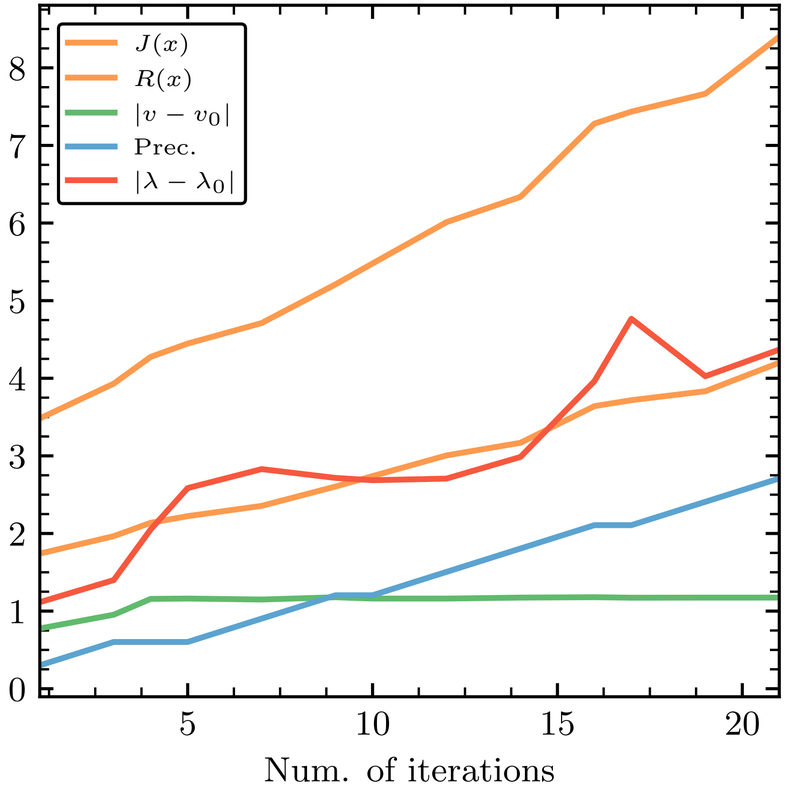


$n = 8$
$\epsilon = 0.001$
Annealing time = 0.54s
$\lambda = 0.188212(0.188204)$
$\frac{\mathbf{v} - \mathbf{v}_c}{\mathbf{v}_c} = 0.067$

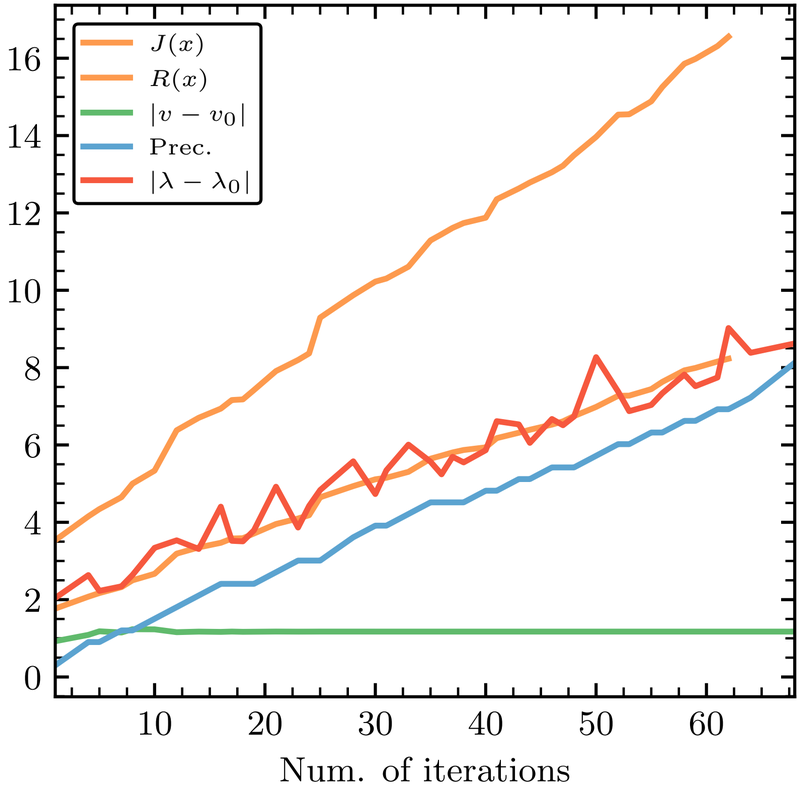


$n = 12$
$\epsilon = 0.001$
Annealing time = 0.69s
$\lambda = 0.188178(0.188204)$
$\frac{\mathbf{v} - \mathbf{v}_c}{\mathbf{v}_c} = 0.079$

