



Universität Augsburg
Mathematisch-Naturwissenschaftlich-
Technische Fakultät

Computer aided Analytical Calculations for Physical Many-Body Problems

- Project Work Presentation -

Jonas Kell
Chair for theoretical Physics III

15th of Mai 2024

Outline

1 Introduction of the problem

2 Solutions: Off the shelf

3 Math-Manipulator

- What did I do?
- How can you use it
- Interactive Demonstration

4 Custom Python Scripts (SymPy)

5 Presentations? - Custom Beamer Template

Physics of what we calculate

Introduction of the problem

- Content of my Practical Training (Fachpraktikum)
- Hubbard Model Hamiltonian
- System under influence of external electric field

$$\mathcal{H} = \mathcal{H}_0 + \hat{V}$$

$$\mathcal{H}_0 = U \cdot \sum_l \hat{n}_{l,\uparrow} \hat{n}_{l,\downarrow} + \sum_{l,\sigma} \underbrace{\left(\vec{E} \cdot \vec{r}_l \right)}_{\varepsilon_l} \hat{n}_{l,\sigma}$$

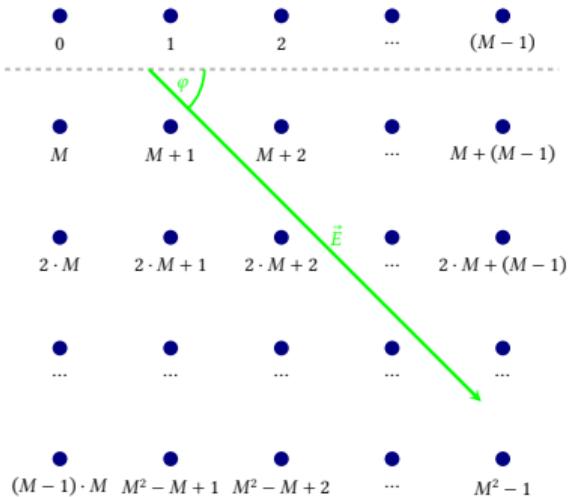
$$\hat{V} = -J \cdot \sum_{\langle l,m \rangle, \sigma} \left(\hat{c}_{l,\sigma}^\dagger \hat{c}_{m,\sigma} + \hat{c}_{m,\sigma}^\dagger \hat{c}_{l,\sigma} \right)$$

Physics of what we calculate

Introduction of the problem

- 2-dimensional geometry
- Square lattice arrangement
- Computation for general size and field angle

Goal: Approximate evaluation of time-evolution using Monte-Carlo Sampling



Physics of what we calculate

Introduction of the problem

- Hard-Core Bosonic operators
 - (Would work analogously with Fermionic operators)
- Two spin-degrees of freedom rewritten in alternate notation

$$\begin{aligned} [\hat{c}_{l,\sigma}, \hat{c}_{l',\sigma'}] &= [\hat{c}_{l,\sigma}^\dagger, \hat{c}_{l',\sigma'}^\dagger] = 0 \\ [\hat{c}_{l,\sigma}, \hat{c}_{l',\sigma'}^\dagger] &= \delta(l,l') \cdot \delta(\sigma,\sigma') \end{aligned}$$

$$\hat{c}_{l,\uparrow}^{(\dagger)} \leftrightarrow \hat{c}_l^{(\dagger)} \quad \hat{c}_{l,\downarrow}^{(\dagger)} \leftrightarrow \hat{d}_l^{(\dagger)}$$

$$\mathcal{H}_0 = U \cdot \sum_l \hat{c}_l^\dagger \hat{c}_l \hat{d}_l^\dagger \hat{d}_l + \sum_l \varepsilon_l \hat{c}_l^\dagger \hat{c}_l \sum_l \varepsilon_l \hat{d}_l^\dagger \hat{d}_l$$

$$\hat{V} = -J \cdot \sum_{\langle l,m \rangle} \left(\hat{c}_l^\dagger \hat{c}_m + \hat{c}_m^\dagger \hat{c}_l + \hat{d}_l^\dagger \hat{d}_m + \hat{d}_m^\dagger \hat{d}_l \right)$$

Working on paper: A Computer-Scientists view

Introduction of the problem

Doing Theoretical Physics often requires lengthy analytical calculations

Advantages of working on paper:

- No barrier to entry (required Skill & technology)
- Maximum liberty how to operate
- Fast iteration for high variation workload
- "Offline" available

Problems of working on paper:

