OPTIMASI PORTOFOLIO BERBASIS MODEL MARKOWITZ MENGGUNAKAN METODE *VARIATIONAL QUANTUM EIGENSOLVER*

Nama	: PUTRA NAUFAL TABASSAM
NRP	: 5001221120
Departemen	: FISIKA
Pembimbing	: Dr. Heru Sukamto, M.Si.
	: Dr. Gagus K. Sunnardianto

Abstrak

Model varians-rerata yang digagas oleh Markowitz merupakan model matematika untuk strategi optimasi portofolio investasi dengan metode pencarian alokasi asset yang memaksimalkan ekspektasi imbal hasil sekaligus meminimalkan varians sebagai risiko. Model tersebut membutuhkan algoritma optimasi yang mumpuni untuk menghasilkan portofolio optimal dari data historis harga pasar. Namun, model optimasi konvensional seringkali mengalami kendala dalam menangkap dinamika pasar modal yang kompleks, terutama saat menangani korelasi antar aset dalam jumlah besar. Untuk mengatasi hal tersebut, tugas akhir ini bertujuan untuk mengoptimalkan alokasi portofolio investasi berdasarkan model varians-rerata yang ditingkatkan lewat kerangka kerja optimasi portofolio hibrida dengan mengintegrasikan prinsip teori permainan potensial eksak ke dalam algoritma *Variational Quantum Eigensolver* (VQE). Kombinasi ini dilakukan dengan pemetaan permasalahan alokasi portofolio ke dalam bentuk Hamiltonian Ising lewat penyetaraan yang diturunkan dari transformasi QUBO. Untuk mempercepat konvergensi proses optimasi pada komputer kuantum, algoritma pencarian ekuilibrium Nash digunakan sebagai inisialisasi keadaan awal alih-alih menggunakan inisialisasi acak. Hasil simulasi menggunakan data saham indeks IHSG periode 2021–2023 menunjukkan bahwa strategi hibrida GT-VQE secara keseluruhan mengungguli metode klasik dan kuantum murni dengan capaian rasio Sharpe sebesar 1,0132 pada sistem empat aset. Integrasi teori permainan terbukti meningkatkan kestabilan pertumbuhan modal di tengah volatilitas pasar serta mengefisiensikan kedalaman sirkuit kuantum. Temuan ini mengukuhkan potensi pendekatan kuantum-klasik hibrida sebagai solusi praktis yang tangguh untuk optimasi portofolio pada era NISQ.

Kata kunci : *Ekonofisika, Game Theory, Hamiltonian Ising, Optimasi Portofolio, VQE*

Halaman ini sengaja dikosongkan

PORTFOLIO OPTIMIZATION BASED ON MARKOWITZ MODEL USING VARIATIONAL QUANTUM EIGENSOLVER METHOD

Name	: PUTRA NAUFAL TABASSAM
NRP	: 5001221120
Department	: FISIKA
Advisor	: Dr. Heru Sukamto, M.Si.
	: Dr. Gagus K. Sunnardianto

Abstract

The Mean-Variance model introduced by Markowitz is a mathematical framework for investment portfolio optimization strategies that seeks an asset allocation to maximizing expected return while minimizing variance as a measure of risk. This model requires a robust optimization algorithm to generate an optimal portfolio from historical market price data. However, conventional optimization models often face challenges in capturing complex capital market dynamics, particularly when handling correlations among a large number of assets. To address this, this undergraduate thesis aims to optimize investment portfolio allocation based on the Mean-Variance model, enhanced by a hybrid portfolio optimization framework that integrates game theory principles (Exact Potential Game) into the Variational Quantum Eigensolver (VQE) algorithm. This combination is achieved by mapping the portfolio allocation problem into an Ising Hamiltonian form through an equivalence derived from the Quadratic Unconstrained Binary Optimization (QUBO) transformation. To accelerate the convergence of the optimization process on a quantum computer, a Nash Equilibrium search algorithm is utilized as an initial state initialization (warm-start) instead of relying on random initialization. Simulation results using IHSG index stock data for the 2021–2023 period demonstrate that the GT-VQE hybrid strategy overall outperforms both classical and pure quantum methods, achieving a Sharpe Ratio of 1.0132 in a four-asset system. The integration of game theory is proven to enhance capital growth stability amidst market volatility and improve quantum circuit depth efficiency. These discoveries confirm the potential of the hybrid quantum-classical approach as a robust practical solution for portfolio optimization in the Noisy Intermediate-Scale Quantum (NISQ) era.

Keywords : *Econophysics, Game Theory, Ising Hamiltonian, Portfolio Optimization, VQE*

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Alhamdulillahi Rabbil 'Alamin. Segala puja dan puji syukur dipanjatkan kehadirat Allah SWT karena telah melimpahkan rahmat, hidayah, petunjuk, kekuatan, dan rezeki kepada penulis. Tak lupa sholawat serta salam penulis haturkan kepada junjungan dan panutan mulia Nabi Muhammad SAW yang membawa dunia dari jalan kegelapan menuju jalan yang terang benderang. Dengan demikian, berkat rahmat Allah SWT serta risalah Nabi Muhammad SAW, penulis dapat menyelesaikan Tugas Akhir yang berjudul:

OPTIMASI PORTOFOLIO BERBASIS MODEL MARKOWITZ MENGGUNAKAN *VARIATIONAL QUANTUM EIGENSOLVER*

sebagai salah satu syarat kelulusan Program Sarjana Fisika FSAD ITS.

Penulisan Tugas Akhir ini dapat terselesaikan dengan baik berkat hadirnya bantuan dan bimbingan dari berbagai pihak. Penulis menyampaikan terima kasih kepada semua pihak yang telah membantu atas terselesaikannya Tugas Akhir ini.

1. Tuhan yang maha Esa karena tanpa kasih sayang dan rahmat-Nya, penulis tidak akan bisa sampai di titik ini.

2. Ibu dan Ayah yang senantiasa memberikan dukungan, doa, dan harapan kepada penulis.

3. Bapak Dr. Heru Sukamto, M.Si. selaku dosen pembimbing I dan Bapak Gagus Ketut Sunnardianto, Ph.D. selaku dosen pembimbing II yang telah meluangkan waktu, tenaga, dan pikiran untuk membimbing penulis.

4. Bapak Prof. Agus Purwanto, D.Sc. dan Ibu Fahmi Astuti, M.Si., Ph.D. selaku dosen penguji yang telah meluangkan energi dan ilmunya sehingga penulis belajar untuk menjadi lebih teliti dan belajar untuk dapat menyampaikan ilmu agar lebih mudah dimengerti.

5. Teman-teman yang meluangkan waktunya untuk hadir pada sidang Tugas Akhir mulai dari sidang terbuka hingga sidang "tertutup" tanggal 9 Juli 2026 karena doa, kehadiran, dan tawa para audiens saat sidang sangat membantu meringankan beban penulis dalam menjawab pertanyaan dari dosen penguji.

6. Teman-teman Fisika ITS angkatan 2022 *Betelgeuse* yang menjadi keluarga penulis selama berkuliah di Fisika ITS. Terima kasih penulis khususnya kepada teman-teman kelompok belajar *RUDAL.CORP* karena telah mewarnai perjuangan selama berkuliah dan membantu penulis dalam menjalani perkuliahan sehingga tidak ada mata kuliah yang mengulang sepanjang masa kuliah.

7. Anggota Laboratorium Fisika Teori dan Fislsafat Alam (LaFTiFA) sebagai tempat penulis berkembang dan belajar selama satu semester penuh. Terkhusus Ilham yang sudah menjerumuskan penulis ke LaFTiFA dan menjadi partner dari maba hingga tugas akhir, Alief yang sudah menjadi "dosen pembimbing III" yang selalu menuntut penulis untuk belajar hal-hal penting dalam menyusun tugas akhir, Lail

yang telah meminjamkan catatan kuliah teorinya agar penulis dapat mengejar ketertinggalan mata kuliah teori, dan Agung yang telah menjadi teman diskusi dalam merencanakan masa depan.

8. Seluruh pihak yang tidak bisa disebuthkan satu-persatu yang selalu memberikan dukungan, doa, dan harapan dalam menyelesaikan tugas akhir ini.

Penulis menyadari bahwa dalam penulisan ini masih terdapat kekurangan, sehingga segala kritik dan saran yang sifatnya membangun sangat diharapkan untuk kesempurnaan Tugas Akhir ini. Penulis berharap Tugas Akhir ini dapat bermanfaat bagi penulis sendiri pada khususnya dan pembaca pada umumnya.

Surabaya, 20 Juli 2026

Penulis

Daftar Isi

Halaman Judul **i**

Lembar Pengesahan **iii**

Approval Sheet **v**

Lembar Orisinalitas **vii**

Statement of Originality **ix**

Abstrak **xi**

Abstract **xiii**

Kata Pengantar **xv**

Daftar Isi **xvii**

Daftar Gambar **xxi**

Daftar Tabel **xxiii**

Daftar Kode **xxiv**

Daftar Istilah **xxvii**

1 Pendahuluan **1**
1.1 Latar Belakang 1
1.2 Rumusan Masalah 3
1.3 Tujuan Penelitian 3
1.4 Batasan Masalah 3
1.5 Manfaat Penelitian 4

2 Tinjauan Pustaka **5**
2.1 Pasar Finansial 5
2.1.1 Sejarah Uang 6
2.1.2 Investasi 7
2.2 Representasi Data Finansial 8
2.2.1 Analisis Kualitatif dalam Pasar Finansial 8

2.2.2 Representasi Data pada Pergerakan Harga 9
2.2.3 Analisis Kuantitatif 10
2.2.4 Model Markowitz 12
2.3 Teori Permainan dan Kerangka Permainan Potensial Eksak 14
2.3.1 Teori Permainan 14
2.3.2 Kerangka Permainan Potensial Eksak sebagai Algoritma Pencarian Ekuilibrium Nash 15
2.3.3 Kerangka EPG dari Model Markowitz 16
2.4 Hamiltonian Ising sebagai Representasi Sistem Kuantum 18
2.4.1 Sejarah dan Evolusi Model Ising 18
2.4.2 Representasi Fisik Aset Finansial 21
2.4.3 Pemetaan Kuantum: QUBO menjadi basis Pauli-Z 22
2.5 Komputer Kuantum 25
2.5.1 Sistem Qubit 25
2.5.2 Manipulasi Informasi pada Gerbang 1 Qubit 27
2.5.3 Manipulasi Informasi pada Gerbang 2 Qubit 31
2.5.4 Sirkuit Kuantum Berparameter 34
2.6 Optimasi Parameter pada Sistem Klasik 36
2.6.1 Metode Pergeseran Parameter (*Parameter Shift Rule*) 36
2.6.2 Optimasi SPSA (*Simultaneous Perturbation Stochastic Approximation*) 38
2.7 Algoritma VQE (*Variational Quantum Eigensolver*) 40
2.7.1 Prinsip Variasi Rayleigh-Ritz dalam Mekanika Kuantum 41
2.7.2 Teorema Rayleigh–Ritz dan Dasar Algoritma VQE 43
2.7.3 Ansatz UCC (*Unitary Coupled Cluster*) 44
2.7.4 Arsitektur Ansatz EfficientSU2 45
2.8 Validasi *Barren Plateau* 46
2.8.1 *Barren Plateau* 46
2.8.2 Entropi Von Neumann 46

3 Metodologi **49**
3.1 Alur Penelitian 49
3.2 Akuisisi dan Pra-pemrosesan Data 50
3.3 Pemodelan Hamiltonian Portofolio dan Pemetaan Qubit 50
3.4 Algoritma Pencarian Ekuilibrium Nash dengan *Sequential Best Response* (SBR) 51
3.5 Optimasi Variasional GT-VQE Adaptif 52
3.6 Simulasi Backtesting dan Metrik Evaluasi 53
3.7 Optimasi Klasik SLSQP (*Sequential Least Squares Programming*) 54
3.8 Validasi Solusi Eksak *Brute Force* 54

4 Hasil dan Pembahasan **57**
4.1 Akuisisi dan Prapemrosesan Data Runtun Waktu Saham 58
4.1.1 Imbal Hasil dan Kovariansi 59
4.1.2 Implementasi Numerik pada Sampel Data 62
4.1.3 Parameter penghindaran risiko Endogen 63
4.2 Formulasi EPG dan Pencarian Ekuilibrium Nash 64
4.2.1 Formulasi EPG 65

4.2.2 Pencarian Ekuilibrium Nash 65
4.3 Pemetaan Hamiltonian Ising 68
4.3.1 Implementasi Numerik Hamiltonian Ising 68
4.4 Arsitektur Ansatz dalam VQE 70
4.4.1 Konstruksi *Hardware-Efficient Ansatz* (HEA) 70
4.4.2 Implementasi Sirkuit Kuantum Variasional 70
4.5 Optimasi *Simultaneous Perturbation Stochastic Approximation* (SPSA) . . 72
4.5.1 Dinamika Pembaruan Parameter SPSA 72
4.5.2 Implementasi Numerik dan Konvergensi Energi Optimal 74
4.6 Evolusi Entropi Von Neumann dan Distribusi Probabilitas 76
4.7 Validasi Solusi melalui Algoritma *Brute Force* 77
4.7.1 Kalkulasi Energi Kombinatorial 77
4.7.2 Implementasi *Brute Force* Berbasis Python 78
4.8 Analisis Performa Portofolio dan Dinamika penyeimbangan kembali 79

5 Penutup 91
5.1 Kesimpulan 91
5.2 Saran 91

Daftar Pustaka 93

Lampiran A Operasi Aljabar Linier pada Ruang Hilbert 101
A.1 Notasi Bra dan Ket 101
A.2 Operasi Perkalian Dalam (*Inner Product*) 102
A.3 Operasi Perkalian Luar (*Outer Product*) 102
A.4 Operasi Konjugasi Kompleks dan Adjoin Hermitian 103
A.4.1 Konjugasi Kompleks 103
A.4.2 Transpose dan Sekawan Hermit 103
A.5 Operasi *Trace* Tereduksi 104
A.5.1 Langkah Perhitungan *Trace* Tereduksi 104
A.5.2 Contoh pada Keadaan Bell 104
A.6 Sifat Dasar Operator Linier 105
A.6.1 Linearitas dan Sifat Distributif 105
A.6.2 Komutator dan Non-Komutativitas 106

Lampiran B Penurunan Matriks Umum Gerbang Uniter Satu Qubit 107

Lampiran C Penurunan Operator Rotasi SU(2) dari Ekspansi Taylor 111

Lampiran D Penurunan Diagonalisasi Operator Densitas dan Fungsi Logaritma Operator 115

Lampiran E Data Harga Harian Saham 119

Lampiran F Data Numerik Hasil Simulasi N=2 157
F.1 Parameter Dinamis Hamiltonian N=2 157
F.2 Riwayat Iterasi SBR N=2 158
F.3 Perbandingan Konvergensi VQE Berdasarkan Depth N=2 159

Lampiran G Visualisasi Perbandingan Alokasi Portofolio N=2 (GT vs No-GT) 161

Lampiran H Data Numerik Hasil Simulasi N=4 169
H.1 Parameter Dinamis Hamiltonian N=4 169
H.2 Riwayat Iterasi SBR N=4 171
H.3 Perbandingan Konvergensi VQE Berdasarkan Depth 172

Lampiran I Visualisasi Perbandingan Alokasi Portofolio N=4 (GT vs No-GT) 175

Lampiran J Skrip Visualisasi dan Pembangkitan Grafik Hasil Simulasi 189

Lampiran K Skrip Utama Orkestrasi dan Algoritma Hibrida Kuantum 199

Biodata Penulis 207

Daftar Gambar

2.1 Contoh ikhtisar laporan posisi keuangan konsolidasian yang merepresentasikan struktur aset dan kesehatan fiskal entitas. 9
2.2 Struktur anatomi grafik lilin dari kenaikan (*bullish*) dan penurunan (*bearish*) yang merangkum fluktuasi harga dalam satu periode waktu. 10
2.3 Representasi visual level *support* dan *resistance* sebagai batas memori sosial dalam dinamika harga aset. 11
2.4 Kurva batas efisien sebagai representasi batas efisiensi alokasi aset dalam kerangka kerja model Markowitz. 13
2.5 Matriks imbal hasil pada model permainan klasik 15
2.6 Representasi visual model Ising satu dimensi (1D) 21
2.7 Representasi Bola Bloch sebagai visualisasi geometris keadaan murni sebuah qubit. Posisi vektor keadaan ditentukan oleh sudut polar θ dan sudut azimutal ϕ, yang dipetakan ke koordinat Kartesian (x, y, z). 27
2.8 Simbolisme gerbang Pauli dan gerbang Hadamard pada sirkuit kuantum yang merepresentasikan transformasi uniter dasar X, Y, Z, dan H pada satu qubit. 30
2.9 Simbolisme gerbang rotasi pada sirkuit kuantum yang merepresentasikan rotasi parameter tunggal pada sumbu-X, sumbu-Y, dan sumbu-Z di bola Bloch. 31
2.10 Representasi simbolik gerbang keterbelitan CNOT yang mengilustrasikan interaksi non-lokal antar qubit dalam sistem sirkuit. 32
2.11 Representasi alur kerja sirkuit kuantum berparameter (PQC) yang mengilustrasikan tahap penyandian data, sirkuit variasional, pengukuran, dan pembaruan parameter oleh algoritma optimasi klasik. 35

3.1 Diagram Alir Metodologi Optimasi Portofolio Berbasis GT-VQE (Nash SBR dan Adaptive VQE dengan Penalty Annealing). 51

4.1 Rangkaian Sirkuit Kuantum HEA EfficientSU2 dengan Kedalaman $(d) = 1$. 70
4.2 Pergerakan Harga Saham Individual (*Raw Price*) pada Sistem N=4 (2021–2023). 80
4.3 Visualisasi Pertumbuhan Modal Optimasi Klasik SLSQP pada Sistem N=2. 83
4.4 Visualisasi Pertumbuhan Modal Strategi Nash (Klasik) pada Sistem N=2. 83
4.5 Visualisasi Pertumbuhan Modal VQE Murni (Tanpa GT) pada Sistem N=2. 84
4.6 Visualisasi Pertumbuhan Modal Strategi GT-VQE (Hibrida) pada Sistem N=2. 84
4.7 Visualisasi Pertumbuhan Modal Optimasi Klasik SLSQP pada Sistem N=4. 85
4.8 Visualisasi Pertumbuhan Modal Strategi Nash (Klasik) pada Sistem N=4. 85

4.9 Visualisasi Pertumbuhan Modal VQE Murni (Tanpa GT) pada Sistem N=4. 86
4.10 Visualisasi Pertumbuhan Modal Strategi GT-VQE (Hibrida) pada Sistem N=4. 86
4.11 Perbandingan Dinamika Adaptasi Kedalaman Sirkuit (kedalaman) per Jendela Simulasi antara VQE Murni dan GT-VQE untuk Berbagai Konfigurasi Aset. 87

G.1 Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2021-01-04 161
G.2 Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2022-02-09 162
G.3 Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2022-09-21 163
G.4 Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2022-12-19 164
G.5 Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2023-02-16 165
G.6 Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2023-03-17 166
G.7 Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2023-07-05 167
G.8 Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2023-12-04 168

I.1 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2021-04-06 175
I.2 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2021-07-09 176
I.3 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2021-10-11 177
I.4 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2021-11-10 178
I.5 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-01-10 179
I.6 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-03-14 180
I.7 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-04-12 181
I.8 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-05-23 182
I.9 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-06-23 183
I.10 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-08-23 184
I.11 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-12-19 185
I.12 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2023-08-04 186
I.13 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2023-09-05 187
I.14 Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2023-11-03 188

Daftar Tabel

2.1 Analogi Konsep Fisika Statistik dan Ekonomi Finansial dalam Kerangka Model Ising. 18

3.1 Daftar Simbol dan Notasi Matematis dalam Metodologi. 49

4.1 Matriks Imbal Hasil Strategis Sistem N=2 (Januari 2021). 66
4.2 Ringkasan Performa Finansial Strategi Kuantum, SLSQP, dan *Equal Weight* ($N = 2, 4$). 81

E.1 Data Harga Harian Saham Gabungan (BBCA, ADRO, SMGR, TLKM) . . 119
E.2 Data Simple Return dan Log Return Saham (BBCA, ADRO, SMGR, TLKM) 136
E.3 Ekspektasi Return dan Varians Log Return 36 Periode - Skala 126 Hari (BBCA, ADRO, SMGR, TLKM) 153
E.4 Kovarians Log Return 36 Periode - Skala 126 Hari 154

F.1 Parameter Dinamis Portofolio N=2 (γ, h_{obj}, J_{obj}) 157
F.2 Riwayat Iterasi SBR N=2 158
F.3 Perbandingan Energi Akhir VQE Berdasarkan Depth N=2 159

H.1 Parameter Dinamis Portofolio N=4 (γ, h_{obj}, C_{obj}) 169
H.2 Kompilasi Parameter Bias h_{obj}, Konstanta C_{obj}, dan Koefisien γ N=4 . . . 170
H.3 Riwayat Iterasi SBR N=4 171
H.4 Perbandingan Energi Akhir VQE Berdasarkan Depth N=4 172

Halaman ini sengaja dikosongkan

Daftar Kode

4.1 Skrip ekstraksi data harga historis saham menggunakan API `yfinance` . . 59
4.2 Fungsi prapemrosesan imbal hasil dan ekspektasi pertumbuhan portofolio . 61
4.3 Algoritma kalkulasi parameter penghindaran risiko dinamis berbasis Rasio Sharpe agregat 64
4.4 Algoritma kalkulasi ekspektasi imbal hasil strategis berbasis observasi probabilitas keadaan mikro 65
4.5 Skrip implementasi SBR untuk pencarian konvergensi ekuilibrium Nash . . 66
4.6 Skrip perakitan operator Hamiltonian Ising dinamis menggunakan pustaka `PennyLane` 69
4.7 Fungsi penyusunan arsitektur HEA pada iterasi sirkuit kuantum VQE . . 72
4.8 Implementasi algoritma dasar SPSA 75
4.9 Skrip kalkulasi fungsional matriks evaluasi observabel algoritma Entropi Von Neumann komputasional menggunakan matriks trace tereduksi arsitektur operasional pada ruang fase deterministik fungsi ekuilibrium 77
4.10 Skrip komputasi algoritma *brute force* untuk validasi lanskap energi portofolio pada arsitektur kuantum 79
4.11 Skrip komputasional untuk eksekusi simulasi pengujian historis dengan iterasi jendela bergulir (`backtest_runner.py`) 87
4.12 Fungsi operasional untuk penyesuaian bobot portofolio pada momen evaluasi (`rebalance_portfolio.py`) 90
J.1 Skrip utama generator grafik evaluasi portofolio (`plot_generator.py`) . . 189
J.2 Skrip visualisasi distribusi probabilitas status VQE (`render_vqe_probs.py`) 193
J.3 Skrip visualisasi grafik interaksi antar aset (`project_graph.py`) 195
J.4 Skrip utilitas untuk pengeksporan hasil graf fisis (`export_graph.py`) . . . 196
K.1 Skrip utama orkestrasi keseluruhan parameter dan eksekusi pada sistem N=4 (`main_N4.py`) 199
K.2 Skrip inti loop VQE adaptif (`run_vqe_adaptive.py`) 199
K.3 Skrip eksekusi langkah strategi yang memadukan teori permainan klasik dengan VQE kuantum (`run_strategy_step.py`) 203

Halaman ini sengaja dikosongkan

Daftar Istilah

Istilah	**Definisi**
Uang	Instrumen finansial yang berfungsi sebagai penyimpan nilai dan alat tukar, dalam ekonofisika sering dimodelkan melalui analogi distribusi energi termodinamika.
Pasar	Tempat bertemunya penjuan dan pembeli untuk melakukan transaksi jual beli barang maupun jasa.
Investasi	Komitmen sejumlah modal pada aset finansial dengan ekspektasi memperoleh imbal hasil (*return*) di masa depan, yang melibatkan pertukaran (*trade-off*) antara risiko dan keuntungan.
Hamiltonian	Operator matematika yang merepresentasikan energi total sebuah sistem; dalam konteks optimasi kuantum, *Hamiltonian* Ising digunakan untuk merepresentasikan fungsi biaya (*cost function*).
Ruang Hilbert	Ruang vektor kompleks berdimensi tak hingga dengan dilengkapi produk dalam (*inner product*) yang mampu mendefinisikan panjang vektor dan sudut antara vektor.
Fungsi Biaya	Fungsi yang digunakan untuk mengukur seberapa baik kinerja suatu algoritma optimasi, dengan nilai minimum fungsi biaya menunjukkan solusi optimal.
Return	Tingkat pengembalian atau imbal hasil dari suatu investasi dalam periode waktu tertentu, biasanya dihitung sebagai perubahan persentase harga aset (*simple return*) atau logaritma natural rasio harga (*log return*).
Risiko	Probabilitas atau kemungkinan terjadinya kerugian pada suatu investasi, yang dapat diukur melalui berbagai metode statistik seperti standar deviasi.
Portofolio	Kumpulan atau kombinasi aset keuangan seperti saham, obligasi, dan kas yang dikelola untuk mencapai tujuan investasi tertentu.
Teori Prospek	Model perilaku ekonomi yang menjelaskan pengambilan keputusan dalam kondisi ketidakpastian, di mana individu cenderung lebih sensitif terhadap potensi kerugian dibandingkan keuntungan yang setara (*loss aversion*).
Inflasi	Fenomena kenaikan harga barang dan jasa secara umum dan terus-menerus dalam jangka waktu tertentu yang mengakibatkan penurunan daya beli masyarakat.