Result of the gradient-descent phase via QA



$n = 8$
$\epsilon = 0.001$
Annealing time = 0.54s
$\lambda = 0.188212(0.188204)$
$\frac{\mathbf{v} - \mathbf{v}_c}{\mathbf{v}_c} = 0.067013$



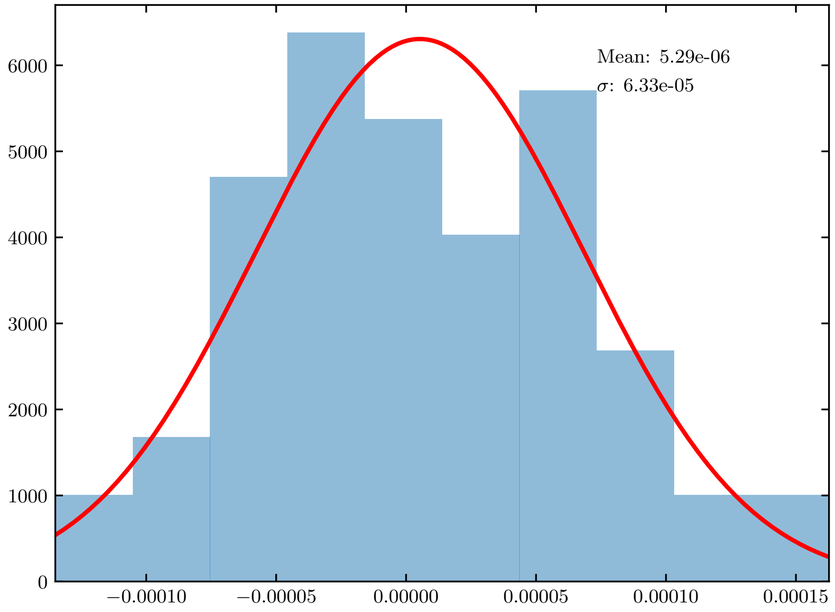
$n = 8$
$\epsilon = 0.000000001$
Annealing time = 1.7s
$\lambda = 0.1882038165(0.1882038169)$
$\frac{\mathbf{v} - \mathbf{v}_c}{\mathbf{v}_c} = 0.067000$

- Weak dependance on n
- The precision of λ is comparable with ϵ
- The precision of the eigenvector does not improve with the precision of λ

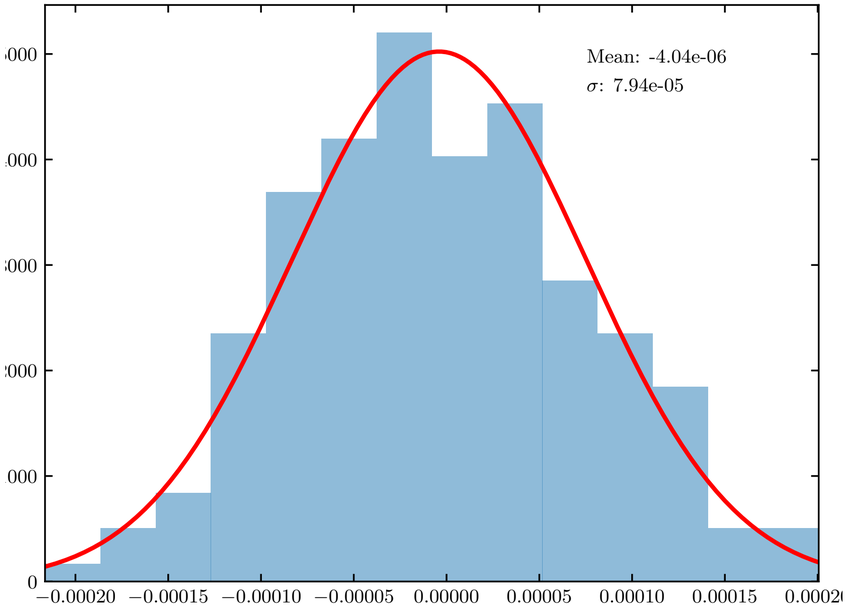
Comparing QA and SA

Distribution of the eigenvalue obtained with the algorithm compared to the eigenvalue obtained with classical computation: $\frac{\lambda - \lambda_c}{\lambda_c}$.