- Repetitive tasks not automizable
- Error-prone and time-consuming to fix mistakes
- Difficult & non-standard to share/archive
- No version-control
- Need to produce final version anyway

Examples from "Theoretical Solid State Physics"

Introduction of the problem

$$[S, H] = [c_1^t c_2 - c_2^t c_1, \epsilon c_1^t c_1 + \Delta (c_1^t c_2 + c_2^t c_1)] \quad \Rightarrow [c_1^t c_2, c_1^t c_2] = 0, \text{ obvious, usu.}$$

$$= \epsilon [c_1^t c_2, c_1^t c_2] - \epsilon [c_2^t c_1, c_1^t c_2] + \Delta [c_1^t c_2, c_1^t c_2] - \Delta [c_2^t c_1, c_1^t c_2]$$

$$\begin{aligned} & c_1^t c_2 c_1^t c_2 - c_1^t c_2 c_1^t c_2 = c_1^t c_2 \\ & \underbrace{c_1^t c_2}_{= -c_2^t c_1} \quad \left| \begin{array}{l} c_1^t c_2 c_1^t c_2 - c_1^t c_2 c_1^t c_2 = \\ + c_1^t c_2 \quad \quad \quad - c_1^t c_2 \end{array} \right. \\ & \quad \quad \quad = c_1^t c_2 - c_1^t c_2 c_1^t c_2 - c_2^t c_1^t c_2 = \end{aligned}$$

$$\begin{aligned} & \quad \quad \quad - c_1^t c_2 \quad \quad \quad - c_1^t c_2 \\ & = c_1^t c_2 + c_1^t c_2 c_1^t c_2 - c_1^t c_2 - c_1^t c_2 + c_1^t c_2 = c_1^t c_2 - c_1^t c_2 \end{aligned}$$

$$= \epsilon c_1^t c_2 - \epsilon c_2^t c_1 + 2\Delta (c_1^t c_2 - c_2^t c_1)$$

$$= \epsilon (c_1^t c_2 - c_2^t c_1) + 2\Delta (c_1^t c_2 - c_2^t c_1)$$

$$= \frac{\epsilon}{\Delta} V + 2\Delta (c_1^t c_2 - c_2^t c_1)$$

Examples from "Theoretical Solid State Physics"

Introduction of the problem

Final transform $\tilde{c}_{k\sigma}^{\dagger}(t) = e^{iHt} c_{k\sigma}^{\dagger} e^{-iHt}$ \rightarrow complicated
 $\tilde{c}_{k\sigma}^{\dagger}(t) = e^{iHt} \tilde{c}_{k\sigma}^{\dagger} e^{-iHt}$ \rightarrow easier

$$\tilde{c}_{k\sigma}^{\dagger} = e^S c_{k\sigma}^{\dagger} e^{-S} \approx c_{k\sigma}^{\dagger} + [S, c_{k\sigma}^{\dagger}] + O(U^2)$$

$$[S, c_{k\sigma}^{\dagger}] = \frac{U}{N} \sum'_{k_1 \dots k_4} \frac{\delta(k_1 - k_2 + k_3 - k_4)}{\epsilon_{k_1} - \epsilon_{k_2} + \epsilon_{k_3} - \epsilon_{k_4}} [c_{k_1\sigma}^{\dagger} c_{k_2\sigma}^{\dagger} c_{k_3\sigma}^{\dagger} c_{k_4\sigma}, c_{k\sigma}^{\dagger}] =$$

$$c_{k_1\sigma}^{\dagger} c_{k_2\sigma}^{\dagger} c_{k_3\sigma}^{\dagger} S_{k_2} S_{k_3} + c_{k_1\sigma}^{\dagger} c_{k_2\sigma}^{\dagger} c_{k_3\sigma}^{\dagger} S_{k_3} S_{k_2}$$