Istilah	**Definisi**
Barren Plateau	Fenomena yang terjadi pada algoritma optimasi kuantum di mana landskap fungsi biaya menjadi sangat datar sehingga gradiennya mendekati nol, menyulitkan algoritma klasik untuk menemukan minimum global.
Aset	Sumber daya ekonomi yang memiliki nilai dan dapat dimiliki secara individu maupun kolektif, mencakup aset riil (real estate, komoditas) dan aset finansial (saham, obligasi).
Asimetri Informasi	Kondisi di mana satu pihak dalam transaksi memiliki informasi yang lebih banyak atau lebih baik daripada pihak lainnya, sehingga menciptakan ketidakseimbangan kekuatan pasar.
Volatilitas	Fluktuasi harga aset yang disebabkan oleh faktor-faktor sistemik yang mempengaruhi pasar secara keseluruhan, seperti kebijakan moneter, krisis finansial, atau bencana alam.
Likuiditas	Kemampuan suatu aset untuk diperdagangkan atau dicairkan menjadi uang tunai tanpa menyebabkan perubahan harga yang signifikan.
Herding Behavior	Kecenderungan individu untuk mengikuti tindakan atau keputusan mayoritas dalam suatu kelompok, terlepas dari penilaian rasional individu tersebut terhadap informasi yang tersedia.
Time series	Runtun waktu dari data yang diamati dalam periode tertentu.
Sharpe Ratio	Rasio keuangan yang digunakan untuk mengukur pengembalian portofolio disesuaikan dengan risiko, yang dihitung dengan membagi selisih antara tingkat pengembalian portofolio dengan tingkat pengembalian bebas risiko dengan standar deviasi portofolio.
Rebalancing	Proses penyesuaian bobot aset dalam portofolio yang dilakukan secara berkala untuk mempertahankan tingkat risiko dan imbal hasil yang diinginkan.
Kapitalisasi Pasar	Nilai pasar keseluruhan ekuitas perusahaan yang beredar, yang dihitung dengan mengalikan harga saham per lembar dengan jumlah saham yang beredar.

Bab 1

Pendahuluan

1.1 Latar Belakang

Pasar keuangan modern merupakan hal yang sangat dipengaruhi oleh dinamika nilai tukar, harga komoditas, dan tingkat suku bunga sehingga sering terdistorsi oleh korelasi antar aset, terutama selama periode krisis ekonomi global yang tidak terduga. Hal ini menyebabkan kesulitan bagi investor dalam mengelola risiko dan memilih aset yang optimal untuk diinvestasikan. Seiring berkembangnya matematika, model varian-rerata atau *Mean-Variance* (MV) yang diperkenalkan oleh Markowitz menjadi fondasi utama teori portofolio modern atau *Modern Portfolio Theory* (MPT) untuk mengoptimasi keputusan investasi. Namun model ini memiliki keterbatasan dalam menangkap dinamika pasar yang non-linear dan ketergantungan antar aset secara akurat saat jumlah aset yang dimasukkan ke dalam model sangat banyak. Ketidakmampuan metode klasik dalam menangani ruang pencarian yang sangat luas pada masalah optimasi mendorong perlunya transformasi diskritisasi fungsi objektif portofolio menjadi kerangka Optimasi Biner Kuadratik Tak Terbatas atau *Quadratic Unconstrained Binary Optimization* (QUBO). Pendekatan ini memungkinkan pemanfaatan teknologi komputasi kuantum dalam mengidentifikasi konfigurasi aset yang optimal dengan menggunakan pemetaan masalah ke dalam sistem fisik yang setara dengan pencarian energi terendah pada sistem kuantum.

Secara historis, eksplorasi literatur dan eksperimen berkembang pesat dalam penerapan algoritma kuantum variasional untuk menjawab kompleksitas alokasi aset pada ekosistem finansial modern. A. Wang dkk. (2025) mengusulkan integrasi fungsi biaya Nilai Risiko Bersyarat Berbobot atau *Weighted Conditional Value-at-Risk* (WCVaR) yang dikombinasikan dengan pengoptimal klasik Strategi Evolusi Adaptasi Matriks Kovarians atau *Covariance Matrix Adaptation Evolution Strategy* (CMA-ES) untuk meningkatkan stabilitas konvergensi energi pada arsitektur algoritma Pencari Nilai Eigen Kuantum Variasional atau *Variational Quantum Eigensolver* (VQE). Sementara itu, Wei dkk. (2025) mengembangkan kerangka kerja Metode Annealing Hibrida Kuantum-Klasik atau *Quantum-Classical Hybrid Annealing* (QCHA) untuk menangani masalah optimasi portofolio diskrit dengan membagi struktur *Quadratic Unconstrained Binary Optimization* (QUBO) yang kompleks menjadi sub-masalah yang lebih kecil untuk dieksekusi pada prosesor kuantum. Selain itu, pengembangan model asumsi (Ansatz) berbasis keadaan Dicke dengan kedalaman sirkuit linear telah diusulkan oleh S. Wang dkk. (2025) untuk mengatasi kendala *hardware* pada era Perangkat Komputasi Kuantum Skala Menengah yang Bising atau *Noisy Intermediate-Scale Quantum* (NISQ) sekaligus memperluas skalabilitas

sistem hingga puluhan bit kuantum atau *quantum bit* (*qubit*) melalui skema komputasi terdistribusi secara hibrida. Meskipun berbagai inovasi tersebut telah berhasil mereduksi beban komputasi melalui pendekatan hibrida, inisialisasi parameter awal yang digunakan pada ketiga paper tersebut masih menggunakan inisialisasi acak.

Dari tinjauan literatur tersebut, terdapat beberapa celah penelitian yang berusaha diisi oleh tugas akhir ini. Pertama, fenomena Dataran Datar Tandus atau *Barren Plateaus* merupakan hal yang sering menghambat efisiensi algoritma kuantum variasional pada sistem dengan dimensi parameter yang besar. Salah satu cara untuk mengurangi risiko fenomena tersebut adalah dengan melakukan strategi inisialisasi awal cerdas (*warm-start*) alih-alih menggunakan inisialisasi acak. Selain itu, penggunaan inisialisasi awal ini juga dapat membantu mempercepat konvergensi optimasi yang dilakukan oleh VQE dibandingkan dengan inisialisasi acak. inisialisasi awal yang diajukan pada tugas akhir ini menggunakan konsep Permainan Potensial Eksak atau *Exact Potential Game* (EPG) dari teori permainan atau *Game Theory* (GT) dengan memanfaatkan titik ekuilibrium Nash (*Nash Equilibrium*) sebagai pemandu awal proses optimasi. Kedua, penggunaan entropi Von Neumann sebagai pendeteksi keacakan yang terpilih selama proses optimasi masih jarang dieksplorasi. Padahal, entropi Von Neumann dapat memberikan informasi historis pemilihan distribusi probabilitas pada Ansatz dan tingkat keacakan dalam keadaan kuantum, sehingga dapat digunakan juga untuk mendeteksi fenomena *Barren Plateaus* dan mengoptimalkan strategi inisialisasi awal.

Prinsip EPG diterapkan dengan mencari titik (ekuilibrium Nash) menggunakan algoritma Respons Terbaik Berurutan atau *Sequential Best Response* (SBR). Algoritma SBR bekerja dengan memutakhirkan strategi setiap agen pemain secara berurutan berdasarkan strategi terbaik dari pemain sebelumnya hingga mencapai stabilitas di titik ekuilibrium Nash. Ekuilibrium Nash yang diperoleh adalah konfigurasi biner strategi beli atau jual suatu aset menggunakan fungsi potensial sebagai dasar penentuan strategi terbaik. Titik ekuilibrium Nash tersebut kemudian digunakan sebagai inisialisasi awal pada sirkuit kuantum untuk memandu proses pencarian energi terendah secara sistematis. Kemudian fungsi potensial tersebut ditransformasi menjadi kerangka ising Hamiltonian yang memiliki dua komponen utama, yaitu parameter medan lokal (h_i) dan koefisien kopling (J_{ij}). Parameter medan lokal (h_i) merepresentasikan bias risiko individu yang berkaitan dengan preferensi untuk membeli atau menjual aset, sedangkan koefisien kopling (J_{ij}) merepresentasikan interaksi atau korelasi antar aset. Transformasi ini akan menghasilkan struktur Hamiltonian yang sesuai dengan ide dari model varians-rerata, yaitu meminimalkan ekspektasi varians sambil tetap mempertahankan ekspektasi imbal hasil yang diinginkan.

Proses optimasi dilanjutkan dengan memasukkan hamiltonian yang diperoleh ke dalam sirkuit kuantum berparameter. Di dalam sirkuit tersebut, dilakukan pembaruan parameter sirkuit secara iteratif menggunakan Ansatz EfficientSU(2). Secara umum, Ansatz ini dirancang untuk meminimalkan energi sistem kuantum pada perangkat keras kuantum. Proses optimasi ini mengoptimalkan parameter sudut rotasi (θ) dengan optimasi klasik algoritma Aproksimasi Stokastik dengan Perturbasi Simultan atau *Simultaneous Perturbation Stochastic Approximation* (SPSA) yang memiliki ketangguhan dan efisiensi tinggi dalam memberikan pembaruan parameter berdasarkan estimasi gradien. Selain itu, penggunaan metode penganilan penalti (*Penalty Annealing*) juga diintegrasikan ke dalam optimasi agar sistem kuantum tetap berada dalam ruang solusi yang memenuhi batasan investasi selama proses transisi energi menuju keadaan dasar (*ground*

state) berlangsung. Secara hipotesis, metode gabungan *game theory* dan algoritma VQE (GT-VQE) yang dilakukan pada tugas akhir ini akan menghasilkan performa yang lebih baik dibandingkan dengan implementasi VQE tanpa menggunakan inisialisasi awal dan mengefisiensi optimasi karena cenderung memiliki kedalaman Ansatz yang lebih rendah sehingga mengurangi risiko terjebak di *barren plateaus*.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, permasalahan yang akan dibahas dalam tugas akhir ini adalah:

1. Bagaimana perumusan bias aset sebagai parameter medan lokal (h_i) dan interaksi antar aset sebagai koefisien kopling (J_{ij}) menggunakan fungsi potensial dalam kerangka EPG memengaruhi akurasi pemetaan Hamiltonian?

2. Bagaimana efektivitas penerapan ekuilibrium Nash sebagai strategi inisialisasi awal dalam mempercepat konvergensi dan mengatasi kendala inisialisasi pada algoritma VQE?

3. Bagaimana performa algoritma VQE dengan optimasi SPSA dan Ansatz EfficientSU(2) dalam menemukan solusi keadaan dasar pada model portofolio hibrida?

1.3 Tujuan Penelitian

Dari rumusan masalah yang diberikan, tujuan dari pembuatan tugas akhir ini antara lain:

1. Menganalisis formulasi bias aset sebagai parameter medan lokal (h_i) dan interaksi antar aset sebagai koefisien kopling (J_{ij}) melalui pendekatan fungsi potensial dalam sistem Hamiltonian berbasis EPG.

2. Mengevaluasi kontribusi strategi inisialisasi awal dari ekuilibrium Nash terhadap efisiensi pencarian solusi optimal pada algoritma VQE.

3. Menguji kapabilitas algoritma VQE yang menggunakan optimasi SPSA dan Ansatz EfficientSU(2) untuk menghasilkan keputusan alokasi aset yang optimal dalam kondisi pasar yang kompleks.

1.4 Batasan Masalah

Agar penelitian ini lebih terfokus, maka pengerjaan tugas akhir ini diberikan batasan-batasan sebagai berikut:

1. Portofolio yang dianalisis dibatasi pada pemilihan 1 aset dari 2 pilihan aset pada sistem $N = 2$ dan 2 aset dari 4 pilihan aset pada sistem $N = 4$.

2. Simulasi dan implementasi algoritma dilakukan menggunakan simulator *Pennylane* tanpa mempertimbangkan derau.

3. Tugas akhir ini hanya merupakan bukti konsep dari implementasi inisialisasi awal dari ekuilibrium Nash terhadap performa algoritma VQE dalam menemukan solusi keadaan dasar pada model portofolio hibrida.

4. Analisis entropi Von Neumann dibatasi sebagai indikator perubahan distribusi probabilitas keadaan kuantum selama proses optimasi VQE dan tidak digunakan sebagai kajian mendalam mengenai sifat keterbelitan dari Ansatz.

5. Analisis berfokus pada pengaruh inisialisasi awal dari ekuilibrium Nash terhadap proses optimasi VQE, bukan untuk membuktikan keunggulan komputasi kuantum terhadap metode klasik.

1.5 Manfaat Penelitian

Hasil dari tugas akhir ini diharapkan dapat memberikan alternatif metode optimasi portofolio yang lebih tangguh terhadap dinamika korelasi pasar serta memberikan kontribusi akademis dalam penerapan komputasi kuantum pada bidang finansial.

Bab 2

Tinjauan Pustaka

Ekonofisika merupakan bidang penelitian interdisipliner yang mengaplikasikan teori dan metode dari fisika untuk memahami fenomena ekonomi yang dinamis dan stokastik (Baaquie dkk., 2003). Pada bab tinjauan pustaka ini disajikan landasan-landasan teoretis secara komprehensif untuk menjembatani dinamika pasar finansial dengan algoritma komputasi kuantum melalui serangkaian pemodelan strategis dan fisisnya. Tinjauan pustaka dimulai dari pengenalan pasar finansial dan teknik analisa lewat representasi data pasar. Kemudian dilanjutkan dengan penjabaran formalisme optimasi portofolio model Markowitz dan perluasan pendekatan optimasi portofolio ke dalam kerangka kerja teori permainan untuk menangkap interaksi antar agen ekonomi. Selanjutnya, dijabarkan pula proses formulasi dari pendekatan tersebut yang ditransformasikan ke dalam struktur Hamiltonian Ising melalui transformasi QUBO agar dapat dieksekusi ke dalam sirkuit kuantum berparameter. Untuk memperdalam karakteristik rangkaian kuantum yang digunakan dalam tugas akhir ini, akan diuraikan pula mengenai arsitektur Ansatz dan mekanisme optimasi hibrida kuantum-klasik dalam VQE untuk mencapai konvergensi energi dasar. Terakhir, dijabarkan pula teknik evaluasi performa rangkaian kuantum menggunakan entropi Von Neumann dan analisis varians gradien untuk mengidentifikasi ketangguhan sistem kuantum terhadap fenomena *Barren Plateau*.

2.1 Pasar Finansial

Dalam perspektif ekonomi formal, uang didefinisikan sebagai alat tukar atau standar pengukur nilai secara sah yang diterbitkan oleh otoritas moneter suatu negara guna memfasilitasi transaksi dalam skala makro maupun mikro secara efisien (Kolmogorov, 1968). Uang berfungsi sebagai entitas fundamental yang menggerakkan sirkulasi kekayaan melalui pasar finansial, yaitu sebuah mekanisme kolektif yang mempertemukan penawaran dan permintaan terhadap berbagai instrumen aset ekonomi (Mantegna dkk., 2000). Pasar finansial berfungsi sebagai tempat pertukaran aset keuangan yang memungkinkan terjadinya realokasi modal dari pihak yang memiliki surplus kepada pihak yang membutuhkan pendanaan agar dapat mendukung produktivitas sektor riil dengan jaminan kepemilikan aset sebagai investasi yang diharapkan dapat kembali dengan nilai yang lebih tinggi di masa depan. Pada dasarnya, pertumbuhan ekonomi dan fluktuasi harga di pasar sangat dipengaruhi oleh interaksi antar pelaku pasar dalam merespons ketidakpastian informasi pasar. Mekanisme pertukaran ini terus berevolusi hingga membentuk arsitektur jaringan makroekonomi yang luas untuk mengatur distribusi sumber daya secara optimal baik

korelasi linear antar aset maupun non-linear bergantung pada kondisi likuiditas yang tersedia di pasar (Mantegna dkk., 2000). Seluruh kumpulan dari berbagai institusi perbankan, instrumen investasi, serta mekanisme regulasi formal yang mengatur sirkulasi dan akumulasi dari arus uang tersebut membentuk sebuah integrasi sistem yang disebut sebagai sistem finansial.

2.1.1 Sejarah Uang

Dalam sejarah manusia, sistem pertukaran nilai mengalami evolusi dimulai dari sistem barter yang masih dapat digunakan untuk memenuhi kebutuhan dasar yang sifatnya terbatas. Namun, manusia juga memiliki kecenderungan untuk memenuhi keinginannya yang bersifat taka terbatas sehingga sumber daya yang digunakan pada sistem barter tidak dapat memenuhi keinginan tersebut secara efisien sehingga menciptakan koflik kelangkaan. Fenomena tersebut mendasari munculnya instrumen uang sebagai alat tukar yang tidak mengalami kerusakan, bersifat langka, dan disepakati oleh seluruh komunitas sebagai alat transaksi yang sah. Seiring dengan berjalannya waktu, munculnya kebutuhan tersebut mendorong terciptanya sistem uang komoditas yang terbuat dari benda-benda berharga seperti logam mulia. Logam mulia menjadi sarana pertukaran nilai yang efisien namun pada akhirnya mengalami masalah akomodasi saat terjadi transaksi besar mengingat logam mulia memiliki bobot yang berat dan dimensi yang cukup besar. Selain itu, saat uang logam dalam proses akomodasi, sering terjadi perampokaan dan pencurian di tengah perjalanan sehingga menyebabkan masalah keamanan dalam akomodasi. Hingga pada dinasti Song di Tiongkok, diciptakanlah uang kertas bernama *jiaozi Wu* sebagai alat tukar yang lebih efisien, aman, dan ringan untuk mengakomodasi transaksi besar di mana uang kertas tersebut dijamin dengan logam yang disimpan di gudang-gudang pemerintah Song.

Adopsi uang kertas di Eropa dimulai melalui penggunaan surat utang oleh para bankir di Swedia dan Inggris sebagai representasi kepemilikan cadangan logam mulia (Shaikh dkk., 2025). Karena efisiensinya, surat utang tersebut pada akhirnya digunakan sebagai sarana pertukaran nilai antar wilayah yang ada di Eropa (Ibrahim dkk., 2020). Sistem standar emas kemudian menjadi landasan stabilitas moneter global sebagai jaminan konvertibilitas nilai mata uang masing-masing negara terhadap emas fisik hingga berakhirnya era kolonialisme dan pecahnya konflik dunia pada awal abad ke-20 (Jiang dkk., 2023). Setelah perang dunia kedua berakhir, di tetapkanlah sistem nilai tukar baru di bawah naungan Perjanjian Bretton Woods yang menetapkan dolar Amerika Serikat sebagai satu-satunya mata uang cadangan dunia yang dapat dikonversikan terhadap nilai emas agar tercipta ketertiban ekonomi internasional (X. Yang dkk., 2024). Namun demikian, tekanan defisit fiskal yang dialami oleh otoritas Amerika Serikat pada akhirnya memicu kebijakan unilateral presiden Richard Nixon pada tahun 1971 yang dikenal sebagai Guncangan Nixon (*Nixon Shock*). Pada peristiwa tersebut, presiden Nixon menyatakan bahwa dolar amerika tidak lagi dapat dikonversikan terhadap nilai emas fisik dengan nilai yang tetap sehingga menandai era uang fiat sepenuhnya. Uang fiat ini lah yang digunakan oleh masyarakat di dunia di mana nilai nominal instrumen moneter tidak lagi didukung oleh komoditas fisik melainkan bersandar pada kredibilitas otoritas pemerintah dan konsensus kepercayaan masyarakat global (DeMiguel dkk., 2009).

Karena sifatnya yang mengambang uang fiat sangat rentan terhadap inflasi jika kebijakan moneter di suatu negara terlampau longgar atau bahkan tidak terkendali. Fe-

nomena inflasi muncul sebagai konsekuensi dari peningkatan jumlah uang beredar yang melebihi laju pertumbuhan barang dan jasa, sehingga mengakibatkan penurunan daya beli masyarakat secara progresif dalam jangka panjang. Dalam sebuah komunitas ekonomi, inflasi yang tidak terkendali dapat menyebabkan kekacauan alokasi sumber daya yang berakibat pada redistribusi kekayaan menjadi tidak efisien antara pihak kreditur dan debitur karena perubahan nilai mata uang. Dampak inflasi juga dapat meluas pada neraca perdagangan dan daya saing produk domestik antar negara karena terjadi fluktuasi nilai tukar (Gabaix, 2020). Untuk merespons tekanan inflasi tersebut, para pelaku ekonomi akan mencari instrumen lindung nilai (*hedging*) yang mampu mempertahankan nilai aset riil dari degradasi daya beli yang disebabkan oleh ketidakstablian moneter yang terjadi di negaranya. Investasi merupakan salah satu cara dalam usaha lindung nilai untuk mengatasi inflasi dengan resiko yang terkendali.