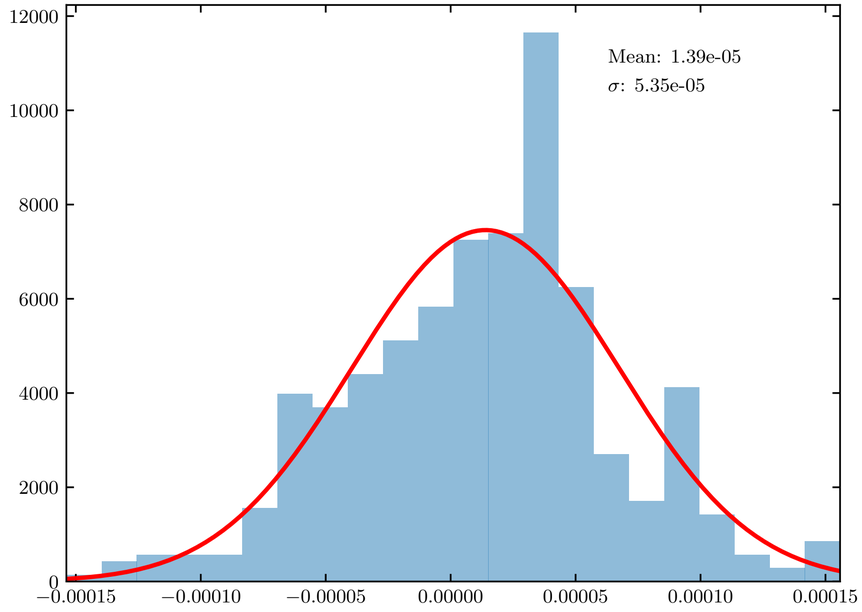
n=4, 200 samples, $\epsilon = 0.001$.



SA “pegasus”