| Change index $k_2 \leftrightarrow k_3$
 $k_3 \leftrightarrow k_4$

$$\begin{aligned} \tilde{c}_{k\sigma}^{\dagger} &= e^S c_{k\sigma}^{\dagger} e^{-S} = c_{k\sigma}^{\dagger} + \frac{U}{N} \sum'_{k_1 \dots k_4} \frac{\delta(k_1 - k_2 + k_3 - k_4)}{\epsilon_{k_1} - \epsilon_{k_2} + \epsilon_{k_3} - \epsilon_{k_4}} \\ &= c_{k\sigma}^{\dagger} + \frac{U}{N} \sum'_{k_1 \dots k_4} \frac{\delta(k_1 - k_2 + k_3 - k_4)}{\epsilon_{k_1} - \epsilon_{k_2} + \epsilon_{k_3} - \epsilon_{k_4}} \delta_{kk_1} c_{k_1\sigma}^{\dagger} c_{k_3\bar{\sigma}}^{\dagger} c_{k_4\bar{\sigma}} \\ &\quad - \frac{U}{N} \sum'_{k_1 \dots k_4} \frac{1}{\epsilon_{k_1} - \epsilon_{k_2} + \epsilon_{k_3} - \epsilon_{k_4}} c_{k_1\sigma}^{\dagger} c_{k_2\bar{\sigma}}^{\dagger} c_{k_3\bar{\sigma}}^{\dagger} c_{k_4\bar{\sigma}} \end{aligned}$$

ii)

$$\bar{H} = u H u^{\dagger} + e^S H e^{-S} = H + [S, H_0] + O(U^2) \quad [S, V] \rightarrow \text{second order}$$

$$[S, H_0] = \frac{U}{N} \sum'_{k_1 \dots k_4} \frac{\delta(k_1 - k_2 + k_3 - k_4)}{\epsilon_{k_1} - \epsilon_{k_2} + \epsilon_{k_3} - \epsilon_{k_4}} \left[c_{k_1\sigma}^{\dagger} c_{k_2\sigma}^{\dagger} c_{k_3\sigma}^{\dagger} c_{k_4\sigma}, \sum_{k\sigma} \epsilon_k n_{k\sigma} \right]$$

$$\rightarrow \text{use } [c_{k\sigma}^{\dagger}, h_{k'\bar{\sigma}}] = -(\epsilon_{k\sigma} - \epsilon_{k_1} + \epsilon_{k_2} - \epsilon_{k_3}) c_{k\sigma}^{\dagger} c_{k_1\bar{\sigma}}^{\dagger} c_{k_2\bar{\sigma}}^{\dagger} c_{k_3\bar{\sigma}}^{\dagger}$$

$$u = e^S, \quad S = \frac{1}{N} \sum_{k_1 \dots k_4} \delta(k_1 - k_2 + k_3 - k_4) \frac{U}{\epsilon_{k_1} - \epsilon_{k_2} + \epsilon_{k_3} - \epsilon_{k_4}} c_{k_1\sigma}^{\dagger} c_{k_2\bar{\sigma}}^{\dagger} c_{k_3\bar{\sigma}}^{\dagger} c_{k_4\bar{\sigma}} \quad (2)$$

we can eliminate to leading order in U the interaction term. Notice that the summation $\sum_{k_1 \dots k_4}$ only involves those contributions for which $\epsilon_{k_1} - \epsilon_{k_2} + \epsilon_{k_3} - \epsilon_{k_4} \neq 0$.

- a) Calculate to the leading order in U the time evolution $c_{k\sigma}^{\dagger}(t)$ of single fermion operators in Heisenberg picture:

$$c_{k\sigma}^{\dagger}(t) = e^{iHt} c_{k\sigma}^{\dagger} e^{-iHt} \quad (3)$$

- i) First transform both the Hamiltonian H and $c_{k\sigma}^{\dagger}$ with u :

$$\hat{H} = u H u^{\dagger}, \quad \hat{c}_{k\sigma}^{\dagger} = u c_{k\sigma}^{\dagger} u^{\dagger} \quad (4)$$

(see previous exercise sheet)

- ii) Show that the transformed Hamiltonian has the following form

$$\hat{H} = u H u^{\dagger} = \sum_{k\sigma} \epsilon_k n_{k\sigma} + \sum_{k,k'} F_{k,k'} n_{k\sigma} n_{k'\bar{\sigma}} + O(U^2) \quad (5)$$

and determine $F_{k,k'}$.

Examples from the Practical Training

Introduction of the problem

- Goal: produce time evolution of operator
- Uses calculation in Interaction Picture
- Workflow:
 - Inserting Definitions
 - Expanding
 - Ordering
 - Rewriting efficiently

$$\hat{V} = -J \cdot \sum_{\langle l,m \rangle} \left(\hat{c}_l^\dagger \hat{c}_m + \hat{c}_m^\dagger \hat{c}_l + \hat{d}_l^\dagger \hat{d}_m + \hat{d}_m^\dagger \hat{d}_l \right)$$

$$\hat{c}_m^{\dagger I}(t) = e^{i \cdot \varepsilon_m \cdot t} \left(1 + (e^{i \cdot U \cdot t} - 1) \hat{d}_m^{\dagger S} \hat{d}_m^S \right) \hat{c}_m^{\dagger S}$$