2.1.2 Investasi

Investasi merupakan usaha mengalokasikan uang sebagai sumber daya finansial ke dalam aset-aset tertentu dengan harapan nilai aset tersebut dapat meningkat di masa mendatang. Tindakan investasi ini memiliki fungsi utama sebagai usaha lindung nilai kekayaan investornya di tengah sistem moneter yang sangat rentan terhadap tekanan inflasi. Selain itu, investasi juga dapat meningkatkan produktivitas ekonomi bagi negara atau perusahaan yang mendapatkan investasi sebagai modal awal untuk dapat dikelola hingga mendapatkan keuntungan di dalam arus kas mereka. Jumlah pendapatan tersebut kemudian dapat diterima investor melalui kenaikan harga di pasar finansial dan distribusi keuntungan berkala yang disebut sebagai deviden. Dalam berinvestasi, investor perlu melakukan mitigasi risiko penurunan daya beli masyarakat yang menyebabkan harga aset akan turun di pasar finansial. Oleh karena itu, para investor perlu membuat strategi investasi yang tepat dan sesuai dengan profil risiko yang mereka miliki (Onnela dkk., 2003).

Strategi investasi yang digunakan oleh para investor sangat bervariasi, mulai dari pendekatan konvensional yang mengutamakan keamanan modal hingga strategi agresif yang mengejar pertumbuhan nilai aset secara eksponensial dalam rentang waktu yang relatif singkat. Keragaman strategi ini didasarkan oleh prinsip kompromi antara risiko dan imbal hasil (*risk-return trade-off*) yang mana menyatakan bahwa ekspektasi imbal hasil yang tinggi selalu berbanding lurus dengan tingkat risiko atau ketidakpastian yang harus ditanggung dalam mempertaruhkan modalnya. Sebagai agen rasional, investor berusaha mencapai potensi keuntungan yang maksimal dengan memilih berbagai instrumen finansial yang sesuai dengan batas toleransi risiko sehingga performa portofolio menjadi optimal namun tetap dengan risiko yang terjaga. Hal tersebut dapat dicapai dengan strategi diversifikasi yaitu menyebarkan modal investasi dengan bobot tertentu ke berbagai macam instrumen finansial untuk menyeimbangkan proporsi antara aset berisiko dan aset aman sehingga dapat meminimalkan dampak volatilitas pasar (DeMiguel dkk., 2009).

Dalam strategi diversifikasi, terdapat parameter penghindaran risiko (*risk aversion*) yang pada umumnya dipilih berdasarakn preferensi investor. Untuk investor konservatif seperti ibu rumah tangga dan pensiunan, parameter penghindaran risiko umumnya bernilai tinggi sehingga cenderung memilih keputusan investasi dengan aset berisiko rendah yang lebih likuid dan stabil seperti obligasi pemerintah. Sedangkan untuk investor agresif, parameter penghindaran risiko akan bernilai rendah sehingga cenderung memper-

taruhkan modal pada aset-aset yang berpotensi mengalami kenaikan signifikan walaupun berisiko tinggi. Namun pada ekonomi modern, nilai parameter tersebut dapat bersifat endogen yaitu nilainya dipilih secara dinamis mengikuti fluktuasi performa aset dan deviasi pergerakan harga aset di pasar modal (Levy & Levy, 1991). Landasan perilaku ini didukung oleh Teori Prospek yang dikembangkan oleh (Kahneman & Tversky, 1979). Teori ini menjelaskan bahwa para agen cenderung menunjukkan respon psikologis yang asimetris, di mana rasa sakit akibat kerugian dirasakan jauh lebih besar dibandingkan kepuasan yang diperoleh dari keuntungan investasi (Kahneman & Tversky, 1979).

2.2 Representasi Data Finansial

Pasar finansial merupakan sistem yang dinamis namun pergerakannya tidak bersifat acak murni karena dipengaruhi oleh interaksi strategis antar pelaku pasar yang terstruktur (Schwert, 2003). Untuk mengidentifikasi peluang profitabilitas dari dinamika tersebut, investor menerapkan metodologi analisis kualitatif dan teknikal untuk mengolah informasi yang relevan dari fluktuasi harga historis dan memanfaatkannya untuk memprediksi pergerakan harga di masa depan. Representasi data finansial dalam memetakan perilaku kolektif para agen ekonomi menjadi bentuk visual bagi para analis pasar profesional. Data mentah tersebut diproses menjadi data yang terstruktur sehingga memungkinkan dilakukannya analisis melalui pendekatan ekonofisika yang menganggap pasar sebagai sistem partikel yang saling berinteraksi menuju titik kesetimbangan energi yang stabil (Hidalgo, 2006).

2.2.1 Analisis Kualitatif dalam Pasar Finansial

Analisis kualitatif atau yang secara konvensional dikenal sebagai analisis fundamental merupakan metode evaluasi kondisi makro dan mikro ekonomi secara komprehensif untuk memahami nilai-nilai fundamental yang mendasari pergerakan harga aset (Schwert, 2003). Pada umumnya, metode analisis ini digunakan para investor untuk membuat keputusan investasi jangka panjang dengan target waktu minimal 3 tahun. Pendekatan ini berfokus pada evaluasi variabel krusial seperti tingkat kesehatan industri, posisi strategis entitas di pasar global, dan estimasi nilai intrinsik perusahaan dari kepemilikan asetnya (Markowitz, 1952). Saat informasi kualitatif tersebut mulai banyak dikenali oleh para pelaku pasar, maka akan terjadi efisiensi pasar karena terjadi distribusi informasi kualitatif tersebut secara merata ke seluruh pelaku pasar sehingga mengurangi asimetri informasi (Schwert, 2003). Interaksi antara berbagai parameter fundamental tersebut pada akhirnya akan membentuk sebuah konsensus kolektif terkait dengan ekspektasi risiko dan imbal hasil yang diharapkan oleh para investor di tengah fluktuasi pasar akibat ketidakpastian global (Sikalo dkk., 2022). Oleh karena itu, analisis fundamental merupakan dasar pemahaman yang harus dikuasai oleh para investor untuk mengidentifikasi peluang investasi yang memiliki basis nilai riil sebelum mulai berinvestasi sehingga mampu mempertahankan stabilitas portofolio terhadap guncangan volatilitas yang ekstrem (La Torre & Maggistro, 2026).

Salah satu metrik ekonomi yang digunakan untuk mengukur skala likuiditas suatu entitas adalah kapitalisasi pasar. Aset yang memiliki kapitalisasi pasar besar sering kali diasosiasikan dengan likuiditas yang tinggi dan stabilitas yang lebih baik, sehingga menjadi pilihan utama bagi para investor tingkat institusi. Kapitalisasi pasar juga dapat

berfungsi sebagai indikator kualitatif yang esensial untuk mengukur tingkat kepercayaan publik terhadap profil risiko suatu entitas finansial (Schwert, 2003). Selain itu, terdapat metrik kinerja laporan keuangan yang memberikan rincian performa aset dan liabilitas yang sangat mendetail, di mana Gambar 2.1 mengilustrasikan ringkasan posisi keuangan suatu perusahaan sebagai landasan utama dalam menentukan derajat kesehatan fiskal mereka (Abdul Hali & Yuliati, 2020). Terdapat pula variabel modern lainnya sepeti ana-

Ikhtisar Keuangan dan Rasio Keuangan

Tabel Ikhtisar Keuangan dan Rasio Keuangan (dalam jutaan Rupiah)

Uraian	2025	2024*	2023	2022	2021
LAPORAN POSISI KEUANGAN KONSOLIDASIAN					
ASET					
Kas	32.044.482	29.783.642	31.603.784	27.407.478	26.299.973
Giro pada Bank Indonesia	31.929.608	88.878.969	101.909.121	150.935.150	56.426.573
Giro dan Penempatan pada bank lain - Netto	63.488.113	83.448.015	87.545.335	91.869.777	73.012.684
Efek-efek, Wesel Ekspor, Reverse Repo dan Tagihan Lainnya	420.454.320	382.903.830	416.411.206	418.685.107	455.174.902
Kredit yang Diberikan, Piutang Syariah, dan Pembiayaan	1.521.485.857	1.354.640.779	1.266.429.247	1.139.077.065	1.042.867.453
CKPN Kredit yang Diberikan, Piutang Syariah, dan Pembiayaan	(83.058.656)	(81.063.511)	(85.501.888)	(93.087.981)	(87.829.417)
Tagihan Derivatif - Netto	1.167.029	1.087.048	911.683	911.405	730.083
Tagihan Akseptasi - Netto	13.046.341	9.783.690	9.967.710	7.031.064	9.066.005
Penyertaan Saham - Netto	8.834.868	8.076.567	7.305.491	6.506.903	6.071.727
Aset Tetap - Netto	63.294.240	62.477.965	59.678.119	55.216.047	47.970.187
Aset Pajak Tangguhan - neto	8.129.522	12.800.660	15.445.977	18.712.994	16.284.898
Aset Lain-lain - neto	54.555.381	39.369.252	53.709.169	42.374.001	32.022.666
TOTAL ASET	**2.135.371.105**	**1.992.186.906**	**1.965.414.954**	**1.865.639.010**	**1.678.097.734**

Gambar 2.1: Contoh ikhtisar laporan posisi keuangan konsolidasian yang merepresentasikan struktur aset dan kesehatan fiskal entitas.

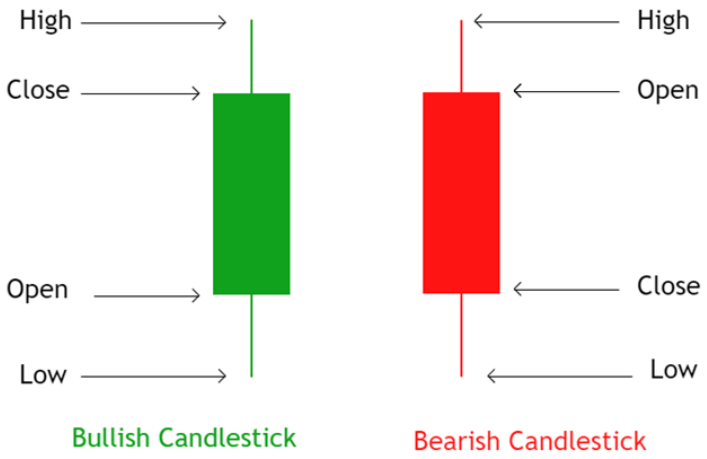
lisis sentimen berita dan data jaringan sosial yang dapat memberikan gambaran mengenai potensi saturasi nilai aset di pasar. Semakin banyak agen yang memiliki sentimen negatif terhadap suatu aset, maka semakin besar potensi penurunan harga aset tersebut di masa depan. Sebaliknya, semakin banyak agen yang memiliki sentimen positif terhadap suatu aset, maka semakin besar potensi kenaikan harga aset tersebut di masa depan.

2.2.2 Representasi Data pada Pergerakan Harga

Di samping analisis fundamental, terdapat analisis teknikal yang berfokus pada pemetaan stokastik aktivitas transaksi ke dalam domain visual untuk memberikan identifikasi pola-pola harga yang sering kali berulang dalam periode waktu tertentu. Pada umumnya, analisis teknikal digunakan oleh para investor untuk membuat keputusan investasi jangka pendek hingga menengah dengan target waktu maksimal 1 tahun. Terdapat berbagai jenis representasi grafis mulai dari grafik garis (*line chart*), grafik batang (*bar chart*), hingga grafik lilin (*candlestick*) yang digunakan sebagai instrumen untuk mengenali derau harga (*price noise*) dalam bentuk visual bagi para analis pasar profesional. Prinsip utama

dari analisis ini didasarkan pada asumsi bahwa harga pasar saat ini telah mencerminkan seluruh informasi historis dan aksi kolektif para pelaku pasar secara efisien (Schwert, 2003). Visualisasi data tersebut dapat membantu investor dalam melakukan ekstrapolasi tren masa depan berdasarkan pengenalan pola-pola geometris yang telah terbukti secara statistik dalam riwayat perdagangan (Sikalo dkk., 2022).

Representasi grafik lilin yang ditunjukkan pada Gambar 2.2 memberikan visualisasi pergerakan harga dalam interval waktu tertentu yang mencakup nilai harga pada pembukaan (*open*), tertinggi (*high*), terendah (*low*), dan (*close*). Analisis data grafik lilin sering digunakan oleh para investor untuk mengidentifikasi tekanan beli maupun jual yang signifikan dalam menentukan potensi pembalikan atau kelanjutan suatu tren harga aset. Saat grafik lilin dideretkan menjadi sebuah deret waktu, akan terbentuk pola-pola

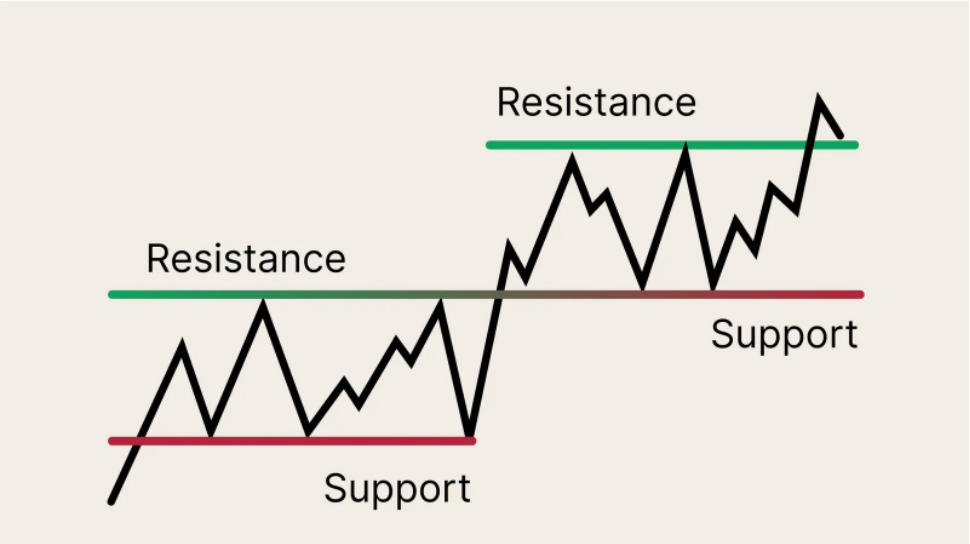


Gambar 2.2: Struktur anatomi grafik lilin dari kenaikan (*bullish*) dan penurunan (*bearish*) yang merangkum fluktuasi harga dalam satu periode waktu.

yang merepresentasikan psikologi kolektif pelaku pasar. Titik-titik balik penting dalam grafik di mana pergerakan harga tertahan atau berbalik arah didefinisikan sebagai *Support* dan *Resistance*. *Support* merupakan tingkat harga di mana tekanan beli cukup kuat untuk mengatasi tekanan jual sehingga mencegah harga jatuh lebih jauh, sementara *Resistance* merujuk pada tingkat di mana tekanan jual mendominasi sehingga mencegah harga naik lebih tinggi. Kedua konsep tersebut mencerminkan keseimbangan kekuatan dinamis antara penawaran dan permintaan dalam pasar, sehingga berfungsi sebagai indikator krusial bagi investor dalam pengambilan keputusan teknis. Analogi ilustrasi interaksi pasar tersebut dapat dilihat pada Gambar 2.3, di mana harga sering kali mengalami pantulan pada level *support* sebelum akhirnya berhasil menembus level *resistance* untuk selanjutnya membentuk tren kenaikan yang baru di pasar. Identifikasi pola teknikal tersebut berfungsi sebagai instrumen untuk mendeteksi fenomena perilaku mengikuti kawanan (*herding behavior*) yang menjadi pemicu utama munculnya volatilitas ekstrem dan perubahan rezim dalam sistem pasar finansial (Hidalgo, 2006).

2.2.3 Analisis Kuantitatif

Analisis kuantitatif merupakan metode yang sering digunakan oleh para analis kuantitatif atau biasa disebut *quants* untuk menentukan alokasi portofolio optimal selain analisis



Gambar 2.3: Representasi visual level *support* dan *resistance* sebagai batas memori sosial dalam dinamika harga aset.

fundamental dan teknikal. Analisis kuantitatif dimulai dari proses transformasi data harga mentah menjadi variabel imbal hasil sederhana (*simple return*) untuk mengurangi efek non-stasioneritas yang sering kali menghambat analisis statistik. Penggunaan imbal hasil logaritmik (*log return*) lebih diutamakan oleh para peneliti karena memungkinkan agregasi nilai secara matematis serta memberikan kemudahan dalam analisis numerik yang kompleks (Saslow, 1999). Imbal hasil sederhana aset ke-i pada hari ke-t ($R_{i,t}$) dihitung sebagai rasio perubahan harga terhadap hargal awal

$$R_{i,t} = \frac{P_{i,t} - P_{i,t-1}}{P_{i,t-1}}. \tag{2.1}$$

Sedangkan imbal hasil logaritmik digunakan untuk menjamin properti stasioneritas pada runtun waktu yang didefinisikan sebagai fungsi logaritma natural dari rasio harga dengan bentuk

$$r_{i,t} = \ln(1 + R_{i,t}) = \ln\left(\frac{P_{i,t}}{P_{i,t-1}}\right). \tag{2.2}$$

Dari kedua data tersebut, dapat diperoleh proses data selanjutnya yakni ekspektasi imbal hasil (μ_i) dan varians (σ_i^2) untuk setiap aset. Ekspektasi imbal hasil diperoleh melalui rata-rata data satu semester perdagangan yang mana memiliki 126 hari perdagangan ($n = 126$) sebagai data pelatihan untuk algoritma dengan rumus

$$\mu_i = \bar{R}_i = \frac{1}{n}\sum_{t=1}^{n} R_{i,t}. \tag{2.3}$$

Jika menggunakan imbal hasil logaritmik pada aset ke-i, rumusnya menjadi

$$\mu_{i,l} = \bar{r}_i = \frac{1}{n}\sum_{t=1}^{n} r_{i,t}. \tag{2.4}$$

Sedangkan untuk varians yang merepresentasikan risiko karena semakin besar varians maka semakin besar pula potensi kerugian yang dapat terjadi, diperoleh dengan bentuk

$$\sigma_i^2 = \frac{1}{n-1}\sum_{t=1}^{n}(R_{i,t} - \bar{R}_i)^2. \tag{2.5}$$

Jika mengunakan imbal hasil logaritmik, rumus variansnya menjadi

$$\sigma_{i,l}^2 = \frac{1}{n-1}\sum_{t=1}^{n}(r_{i,t} - \bar{r}_i)^2. \tag{2.6}$$

Selain itu, kovariansi yang digunakan untuk mengukur tingkat keterkaitan antara dua aset saham memiliki bentuk

$$\sigma_{ij} = \frac{1}{n-1}\sum_{t=1}^{n}(R_{i,t} - \bar{R}_i)(R_{j,t} - \bar{R}_j). \tag{2.7}$$

Jika mengunakan imbal hasil logaritmik, rumus kovariansinya menjadi

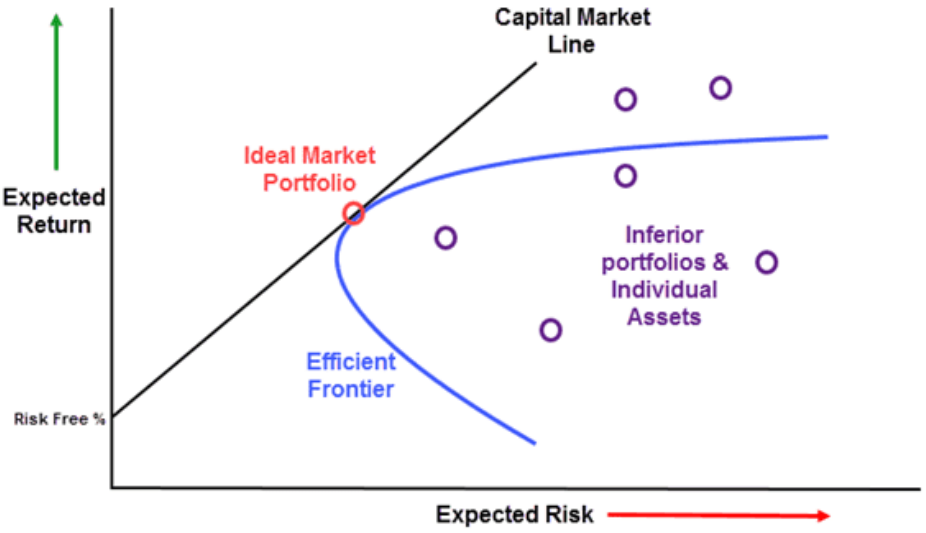
$$\sigma_{ij,l} = \frac{1}{n-1}\sum_{t=1}^{n}(r_{i,t} - \bar{r}_i)(r_{j,t} - \bar{r}_j). \tag{2.8}$$

Kovariansi positif menunjukkan bahwa kedua aset bergerak searah, sedangkan kovariansi negatif menunjukkan bahwa kedua aset bergerak berlawanan arah. Sebelum masuk ke dalam formulasi optimasi portofolio, perlu dipahami konsep efisiensi alokasi sumber daya melalui kerangka Optimum Pareto. Sebuah alokasi portofolio dikatakan optimal secara Pareto jika tidak ada cara untuk meningkatkan ekspektasi imbal hasil tanpa meningkatkan risiko, atau sebaliknya, tidak ada cara untuk menurunkan risiko tanpa menurunkan imbal hasil yang diharapkan.

Dalam ekonomi kesejahteraan, Optimum Pareto merepresentasikan titik-titik di mana utilitas sistem tidak dapat ditingkatkan lagi melalui realokasi aset sederhana tanpa merugikan salah satu parameter objektif (Ahmed dkk., 2017). Konsep ini mendasari pembentukan batas efisien (*Efficient Frontier*) dalam teori portofolio yang mana merupakan kumpulan dari semua portofolio yang menawarkan imbal hasil tertinggi untuk tingkat risiko tertentu (Cohen dkk., 2020). Bagi investor rasional, pemilihan alokasi modal harus selalu berada pada kurva efisiensi ini guna memastikan penggunaan sumber daya finansial yang maksimal di tengah ketidakpastian pasar (Innan dkk., 2025).