QA



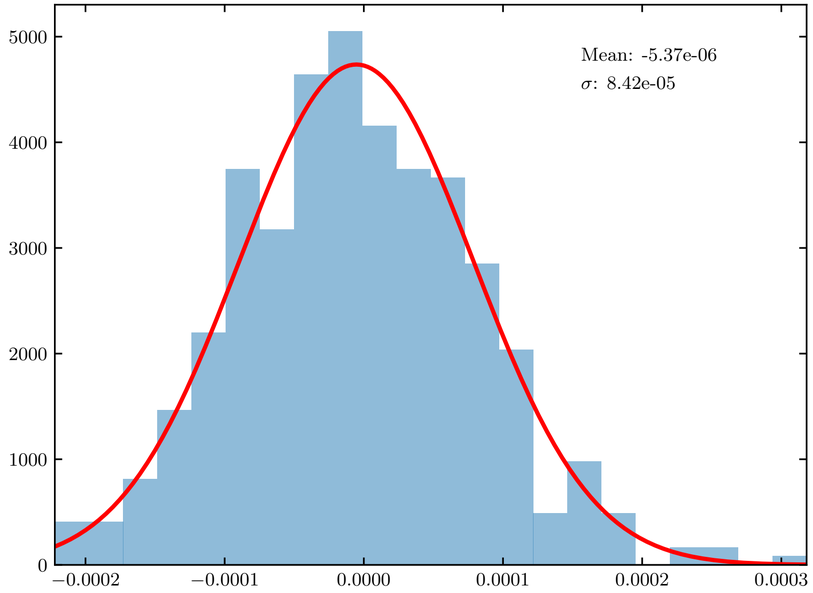
SA

Scaling with the dimension

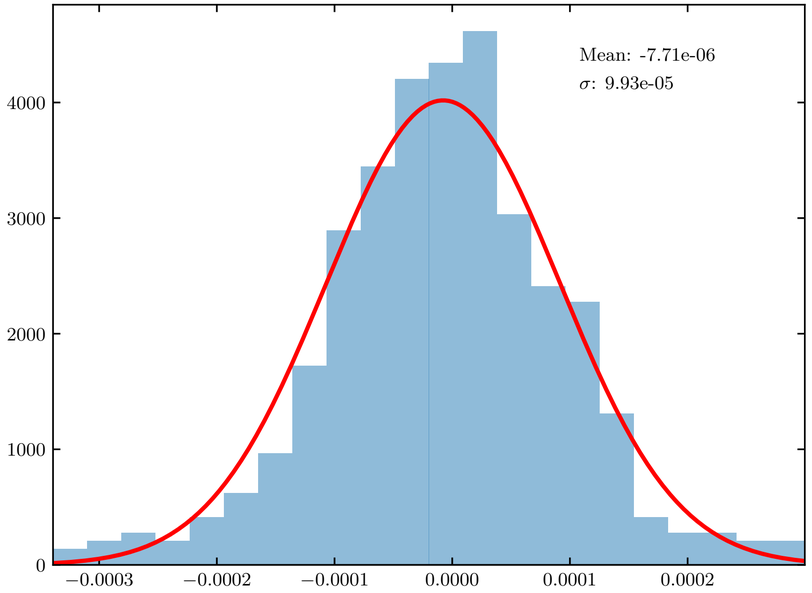
Samples = 500, $\epsilon = 0.001$

We can't see a strong dependance on n

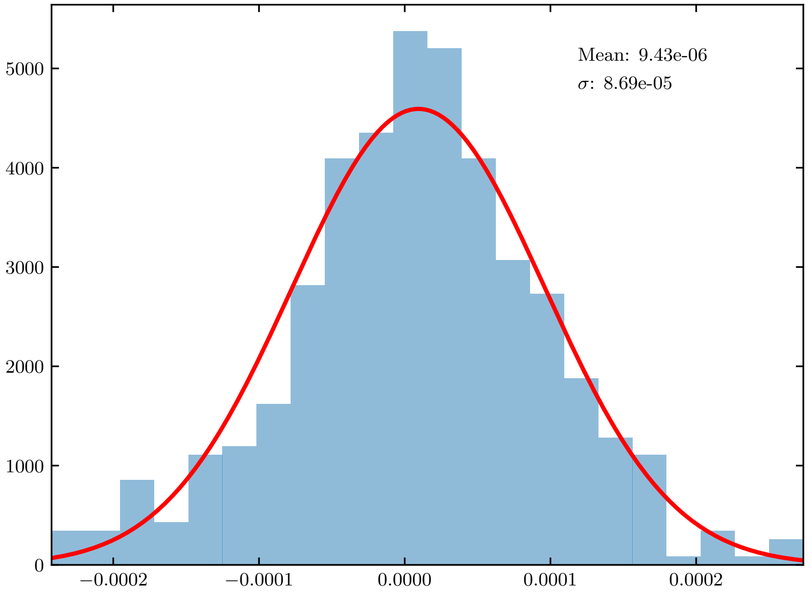
For every n we have $\dfrac{\epsilon}{\lambda_c} > 5\sigma$



n=6, SA



n=12, SA



n=24, SA

Conclusions

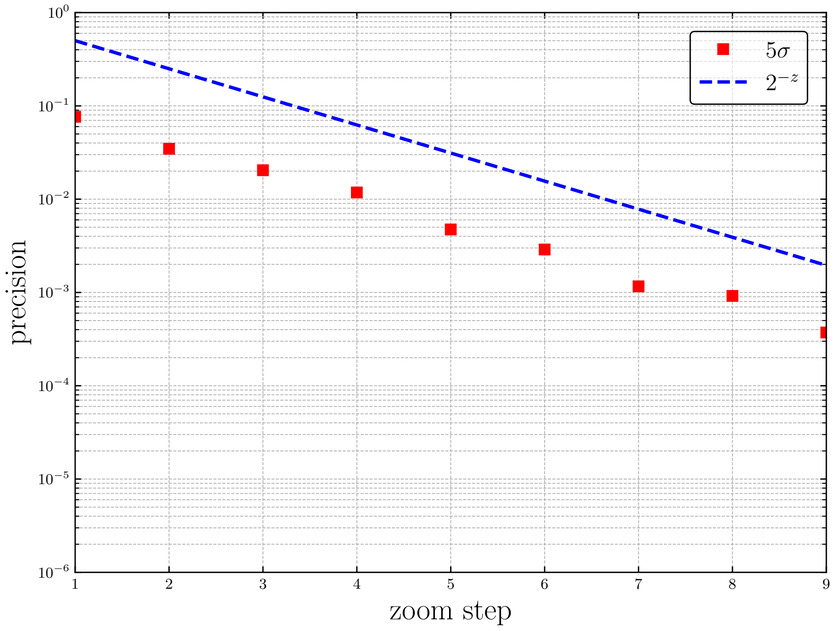
- The algorithm reproduces the eigenvalues with good agreement
- The precision on the eigenvalues does not depend on the dimension of the matrices