$$\hat{c}_m^I(t) = e^{-i \cdot \varepsilon_m \cdot t} \left(1 + (e^{-i \cdot U \cdot t} - 1) \hat{d}_m^{\dagger S} \hat{d}_m^S \right) \hat{c}_m^S$$

$$\hat{d}_m^{\dagger I}(t) = e^{i \cdot \varepsilon_m \cdot t} \left(1 + (e^{i \cdot U \cdot t} - 1) \hat{c}_m^{\dagger S} \hat{c}_m^S \right) \hat{d}_m^S$$

$$\hat{d}_m^I(t) = e^{-i \cdot \varepsilon_m \cdot t} \left(1 + (e^{-i \cdot U \cdot t} - 1) \hat{c}_m^{\dagger S} \hat{c}_m^S \right) \hat{d}_m^S$$

Paper-like writing on digital devices

Solutions: Off the shelf

- Can give benefits that paper lacks
 - Copy/Paste
 - Collaborate
 - Convert to text
 - Searching and ordering large amounts of notes
- HOWEVER
 - Still no proper automatization
 - Easily access is lost when using proprietary technology
 - "Vendor-Locking"

TIP:

Always export to a standard format (like pdf) for manual backup!

Computer-Algebra Systems

Solutions: Off the shelf

- Proprietary Software (Often payed):
 - Wolfram-Alpha
 - Mathematica
 - Chat-GPT
 - ...
- Standalone Online Calculators
 - Integralrechner.de (Integral-Calculator)
 - Matrix-Calculators
 - ...
- ... and many more

What is Math-Manipulator?

Math-Manipulator - What did I do?

```
sum((n = 0); 100; int(-inf; inf; (123+(A*4)/100); x))
```

$$\sum_{n=0}^{100} \int_{-\infty}^{\infty} \left(123 + \frac{(A \cdot 4)}{100} \right) dx$$

[Sel. Parent](#) [Sel. First Child](#) [Sel. Previous Sibling](#) [Sel. Next Sibling](#)

[Replace](#) [Replace All Equal](#) [Replace \(With Export\)](#) [Define Variable](#) [Commute with subsequent](#) [Fold Numbers](#) [Cleanup Terms](#) [Pull Out Minus](#) [Reduce](#)

[Rename With Swap](#) [Pack Same Variables](#) [Replace All With](#)

[Selected Export to Clipboard](#) [Show Export Structure \(UUIDs\)](#) [Show Export Structure](#) [Show Latex](#)

T

$$\sum_{n=0}^{100} \int_{-\infty}^{\infty} (123 + T) dx$$

In the Browser

Math-Manipulator - How can you use it

- Available open-source on GitHub
- Self-hosting possible
- Hosted version available in every modern browser

Available here

<https://jonas-kell.github.io/math-manipulator/>

Usage as VS-Code Extension

Math-Manipulator - How can you use it

- Offline-Use after installed once
- Store progress in files
- Directly integrated in powerful IDE, and with that version control

Note

Do not allow a spellchecker to run on the file, it will break everything performance-wise

Demo Problems from before

Math-Manipulator - Interactive Demonstration

$$\begin{aligned} &= \epsilon [c_1^t c_2, c_1^+ c_2] - \epsilon [c_2^t c_1 - c_1^t c_2] + D [c_1^t c_2, c_2^+ c_1] - D [c_1^+ c_1, c_2^t c_2] \\ &= \epsilon (c_1^t c_2 - c_2^t c_1) + 2D (c_1^t c_1 - c_2^t c_2) \end{aligned}$$

$$\begin{aligned} &\left[c_{41}^t c_{42} + c_{42}^t c_{43} + c_{43}^t c_{44}, \sum_{k \neq 4} \epsilon_{kk} u_{kk} \right] \\ &- (\epsilon_{41} - \epsilon_{42} - \epsilon_{43} - \epsilon_{44}) c_{41}^t c_{42} + c_{42}^t c_{43} \end{aligned}$$