2.2.4 Model Markowitz

Model varians-rerata yang diperkenalkan oleh Harry Markowitz pada tahun 1952 mentransformasikan konsep abstrak optimum pareto untuk mencari titik keseimbangan antara ekspektasi imbal hasil dan tingkat risiko (Markowitz, 1952). Titik-titik keseimbangan optimal tersebut membentuk sebuah kurva yang dikenal sebagai kurva batas efisien sebagaimana divisualisasikan pada Gambar 2.4. Titik-titik yang terletak pada kurva tersebut merepresentasikan konfigurasi portofolio yang paling efisien, di mana setiap pergeseran di sepanjang kurva mencerminkan pertukaran antara risiko dan imbal hasil. Jika ingin mendapatkan imbal hasil tinggi, maka investor harus siap dengan risiko yang tinggi juga.



Gambar 2.4: Kurva batas efisien sebagai representasi batas efisiensi alokasi aset dalam kerangka kerja model Markowitz.

Namun jika investor menginginkan risiko yang rendah, maka investor harus memilkih aset dengan potensi imbal hasil yang rendah pula. Oleh karena itu, pemilihan titik spesifik pada kurva ini sangat bergantung pada tingkat profil toleransi risiko dari masing-masing investor untuk mencapai utilitas maksimal.

Sebagai bentuk formulasi matematis dari gambar kurva batas efisien tersebut, analisis kuantitatif pada model ini bertujuan untuk menentukan bobot alokasi modal yang tepat pada setiap instrumen untuk memaksimalkan utilitas portofolio (Abdul Hali & Yuliati, 2020). Fungsi biaya dalam model Markowitz menggabungkan matriks kovarians yang merepresentasikan risiko dan vektor ekspektasi imbal hasil dari semua aset yang ingin dipilih (Schwert, 2003). Fungsi objektif model Markowitz dapat dinyatakan dalam bentuk fungsi biaya (cost function) sebagai

$$f(\omega) = \sum_{i,j} \sigma_{ij}\omega_i\omega_j - \lambda \sum_i \mu_i\omega_i \tag{2.9}$$

dengan ω_i merepresentasikan bobot untuk setiap aset i, σ_{ij} merupakan elemen matriks kovarians, dan μ_i menunjukkan ekspektasi imbal hasil aset. Parameter λ berfungsi sebagai tingkat toleransi risiko yang mana secara matematis menentukan gradien utilitas investor pada kurva efisiensi tersebut. Untuk mempermudah optimasi model, parameter ω_i yang memiliki nilai kontinu antara 0 hingga 1 dan memiliki syarat $\sum \omega_i = 1$, dapat diubah menjadi variabel biner x_i yang hanya bernilai 0 dan 1 untuk menyatakan aset ke-i dibeli jika bernilai 1 dan tidak dibeli jika bernilai 0. Hubungan perubahan variabel tersebut dinyatakan dengan bentuk

$$\omega_i = \frac{x_i}{K} \tag{2.10}$$

dimana K merepresentasikan jumlah aset yang dipilih dari total pilihan aset yang ada (N). Jika jumlah aset yang dipilih adalah setengah dari pilihan aset maka $K = N/2$.

Sehingga, fungsi objektif dapat diubah menjadi

$$f(\mathbf{x}) = \sum_{i,j} \sigma_{ij} \frac{x_i x_j}{K^2} - \lambda \sum_i \mu_i \frac{x_i}{K}. \tag{2.11}$$

2.3 Teori Permainan dan Kerangka Permainan Potensial Eksak

Pada dasarnya, manusia tidak bisa hidup dan bekerja sendirian karena manusia sangat membutuhkan orang lain mengingat sifatnya sebagai makhluk sosial yang memiliki ketergantungan interaktif satu sama lain. Namun faktanya, interaksi sosial yang dilakukan oleh manusia tidak selalu bersifat harmonis dan saling menguntungkan, melainkan sering kali terjebak dalam dilema kepentingan yang memicu persaingan strategis. Ketidakpastian hasil dari interaksi ini membutuhkan formalisme ke dalam suatu metode analisis yang mampu memetakan perilaku agen rasional secara matematis dalam menghadapi skenario pilihan untuk mengoptimalkan utilitas kolektif. Oleh karena itu, dalam buku "*Theory of Games and Economic Behavior*", matematikawan John Von Neumann dan Oskar Morgenstern merumuskan formalisme matematika aksiomatik yang mengurai logika strategi di bawah kondisi ketergantungan secara sistematis (John Von Neumann & Oskar Morgenster, 1953). Paradigma ini kemudian berkembang dan memberikan landasan bagi pemodelan perilaku agen hingga tercetuslah konsep *game theory* atau teori permainan yang menjadi pilar utama dalam analisis interaksi strategis (Galam & Walliser, 2010).

2.3.1 Teori Permainan

Teori permainan merupakan kerangka kerja matematis aksiomatik yang digunakan untuk menganalisis interaksi strategis yang melibatkan sekumpulan agen rasional dalam lingkungan yang kompetitif. Sebuah permainan didefinisikan oleh tiga komponen fundamental yang meliputi himpunan pemain (N), ruang strategi (S), dan fungsi utilitas (U) bagi setiap partisipan yang terlibat. Interaksi antar pemain terjadi ketika keputusan yang diambil oleh satu agen mempengaruhi nilai utilitas atau keuntungan yang diperoleh oleh agen lainnya secara langsung, sehingga menciptakan ketergantungan yang beragam. Terdapat berbagai fenomena interaksi strategis dalam ekosistem ekonomi yang dapat direpresentasikan secara akurat melalui model permainan klasik (Monderer & Shapley, 1996). Model permainan klasik tersebut seperti dilema tahanan (*Prisoner's Dilemma*), Berburu Rusa (*Stag Hunt*), dan Pertempuran Antargender (*Battle of the Sexes*), menggambarkan esensi konflik kepentingan antar agen. Meskipun memiliki narasi konflik yang bervariasi, model-model tersebut secara formal dapat diklasifikasikan sebagai Permainan Potensial Eksak atau *Exact Potential Game* (EPG) karena memenuhi syarat penyelarasan antara utilitas marginal pemain dan perubahan fungsi potensial global yang akan dibahas di kedua subbab selanjutnya (Monderer & Shapley, 1996). Dalam kerangka *Prisoner's Dilemma*, pendekatan EPG memungkinkan identifikasi ekuilibrium yang stabil meskipun sering kali bersifat sub-optimal, sementara dalam *Stag Hunt*, metode ini mampu memfasilitasi penemuan titik koordinasi yang optimal bagi seluruh agen (Sarkar & Benjamin, 2018).

Ketiga model permainan tersebut memberikan kerangka umum terhadap dinamika ekuilibrium dalam sistem multi-agen dengan ketergantungan utilitas yang bersifat non-

linear (Monderer & Shapley, 1996). Gambar 2.5(a) mengilustrasikan permainan *Stag Hunt* di mana strategi berburu rusa (*Stag*) merepresentasikan koordinasi berisiko tinggi dengan imbal hasil maksimal, sedangkan strategi berburu kelinci (*Hare*) merupakan pilihan individualistik yang aman namun menghasilkan utilitas rendah bagi sistem (Sarkar & Benjamin, 2018). Dalam model *Prisoner's Dilemma* yang ditunjukkan pada Gambar 2.5(b), konflik muncul antara pilihan bekerja sama (*Cooperation*) untuk mencapai keuntungan kolektif optimal berhadapan dengan strategi tidak bekerja sama (*Non-cooperation*) yang secara individual dominan namun berujung pada ekuilibrium yang bersifat *Pareto-inferior*. Sementara itu, Gambar 2.5(c) merepresentasikan *Battle of the Sexes* yang mensyaratkan agen untuk melakukan koordinasi antara opsi menonton balapan anjing (*Dog race*) sebagai D dan menonton balet (*Ballet*) sebagai B meskipun terdapat preferensi imbal hasil yang berbeda di antara para pemain yang terlibat.

		Hunter 2	
		Stag	Hare
Hunter 1	Stag	3, 3	0, 2
	Hare	2, 0	2, 2

(a) Stag Hunt

		Player B	
		Cooperation	Non-cooperation
Player A	Cooperation	3, 3	0, 5
	Non-cooperation	5, 0	1, 1

(b) Prisoner's Dilemma

		Woman	
		D	B
Man	D	(4, 3)	(2, 2)
	B	(1, 1)	(3, 4)

(c) Battle of the Sexes

Gambar 2.5: Matriks imbal hasil pada model permainan klasik: (a) *Stag Hunt*, (b) *Prisoner's Dilemma*, dan (c) *Battle of the Sexes* yang merepresentasikan variasi struktur ekuilibrium (Szabó & Borsos, 2016).

2.3.2 Kerangka Permainan Potensial Eksak sebagai Algoritma Pencarian Ekuilibrium Nash

Ekuilibrium Nash merupakan konsep stabilitas fundamental dalam teori permainan di mana tidak ada satu pun pemain yang dapat meningkatkan utilitasnya dengan mengubah strateginya secara sepihak tanpa adanya perubahan dari pihak lain. Dalam sistem finansial, sebagian besar interaksi strategis antar agen ekonomi direpresentasikan melalui model permainan koordinasi (*coordination game*) yang mana setara dengan struktur *Stag Hunt* (Szabó & Borsos, 2016). Pemilihan model ini didasarkan pada fakta bahwa pasar sangat bergantung pada sinkronisasi keputusan para investor. Kerja sama kolektif untuk mempertahankan aset akan memberikan imbal hasil lebih tinggi dibandingkan pilihan individualis untuk keluar dari pasar (Sarkar & Benjamin, 2018). Namun, identifikasi titik ekuilibrium tersebut sering kali menghadapi kendala komputasi yang berat karena ruang solusi yang bersifat non-konveks serta diskrit, terutama pada sistem yang memiliki jumlah agen sangat besar.

Kerangka Permainan Potensial Eksak atau *Exact Potential Game* (EPG) hadir untuk mengatasi masalah tersebut (Ahmadi dkk., 2023). EPG didefinisikan sebagai kelas permainan strategis khusus di mana perubahan utilitas marginal setiap pemain selaras

dengan variasi sebuah fungsi potensial global (Monderer & Shapley, 1996). Secara matematis, eksistensi fungsi potensial (P) menunjukkan bahwa setiap insentif agen untuk meningkatkan keuntungan lokal akan berujung pada peningkatan nilai potensial global (Y. Yang dkk., 2012). Karakteristik penyelarasan tersebut dapat menyederhanakan masalah optimasi multi-agen yang sangat kompleks menjadi masalah maksimisasi fungsi tunggal yang jauh lebih efisien untuk dikelola secara komputasional. Dalam sistem pasar finansial, EPG berfungsi sebagai algoritma pencarian ekuilibrium Nash yang andal melalui pemanfaatan mekanisme dinamika respons-terbaik (*Best-Response Dynamics*) secara iteratif (Y. Yang dkk., 2012). Sebagai contoh, implementasi dinamika respons-terbaik oleh (Y. Yang dkk., 2012) memiliki prosedur lewat perumusan sebuah sistem permainan dalam bentuk algoritma yang memanfaatkan variabel ω_n^0 sebagai representasi bobot portofolio awal bagi setiap agen n di dalam himpunan strategi K_n. Indeks q digunakan untuk melacak kemajuan tahapan iteratif, di mana profil strategi kolektif ω^q dievaluasi terhadap fungsi utilitas u_n untuk mendapatkan konvergensi sistem menuju titik stabil. Mekanisme pembaruan sekuensial (Gauss-Seidel) beroperasi dengan mengintegrasikan informasi strategis terbaru dari agen sebelumnya, sementara pembaruan simultan (Jacobi) menggunakan data dari iterasi sebelumnya untuk menghitung respon terbaik secara paralel bagi seluruh partisipan pasar (Freitas dkk., 2024). Struktur operasional algoritma iteratif tersebut dapat ditinjau pada algoritma 1.

Algorithm 1 Iterative Best Response Algorithm (Y. Yang dkk., 2012)

1: **Data:** Pilih strategi awal $\omega_n^0 \in K_n$ untuk semua $n = 1, \ldots, N$, dan set $q = 0$.
2: **Step 1:** Jika ω^q memenuhi kriteria penghentian yang sesuai: **STOP**.
3: **Step 2:** Perbarui strategi ω_n^{q+1} untuk $n = 1, \ldots, N$ menggunakan salah satu metode:
4: **Sequential (Gauss-Seidel) Update:**
5: $\omega_n^{q+1} \leftarrow \arg\min_{\omega_n \in K_n} u_n(\omega_1^{q+1}, \ldots, \omega_{n-1}^{q+1}, \omega_n, \omega_{n+1}^q, \ldots, \omega_N^q)$
6: **Simultaneous (Jacobi) Update:**
7: $\omega_n^{q+1} \leftarrow (1 - \frac{1}{N})\omega_n^q + \frac{1}{N} \cdot \arg\min_{\omega_n \in K_n} u_n(\omega_1^q, \ldots, \omega_{n-1}^q, \omega_n, \omega_{n+1}^q, \ldots, \omega_N^q)$
8: **Step 3:** Set $q \leftarrow q + 1$; dan kembali ke **Step 1** (Monderer & Shapley, 1996).

2.3.3 Kerangka EPG dari Model Markowitz

Implementasi kerangka kerja EPG dalam pemilihan aset strategis diawali dengan transformasi fungsi objektif dari kontinu menjadi diskrit versi biner yang telah dijabarkan sebelumnya pada Persamaan (2.11). Formulasi fungsi objektif tersebut selanjutnya dikonversi menjadi representasi fungsi utilitas (U) yang harus dimaksimalkan oleh setiap agen di dalam sistem. Transformasi ini dilakukan dengan mendefinisikan sebuah faktor baru yaitu faktor penghindaran risiko (γ) yang memiliki hubungan $\gamma = 2/\lambda$, sehingga profil nilai utilitas portofolio sistem dapat dinyatakan dengan hubungan

$$U(\mathbf{x}) = -\frac{1}{\lambda} f(\mathbf{x})$$

sehingga menjadi

$$U(\mathbf{x}) = \sum_i \frac{\mu_i}{K} x_i - \frac{\gamma}{2K^2} \sum_{i,j} \sigma_{ij} x_i x_j. \tag{2.12}$$

dengan utilitas individu yang dimiliki oleh agen ke-i didefinisikan sebagai

$$u_i(\mathbf{x}) = \frac{\mu_i}{K}x_i - \frac{\gamma}{2K^2}x_i\left(\sum_{j\neq i}\sigma_{ij}x_j\right) \tag{2.13}$$

yang mana jika varians dipisah dari matriks kovarinas, maka diperoleh persamaan

$$u_i(\mathbf{x}) = \frac{\mu_i}{K}x_i - \frac{\gamma}{2K^2}x_i\left(\sigma_{ii} + 2\sum_{j\neq i}\sigma_{ij}x_j\right). \tag{2.14}$$

Hubungan $\gamma = 2/\lambda$ ini merupakan konsekuensi langsung dari struktur formulasi fungsi objektif Markowitz sebagai konvensi umum. Jika menggunakan hubungan matematis $\lambda = 1/\gamma$, maka akan menghasilkan inkonsistensi karena nilai penalti risiko pada fungsi utilitas akan mengecil ketika nilai penghindaran risiko γ membesar, sehingga bertentangan dengan makna ekonomisnya. Pada formulasi utilitas tersebut, kehadiran parameter skalar γ didapat menggunakan parameter penghindaran risiko secara endogen menggunakan fungsi sigmoid dengan bentuk

$$\gamma = \frac{1}{1 + e^{(\bar{\mu}_l/\bar{\sigma}_l)}} \tag{2.15}$$

dimana $\bar{\mu}_l$ adalah rata-rata imbal hasil dari seluruh aset pilihan dan $\bar{\sigma}_l$ adalah rata-rata deviasi standar dari seluruh aset pilihan yang keduanya dibangun menggunakan imbal hasil logaritmik.

Berdasarkan teorema yang digagas oleh Monderer dan Shapley (Monderer & Shapley, 1996), sebuah permainan dapat dikategorikan sebagai EPG apabila terdapat sebuah fungsi potensial skalar $\Phi(\mathbf{x})$ sedemikian rupa sehingga selisih utilitas individu Δu_i akibat perubahan strategi satu agen secara eksak sama dengan selisih nilai fungsi potensial tersebut. Kesetaraan ini dapat dibuktikan dengan membandingkan Persamaan (2.14) dengan suku yang bergantung pada x_i pada hasil dekomposisi Persamaan (2.17), yang menunjukkan bahwa kedua ekspresi tersebut identik secara struktural sehingga $\Delta u_i = \Delta\Phi$ terjamin berlaku untuk setiap perubahan strategi x_i secara sepihak (John Von Neumann & Oskar Morgenster, 1953). Dari persamaan (2.12), fungsi potensial diperoleh dari ekspansi fungsi biaya portofolio yang terasosiasi langsung dengan identitas utilitas sistem:

$$\Phi(\mathbf{x}) = \sum_{l=1}^{N}\mu_l\frac{x_l}{K} - \frac{\gamma}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\sigma_{ij}\frac{x_i}{K}\frac{x_j}{K}. \tag{2.16}$$

Untuk mengidentifikasi kontribusi marginal agen ke-i, dilakukan dekomposisi jumlahan dengan memisahkan suku-suku yang mengandung variabel x_i. Karena $x_i \in \{0, 1\}$ dan jika dikuadratkan hasilnya akan tetap $x_i^2 \in \{0, 1\}$, maka $x_i^2 = x_i$. Sehingga dekomposisi fungsi potensial menjadi

$$\Phi(\mathbf{x}) = \frac{\mu_i}{K}x_i - \frac{\gamma}{2}\left(\frac{\sigma_{ii}}{K^2}x_i + \frac{2}{K^2}x_i\sum_{j\neq i}\sigma_{ij}x_j\right) + \sum_{l\neq i}\frac{\mu_l}{K}x_l - \frac{\gamma}{2}\sum_{l\neq i}\sum_{m\neq i}\frac{\sigma_{lm}}{K^2}x_l x_m. \tag{2.17}$$

Persamaan (2.17) akan menghasilkan selisih fungsi potensial $\Delta\Phi$ sebagai insentif marginal ketika agen i mengubah status dari 0 menjadi 1 secara sepihak. Insentif bagi agen untuk meningkatkan utilitas individualnya akan meningkatkan nilai potensial global sistem

melalui relasi

$$\Delta\Phi(\mathbf{x}) = \Phi(1, \mathbf{x}_{-i}) - \Phi(0, \mathbf{x}_{-i}) = \frac{\mu_i}{K} - \frac{\gamma\sigma_{ii}}{2K^2} - \frac{\gamma}{K^2}\sum_{j\neq i}\sigma_{ij}x_j. \tag{2.18}$$

Hubungan tersebut menyatakan bahwa setiap langkah peningkatan utilitas lokal agen akan meningkatkan potensial global sistem menuju titik kesetimbangan. Ekuilibrium Nash akan tercapai ketika tidak ada satu pun agen yang memiliki insentif positif ($\Delta\Phi > 0$) untuk mengubah status keputusannya secara sepihak (Mukhopadhyay dkk., 2025). Konfigurasi yang diperoleh dari kerangka EPG ini selanjutnya dimanfaatkan sebagai tebakan awal untuk memandu proses optimasi kuantum pada bab selanjutnya (Y. Yang dkk., 2012).

2.4 Hamiltonian Ising sebagai Representasi Sistem Kuantum

Setiap sistem fisik yang perilakunya didorong oleh prinsip minimisasi energi dapat direpresentasikan ke dalam Hamiltonian operator secara universal, tidak terkecuali pada konfigurasi seleksi aset finansial yang memiliki kompleksitas variabel yang sangat tinggi. Dalam kerangka ekonofisika, proses pengambilan keputusan untuk alokasi portofolio dapat dipetakan ke dalam lanskap energi di mana solusi optimal setara dengan konfigurasi pada energi terendah sebagai keadaan dasar (*ground state*) dari sistem tersebut (Lucas, 2014). Pemetaan ini menganggap setiap instrumen investasi sebagai partikel spin yang saling berinteraksi, di mana derajat korelasi antar pergerakan harga dimodelkan sebagai koefisien kopling di dalam sebuah kisi kristal yang terstruktur (Galam & Walliser, 2010). Penerapan model Ising dalam ekonomi didasarkan pada analogi formal antara parameter fisik sistem magnetik dan variabel-variabel pasar modal. Tabel 2.1 merangkum hubungan fungsional antara kedua domain tersebut.

Tabel 2.1: Analogi Konsep Fisika Statistik dan Ekonomi Finansial dalam Kerangka Model Ising.

Konsep Fisika Statistik	**Analogi Ekonomi/Finansial**
Sistem Partikel	Sekumpulan Aset Saham
Spin ($s_i \in \{-1, 1\}$)	Keputusan Investasi/Seleksi Aset ($x_i \in \{0, 1\}$)
Interaksi Kopling (J_{ij})	Korelasi atau Kovarians antar Aset (σ_{ij})
Medan Eksternal (h_i)	Ekspektasi Imbal Hasil
Energi Total (H)	Fungsi Biaya atau Risiko Portofolio
Keadaan Dasar ($\mathcal{H}_{min}$)	Alokasi Portofolio Optimal

2.4.1 Sejarah dan Evolusi Model Ising

Model Ising pada awalnya dikonseptualisasikan oleh Wilhelm Lenz dan diselesaikan oleh Ernst Ising pada awal tahun 1925 sebagai instrumen teoretis revolusioner untuk menyelidiki asal-usul mikroskopis dari fenomena feromagnetisme (Niss, 2005). Fokus fundamental dari model ini terletak pada penyederhanaan interaksi fisik antara momen

magnetik diskrit yang disusun dalam kisi reguler guna mengevaluasi munculnya keteraturan makroskopik dari fluktuasi termal lokal. Melalui pendekatan mekanika statistik, model ini berupaya menjawab pertanyaan krusial mengenai bagaimana orientasi mandiri dari partikel-partikel spin yang saling bertetangga dapat memicu transisi fase sistemik pada suhu kritis tertentu. Keberhasilan awal dalam merumuskan kerangka kerja tersebut memberikan fondasi yang sangat kokoh bagi pengembangan teori fenomena kooperatif yang menjadi pilar utama dalam fisika statistik modern (Feynman, 1981).