To do next:

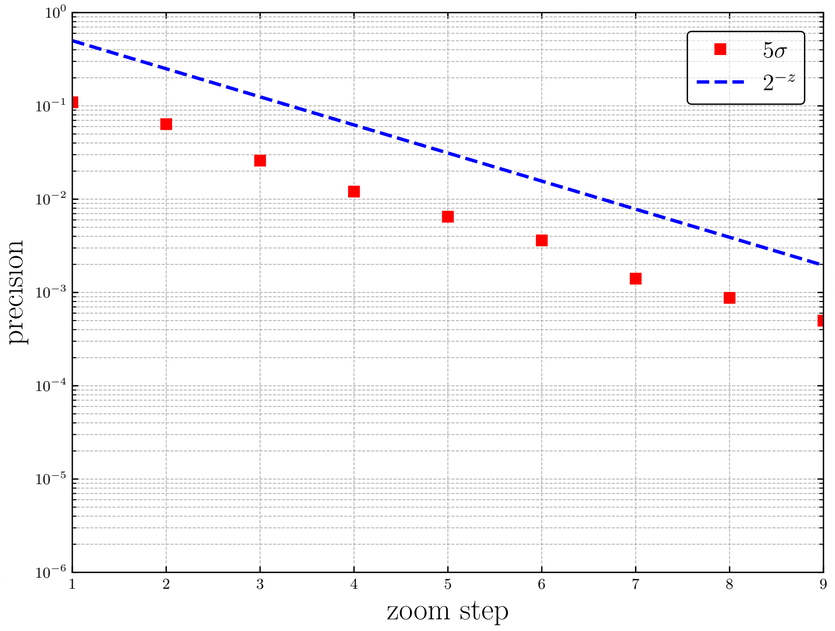
- Analysis of the dependance on the parameters $(\lambda_g, b, \epsilon, \ldots)$
- Run the code with QA to obtain statistical distributions for higher dimensions
- Try to use a pure quantum algorithm for the GEVP

Comparing QA and SA

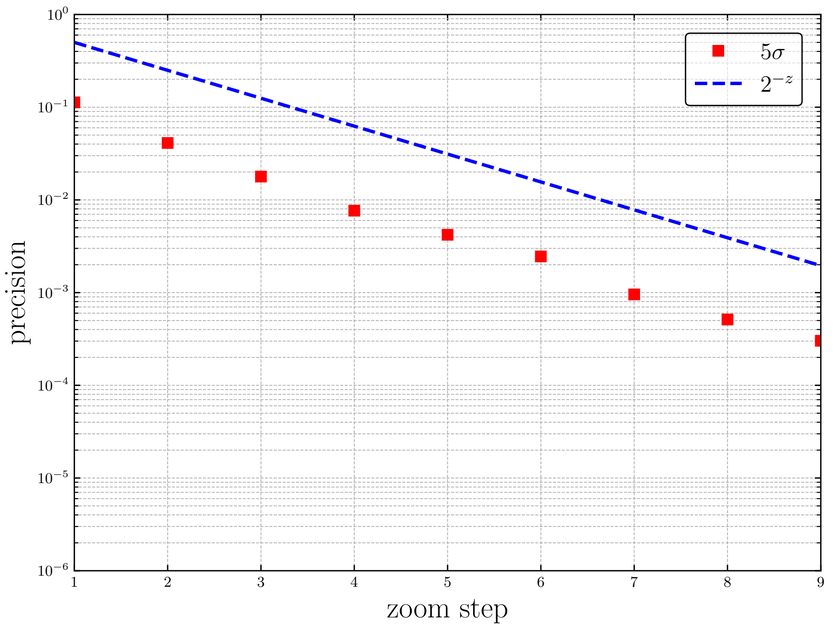
σ extracted for each zoom step vs 2^{-z} (nominal precision of the algorithm)
5σ is always below the nominal precision of the algorithm



SA “pegasus"