Usage for the Practical Training

Math-Manipulator - Interactive Demonstration

$$\hat{V}^I(t) = \left\{ \hat{V}^S \right\}(t) \stackrel{2.11}{=} -J \cdot \sum_{[l,m]} \left\{ \left(\hat{c}_l^{\dagger S} \hat{c}_m^S + \hat{d}_l^{\dagger S} \hat{d}_m^S \right) \right\}(t)$$

$$= -J \cdot \sum_{[l,m]} \left(\hat{c}_l^{\dagger I}(t) \hat{c}_m^I(t) + \hat{d}_l^{\dagger I}(t) \hat{d}_m^I(t) \right)$$

$$\stackrel{MM}{=} -J \cdot \sum_{[l,m]} \left[e^{i(\varepsilon_m - \varepsilon_l)t} \cdot \hat{V}_{\text{Part A}}(l, m) + e^{i(\varepsilon_m - \varepsilon_l + U)t} \cdot \hat{V}_{\text{Part B}}(l, m) + e^{i(\varepsilon_m - \varepsilon_l - U)t} \cdot \hat{V}_{\text{Part C}}(l, m) \right]$$

$$\hat{V}_{\text{Part A}}(l, m) \stackrel{MM}{=} \left(5 \cdot \hat{c}_l^S \hat{c}_m^{\dagger S} \right) + \left(5 \cdot \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(2 \cdot \hat{c}_l^S \hat{c}_m^S \hat{c}_l^{\dagger S} \hat{c}_m^{\dagger S} \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(2 \cdot \hat{c}_l^S \hat{c}_m^{\dagger S} \hat{d}_l^S \hat{d}_m^S \hat{d}_l^{\dagger S} \hat{d}_m^{\dagger S} \right) + \left(-3 \cdot \hat{c}_l^S \hat{c}_l^{\dagger S} \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(-3 \cdot \hat{c}_m^S \hat{c}_m^{\dagger S} \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(-3 \cdot \hat{c}_l^S \hat{c}_m^S \hat{d}_l^S \hat{d}_l^{\dagger S} \right) + \left(-3 \cdot \hat{c}_l^S \hat{c}_m^{\dagger S} \hat{d}_m^S \hat{d}_m^{\dagger S} \right)$$

$$\hat{V}_{\text{Part B}}(l, m) \stackrel{MM}{=} \left(-2 \cdot \hat{c}_l^S \hat{c}_m^{\dagger S} \right) + \left(-2 \cdot \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(-1 \cdot \hat{c}_l^S \hat{c}_m^S \hat{c}_l^{\dagger S} \hat{c}_m^{\dagger S} \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(-1 \cdot \hat{c}_l^S \hat{c}_m^{\dagger S} \hat{d}_l^S \hat{d}_m^S \hat{d}_l^{\dagger S} \hat{d}_m^{\dagger S} \right) + \left(1 \cdot \hat{c}_l^S \hat{c}_l^{\dagger S} \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(2 \cdot \hat{c}_m^S \hat{c}_m^{\dagger S} \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(1 \cdot \hat{c}_l^S \hat{c}_m^{\dagger S} \hat{d}_l^S \hat{d}_l^{\dagger S} \right) + \left(2 \cdot \hat{c}_l^S \hat{c}_m^{\dagger S} \hat{d}_m^S \hat{d}_m^{\dagger S} \right)$$

$$\hat{V}_{\text{Part C}}(l, m) \stackrel{MM}{=} \left(-2 \cdot \hat{c}_l^S \hat{c}_m^{\dagger S} \right) + \left(-2 \cdot \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(-1 \cdot \hat{c}_l^S \hat{c}_m^S \hat{c}_l^{\dagger S} \hat{c}_m^{\dagger S} \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(-1 \cdot \hat{c}_l^S \hat{c}_m^{\dagger S} \hat{d}_l^S \hat{d}_m^S \hat{d}_l^{\dagger S} \hat{d}_m^{\dagger S} \right) + \left(2 \cdot \hat{c}_l^S \hat{c}_l^{\dagger S} \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(1 \cdot \hat{c}_m^S \hat{c}_m^{\dagger S} \hat{d}_l^S \hat{d}_m^{\dagger S} \right) + \left(2 \cdot \hat{c}_l^S \hat{c}_m^{\dagger S} \hat{d}_l^S \hat{d}_l^{\dagger S} \right) + \left(1 \cdot \hat{c}_l^S \hat{c}_m^{\dagger S} \hat{d}_m^S \hat{d}_m^{\dagger S} \right)$$

Optimization: Difference of $V(t)$ for Hopping

Custom Python Scripts (SymPy)