Secara matematis, Lenz dan Ising memodelkan material feromagnetik sebagai sekumpulan variabel spin biner yang hanya dapat berada pada dua orientasi, yaitu sejajar $(+1)$ atau berlawanan (-1) terhadap suatu arah acuan. Model ini berangkat dari energi potensial sebuah dipol magnetik yang berada di dalam medan magnet eksternal, yaitu

$$E_i = -\boldsymbol{\mu}_i \cdot \mathbf{B}, \tag{2.19}$$

dengan $\boldsymbol{\mu}_i$ menyatakan momen magnetik partikel ke-i dan $\mathbf{B}$ merupakan medan magnet luar. Sistem akan mencapai keadaan energi minimum ketika arah momen magnetik sejajar dengan medan magnet yang diberikan sehingga diberikan tanda negatif pada persamaan. Pada Model Ising klasik, setiap momen magnetik diasumsikan hanya dapat memiliki dua orientasi terhadap arah medan magnet, yaitu sejajar atau berlawanan arah. Kedua keadaan tersebut direpresentasikan oleh variabel spin biner

$$s_i = \pm 1, \tag{2.20}$$

dengan $s_i = +1$ menyatakan spin sejajar terhadap medan magnet, sedangkan $s_i = -1$ menyatakan spin yang berlawanan arah. Dengan mengasumsikan medan magnet eksternal homogen dan hanya bekerja sepanjang sumbu-z, maka besar energi setiap spin hanya bergantung pada orientasi spin tersebut terhadap medan. Oleh karena itu, kontribusi energi akibat medan magnet eksternal dapat dituliskan dalam bentuk

$$E_i = -hs_i, \tag{2.21}$$

dengan

$$h = \frac{\gamma \hbar B_z}{2} \tag{2.22}$$

merupakan parameter medan efektif yang nilainya sama untuk seluruh spin karena rasio giromagnetik γ, konstanta Planck tereduksi $\hbar$, dan medan magnet eksternal B_z diasumsikan homogen pada seluruh sistem. dengan $s_i = \pm 1$ menyatakan orientasi spin pada lokasi ke-i. Selain dipengaruhi oleh medan luar, model Ising juga mengasumsikan bahwa setiap pasangan spin saling berinteraksi. Energi interaksi antar pasangan spin dinyatakan sebagai

$$E_{ij} = -J_{ij} s_i s_j, \tag{2.23}$$

dengan J_{ij} merupakan konstanta kopling antara spin ke-i dan ke-j. Ketika kedua spin memiliki orientasi yang sama, maka nilai $s_i s_j$ akan bernilai 1 yang mana merepresentasikan konfigurasi sejajar sempurna, sedangkan apabila keduanya berlawanan arah maka $s_i s_j$ akan bernilai -1 yang mana merepresentasikan konfigurasi anti-sejajar sempurna.

Dengan menjumlahkan seluruh kontribusi energi medan luar dan interaksi antar pasangan spin, maka diperoleh Hamiltonian Ising pada sistem klasik berbentuk

$$\mathcal{H} = -h \sum_{i=1}^{N} s_i \tag{2.24}$$

yang mana merupakan bentuk asli Model Ising yang diperkenalkan oleh Lenz dan Ising dalam kerangka mekanika statistik klasik. Pada formulasi mekanika kuantum, variabel spin klasik s_i dipromosikan menjadi operator yang bekerja pada ruang Hilbert dua dimensi. Untuk sistem spin-$\frac{1}{2}$, operator spin sepanjang sumbu-z memenuhi hubungan

$$\hat{S}_z = \frac{\hbar}{2}\hat{Z}, \tag{2.25}$$

dengan $\hat{Z}$ merupakan operator Pauli-Z. Karena operator tersebut memiliki nilai eigen ± 1, sama seperti variabel spin pada Model Ising klasik, maka setiap variabel spin klasik s_i dapat direpresentasikan oleh operator $\hat{Z}_i$. Proses ini merupakan bentuk kuantisasi dari Model Ising klasik sehingga Hamiltonian dapat dituliskan dalam bentuk operator kuantum. Operator Pauli-Z untuk sebuah spin-$\frac{1}{2}$ dinyatakan sebagai

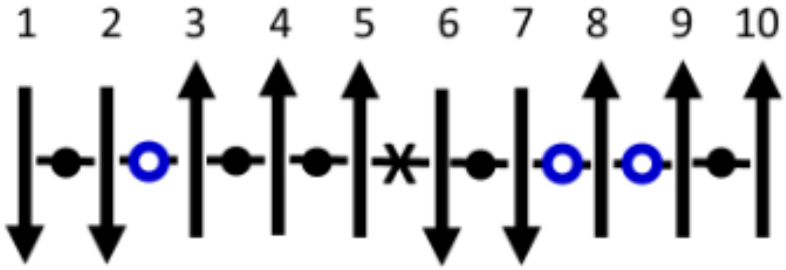
$$\hat{Z} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{2.26}$$

karena operator tersebut memiliki nilai eigen ± 1 yang sesuai dengan orientasi spin pada model Ising klasik. Melalui substitusi tersebut, Hamiltonian Ising akhir untuk sistem kuantum dapat dituliskan dalam bentuk

$$\hat{\mathcal{H}} = -h \sum_{i=1}^{N} \hat{Z}_i - \sum_{i<j}^{N} J_{ij} \hat{Z}_i \hat{Z}_j. \tag{2.27}$$

Pada pengembangan selanjutnya, parameter medan luar homogen h dapat diperumum menjadi parameter lokal h_i sehingga setiap spin memiliki bias energi yang berbeda. Bentuk Hamiltonian tersebut banyak digunakan pada formulasi optimisasi kombinatorial, termasuk representasi masalah QUBO yang menjadi dasar implementasi algoritma kuantum seperti VQE.

Meskipun memiliki landasan yang kuat, studi orisinal oleh Ernst Ising pada sistem satu dimensi (1D) menghasilkan kesimpulan pesimistis bahwa model tersebut tidak mampu mendemonstrasikan transisi fase pada suhu di atas nol mutlak. Analisis terhadap rantai interaksi biner berukuran hingga menunjukkan bahwa fluktuasi termal selalu mendominasi energi interaksi sehingga mencegah terbentuknya keteraturan magnetik yang stabil dalam skala makroskopik (Chamberlin & Lindsay, 2024). Figure 2.6 mengilustrasikan konfigurasi spin pada model satu dimensi tersebut, di mana distribusi orientasi partikel cenderung bersifat acak akibat lemahnya korelasi jangka panjang pada dimensi yang rendah. Representasi visual ini merinci interaksi antar-partikel melalui tiga simbol utama, yaitu titik hitam (•) untuk interaksi energi rendah, tanda silang (**X**) untuk interaksi energi tinggi, serta lingkaran biru kosong (∘) sebagai pemutusan (*break*) tanpa energi. Titik hitam merepresentasikan pasangan spin paralel yang meminimalkan energi sistem, sedangkan tanda silang menunjukkan konfigurasi anti-paralel yang memicu peningkatan tegangan energi lokal (Chamberlin & Lindsay, 2024). Lingkaran biru (∘) secara khusus menandai adanya pemutusan jalinan (*subdivision*) yang memisahkan rantai menjadi subsistem independen agar dapat mencapai kestabilan termal pada skala nano. Pemetaan parameter interaksi mikroskopis ini memungkinkan pemodelan perilaku kolektif dan transisi fase pada sistem berukuran terbatas melalui perhitungan energi total yang mengombinasikan jumlah interaksi dan pemutusan tersebut secara sistematis. Kekakuan hasil



Gambar 2.6: Representasi visual model Ising satu dimensi (1D) yang menunjukkan dinamika orientasi spin dan klasifikasi energi interaksi (•, **X**, ∘) di bawah pengaruh fluktuasi termal (Chamberlin & Lindsay, 2024).

matematis pada dimensi tunggal ini sempat memicu skeptisisme terhadap relevansi fisik model Ising, sebelum akhirnya paradigma tersebut berubah total setelah Lars Onsager merumuskan solusi eksak untuk kasus dua dimensi yang sangat kompleks (Preskill, 2018).

Kebangkitan minat terhadap model Ising terjadi pada era 1930-an hingga 1940-an setelah para fisikawan menyadari kapasitas model tersebut dalam merepresentasikan berbagai fenomena kooperatif di luar bidang magnetisme (Niss, 2005). Struktur energi yang dibentuk oleh interaksi antar spin ternyata memiliki kemiripan topologi dengan berbagai masalah optimasi kombinatorial yang ditemukan dalam sistem biologis, kimia, maupun ekonomi global. Pada tahun 1944, Lars Onsager berhasil menyimpulkan bahwa transisi-fase pada sistem dua dimensi membuktikan bahwa interaksi jarak pendek yang sederhana mampu menghasilkan perilaku kolektif yang sangat kaya dan tidak terduga lewatpenurunan rumusnya yang sangat panjang (Kabelac, 2021). Saat ini, model Ising telah bertransformasi menjadi bahasa standar dalam ranah komputasi berbasis model ising yang dapat menghubungkan masalah optimasi dunia nyata dengan arsitektur komputer sebagai contoh pada kasus penganilan kuantum (*Quantum Annealing*) (Lucas, 2014). Pendekatan tersebut mampu membuat komputasi dengan ruang solusi yang tumbuh secara eksponensial menjadi efisien melalui pemetaan parameter-parameter Hamiltonian Ising (Navarrete & Davis, 2022).

2.4.2 Representasi Fisik Aset Finansial

Dalam bidang ekonofisika, para agen ekonomi dianalogikan sebagai sistem partikel diskrit yang saling berinteraksi secara dinamis guna mencapai titik kesetimbangan energi yang paling stabil di tengah fluktuasi pasar. Setiap aset dalam portofolio investasi kemudian direpresentasikan sebagai entitas fisik spin (s_i) yang memiliki dua arah keadaan tetap di dalam ruang Hilbert yang terdefinisi secara matematis (Galam & Walliser, 2010). Berdasarkan transformasi variabel yang digunakan, keadaan -1 merepresentasikan posisi beli (*long*), sementara keadaan $+1$ digunakan untuk merepresentasikan posisi jual (*short*) atau pengabaian aset dalam strategi alokasi modal (Lucas, 2014). Interaksi antar spin tersebut ditentukan oleh kekuatan korelasi yang dimiliki masing-masing aset melalui data ekspektasi imbal hasil dan matriks kovarians yang telah diperoleh sebelumnya.

Parameter utama dari model Ising adalah parameter kopling dan parameter medan magnet eksternal yang selanjutnya dianalogikan sebagai parameter quantitative finance untuk mendapatkan keputusan aset terbaik (H. Li dkk., 2023). Nilai parameter kopling

coupling (J_{ij}) didasarkan pada matriks kovarians antar aset yang menjadi representasi fisik dari hubungan interaksi antar spin (Galam & Walliser, 2010). Sementara itu, ekspektasi imbal hasil dari setiap aset berperan sebagai parameter medan magnet eksternal (h_i) yang memberikan dorongan pada arah orientasi spin tertentu untuk memaksimalkan utilitas investasi (Lucas, 2014). Hamiltonian sistem kemudian digunakan untuk menghitung total energi yang merepresentasikan total ketidakstabilan strategi investasi. Nilai energi minimum yang didapat setelah berbagai kombinasi spin dihitung akan menjadi indikator utama dari solusi portofolio yang paling optimal (Pagni dkk., 2026).

2.4.3 Pemetaan Kuantum: QUBO menjadi basis Pauli-Z

Agar dapat diimplementasikan ke dalam basis komputasi kuantum, maka fungsi optimasi pada domain biner klasik perlu ditransformasi menjadi fungsi yang mendukung arsitektur komputasi kuantum. Proses transformasi tersebut dapat dilakukan menggunakan tranformasi QUBO yang mengubah ruang solusi dari domain biner klasik menjadi basis operator kuantum (Lucas, 2014). Tahap awal transformasi dilaksanakan dengan mengonversi rumusan fungsi utilitas portofolio sebelumnya menjadi model fungsi energi

$$E(\mathbf{x}) = -\Phi(\mathbf{x}). \tag{2.28}$$

Proses transformasi tersebut berguna untuk menyelaraskan arah pergerakan optimasi yang awalnya berupa maksimasi menjadi minimasi agar dapat mendukung sistem dengan prinsip operasional pencarian energi terendah. Sehingga persamaan (2.16) dapat disubstitusi ke dalam persamaan (4.20) menjadi:

$$E(\mathbf{x}) = \frac{\gamma}{2K^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\sigma_{ij}x_ix_j - \sum_{i=1}^{N}\frac{\mu_i}{K}x_i. \tag{2.29}$$

Mengingat kerangka matriks kovarians selalu memiliki sifat parameter numerik yang saling simetris, persamaan energi dasar portofolio ini dapat segera didekomposisi untuk mengisolasi komponen keputusan pembelian masing-masing aset. Dengan identitas variabel biner $x_i^2 = x_i$, total fungsional energi matematika di atas dapat dikonstruksi ulang melalui prosedur penjabaran sebagai berikut:

$$\begin{aligned}
E(\mathbf{x}) &= \frac{\gamma}{2K^2}\left(\sum_{i=1}^{N}\sigma_{ii}x_i^2 + \sum_{i\neq j}\sigma_{ij}x_ix_j\right) - \sum_{i=1}^{N}\frac{\mu_i}{K}x_i \\
&= \frac{\gamma}{2K^2}\sum_{i=1}^{N}\sigma_{ii}x_i^2 + \frac{\gamma}{2K^2}\sum_{i\neq j}\sigma_{ij}x_ix_j - \sum_{i=1}^{N}\frac{\mu_i}{K}x_i \\
&= \frac{\gamma}{2K^2}\sum_{i=1}^{N}\sigma_{ii}x_i + \frac{\gamma}{2K^2}\sum_{i\neq j}\sigma_{ij}x_ix_j - \sum_{i=1}^{N}\frac{\mu_i}{K}x_i
\end{aligned}$$

$$\therefore E(\mathbf{x}) = \sum_{i=1}^{N}\left(\frac{\gamma\sigma_{ii}}{2K^2} - \frac{\mu_i}{K}\right)x_i + \sum_{i\neq j}\frac{\gamma\sigma_{ij}}{2K^2}x_ix_j. \tag{2.30}$$

Untuk memastikan bahwa konfigurasi portofolio rasional selalu mematuhi batasan sistem yang mengharuskan pemilihan instrumen sebesar batasan K, maka diperlukan

penambahan suku penalti kuadratik. Komponen suku penalti disimbolkan sebagai $P(\mathbf{x})$ dilengkapi dengan pemberian faktor skalar pengali A yang bertindak sangat dominan dalam menjustifikasi arsitektur komputasi kuantum (Suzuki, 2025). Konfigurasi perlakuan energi tersebut dimaksudkan untuk menekan nilai probabilitas kombinasi yang melanggar ketentuan rasio alokasi pembelian aset (Schultz dkk., 1964). Bentuk dari fungsi penalti kuadratik tersebut secara umum dapat dirumuskan sebagai

$$P(\mathbf{x}) = A\left(\sum_{i=1}^{N} x_i - K\right)^2 \tag{2.31}$$

yang mana jika dijabarkan lebih lanjut, maka akan menjadi

$$\begin{aligned} P(\mathbf{x}) &= A\left(\sum_{i=1}^{N} x_i^2 + \sum_{i\neq j} x_i x_j - 2K\sum_{i=1}^{N} x_i + K^2\right) \\ &= A\sum_{i=1}^{N} x_i^2 + A\sum_{i\neq j} x_i x_j - 2AK\sum_{i=1}^{N} x_i + AK^2. \end{aligned}$$

Selanjutnya, fungsi energi finansial dasar dan komponen penalti disatukan sehingga menghasilkan sebuah arsitektur lanskap energi utuh dalam format optimasi alokasi QUBO. Penggabungan kedua fungsi tersebut merumuskan fungsi objektif total bernotasi $E_{total}(\mathbf{x})$ yang merangkum seluruh perhitungan variabel dinamis dari pasar yang akan dioptimasi beserta batasannya

$$\begin{aligned} E_{total}(\mathbf{x}) =& E_(\mathbf{x}) + P(\mathbf{x}) \\ =& \sum_i \left(\frac{\gamma\sigma_{ii}}{2K^2} - \frac{\mu_i}{K}\right) x_i + \sum_{i\neq j} \frac{\gamma\sigma_{ij}}{2K^2} x_i x_j + A\sum_{i=1}^{N} x_i^2 + A\sum_{i\neq j} x_i x_j \\ & - 2AK\sum_{i=1}^{N} x_i + AK^2 \end{aligned}$$

$$\therefore E_{total}(\mathbf{x}) = \sum_i \left(\frac{\gamma\sigma_{ii}}{2K^2} + A - \frac{\mu_i}{K} - 2AK\right) x_i + \sum_{i\neq j}\left(\frac{\gamma\sigma_{ij}}{2K^2} + A\right) x_i x_j + AK^2. \tag{2.32}$$

Untuk menghindari proses aljabar yang panjang pada persamaan (2.32), dapat didefinisikan notasi koefisien Q_{ii} dan Q_{ij} sebagai berikut:

$$Q_{ii} = \frac{\gamma\sigma_{ii}}{2K^2} + A - \frac{\mu_i}{K} - 2AK \quad \text{dan} \quad Q_{ij} = \frac{\gamma\sigma_{ij}}{2K^2} + A \ (Menezes\ \&Quiggin, 2025).$$

Sehingga komposisi arsitektur fungsi energi (2.32) dapat diringkas menjadi

$$E_{total}(\mathbf{x}) = \sum_i Q_{ii} x_i + \sum_{i\neq j} Q_{ij} x_i x_j + AK^2. \tag{2.33}$$

Selanjutnya, dilakukan prosedur mekanika kuantum untuk memetakan variabel biner dari komputasi klasik $x_i \in \{0, 1\}$ menuju formulasi permodelan energi elemen rotasi spin yang

berniali $s_i \in \{-1, 1\}$. Perubahan domain matematika tersebut dapat dilakukan dengan konversi bentuk linearisasi afinitas

$$\begin{aligned} x_i &= \frac{(1 - s_i)}{2} \\ x_i x_j &= \left(\frac{1 - s_i}{2}\right)\left(\frac{1 - s_j}{2}\right). \end{aligned} \tag{2.34}$$

Sehingga, dengan simetri korelasi $Q_{ij} = Q_{ji}$, maka fungsi energi dalam domain spin menjadi

$$\begin{aligned} E_{total}(\mathbf{s}) &= \sum_i Q_{ii}\left(\frac{1 - s_i}{2}\right) + \sum_{i \neq j} Q_{ij}\left(\frac{1 - s_i - s_j + s_i s_j}{4}\right) + AK^2 \\ &= \sum_i Q_{ii}\frac{1}{2} - \sum_i Q_{ii}\frac{s_i}{2} + \sum_{i \neq j} Q_{ij}\frac{1}{4} - \sum_{i \neq j} Q_{ij}\frac{s_i}{4} - \sum_{i \neq j} Q_{ij}\frac{s_j}{4} \\ &\quad + \sum_{i \neq j} Q_{ij}\frac{s_i s_j}{4} + AK^2 \end{aligned}$$

$$\begin{aligned} \therefore E_{total}(\mathbf{s}) = &\sum_i \left(-\frac{Q_{ii}}{2} - \sum_{i \neq j}\frac{Q_{ij}}{2}\right) s_i + \sum_{i \neq j}\frac{Q_{ij}}{4} s_i s_j \\ &+ \left(\sum_i \frac{Q_{ii}}{2} + \sum_{i \neq j}\frac{Q_{ij}}{4} + AK^2\right). \end{aligned} \tag{2.35}$$

Untuk mengubah fungsi energi dalam domain spin menjadi formalisme Hamiltonian, maka komponen parameter Hamiltonian dapat didefinisikan dari persamaan (2.35) yang terdiri dari parameter bias, parameter kopling, dan parameter konstanta (Coyle dkk., 2021). Parameter bias h_i didapat dari koefisien komponen linear pada persamaan (2.35)

$$h_i = \left(-\frac{Q_{ii}}{2} - \sum_{i \neq j}\frac{Q_{ij}}{2}\right). \tag{2.36}$$

Sedangkan komponen parameter kopling J_{ij} didapat dari koefisien komponen kuadratik pada persamaan (2.35)

$$J_{ij} = \frac{Q_{ij}}{4}. \tag{2.37}$$

Untuk suku sisa yang tidak mengandung variabel s_i dan $s_i s_j$ pada persamaan (2.35), dapat didefinisikan sebagai parameter konstanta C dengan bentuk

$$C = \sum_i \frac{Q_{ii}}{2} + \sum_{i \neq j}\frac{Q_{ij}}{4} + AK^2 \tag{2.38}$$

Sehingga semua parameter tersebut dapat dikonfigurasi ke dalam persamaan fungsi Hamiltonian Ising dengan bentuk

$$\hat{\mathcal{H}} = -\sum_{i=1}^{N} h_i \hat{Z}_i - \sum_{i<j}^{N} J_{ij}\hat{Z}_i\hat{Z}_j + C. \tag{2.39}$$

Pada dasarnya, komponen parameter medan h_i pada ekspresi Hamiltonian tersebut menggambarkan kecenderungan arah orientasi pergeseran medan spin lokal (Lucas, 2014). Sedangkan komponen interaksi J_{ij} merepresentasikan interaksi timbal balik antar elemen spin. Dalam kasus optimasi keputusan investasi, interaksi timbal balik antar elemen spin merepresentasikan korelasi antar aset finansial yang akan diinvestasikan. Dimana ketika interaksi timbal balik antar elemen spin negatif maka korelasi antar aset finansial yang akan diinvestasikan berbanding terbalik dan ketika interaksi timbal balik antar elemen spin positif maka korelasi antar aset finansial yang akan diinvestasikan berbanding lurus (Datta & Hansda, 2015). Sedangkan untuk parameter medan h_i, merepresentasikan kecenderungan harga pasar aset finansial berfluktuasi (Reifenstein dkk., 2021). Sementara itu, parameter konstanta C merupakan suku pergeseran skala energi yang nilainya tidak memengaruhi konfigurasi terbaik dari orientasi spin, sehingga nilainya dapat diabaikan dalam analisis optimasi namun tetap harus dilibatkan untuk analisis numerik lebih lanjut.

2.5 Komputer Kuantum

Dalam pencarian keadaan dasar Hamiltonian Ising, diperlukan komputasi yang mampu memecahkan masalah optimasi kombinatorial yang kompleks. Komputasi klasik dapat membantu memecahkan masalah kombinatorial namun memiliki keterbatasan dalam memecahkan masalah saat menghadapi kompleksitas permasalahan yang terlalu tinggi. Untuk mengatasi hal tersebut, komputer kuantum dimanfaatkan sebagai alat pengukuran energi kombinasi yang ada secara lebih efisien dari pemodelan Hamiltonian Ising yang sudah terkonstruksi. Selain itu, komputer kuantum dapat memproses dan memanipulasi informasi dalam ruang Hilbert yang sangat luas menggunakan rangkaian kuantum khusus lewat gerbang kuantum yang sangat berbeda dari gerbang klasik (Rajak dkk., 2023). Komputer kuantum memanfaatkan prinsip superposisi dan keterbelitan kuantum untuk menavigasi ruang solusi dari masalah optimasi kombinatorial kompleks yang mana sangat sulit dijangkau oleh sistem komputasi klasik (Preskill, 2018).

Bab ini akan menguraikan bagaimana sistem qubit dapat menjadi fondasi utama teknis komputer kuantum lalu dilanjutkan dengan gerbang kuantum 1 qubit dan 2 qubit. Kemudian dikenalkan pula keterbelitan kuantum (*Quantum entanglement*) sebagai salah satu akibat dari penggunaan gerbang kuantum. Selain itu, juga akan dikenalkan tentang pembelajaran mesin kuantum (*Quantum Machine Learning*) dan rangkaian kuantum berparameter (*Parameterized Quantum Circuit*) sebagai dasar dari algoritma *Variational Quantum Eigensolver* (VQE) yang akan digunakan pada tugas akhir ini.