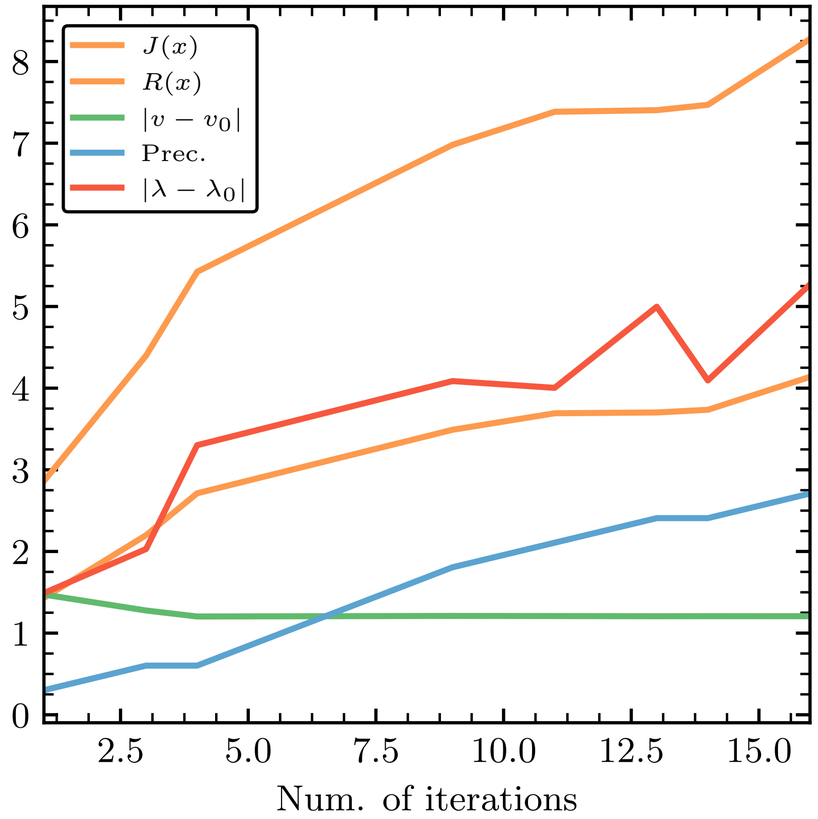


QA



SA

Best solution

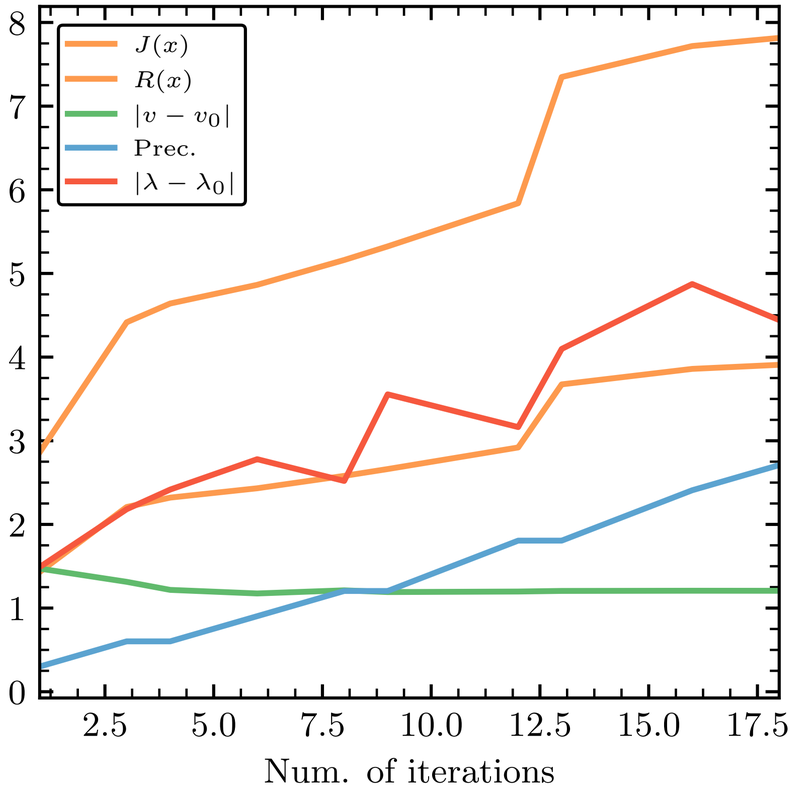


Noiseless simulation

n=4

$\epsilon = 0.001$

$\lambda = 0.475331(0.475328)$



QA

n=4

$\epsilon = 0.001$

$\lambda = 0.475311(0.475328)$