- Monte-Carlo sampling requires transition probabilities between "adjacent" states
- These require differences of $\hat{V}^I(t)$ and E_0
- Too many terms for simple evaluation by hand
- Intelligent pre-computation required to speed up processing

$$\begin{aligned}\alpha &= \frac{P(\tilde{N}, t)}{P(N, t)} = \frac{f(\tilde{N}, t)}{f(N, t)} \stackrel{2.39}{=} \frac{\left| e^{\mathcal{H}_{\text{eff}}(\tilde{N}, t)} \right|^2 |\Psi_{\tilde{N}}|^2}{\left| e^{\mathcal{H}_{\text{eff}}(N, t)} \right|^2 |\Psi_N|^2} \\ &= \frac{|\Psi_{\tilde{N}}|^2 e^{\Re(\mathcal{H}_{\text{eff}}(\tilde{N}, t)) + i\Im(\mathcal{H}_{\text{eff}}(\tilde{N}, t)) + \Re(\mathcal{H}_{\text{eff}}(\tilde{N}, t)) - i\Im(\mathcal{H}_{\text{eff}}(\tilde{N}, t))}}{|\Psi_N|^2 e^{\Re(\mathcal{H}_{\text{eff}}(N, t)) + i\Im(\mathcal{H}_{\text{eff}}(N, t)) + \Re(\mathcal{H}_{\text{eff}}(N, t)) - i\Im(\mathcal{H}_{\text{eff}}(N, t))}} \\ &= \frac{|\Psi_{\tilde{N}}|^2}{|\Psi_N|^2} e^{2 \cdot \Re(\mathcal{H}_{\text{eff}}(\tilde{N}, t)) - 2 \cdot \Re(\mathcal{H}_{\text{eff}}(N, t))} \\ &= \frac{|\Psi_{\tilde{N}}|^2}{|\Psi_N|^2} e^{2 \cdot \Re(\mathcal{H}_{\text{eff}}(\tilde{N}, t) - \mathcal{H}_{\text{eff}}(N, t))}\end{aligned}$$

The process: Difference of $V(t)$ for Hopping

Custom Python Scripts (SymPy)

$$\begin{aligned} E_0(N) - E_0(\tilde{N}) &= U \sum_l n_{l,\downarrow} n_{l,\uparrow} - U \sum_l \tilde{n}_{l,\downarrow} \tilde{n}_{l,\uparrow} + \sum_{l,\sigma} \varepsilon_l n_{l,\sigma} - \sum_{l,\sigma} \varepsilon_l \tilde{n}_{l,\sigma} \\ &\stackrel{2.45}{=} (\varepsilon_i - \varepsilon_j) (n_{i,\sigma_i} - n_{j,\sigma_j}) + U \cdot \begin{cases} (n_{i,\uparrow} - n_{j,\uparrow}) (n_{i,\downarrow} - n_{j,\downarrow}) & : \sigma_i = \sigma_j \\ (n_{i,\uparrow} - n_{j,\downarrow}) (n_{i,\downarrow} - n_{j,\uparrow}) & : \sigma_i \neq \sigma_j \end{cases} \quad \tilde{n}_{l,\sigma} = \begin{cases} n_{i,\sigma_i} & : l = j \wedge \sigma = \sigma_j \\ n_{j,\sigma_j} & : l = i \wedge \sigma = \sigma_i \\ n_{l,\sigma} & : \text{else} \end{cases} \\ &= (\varepsilon_i - \varepsilon_j) (n_{i,\sigma_i} - n_{j,\sigma_j}) + U (n_{i,\sigma_i} - n_{j,\sigma_j}) (n_{i,\bar{\sigma}_i} - n_{j,\bar{\sigma}_j}) \end{aligned}$$

- Setup difference
- Replace one by appropriate new occupation
- Simplify as much as possible
- Sum over all possible neighbor combinations

Problem: number of terms/combinations: $3 \cdot 8 \cdot 4 \cdot 2 = 192$

Script-Generation: Difference of $V(t)$ for Hopping

Custom Python Scripts (SymPy)

Generating Script

simplificationtermhelper.py

Generated Script

analyticalcalcfunctions.py

Beamer: LaTeX way of writing Presentations

Presentations? - Custom Beamer Template

- Write presentations like your papers/thesis in \LaTeX
- Reuse formulas/images/code/sources
- Consistent style & references
- Version control
- Easier collaboration