2.5.1 Sistem Qubit

Qubit didefinisikan sebagai satuan informasi dasar dalam komputasi kuantum yang memperluas konsep bit klasik menjadi sistem dua-keadaan mekanika kuantum sehingga mampu eksis dalam kondisi superposisi. Berbeda dengan bit klasik yang hanya memiliki dua status 0 dan 1 secara eksklusif, qubit didefinisikan secara matematis melalui vektor kolom di ruang vektor kompleks dua-dimensi $\mathbb{C}^2$ (Nielsen & Chuan, 2010). Qubit dapat direpresentasikan dengan basis komputasi ortogonal $|0\rangle$ dan $|1\rangle$ yang dinyatakan melalui

vektor kolom ortonormal dengan bentuk matematis

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{2.40}$$

Keberadaan vektor keadaan qubit yang bersifat superposisi dinyatakan pada vektor keadaan ($|\psi\rangle$) dari keadaan qubit dengan bentuk

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{2.41}$$

di mana α dan β merupakan amplitudo probabilitas sistem kuantum berada di keadaan $|0\rangle$ dan $|1\rangle$. Karena sistem fisik harus memenuhi syarat kekekalan probabilitas yang menyatakan bahwa total probabilitas dari keadaan $|0\rangle$ dan $|1\rangle$ harus bernilai 1 yang berarti pasti ada, dengan bentuk persamaan

$$|\alpha|^2 + |\beta|^2 = 1 \tag{2.42}$$

dimana kuadrat dari amplitudo probabilitas menyatakan peluang sistem kuantum berada di keadaan $|0\rangle$ dan $|1\rangle$ yang bernilai 0 hingga 1 (Purwanto, 2016). Sifat superposisi inilah yang memungkinkan qubit berada dalam kombinasi linier dari kedua basis keadaan dalam satu waktu yang sama, sehingga memberikan potensi komputasi paralel yang jauh melampaui kapabilitas bit klasik.

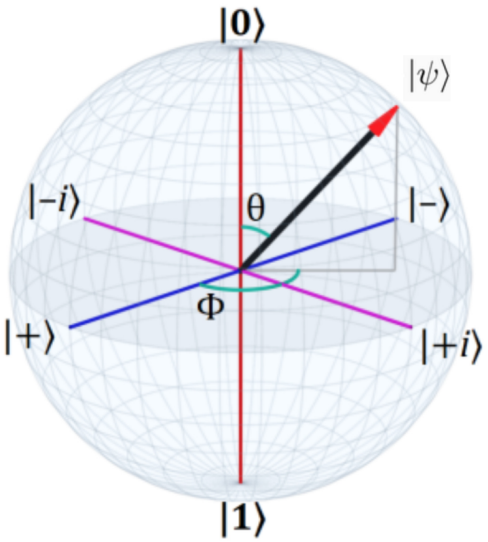
Karena amplitudo probabilitas α dan β merupakan bilangan kompleks yang memenuhi syarat normalisasi, keadaan umum sebuah qubit dapat diparameterkan menggunakan dua parameter sudut, yaitu sudut polar θ dan sudut fase relatif ϕ. Dengan mengabaikan fase global yang tidak memengaruhi hasil pengukuran, vektor keadaan qubit dapat dituliskan dalam bentuk

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle, \tag{2.43}$$

dengan $0 \leq \theta \leq \pi$ dan $0 \leq \phi < 2\pi$. Parameter θ menentukan distribusi amplitudo probabilitas antara keadaan $|0\rangle$ dan $|1\rangle$, sedangkan parameter ϕ menyatakan fase relatif di antara kedua basis tersebut. Keadaan sebuah qubit tersebut dapat direpresentasikan menggunakan representasi Bola Bloch yang memvisualisasikan bagaimana vektor keadaan sebuah qubit terletak di dalam ruang keadaan 3D (Padilla, 2025). Pada gambar 2.7, vektor keadaan pada sebuah qubit direpresentasikan oleh arah garis $|\psi\rangle$ dari tengah menuju permukaan bola (Nakahara, 2008). Vektor tersebut dipengaruhi oleh dua komponen yaitu parameter polar θ yang bergantung pada amplitudo probabilitas dan parameter azimutal sebagai fase relatif ϕ. Ketika vektor keadaan bergerak dari kutub utara menuju kutub selatan sepanjang sumbu z, probabilitas pengukuran pada basis komputasi berubah secara kontinu dari keadaan $|0\rangle$ menjadi $|1\rangle$. Sementara itu, semua keadaan yang terletak pada ekuator Bola Bloch memiliki probabilitas pengukuran yang sama, yaitu 0,5 untuk $|0\rangle$ dan 0,5 untuk $|1\rangle$, namun dibedakan oleh fase relatif yang dinyatakan oleh sudut ϕ. Jika diformalisasikan secara aljabar, posisi geometris dari vektor keadaan tersebut dapat dipetakan ke dalam koordinat Kartesian tiga dimensi pada ruang bola Bloch (Gong dkk., 2025). Koordinat Kartesian x, y, dan z dari titik ujung vektor keadaan didefinisikan secara unik melalui hubungan sudut polar θ dan sudut azimutal ϕ dengan hubungan

$$x = \sin\theta\cos\phi, \quad y = \sin\theta\sin\phi, \quad z = \cos\theta. \tag{2.44}$$

Hubungan koordinat spasial tersebut memetakan ruang status Hilbert dua dimensi ke dalam representasi ruang nyata tiga dimensi secara konsisten untuk mempermudah visualisasi fisis.



Gambar 2.7: Representasi Bola Bloch sebagai visualisasi geometris keadaan murni sebuah qubit. Posisi vektor keadaan ditentukan oleh sudut polar θ dan sudut azimutal ϕ, yang dipetakan ke koordinat Kartesian (x, y, z).

2.5.2 Manipulasi Informasi pada Gerbang 1 Qubit

Manipulasi informasi dalam arsitektur komputasi kuantum dilakukan secara esensial melalui penerapan berbagai gerbang logika yang direpresentasikan sebagai operator uniter U pada ruang vektor kompleks dua dimensi. Secara matematis, representasi matriks uniter paling umum untuk gerbang satu qubit diformulasikan melalui parameterisasi fisis yang melibatkan sudut rotasi dan tiga parameter fase kompleks independen yaitu α, β, serta γ. Konfigurasi operasional dari parameterisasi matriks uniter tersebut dinyatakan ke dalam sebuah representasi persamaan matematis yang merepresentasikan transformasi kuantum umum

$$U = e^{i\alpha} \begin{pmatrix} e^{i\beta}\cos\theta & e^{i\gamma}\sin\theta \\ -e^{-i\gamma}\sin\theta & e^{-i\beta}\cos\theta \end{pmatrix}. \tag{2.45}$$

Penjabaran mengenai proses parameterisasi matriks uniter dua dimensi tersebut dari definisi awal hingga bentuk ekuivalen akhir disajikan secara komprehensif pada Lampiran B. Melalui pemanfaatan bentuk matriks umum tersebut, observasi fisis terhadap berbagai macam gerbang uniter dasar kuantum dapat didemonstrasikan secara sistematis hanya dengan menyesuaikan konfigurasi parameter fase α, β, γ, dan sudut rotasi fisis θ.

Matriks Pauli-X yang bertindak sebagai operator pembalik status keadaan $|0\rangle$ ke $|1\rangle$ dan sebaliknya, dapat diperoleh dengan menetapkan nilai sudut rotasi θ sebesar $\pi/2$. Pemilihan parameter sudut tersebut berimplikasi pada hilangnya kontribusi fungsi trigonometri kosinus dan menyisakan fungsi sinus bernilai satu pada elemen matriks. Untuk menyesuaikan elemen matriks agar bernilai satu, parameter fase relatif γ dipilih sebesar $\pi/2$ sedangkan fase β diatur bernilai nol. Dengan menetapkan nilai fase global α sebesar minus $\pi/2$, diperoleh matriks Pauli-X:

$$X = e^{-i\pi/2} \begin{pmatrix} 0 & e^{i\pi/2} \\ -e^{-i\pi/2} & 0 \end{pmatrix} = -i \begin{pmatrix} 0 & i \\ i & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \tag{2.46}$$

Matriks Pauli-Y juga dapat diperoleh dengan penyesuaian parameter sudut dan fase dari representasi uniter umum. Nilai sudut polar θ kembali dipilih sebesar $\pi/2$, sedangkan

parameter fase relatif γ diatur bernilai nol dan fase β diatur bernilai nol. Penetapan nilai parameter tersebut menghasilkan bentuk matriks uniter yang didominasi oleh elemen bernilai riil pada bagian diagonal kedua matriks tersebut. Selanjutnya, dengan mengimplementasikan nilai fase global α sebesar $\pi/2$, diperoleh matriks Pauli-Y dengan penyimpangan fase global sebesar minus satu:

$$e^{i\pi/2}\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} = i\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & i \\ -i & 0 \end{pmatrix} = -Y. \tag{2.47}$$

Sementara itu, matriks Pauli-Z yang merepresentasikan pergeseran fase relatif pada keadaan qubit dapat diturunkan dengan menetapkan parameter sudut rotasi θ bernilai nol. Pemilihan nilai parameter sudut tersebut mengakibatkan elemen di luar diagonal utama bernilai nol karena kontribusi fungsi trigonometri sinus yang menghilang. Untuk memunculkan perbedaan tanda pada diagonal utama matriks, parameter fase relatif β dipilih sebesar $\pi/2$ sedangkan fase γ bernilai bebas. Selanjutnya, dengan mengalikan matriks tersebut dengan fase global α sebesar minus $\pi/2$, diperoleh representasi matriks Pauli-Z secara presisi tanpa adanya faktor fase imajiner tambahan:

$$Z = e^{-i\pi/2}\begin{pmatrix} e^{i\pi/2} & 0 \\ 0 & e^{-i\pi/2} \end{pmatrix} = -i\begin{pmatrix} i & 0 \\ 0 & -i \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.48}$$

Langkah penyesuaian parameter fase ini secara konsisten membuktikan bahwa seluruh operator Pauli dasar merupakan representasi uniter dari grup khusus kuantum.

Matriks Pauli dapat dihimpun dan dinotasikan dengan σ_x, σ_y, dan σ_z yang tetap merupakan operator Hermitian dan uniter dengan bentuk

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \text{dan} \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.49}$$

Setiap representasi matriks Pauli tersebut difungsikan secara komputasional untuk memanipulasi informasi pada suatu qubit tunggal melalui mekanisme pemetaan basis keadaan secara deterministik. Secara fungsional, gerbang Pauli-X setara dengan gerbang NOT pada sistem komputasi klasik, di mana status keadaan dari $|0\rangle$ ditukar menjadi $|1\rangle$ dan sebaliknya melalui operasi uniter tersebut. Secara aljabar, aksi pemetaan fisis dari operator Pauli-X terhadap himpunan basis komputasi tersebut dapat dinyatakan secara komprehensif melalui relasi kesamaan fisis berikut:

$$X|0\rangle = |1\rangle, \quad X|1\rangle = |0\rangle. \tag{2.50}$$

Di sisi lain, gerbang Pauli-Y melakukan pertukaran status keadaan komputasi sekaligus memberikan fase kompleks imajiner pada konfigurasi amplitudo probabilitas qubit. Karakteristik operasional dari gerbang Pauli-Y tersebut secara geometris merepresentasikan rotasi fisis sebesar 180° terhadap sumbu-y pada koordinat bola Bloch, yang aksi pemetaannya secara matematis dirumuskan sebagai:

$$Y|0\rangle = i|1\rangle, \quad Y|1\rangle = -i|0\rangle. \tag{2.51}$$

Sementara itu, stabilitas nilai amplitudo probabilitas pada setiap basis komputasi dipertahankan secara teknis oleh gerbang Pauli-Z, tetapi tanda fase relatif antara keadaan $|0\rangle$

dan $|1\rangle$ dibalikkan oleh operator tersebut. Mekanisme pergeseran fase asimetris tersebut direpresentasikan dengan rotasi sebesar 180° terhadap sumbu-z pada ruang Hilbert melalui pemetaan fisis:

$$Z|0\rangle = |0\rangle, \quad Z|1\rangle = -|1\rangle. \tag{2.52}$$

Selain variasi gerbang Pauli, gerbang Hadamard (H) direpresentasikan sebagai gerbang logika satu qubit yang mana berasal dari pencarian vektor eigen dari gerbang Pauli-X. Pencarian vektor eigen tersebut diawali dengan persamaan untuk mencari nilai eigen λ dari matriks Pauli-X yang beroperasi di dalam ruang Hilbert dua dimensi. Nilai eigen dari sebuah operator kuantum dapat diperoleh dengan menghitung nilai determinan dari selisih antara matriks Pauli-X dan perkalian skalar matriks identitas yang disamakan dengan nilai nol (Sim dkk., 2019). Pencarian himpunan nilai eigen tersebut dapat dirumuskan melalui persamaan

$$\det(X - \lambda I) = \det \begin{pmatrix} -\lambda & 1 \\ 1 & -\lambda \end{pmatrix} = \lambda^2 - 1 = 0. \tag{2.53}$$

Sehingga diperoleh dua buah nilai eigen, yaitu $\lambda_1 = 1$ dan $\lambda_2 = -1$.

Setelah seluruh nilai eigen berhasil diperoleh, masing-masing vektor eigen dievaluasi dengan mensubstitusikan nilai λ kembali ke dalam persamaan nilai eigen. Hubungan amplitudo antara komponen basis keadaan atas dan komponen basis keadaan bawah dihasilkan dari persamaan untuk representasi nilai eigen pertama bernilai $\lambda_1 = 1$. Vektor eigen pertama yang mewakili keadaan kuantum positif dapat diperoleh dari persamaan

$$|+\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{|0\rangle + |1\rangle}{\sqrt{2}}. \tag{2.54}$$

Di sisi lain, sistem persamaan linier mengharuskan amplitudo dari komponen basis yang memiliki tanda sebaliknya memiliki nilai eigen kedua bernilai $\lambda_2 = -1$. Dengan syarat normalisasi, vektor eigen kedua yang merepresentasikan keadaan kuantum negatif dapat diperoleh lewat persamaan

$$|-\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \frac{|0\rangle - |1\rangle}{\sqrt{2}}. \tag{2.55}$$

Hasil kedua vektor eigen yang saling ortogonal ini digunakan sebagai landasan utama untuk mendefinisikan susunan elemen matriks transformasi basis pada arsitektur sistem komputasi kuantum.

Representasi matriks dari gerbang uniter Hadamard dapat dikonstruksi dengan menyusun kedua vektor eigen hasil dari operator Pauli-X tersebut sebagai komponen kolom. Sebuah bentuk perkalian konstanta terhadap elemen matriks yang memiliki konfigurasi tanda asimetris dapat diperoleh lewat mekanisme penyusunan vektor eigen ortogonal ini dengan relasi matematis

$$H = \begin{pmatrix} |+\rangle & |-\rangle \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{2.56}$$

Dari konstruksi matriks tersebut, basis komputasi gerbang Hadamard menunjukkan suatu kombinasi linier dari superposisi untuk memfasilitasi algoritma komputasi yang bersifat

paralel. Oleh karena itu, operator Hadamard dapat dipetakan terhadap masing-masing basis ortogonal di dalam ruang Hilbert dua dimensi melalui hubungan

$$H|0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \quad H|1\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}. \tag{2.57}$$

Representasi skematis dari operasi matriks Pauli dan gerbang Hadamard tersebut dapat divisualisasikan pada Gambar 2.8, di mana kotak berlabel X, Y, Z, dan H menunjukkan masing-masing gerbang logika kuantum dasar yang bekerja pada alur data qubit.

Gambar 2.8: Simbolisme gerbang Pauli dan gerbang Hadamard pada sirkuit kuantum yang merepresentasikan transformasi uniter dasar X, Y, Z, dan H pada satu qubit.

Matriks-matriks generator Pauli tersebut selanjutnya dapat digunakan sebagai basis operator untuk merepresentasikan arah rotasi dari vektor keadaan qubit pada koordinat bola Bloch. Setiap elemen dari grup khusus uniter $SU(2)$ dapat diekspresikan melalui kombinasi linier dari matriks Pauli:

$$U = e^{-i\frac{\theta}{2}(n_x X + n_y Y + n_z Z)} = e^{-i\frac{\theta}{2}(\hat{n}\cdot\vec{\sigma})}, \tag{2.58}$$

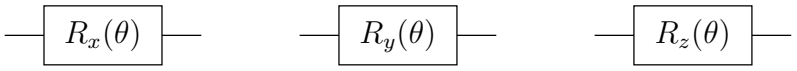
dengan $\vec{\sigma} = (X, Y, Z)$ merupakan vektor matriks Pauli dan $\hat{n} = (n_x, n_y, n_z)$ menyatakan parameter vektor satuan arah rotasi fisis. Evolusi uniter dari keadaan kuantum di bawah pengaruh generator Pauli tersebut dinyatakan dalam bentuk fungsi eksponensial dari operator proyeksi terkait $\hat{p} = \hat{n}\cdot\vec{\sigma}$. Menggunakan ekspansi deret Taylor dan pemisahan suku ganjil-genap dengan sifat $\hat{p}^2 = I$, operator eksponensial tersebut dapat ditransformasikan menjadi bentuk identitas Euler kuantum melalui proses berikut:

$$\begin{aligned}
e^{-i\frac{\theta}{2}\hat{p}} &= \sum_{n=0}^{\infty} \frac{(-i\frac{\theta}{2}\hat{p})^n}{n!} \\
&= \sum_{k=0}^{\infty} \frac{(-i\frac{\theta}{2})^{2k}\hat{p}^{2k}}{(2k)!} + \sum_{k=0}^{\infty} \frac{(-i\frac{\theta}{2})^{2k+1}\hat{p}^{2k+1}}{(2k+1)!} \\
&= \left(\sum_{k=0}^{\infty} \frac{(-1)^k(\frac{\theta}{2})^{2k}}{(2k)!}\right) I + \left(\sum_{k=0}^{\infty} \frac{(-1)^k(\frac{\theta}{2})^{2k+1}}{(2k+1)!}\right) \hat{p} \\
&= \cos\left(\frac{\theta}{2}\right) I - i\sin\left(\frac{\theta}{2}\right) \hat{p}
\end{aligned} \tag{2.59}$$

Dengan mensubstitusikan operator proyeksi arah $\hat{p}$ dengan masing-masing matriks generator Pauli (σ_x, σ_y, σ_z) ke dalam identitas Euler tersebut, diperoleh representasi matriks eksplisit untuk gerbang rotasi kuantum satu qubit sebagai berikut:

$$\begin{aligned}
R_x(\theta) &= e^{-i\theta X/2} = \cos\frac{\theta}{2} I - i\sin\frac{\theta}{2} X = \begin{pmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}, \\
R_y(\theta) &= e^{-i\theta Y/2} = \cos\frac{\theta}{2} I - i\sin\frac{\theta}{2} Y = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}, \\
R_z(\theta) &= e^{-i\theta Z/2} = \cos\frac{\theta}{2} I - i\sin\frac{\theta}{2} Z = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix}.
\end{aligned} \tag{2.60}$$

Penurunan rumus penuh dari persamaan (2.58) menjadi persamaan (2.60) dapat ditinjau pada Lampiran C. Parameter sudut rotasi θ tersebut menjadi parameter yang dioptimasi bagi algoritma klasik untuk mencari konfigurasi energi terendah dari Hamiltonian Ising. Representasi skematik dari gerbang rotasi satu qubit divisualisasikan pada Gambar 2.9, di mana kotak berlabel $R_x(\theta)$, $R_y(\theta)$, dan $R_z(\theta)$ menunjukkan masing-masing operasi transformasi uniter yang bekerja pada alur data qubit.



Gambar 2.9: Simbolisme gerbang rotasi pada sirkuit kuantum yang merepresentasikan rotasi parameter tunggal pada sumbu-X, sumbu-Y, dan sumbu-Z di bola Bloch.

2.5.3 Manipulasi Informasi pada Gerbang 2 Qubit

Manipulasi informasi yang dilakukan oleh gerbang kuantum tidak terbatas hanya pada 1 qubit, tetapi juga dapat diterapkan pada 2 qubit untuk menghasilkan efek-efek yang lebih kompleks (W. Li dkk., 2025). Secara matematis, bentuk matriks dari 2 qubit didapatkan dari hasil kali tensor dari matriks semua basis 1 qubit yang terlibat (Revythi & Koukiou, 2025). Perkalian tensor tersebut memungkinakn kombinasi status keadaan fungsi gelombang individual menjadi status basis gabungan yang merepresentasikan ruang probabilitas Hilbert empat dimensi. Keempat vektor basis gabungan tersebut membangun landasan ruang operasional matematis sistem komputasi fisis gabungan, yang direpresentasikan ke dalam bentuk persamaan fungsi pemetaan komputasional sebagai berikut:

$$\begin{aligned}
|00\rangle &= |0\rangle \otimes |0\rangle = \begin{pmatrix}1 & 0 & 0 & 0\end{pmatrix}^T, \\
|01\rangle &= |0\rangle \otimes |1\rangle = \begin{pmatrix}0 & 1 & 0 & 0\end{pmatrix}^T, \\
|10\rangle &= |1\rangle \otimes |0\rangle = \begin{pmatrix}0 & 0 & 1 & 0\end{pmatrix}^T, \\
|11\rangle &= |1\rangle \otimes |1\rangle = \begin{pmatrix}0 & 0 & 0 & 1\end{pmatrix}^T.
\end{aligned} \tag{2.61}$$

Salah satu gerbang dua qubit yang berfungsi sebagai elemen dasar untuk mengendalikan interaksi dalam ruang probabilitas adalah operator *Controlled-NOT* (CNOT). Gerbang transformasi CNOT merupakan sebuah gerbang uniter yang memiliki aksi pemetaan langsung terhadap keadaan basis komputasi dari sistem yang terdiri dari qubit kontrol dan qubit target. Gerbang CNOT bekerja dengan mengubah keadaan qubit target menjadi keadaan sebaliknya apabila syarat dari qubit kontrol terpenuhi, yaitu $|1\rangle$. Jika qubit pertama menjadi qubit kontrol dan qubit kedua adalah qubit target, maka secara matematis gerbangnya akan menjadi

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{2.62}$$

Sehingga jika difungsikan ke dalam ruang probabilitas ruang keadaan gabungan, aksi CNOT akan menghasilkan masing-masing keluaran keadaan basis gabungan sebagai ber-

ikut:

$$\text{CNOT}|00\rangle = |00\rangle, \quad \text{CNOT}|01\rangle = |01\rangle, \quad \text{CNOT}|10\rangle = |11\rangle, \quad \text{CNOT}|11\rangle = |10\rangle. \tag{2.63}$$

Dari pemetaan komputasi fisis tersebut, maka dapat dibentuk persamaan CNOT berdasarkan mekanisme operasi outer product dari operator transformasi CNOT. Bentuk representasi konstruksi turunan matriks komputasi fisis ini meregulasi orientasi aksi interaktif secara matematis berdasarkan skema polaritas penetapan arah operasi parameter kontrol utamanya menjadi

$$\text{CNOT} = \sum_{i,j} |\text{output}\rangle\langle\text{input}|. \tag{2.64}$$

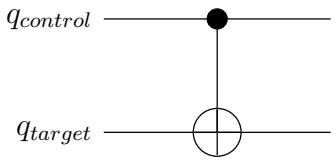
Berdasarkan mekanisma operasi outer product tersebut, maka dapat dibentuk matriks CNOT sebagai berikut:

$$\begin{aligned} \text{CNOT}_{01} &= |00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 11| + |11\rangle\langle 10|, \\ \text{CNOT}_{10} &= |00\rangle\langle 00| + |11\rangle\langle 01| + |01\rangle\langle 10| + |10\rangle\langle 11|. \end{aligned} \tag{2.65}$$

Sehingga secara matematis dapat dibentuk CNOT qubit kontrol pertama dengan qubit target kedua dan sebaliknya menjadi

$$\text{CNOT}_{01} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad \text{CNOT}_{10} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}. \tag{2.66}$$

Pada rangkaian kuantum, CNOT memiliki skema pada Gambar 2.10. Gambar tersebut memberikan notasi pada lintasan pemrosesan kuantum di mana inisiasi qubit kontrol ditandai dengan representasi koordinat titik solid (•), sedangkan qubit target ditandai dengan penggunaan bentuk plus di dalam lingkara (⊕).



Gambar 2.10: Representasi simbolik gerbang keterbelitan CNOT yang mengilustrasikan interaksi non-lokal antar qubit dalam sistem sirkuit.

Selain dapat memanipulasi informasi kuantum, gerbang CNOT juga berperan penting dalam menghasilkan keterbelitan kuantum (entanglement). Suatu sistem kuantum dikatakan terbelit apabila probabilitas hasil pengukuran pada setiap subsistem tidak lagi dapat dijelaskan secara independen, Namun menunjukkan korelasi kuantum yang tidak akan dapat diobservasi pada sistem klasik. Salah satu contoh keadaan terbelit maksimum adalah Keadaan Bell (Bell States), yaitu himpunan empat keadaan dua qubit. Pada umumnya, keadaan-keadaan yang terhimpun dalam (Bell States) dapat dibangkitkan menggunakan kombinasi gerbang Hadamard dan CNOT. Pertama, gerbang Hadamard

diterapkan pada qubit pertama untuk menghasilkan superposisi, sedangkan qubit kedua tidak mengalami perubahan. Untuk keadaan awal $|00\rangle$, proses tersebut menghasilkan

$$(H \otimes I)|00\rangle = \frac{|00\rangle + |10\rangle}{\sqrt{2}} \tag{2.67}$$

Keadaan tersebut merupakan superposisi dari dua basis komputasi yang masih dapat dipisahkan. Selanjutnya, gerbang CNOT diterapkan dengan qubit pertama sebagai kontrol dan qubit kedua sebagai target. Operasi ini membalik keadaan qubit target ketika qubit kontrol berada pada keadaan $|1\rangle$, sehingga diperoleh:

$$\text{CNOT}\left(\frac{|00\rangle + |10\rangle}{\sqrt{2}}\right) = \frac{|00\rangle + |11\rangle}{\sqrt{2}} = |\Phi^+\rangle \tag{2.68}$$

Keadaan $|\Phi^+\rangle$ merupakan salah satu dari empat Keadaan Bell dan tidak dapat dituliskan sebagai hasil kali tensor dua keadaan qubit tunggal, sehingga keadaan tersebut merepresentasikan sistem dua qubit yang terbelit secara maksimum.

Dengan menerapkan kombinasi gerbang Hadamard dan CNOT pada basis komputasi yang berbeda, dapat diperoleh empat keadaan Bell yang saling ortogonal. Keempat keadaan ini membentuk suatu basis ortonormal pada ruang Hilbert dua qubit dan merepresentasikan bentuk keterbelitan maksimum. Karena membentuk basis lengkap, setiap keadaan dua qubit dapat dinyatakan sebagai kombinasi linier dari Keadaan Bell. Secara matematis, keempat Keadaan Bell tersebut dapat dinyatakan sebagai

$$\begin{aligned} |\Phi^\pm\rangle &= \frac{|00\rangle \pm |11\rangle}{\sqrt{2}} = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 0 & 0 & \pm 1\end{pmatrix}^T, \\ |\Psi^\pm\rangle &= \frac{|01\rangle \pm |10\rangle}{\sqrt{2}} = \frac{1}{\sqrt{2}}\begin{pmatrix}0 & 1 & \pm 1 & 0\end{pmatrix}^T \end{aligned} \tag{2.69}$$

Keempat keadaan tersebut memiliki amplitudo probabilitas yang sama besar pada dua basis komputasi penyusunnya, sehingga pengukuran pada salah satu qubit akan langsung menentukan hasil pengukuran pada qubit lainnya. Sifat inilah yang menjadikan Keadaan Bell sebagai sumber daya fundamental dalam berbagai protokol komputasi dan komunikasi kuantum, seperti teleportasi kuantum, superdense coding, dan distribusi kunci kuantum.

Selain dapat dibangkitkan lewat kombinasi gerbang Hadamard dan CNOT, keterbelitan juga dapat dibangkitkan menggunakan kombinasi gerbang rotasi yang dikombinasikan dengan gerbang CNOT. Hal tersebut dapat terjadi karena gerbang Hadamard memiliki hubungan dengan gerbang rotasi. Secara matematis, gerbang Hadamard ekuivalen dengan kombinasi gerbang rotasi terhadap sumbu-Y dan sumbu-Z. Hubungan tersebut dapat dinyatakan sebagai

$$H = e^{i\pi/2} R_z(\pi) R_y\left(\frac{\pi}{2}\right) \sim R_z(\pi) R_y\left(\frac{\pi}{2}\right) \tag{2.70}$$

Karena fase global tidak memiliki konsekuensi fisik yang dapat diamati, faktor $e^{i\pi/2}$ umumnya dapat diabaikan. Ketika keadaan awal sistem adalah $|0\rangle$, pengaruh $R_Z(\pi)$ hanya berupa perubahan fase yang tidak mengubah distribusi probabilitas hasil pengukuran. Oleh karena itu, pembentukan superposisi sering kali cukup dianalisis melalui aksi gerbang rotasi $R_Y(\theta)$. Untuk menunjukkan bahwa gerbang R_Y dapat menghasilkan superposisi yang ekuivalen dengan gerbang Hadamard, aksi gerbang tersebpat ditinjau

terhadap keadaan dasar $|0\rangle$. Variabel sudut rotasi pada matriks gerbang rotasi terhadap sumbu-Y yang didefinisikan pada persamaan (2.45) disubstitusikan dengan nilai parameter rotasi $\theta = \pi/2$, sehingga diperoleh

$$R_y\left(\frac{\pi}{2}\right)|0\rangle = \begin{pmatrix} \cos\left(\frac{\pi}{2}\right) & -\sin\left(\frac{\pi}{2}\right) \\ \sin\left(\frac{\pi}{2}\right) & \cos\left(\frac{\pi}{2}\right) \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \tag{2.71}$$

Hasil tersebut menunjukkan bahwa gerbang $R_Y(\pi/2)$ menghasilkan superposisi dengan amplitudo yang sama besar pada basis $|0\rangle$ dan $|1\rangle$. Hal tersebut serupa dengan gerbang Hadamard dalam membangkitkan superposisi awal pada suatu qubit.

Keadaan superposisi tersebut selanjutnya dapat diperluas ke sistem dua qubit dengan menerapkan gerbang $R_Y(\pi/2)$ pada qubit kontrol dan gerbang identitas pada qubit target. Untuk keadaan awal $|00\rangle$, diperoleh

$$\left(R_y\left(\frac{\pi}{2}\right) \otimes I\right)|00\rangle = \frac{|00\rangle + |10\rangle}{\sqrt{2}} \tag{2.72}$$

yang mana identik dengan pembangkitan superposisi melalui penerapan gerbang Hadamard pada qubit pertama. Selanjutnya, dengan menerapkan gerbang CNOT, diperoleh

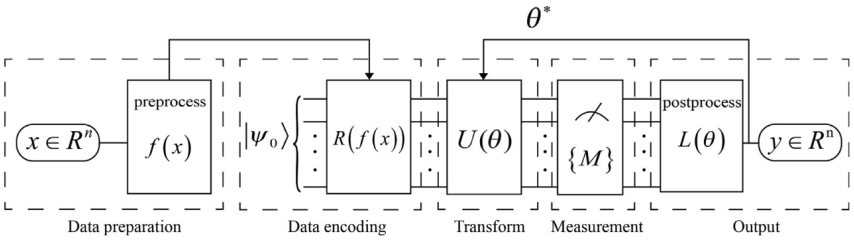
$$\text{CNOT}\left(R_y\left(\frac{\pi}{2}\right) \otimes I\right)|00\rangle = \text{CNOT}\frac{|00\rangle + |10\rangle}{\sqrt{2}} = |\Phi^+\rangle. \tag{2.73}$$

2.5.4 Sirkuit Kuantum Berparameter

Pembelajaran Mesin Kuantum atau *Quantum Machine Learning* (QML) merupakan bidang multidisipliner yang menggabungkan prinsip komputasi kuantum dengan metode pemelajaran mesin. Salah satu pendekatan yang berkembang dalam QML adalah Pembelajaran Mendalam Kuantum atau *Quantum Deep Learning* (QDL), yaitu pemanfaatan model kuantum dengan parameter khusus untuk mempelajari pemodelan data kompleks. Implementasi model QDL umumnya diwujudkan melalui Sirkuit Kuantum Berparameter atau *Parameterized Quantum Circuit* (PQC), yaitu sirkuit kuantum yang terdiri atas gerbang-gerbang kuantum yang memiliki parameter yang nilainya dapat dioptimasi. PQC menjadi komponen utama dalam keluarga Algoritma Kuantum Variasional atau *Variational Quantum Algorithms* (VQA), yaitu algoritma hibrida yang menggabungkan pemrosesan kuantum dan optimasi klasik. Keluarga VQA terdiri dari algoritma Pencari Nilai Eigen Kuantum Variasional atau *Variational Quantum Eigensolver* (VQE), Algoritma Optimisasi Aproksimasi Kuantum atau *Quantum Approximate Optimization Algorithm* (QAOA), dan Klasifikasi Kuantum Variasional atau *Variational Quantum Classifier* (VQC), yang mana ketiganya memanfaatkan PQC sebagai representasi solusi yang dievaluasi untuk kemudian nilainya diperbarui oleh algoritma optimasi klasik secara iteratif (Chen dkk., 2025).

PQC adalah rangkaian gerbang kuantum yang memiliki parameter kontinu, umumnya berupa sudut rotasi (Benedetti dkk., 2019). Dalam algoritma variasional, rangkaian PQC berperan sebagai Ansatz yang mengeksplorasi parameter pada ruang solusi untuk fungsi gelombang dari suatu permasalahan. Pada umumnya, alur kerja PQC ditunjukkan pada Gambar 2.11. Secara singkat, proses diawali dengan penyandian data klasik $x \in \mathbb{R}^n$ ke dalam keadaan kuantum $|\psi_0\rangle$ melalui fungsi prapemrosesan $f(x)$ dan operasi penyandian

$R(f(x))$. Keadaan kuantum yang telah terbentuk kemudian diproses oleh sirkuit variasional $U(\boldsymbol{\theta})$, yang bertugas mentransformasikan keadaan kuantum melalui serangkaian operasi kuantum berparameter. Parameter $\boldsymbol{\theta}$ pada sirkuit ini akan diperbarui secara iteratif selama proses optimasi untuk menghasilkan suatu keadaan kuantum $|\psi(\boldsymbol{\theta})\rangle$ yang merepresentasikan solusi optimum. Selanjutnya, keadaan akhir diproyeksikan melalui pengukuran $\{M\}$ untuk memperoleh nilai ekspektasi atau probabilitas observabel yang relevan. Hasil pengukuran tersebut kemudian dievaluasi menggunakan fungsi biaya $L(\boldsymbol{\theta})$, yang digunakan oleh algoritma optimasi klasik untuk menentukan pembaruan parameter hingga diperoleh parameter optimal $\boldsymbol{\theta}$ dan keluaran model $y \in \mathbb{R}^n$.



Gambar 2.11: Representasi alur kerja sirkuit kuantum berparameter (PQC) yang mengilustrasikan tahap penyandian data, sirkuit variasional, pengukuran, dan pembaruan parameter oleh algoritma optimasi klasik.

Setelah data dikodekan ke dalam keadaan kuantum $|\psi(\boldsymbol{\theta})\rangle$, sirkuit variasional $U(\boldsymbol{\theta})$ diterapkan untuk menghasilkan keadaan kuantum yang bergantung pada parameter-parameter optimasi. Berdasarkan sifat evolusi uniter dalam mekanika kuantum, keadaan keluaran sirkuit dapat dituliskan sebagai

$$|\psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|\psi_0\rangle. \tag{2.74}$$

di mana operator variasional $U(\boldsymbol{\theta})$ dibangun dari kombinasi gerbang rotasi satu-qubit dan gerbang keterbelitan dua-qubit. Secara umum, operator tersebut dapat dinyatakan

$$U(\boldsymbol{\theta}) = \prod_{l=1}^{L} U_{(ent)}^{(l)} U_{(rot)}^{(l)}(\boldsymbol{\theta}^{(l)}) \tag{2.75}$$

dengan $U_{(rot)}^{(l)}$ merepresentasikan lapisan gerbang rotasi berparameter, seperti $Rx(\theta)$, $Ry(\theta)$, dan $Rz(\theta)$, sedangkan $U_{(ent)}^{(l)}$ menyatakan lapisan gerbang keterbelitan, misalnya gerbang CNOT. Struktur berlapis ini memungkinkan sirkuit menghasilkan keadaan kuantum yang semakin kompleks seiring bertambahnya jumlah parameter dan lapisan yang digunakan. Variasi nilai parameter $\boldsymbol{\theta}$ akan menghasilkan fungsi gelombang yang berbeda sehingga memungkinkan eksplorasi berbagai kandidat solusi di dalam ruang pencarian.

Kualitas suatu kandidat solusi kemudian dievaluasi melalui pengukuran nilai ekspektasi suatu operator observabel $\hat{O}$, yang umumnya merepresentasikan fungsi biaya atau besaran fisik yang ingin dioptimalkan (Tomaz dkk., 2025). Nilai fungsi biaya tersebut dirumuskan sebagai

$$C(\boldsymbol{\theta}) = \langle\psi(\boldsymbol{\theta})|\hat{O}|\psi(\boldsymbol{\theta})\rangle. \tag{2.76}$$

Nilai hasil pengukuran tersebut kemudian dikirimkan kepada algoritma optimasi klasik untuk memperbarui parameter sirkuit, kemudian proses evaluasi dan pembaruan diulang hingga kriteria konvergensi terpenuhi. Proses ini dikenal sebagai Optimisasi Hibrida Kuantum-Klasik *hybrid quantum-classical optimization* atau optimasi hibrida kuantum-klasik (C dkk., 2025).

2.6 Optimasi Parameter pada Sistem Klasik

Dalam pemodelan sistem makroskopik, pencarian titik ekuilibrium pada algoritma kuantum merupakan proses yang sangat rentan terhadap pengaruh derau (*noise*) lingkungan. Komputasi pada perangkat keras kuantum era NISQ sering kali menghasilkan distorsi sinyal informasi yang signifikan sehingga memerlukan metode optimasi parameter yang stabil untuk menavigasi kompleksitas ruang solusi secara kokoh. Algoritma optimasi klasik digunakan sebagai pendukung sistem hibrida untuk melakukan kalibrasi model secara iteratif untuk meminimalkan fungsi biaya yang ditentukan. Oleh karena itu, diperlukan pemilihan metodologi evaluasi gradien dengan mempertimbangkan efisiensi sumber daya dan ketahanan terhadap fluktuasi statistik yang melekat pada pengukuran observabel kuantum skala industri (Spall, 1998).

Metode optimasi yang tepat memungkinkan sistem untuk mencapai konvergensi energi minimum meskipun beroperasi di tengah keterbatasan koherensi sirkuit (Fontana dkk., 2021). Algoritma gradien klasik konvensional sering kali mengalami kendala teknis saat diterapkan langsung pada lanskap energi kuantum yang bergejolak akibat derau pengukuran (Sharma dkk., 2020). Ketidakstabilan numerik ini dapat diatasi dengan menerapkan teknik estimasi gradien yang mengeksploitasi properti analitis dari operator uniter tanpa memperburuk eror yang terjadi pada sistem (Wierichs dkk., 2022). Bab ini akan menguraikan implementasi teknis dari metode aturan pergeseran parameter (*Parameter Shift Rule*) dan optimasi SPSA sebagai pilar utama dalam pembaruan parameter sirkuit secara efisien dan akurat.

2.6.1 Metode Pergeseran Parameter (*Parameter Shift Rule*)

Dalam setiap proses optimasi berparameter pada arsitektur komputasi kuantum, diperlukan perhitungan gradien analitis yang cepat dan tepat untuk mengarahkan pembaruan matriks parameter menuju titik konfigurasi energi minimum. Pendekatan konvensional dengan menggunakan metode diferensiasi beda hingga (*finite difference*) sering kali sulit diimplementasikan karena algoritma tersebut sangat bergantung pada presisi pengukuran selisih ruang parameter berskala sangat kecil. Ketergantungan terhadap interval pergeseran fasa infinitesimal pada perangkat keras sirkuit dunia nyata selalu memicu munculnya ketidakstabilan akibat derau statistik yang merusak akurasi perhitungan gradien. Untuk mengatasi hambatan teknis tersebut, metode pergeseran parameter diimplementasikan ke dalam simulasi karena mampu mengeksploitasi sifat periodik dari operator uniter pada gerbang rotasi kuantum secara eksak (Mitarai dkk., 2018). Penerapan algoritma perhitungan analitis ini diawali secara sistematis dengan merumuskan turunan parsial dari nilai ekspektasi energi terhadap variabel sudut representasi θ melalui hubungan:

$$\begin{aligned}\frac{\partial E(\theta)}{\partial \theta} &= \frac{\partial}{\partial \theta}\langle\psi(\theta)|\hat{\mathcal{H}}|\psi(\theta)\rangle \\ &= \left\langle \frac{\partial \psi(\theta)}{\partial \theta}\middle| \hat{\mathcal{H}}|\psi(\theta)\rangle + \langle\psi(\theta)|\hat{\mathcal{H}}\middle|\frac{\partial \psi(\theta)}{\partial \theta}\right\rangle .\end{aligned} \tag{2.77}$$

Untuk suatu gerbang rotasi yang bergantung pada parameter θ, keadaan kuantum dapat dituliskan sebagai

$$|\psi(\theta)\rangle = \hat{U}_B\hat{U}(\theta)\hat{U}_A|0\rangle, \qquad \hat{U}(\theta) = e^{-i\frac{\theta}{2}\sigma}, \tag{2.78}$$

dengan $\hat{U}(\theta)$ merupakan generator gerbang rotasi yang berbasis pada generator pauli $\sigma \in \{I, X, Y, Z\}$ sebagaimana persamaan (2.60) dengan ukuran matriks sebesar $2^n \times 2^n$ dengan n merupakan jumlah qubit. Definisikan

$$|\phi\rangle = \hat{U}_A|0\rangle, \qquad \hat{M} = \hat{U}_B^\dagger \hat{\mathcal{H}} \hat{U}_B \tag{2.79}$$

di mana $\hat{U}_A$ merupakan operator uniter sebelum gerbang rotasi dan $\hat{U}_B$ merupakan operator uniter sesudah gerbang rotasi. Nilai ekspektasi energi kemudian dapat ditulis ulang sebagai

$$\begin{aligned} E(\theta) &= \langle\psi(\theta)|\hat{\mathcal{H}}|\psi(\theta)\rangle \\ &= \langle\phi|\hat{U}^\dagger(\theta)\hat{M}\hat{U}(\theta)|\phi\rangle. \end{aligned} \tag{2.80}$$

Turunannya terhadap θ adalah

$$\frac{\partial E(\theta)}{\partial\theta} = \langle\phi|\frac{\partial\hat{U}^\dagger(\theta)}{\partial\theta}\hat{M}\hat{U}(\theta)|\phi\rangle + \langle\phi|\hat{U}^\dagger(\theta)\hat{M}\frac{\partial\hat{U}(\theta)}{\partial\theta}|\phi\rangle. \tag{2.81}$$

Karena

$$\frac{\partial\hat{U}(\theta)}{\partial\theta} = -\frac{i}{2}\sigma\hat{U}(\theta), \tag{2.82}$$

maka diperoleh

$$\begin{aligned} \frac{\partial E(\theta)}{\partial\theta} &= \frac{i}{2}\langle\psi(\theta)|\sigma\hat{M}|\psi(\theta)\rangle - \frac{i}{2}\langle\psi(\theta)|\hat{M}\sigma|\psi(\theta)\rangle \\ &= \frac{i}{2}\langle\psi(\theta)|[\sigma, \hat{M}]|\psi(\theta)\rangle. \end{aligned} \tag{2.83}$$

Bentuk komutator tersebut sulit diukur secara langsung pada perangkat NISQ. Oleh karena itu digunakan pergeseran parameter sebesar s:

$$\begin{aligned} \hat{U}(\theta \pm s) &= \exp\left[-i\frac{\theta \pm s}{2}\sigma\right] \\ &= \hat{U}(\theta)\hat{U}(\pm s). \end{aligned} \tag{2.84}$$

Untuk pergeseran positif,

$$\begin{aligned} E(\theta + s) &= \langle\phi|\hat{U}^\dagger(\theta)\hat{U}^\dagger(s)\hat{M}\hat{U}(\theta)\hat{U}(s)|\phi\rangle \\ &= \langle\psi(\theta)|\hat{U}^\dagger(s)\hat{M}\hat{U}(s)|\psi(\theta)\rangle. \end{aligned} \tag{2.85}$$

Dengan menggunakan identitas Euler

$$\hat{U}(s) = \cos\left(\frac{s}{2}\right)\hat{I} - i\sin\left(\frac{s}{2}\right)\sigma, \tag{2.86}$$

diperoleh

$$E(\theta + s) = \cos^2\left(\frac{s}{2}\right)\langle\hat{M}\rangle + \sin^2\left(\frac{s}{2}\right)\langle\sigma\hat{M}\sigma\rangle + i\sin\left(\frac{s}{2}\right)\cos\left(\frac{s}{2}\right)\langle[\sigma, \hat{M}]\rangle. \tag{2.87}$$

Dengan cara yang sama,

$$E(\theta - s) = \cos^2\left(\frac{s}{2}\right)\langle\hat{M}\rangle + \sin^2\left(\frac{s}{2}\right)\langle\sigma\hat{M}\sigma\rangle - i\sin\left(\frac{s}{2}\right)\cos\left(\frac{s}{2}\right)\langle[\sigma, \hat{M}]\rangle. \tag{2.88}$$

Mengurangkan kedua persamaan menghasilkan

$$\begin{aligned} E(\theta+s) - E(\theta-s) &= 2i\sin\left(\frac{s}{2}\right)\cos\left(\frac{s}{2}\right)\langle[\sigma,\hat{M}]\rangle \\ &= i\sin(s)\langle[\sigma,\hat{M}]\rangle. \end{aligned} \tag{2.89}$$

Jika dipilih $s = \pi/2$, maka hubungan gradien menjadi

$$\frac{\partial E(\theta)}{\partial\theta} = \frac{1}{2}\left[E\left(\theta+\frac{\pi}{2}\right) - E\left(\theta-\frac{\pi}{2}\right)\right]. \tag{2.90}$$

Persamaan (2.90) menunjukkan bahwa gradien dapat dihitung hanya dari dua evaluasi sirkuit yang digeser sebesar $\pm\pi/2$, sehingga sangat sesuai untuk optimasi pada perangkat kuantum NISQ (Mitarai dkk., 2018).