Minimal example for beamer presentation

Presentations? - Custom Beamer Template

```
1 \documentclass[aspectratio=169]{beamer}
2
3 \usetheme[neutralbackground]{uniamntf}
4
5 \title[]{Title}
6 \subtitle{subtitle}
7 \author{Jonas Kell}
8 \institute[TP III]{Chair for theoretical Physics III}
9 \date[01.05.2024]{\$1^{\text{st}}\$ of Mai 2024}
10
11 \acknowledgement{
12   Jonas Kell\\ Augsburg University\\
13   jonas.kell@student.uni-augsburg.de\\ www.uni-augsburg.de
14 }
15
16 \begin{document}
17
18 \begin{frame}[t,plain]
19   \maketitle
20 \end{frame}
21
22 \section*{Summary \& Conclusion}
23 {
24   \setbeamertemplate{frametitle}{uniamntfack}
25   \begin{frame}[plain]{Acknowledgment}
26     Thank you for your kind attention
27   \end{frame}
28 }
29 \end{document}
```



Thank you for your kind attention



Jonas Kell
Augsburg University
jonas.kell@student.uni-augsburg.de
www.uni-augsburg.de

References I

Summary & Conclusion

- [1] J. Kell, *Math-manipulator tool - info and repository*, (2024)
<https://github.com/jonas-kell/math-manipulator>.
- [2] CTAN: package beamer, <https://ctan.org/pkg/beamer> (visited on 04/11/2024).
- [3] J. Kell, <Https://github.com/jonas-kell/beamer-theme-unia-mntf>, (2024)
<https://github.com/jonas-kell/math-manipulator>.
- [4] J. Kell, *Latex sources for this presentation, the thesis and report*, (2024)
<https://github.com/jonas-kell/master-thesis-documents>.
- [5] J. Kell, *Thesis code implementation and reference*, (2024)
<https://github.com/jonas-kell/master-thesis-code>.
- [6] J. Kell, *Supplementary calculations perfomed with math-manipulator*, (2024)
<https://github.com/jonas-kell/master-thesis-mm-calculations>.

Theme alternative: Faculty of App. Computer Science

Extra slides



Random Text
Testsection
because of the [t] option, text starts at the top. Default is centered.
Hello
This is Text with a delayed show animation.

Title-Here
Testsection Hello

This is a text in first column.
 $E = mc^2$

This text will be in the second column and on a second thoughts, this is a nice looking layout in some cases.

- First item
- Second item
 - aaaaa
 - aaaaaaaaaa
 - aaaaaaaaaa

John Doe (TP III) 1st of Mai 2024 1 / 10

Blocks
Testsection

Block
Sample text

Example
Sample text

Conclusion
Sample text

John Doe (TP III) 1st of Mai 2024 1 / 10

Outline

- 1 Testsection
- 2 Testsection Hello
- 3 Testsection Test
 - sub-1
 - sub-2
- 4 Summary

John Doe (TP III)

Thank you for your kind attention

John Doe
Augsburg University
john.doe@uni-augsburg.de
www.uni-augsburg.de

John Doe (TP III) 1st of Mai 2024 1 / 10

Backup: Simplification V-Parts

Extra slides

-2 c l c# m -1 c l c# m d l d m d# l d# m + 2 c l c# m d l l d# l +1 c l c# m d n d# n

$$\begin{aligned}
 & \left((-2 \cdot c_l c_m^l) + (-1 \cdot c_l c_m^l d_l d_m d_l^l d_m^l) + (2 \cdot c_l c_m^l d_l d_l^l) + (1 \cdot c_l c_m^l d_m d_m^l) \right) \\
 & \left((-2 \cdot c_l c_m^l) + (-1 \cdot c_l c_m^l d_l d_m d_l^l d_m^l) + (2 \cdot c_l c_m^l \cdot (1 + d_l^l d_l)) + (1 \cdot c_l c_m^l d_m d_m^l) \right) \\
 & \left((-2 \cdot c_l c_m^l) + (-1 \cdot c_l c_m^l d_l d_m d_l^l d_m^l) + (2 \cdot c_l c_m^l \cdot (1 + d_l^l d_l)) + (c_l c_m^l \cdot (1 + d_m^l d_m)) \right) \\
 & \left((-2 \cdot c_l c_m^l) - (1 \cdot c_l c_m^l d_l d_l^l d_m^l) + (2 \cdot c_l c_m^l \cdot (1 + d_l^l d_l)) + (c_l c_m^l \cdot (1 + d_m^l d_m)) \right) \\
 & \left((-2 \cdot c_l c_m^l) - (c_l c_m^l \cdot (1 + d_l^l d_l)) \cdot (d_m d_m^l) + (2 \cdot c_l c_m^l \cdot (1 + d_l^l d_l)) + (c_l c_m^l \cdot (1 + d_m^l d_m)) \right) \\
 & \left((-2 \cdot c_l c_m^l) - (c_l c_m^l \cdot (1 + d_l^l d_l)) \cdot (1 + d_m^l d_m) + (2 \cdot c_l c_m^l \cdot (1 + d_l^l d_l)) + (c_l c_m^l \cdot (1 + d_m^l d_m)) \right) \\
 & \left((-2 \cdot c_l c_m^l) - (c_l c_m^l \cdot 1 \cdot 1) - (c_l c_m^l \cdot 1 \cdot d_l^l d_m) - (c_l c_m^l d_l^l d_l \cdot 1) - c_l c_m^l d_l^l d_l d_m^l d_m + (2 \cdot c_l c_m^l \cdot (1 + d_l^l d_l)) + (c_l c_m^l \cdot (1 + d_m^l d_m)) \right) \\
 & \left((-2 \cdot c_l c_m^l) - (c_l c_m^l \cdot 1 \cdot 1) - (c_l c_m^l \cdot 1 \cdot d_m^l d_m) - (c_l c_m^l d_l^l d_l \cdot 1) - c_l c_m^l d_l^l d_l d_m^l d_m + (2 \cdot c_l c_m^l \cdot (1 + d_l^l d_l)) + (2 \cdot c_l c_m^l \cdot (1 + d_m^l d_m)) \right) \\
 & \left((-2 \cdot c_l c_m^l) - (c_l c_m^l \cdot 1 \cdot 1) - (c_l c_m^l \cdot 1 \cdot d_m^l d_m) - (c_l c_m^l d_l^l d_l \cdot 1) - c_l c_m^l d_l^l d_l d_m^l d_m + (2 \cdot c_l c_m^l \cdot 1) + (2 \cdot c_l c_m^l d_l^l d_l) + (c_l c_m^l \cdot (1 + d_m^l d_m)) \right) \\
 & \left((-2 \cdot c_l c_m^l) - (c_l c_m^l \cdot 1 \cdot 1) - (c_l c_m^l \cdot 1 \cdot d_m^l d_m) - (c_l c_m^l d_l^l d_l \cdot 1) - c_l c_m^l d_l^l d_l d_m^l d_m + (2 \cdot c_l c_m^l \cdot 1) + (2 \cdot c_l c_m^l d_l^l d_l) + (c_l c_m^l \cdot 1) + c_l c_m^l d_l^l d_m \right) \\
 & \left(-c_l c_m^l d_l^l d_l - c_l c_m^l d_l^l d_l d_m^l d_m + (2 \cdot c_l c_m^l d_l^l d_l) \right) \\
 & \left(c_l c_m^l d_l^l d_l - c_l c_m^l d_l^l d_l d_m^l d_m \right)
 \end{aligned}$$

Variables:

Macros:	
c ⁺	<code>bc("c+", #0)</code>
c	<code>bc("c", #0)</code>
d ⁺	<code>bc("d+", #0)</code>
d	<code>bc("d", #0)</code>
displayl	<code>\#mathrm{l}</code>
displaym	<code>\#mathrm{m}</code>
l	<code>dist(displayl)</code>
m	<code>dist(displaym)</code>

$$\hat{V}_{\text{Part A}}(l, m) \stackrel{\text{MM}}{=} \sum_{\sigma \in \{\uparrow, \downarrow\}} \hat{c}_{l, \sigma}^S \hat{c}_{m, \sigma}^{\dagger S} \left(1 + 2 \cdot \hat{n}_{l, \bar{\sigma}}^S \hat{n}_{m, \bar{\sigma}}^S - \hat{n}_{l, \bar{\sigma}}^S - \hat{n}_{m, \bar{\sigma}}^S \right)$$

$$\hat{V}_{\text{Part B}}(l, m) \stackrel{\text{MM}}{=} \sum_{\sigma \in \{\uparrow, \downarrow\}} \hat{c}_{l, \sigma}^S \hat{c}_{m, \sigma}^{\dagger S} \left(\hat{n}_{m, \bar{\sigma}}^S - \hat{n}_{l, \bar{\sigma}}^S \hat{n}_{m, \bar{\sigma}}^S \right)$$

$$\hat{V}_{\text{Part C}}(l, m) \stackrel{\text{MM}}{=} \sum_{\sigma \in \{\uparrow, \downarrow\}} \hat{c}_{l, \sigma}^S \hat{c}_{m, \sigma}^{\dagger S} \left(\hat{n}_{l, \bar{\sigma}}^S - \hat{n}_{l, \bar{\sigma}}^S \hat{n}_{m, \bar{\sigma}}^S \right)$$