Algoritma utama yang menggunakan prinsip dari parameter shift rule adalah algoritma Gradient Descent (GD). Algoritma GD merupakan metode optimasi yang memperbarui parameter pada setiap langkah iterasi ke-t menggunakan vektor gradien energi $\nabla E(\theta_t)$ dan laju pembelajaran η. Proses pembaruan parameter ini dilakukan secara repetitif untuk mengarahkan sistem menuju lembah fungsi biaya dengan mengikuti arah kemiringan negatif dari permukaan energi. Secara matematis, prosedur iteratif untuk memperbarui konfigurasi parameter sirkuit dinyatakan melalui persamaan pembaruan berikut:

$$\theta_{t+1} = \theta_t + \eta\nabla E(\theta_t). \tag{2.91}$$

Meskipun Gradient Descent yang dikombinasikan dengan metode pergeseran parameter memberikan hasil yang sangat akurat, implementasinya pada sistem berdimensi tinggi sering kali kurang efisien dalam penggunaan sumber daya. Hal ini disebabkan oleh kebutuhan dua kali evaluasi sirkuit untuk setiap komponen parameter tunggal yang ingin dioptimasi dalam jaringan sirkuit berukuran besar.

2.6.2 Optimasi SPSA (*Simultaneous Perturbation Stochastic Approximation*)

Untuk mengatasi keterbatasan kendala efisiensi yang melekat pada algoritma GD tersebut, algoritma SPSA digunakan sebagai metode optimasi stokastik yang mampu mengestimasi seluruh komponen gradien hanya menggunakan dua evaluasi fungsi biaya pada setiap iterasi (Spall, 1998). Pada iterasi ke-k, SPSA membangkitkan vektor perturbasi acak

$$\Delta_k = (\Delta_{k,1}, \Delta_{k,2}, \ldots, \Delta_{k,p}), \tag{2.92}$$

dengan setiap komponen perturbasi mengikuti distribusi Rademacher

$$\Delta_{k,i} = \begin{cases} +1, & P = \frac{1}{2}, \\ -1, & P = \frac{1}{2}. \end{cases} \tag{2.93}$$

Vektor perturbasi tersebut digunakan untuk menghasilkan dua himpunan parameter yang digeser secara simultan ke arah positif dan negatif,

$$\boldsymbol{\theta}_k^+ = \boldsymbol{\theta}_k + c_k\Delta_k, \tag{2.94}$$

$$\boldsymbol{\theta}_k^- = \boldsymbol{\theta}_k - c_k\Delta_k, \tag{2.95}$$

dengan c_k menyatakan amplitudo perturbasi pada iterasi ke-k. Evaluasi fungsi biaya pada kedua titik tersebut menghasilkan

$$E_+ = E(\boldsymbol{\theta}_k + c_k\Delta_k), \tag{2.96}$$

dan

$$E_- = E(\boldsymbol{\theta}_k - c_k\Delta_k). \tag{2.97}$$

Hubungan antara kedua evaluasi energi tersebut dan gradien fungsi biaya dapat diperoleh melalui ekspansi deret Taylor di sekitar titik $\boldsymbol{\theta}_k$. Untuk arah perturbasi positif diperoleh

$$\begin{aligned} E_+ &= E(\boldsymbol{\theta}_k + c_k\Delta_k) \\ &= E(\boldsymbol{\theta}_k) + c_k\sum_{j=1}^{p}\Delta_j\frac{\partial E}{\partial\theta_j} + \frac{1}{2}c_k^2\sum_{j=1}^{p}\Delta_j^2\frac{\partial^2 E}{\partial\theta_j^2} + O(c_k^3), \end{aligned} \tag{2.98}$$

sedangkan untuk arah perturbasi negatif diperoleh

$$\begin{aligned} E_- &= E(\boldsymbol{\theta}_k - c_k\Delta_k) \\ &= E(\boldsymbol{\theta}_k) - c_k\sum_{j=1}^{p}\Delta_j\frac{\partial E}{\partial\theta_j} + \frac{1}{2}c_k^2\sum_{j=1}^{p}\Delta_j^2\frac{\partial^2 E}{\partial\theta_j^2} + O(c_k^3). \end{aligned} \tag{2.99}$$

Dengan mengurangkan Persamaan (2.98) dan (2.99), seluruh suku berpangkat genap saling menghilangkan sehingga diperoleh

$$E_+ - E_- = 2c_k\sum_{j=1}^{p}\Delta_j\frac{\partial E}{\partial\theta_j} + O(c_k^3). \tag{2.100}$$

Berdasarkan hasil tersebut, estimator gradien SPSA untuk komponen parameter ke-i didefinisikan sebagai

$$\hat{g}_i = \frac{E_+ - E_-}{2c_k\Delta_i}. \tag{2.101}$$

Dengan mensubstitusikan Persamaan (2.100) ke dalam Persamaan (2.101), diperoleh

$$\begin{aligned} \hat{g}_i &= \frac{1}{\Delta_i}\sum_{j=1}^{p}\Delta_j\frac{\partial E}{\partial\theta_j} \\ &= \frac{\Delta_i}{\Delta_i}\frac{\partial E}{\partial\theta_i} + \sum_{j\neq i}\frac{\Delta_j}{\Delta_i}\frac{\partial E}{\partial\theta_j} \\ &= \frac{\partial E}{\partial\theta_i} + \sum_{j\neq i}\frac{\Delta_j}{\Delta_i}\frac{\partial E}{\partial\theta_j}. \end{aligned} \tag{2.102}$$

Karena setiap komponen perturbasi bersifat independen dan mengikuti distribusi Rademacher, maka berlaku

$$\mathbb{E}\left[\frac{\Delta_j}{\Delta_i}\right] = 0, \qquad j \neq i. \tag{2.103}$$

Dengan mengambil nilai ekspektasi pada Persamaan (2.102), diperoleh

$$\begin{aligned}\mathbb{E}[\hat{g}_i] &= \frac{\partial E}{\partial \theta_i} + \sum_{j \neq i} \mathbb{E}\left[\frac{\Delta_j}{\Delta_i}\right] \frac{\partial E}{\partial \theta_j} \\ &= \frac{\partial E}{\partial \theta_i}.\end{aligned} \tag{2.104}$$

Hasil tersebut menunjukkan bahwa estimator SPSA merupakan estimator tak-bias (*unbiased estimator*) terhadap gradien sebenarnya. Sehingga meskipun estimasi gradien diperoleh hanya dari dua evaluasi fungsi biaya, rata-rata statistik estimator tersebut tetap konvergen menuju gradien eksak (Spall, 1998).

Setelah gradien diperoleh, parameter sirkuit diperbarui menggunakan aturan

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - a_k \hat{\mathbf{g}}_k, \tag{2.105}$$

dengan $\hat{\mathbf{g}}_k$ menyatakan vektor gradien hasil estimasi SPSA. Untuk menjaga stabilitas konvergensi, SPSA menggunakan hiperparameter yang mengalami peluruhan secara bertahap sepanjang iterasi optimasi,

$$a_k = \frac{a}{(A + k + 1)^{\alpha}}, \qquad c_k = \frac{c}{(k + 1)^{\gamma}}, \tag{2.106}$$

dengan a, A, c, α, dan γ merupakan konstanta yang ditentukan sebelum proses optimasi dimulai. Karena jumlah evaluasi fungsi biaya pada setiap iterasi tidak bergantung pada banyaknya parameter sirkuit, kompleksitas estimasi gradien SPSA tetap konstan terhadap dimensi ruang parameter. Karakteristik ini menjadikan SPSA sangat sesuai untuk optimasi PQC berskala besar pada perangkat NISQ yang rentan terhadap derau statistik dan memiliki keterbatasan sumber daya pengukuran (Spall, 1998).

2.7 Algoritma VQE (*Variational Quantum Eigensolver*)

VQE merupakan algoritma komputasi kuantum hibrida yang menggabungkan kemampuan pemrosesan kuantum dan optimasi klasik untuk menyelesaikan permasalahan pencarian keadaan dasar suatu Hamiltonian (Peruzzo dkk., 2014). Pendekatan ini dirancang untuk memanfaatkan keunggulan perangkat kuantum dalam merepresentasikan keadaan pada ruang Hilbert berdimensi tinggi, sekaligus memanfaatkan efisiensi komputer klasik dalam melakukan optimasi numerik. Dalam ber bagai bidang, termasuk kimia kuantum, ilmu material, dan optimasi kombinatorial, VQE menjadi salah satu algoritma yang paling banyak digunakan pada era perangkat NISQ karena kebutuhan sumber dayanya relatif lebih rendah dibandingkan algoritma kuantum yang sepenuhnya bergantung pada koreksi kesalahan.

Pada skema VQE, unit pemrosesan kuantum atau *Quantum Processing Unit* (QPU) digunakan untuk menyiapkan keadaan kuantum serta mengevaluasi nilai ekspektasi Hamiltonian yang berperan sebagai fungsi biaya (Fedorov dkk., 2022). Nilai ekspektasi tersebut kemudian dikirimkan ke Unit Pemrosesan Pusat atau *Central Processing Unit* (CPU), yang bertugas memperbarui parameter sirkuit menggunakan algoritma optimasi klasik (S. Wang dkk., 2025). Interaksi berulang antara kedua komponen ini membentuk suatu

siklus umpan-balik yang secara bertahap mengarahkan parameter menuju nilai optimum. Bab ini akan membahas beberapa landasan teoritis yang mendasari mekanisme kerja VQE (Dao, 2025). Pembahasan diawali dengan prinsip variasi dalam mekanika kuantum yang menjadi dasar formulasi masalah optimasi energi (Regadío, 2025). Selanjutnya akan dijelaskan konstruksi keadaan kuantum melalui Ansatz berparameter, serta mekanisme optimasi yang menghubungkan evaluasi kuantum dan pemrosesan klasik (Scursulim dkk., 2026). Keseluruhan komponen tersebut membentuk kerangka kerja VQE dalam mencari pendekatan terbaik terhadap keadaan dasar suatu sistem kuantum (Y. Wang dkk., 2022).

2.7.1 Prinsip Variasi Rayleigh-Ritz dalam Mekanika Kuantum

Salah satu permasalahan utama dalam mekanika kuantum adalah menentukan keadaan dasar (ground state) dari suatu sistem yang dideskripsikan oleh Hamiltonian $\mathcal{H}$. Untuk sistem sederhana, keadaan dasar dapat diperoleh dengan menyelesaikan persamaan Schrödinger secara analitik. Namun, pada sistem yang memiliki banyak partikel atau derajat kebebasan yang besar, solusi eksak umumnya sulit diperoleh sehingga diperlukan pendekatan numerik yang lebih efisien. Prinsip variasi menyediakan kerangka matematis untuk memperkirakan energi keadaan dasar tanpa harus mengetahui solusi eksak persamaan Schrödinger. Prinsip ini menyatakan bahwa energi keadaan dasar dapat diperoleh melalui proses minimisasi nilai ekspektasi Hamiltonian terhadap seluruh keadaan kuantum yang memenuhi syarat normalisasi. Dengan demikian, pencarian keadaan dasar dapat diformulasikan sebagai suatu masalah optimasi yang mencari keadaan dengan energi serendah mungkin.

Misalkan suatu sistem kuantum dideskripsikan oleh vektor keadaan $|\psi\rangle$ dalam ruang Hilbert dan memiliki Hamiltonian $\mathcal{H}$ yang memuat seluruh informasi energi sistem. Untuk keadaan tersebut, energi rata-rata sistem dinyatakan melalui fungsional energi

$$E = \langle\psi|\mathcal{H}|\psi\rangle. \tag{2.107}$$

Notasi $E[\psi]$ menunjukkan bahwa energi merupakan suatu fungsional yang bergantung pada bentuk keadaan kuantum yang dipilih. Berbeda dengan fungsi biasa yang bergantung pada suatu variabel, fungsional bergantung pada keseluruhan fungsi atau vektor keadaan. Oleh karena itu, perubahan pada bentuk fungsi gelombang akan menghasilkan perubahan pada nilai energi yang dihitung melalui fungsional tersebut. Karena interpretasi probabilistik mekanika kuantum mensyaratkan total probabilitas bernilai satu, maka keadaan yang diizinkan harus memenuhi kondisi normalisasi

$$\langle\psi|\psi\rangle = 1. \tag{2.108}$$

Kondisi ini bertindak sebagai kendala dalam proses optimasi karena tidak semua variasi fungsi gelombang dapat dipilih secara bebas. Setiap keadaan yang digunakan dalam proses minimisasi harus tetap mempertahankan syarat normalisasi tersebut. Sehingga, pencarian energi minimum harus dilakukan pada himpunan keadaan kuantum.

Untuk menangani kendala tersebut, digunakan metode pengali Lagrange. Metode ini memungkinkan proses optimasi dilakukan tanpa harus menghilangkan kendala normalisasi. Sebagai akibatnya, perlu ditambahkan suatu parameter tambahan λ yang berfungsi memastikan bahwa keadaan hasil optimasi tetap memenuhi syarat normalisasi. Dengan menggunakan pengali Lagrange, didefinisikan suatu fungsional baru

$$\mathcal{F}[\psi] = \langle\psi|\mathcal{H}|\psi\rangle - \lambda\left(\langle\psi|\psi\rangle - 1\right). \tag{2.109}$$

Adapun fungsi dari $\mathcal{F}[\psi]$ merupakan fungsi objektif yang akan dioptimalkan. Pada persamaan 2.109, suku pertama menyatakan energi sistem, sedangkan suku kedua berfungsi mempertahankan kondisi normalisasi selama proses variasi. Nilai parameter λ belum diketahui pada tahap ini dan akan muncul secara alami sebagai bagian dari solusi optimasi.

Karena konstanta 1 tidak memberikan kontribusi terhadap proses variasi, suku tersebut dapat diabaikan sehingga fungsional di atas dapat dituliskan kembali dalam bentuk yang lebih ringkas sebagai

$$\mathcal{F}[\psi] = \langle\psi|\mathcal{H}|\psi\rangle - \lambda\langle\psi|psi\rangle. \tag{2.110}$$

Bentuk inilah yang akan digunakan untuk memperoleh kondisi stasioner dari fungsional energi. Pada bagian berikutnya akan ditunjukkan bahwa penerapan prinsip variasi terhadap fungsional tersebut menghasilkan persamaan nilai eigen Hamiltonian yang menjadi dasar teoritis bagi pencarian keadaan dasar dalam algoritma VQE.

Kondisi energi minimum diperoleh dengan mencari titik stasioner dari fungsional $\mathcal{F}[\psi]$. Dalam kalkulus variasi, titik stasioner adalah keadaan ketika variasi orde pertama dari suatu fungsional bernilai nol terhadap perubahan infinitesimal fungsi yang diizinkan. Secara fisik, kondisi ini menyatakan bahwa perubahan yang sangat kecil pada keadaan kuantum tidak menghasilkan perubahan linear pada nilai energi. Oleh karena itu, keadaan stasioner merupakan kandidat solusi yang dapat merepresentasikan keadaan fisis sistem. Kondisi stasioner tersebut dinyatakan melalui

$$\frac{\delta\mathcal{F}}{\delta\psi} = 0. \tag{2.111}$$

dengan simbol δ menyatakan operator variasi. Berbeda dengan diferensial biasa (d) yang mengukur perubahan suatu fungsi terhadap variabel independen tertentu, operator variasi mengukur perubahan suatu fungsional akibat deformasi kecil pada fungsi atau keadaan kuantum itu sendiri. Dengan demikian, operator variasi memungkinkan analisis terhadap bagaimana nilai fungsional berubah ketika bentuk fungsi gelombang dimodifikasi secara infinitesimal.

Secara konseptual, variasi keadaan kuantum dapat dituliskan sebagai

$$|\psi\rangle \to |\psi\rangle + \epsilon|\eta\rangle, \tag{2.112}$$

dengan ϵ merupakan parameter real yang sangat kecil dan $|\eta\rangle$ adalah fungsi variasi arbitrer yang memenuhi syarat batas yang sama dengan $|\psi\rangle$. Variasi pertama dari suatu fungsional didefinisikan sebagai koefisien linear terhadap ϵ ketika $\epsilon \to 0$. Apabila koefisien linear tersebut bernilai nol untuk setiap variasi yang diperbolehkan, maka keadaan yang ditinjau merupakan titik stasioner dari fungsional.

Karena fungsi gelombang pada umumnya bernilai kompleks, maka $|\psi\rangle$ dan dual vektornya $\langle\psi|$ diperlakukan sebagai variabel independen selama proses variasi. Dengan melakukan variasi terhadap fungsional $\mathcal{F}[\psi]$, diperoleh

$$\delta\mathcal{F} = \langle\delta\psi|\mathcal{H}|\psi\rangle + \langle\psi|\mathcal{H}|\delta\psi\rangle - \lambda\left(\langle\delta\psi|\psi\rangle + \langle\psi|\delta\psi\rangle\right). \tag{2.113}$$

Seluruh suku yang bergantung pada variasi dapat kemudian dikelompokkan sehingg diperoleh

$$\delta\mathcal{F} = \langle\delta\psi|\left(\mathcal{H} - \lambda\right)|\psi\rangle + c.c., \tag{2.114}$$

dengan *c.c.* konjugat kompleks (*complex conjugate*) menyatakan suku konjugat kompleks yang berasal dari variasi terhadap $|\psi\rangle$. Karena $\langle\delta\psi|$ bersifat arbitrer, syarat $\delta\mathcal{F} = 0$ hanya dapat dipenuhi apabila faktor yang mengalikan variasi tersebut bernilai nol. Akibatnya diperoleh hubungan

$$\begin{aligned} (\mathcal{H} - \lambda)\,|\psi\rangle &= 0, \\ \mathcal{H}|\psi\rangle &= \lambda|\psi\rangle. \end{aligned} \tag{2.115}$$

Persamaan di atas identik dengan masalah nilai eigen (eigenvalue problem) dalam mekanika kuantum. Vektor keadaan $|\psi\rangle$ berperan sebagai fungsi eigen Hamiltonian, sedangkan λ merupakan nilai eigennya. Karena Hamiltonian merepresentasikan energi sistem, nilai eigen tersebut memiliki interpretasi fisik sebagai tingkat energi yang diizinkan oleh sistem kuantum.

Di antara seluruh nilai eigen Hamiltonian, nilai yang paling kecil merepresentasikan energi keadaan dasar. Dengan mengidentifikasi λ sebagai energi eigen dan memilih nilai minimum yang mungkin dicapai, diperoleh persamaan keadaan dasar

$$\mathcal{H}|\psi_0\rangle = E_0|\psi_0\rangle. \tag{2.116}$$

Hasil ini menunjukkan bahwa pencarian energi minimum melalui prinsip variasi ekuivalen dengan pencarian keadaan dasar Hamiltonian. Namun, pada sistem kuantum yang kompleks, bentuk eksak keadaan dasar umumnya tidak diketahui sehingga penyelesaian analitik menjadi sulit dilakukan. Oleh karena itu, diperlukan pendekatan yang memungkinkan energi keadaan dasar diperkirakan melalui suatu keadaan percobaan (*trial state*) yang lebih mudah dibangun.

2.7.2 Teorema Rayleigh–Ritz dan Dasar Algoritma VQE

Misalkan $|\phi\rangle$ merupakan sembarang keadaan kuantum ternormalisasi yang digunakan sebagai pendekatan terhadap keadaan dasar sistem. Keadaan tersebut tidak harus merupakan fungsi eigen Hamiltonian, tetapi cukup berada dalam ruang Hilbert yang sama dengan sistem yang sedang ditinjau. Untuk keadaan tersebut dapat didefinisikan nilai ekspektasi energi sebagai

$$E[\phi] = \frac{\langle\phi|\mathcal{H}|\phi\rangle}{\langle\phi|\phi\rangle}. \tag{2.117}$$

Jika keadaan percobaan diekspansikan ke dalam basis fungsi eigen Hamiltonian, maka nilai ekspektasi energi selalu dapat dinyatakan sebagai kombinasi berbobot dari nilai-nilai eigen energi sistem. Karena seluruh koefisien probabilitas bernilai tidak negatif dan jumlahnya sama dengan satu, energi yang diperoleh tidak mungkin lebih kecil daripada energi keadaan dasar. Hasil ini dikenal sebagai teorema Rayleigh–Ritz dan dinyatakan sebagai

$$\frac{\langle\phi|\mathcal{H}|\phi\rangle}{\langle\phi|\phi\rangle} \geq E_0. \tag{2.118}$$

Teorema Rayleigh–Ritz memberikan batas bawah yang sangat penting dalam mekanika kuantum. Semakin dekat keadaan percobaan terhadap keadaan dasar sesungguhnya,

semakin dekat pula nilai ekspektasi energi terhadap E_0. Dengan demikian, pencarian keadaan dasar dapat diubah menjadi permasalahan optimasi yang bertujuan meminimalkan nilai ekspektasi energi. Algoritma VQE memanfaatkan prinsip tersebut dengan menggantikan keadaan percobaan umum $|\phi\rangle$ menjadi keadaan kuantum berparameter $|\psi(\boldsymbol{\theta})\rangle$ yang dibangun menggunakan suatu sirkuit kuantum. Energi sistem kemudian dituliskan sebagai fungsi parameter

$$E(\boldsymbol{\theta}) = \langle\psi(\boldsymbol{\theta})|\mathcal{H}|\psi(\boldsymbol{\theta})\rangle. \tag{2.119}$$

Tujuan utama VQE adalah mencari parameter optimal $\boldsymbol{\theta}^*$ yang meminimalkan nilai energi tersebut. Berdasarkan Persamaan (2.118), energi minimum yang diperoleh akan selalu berada di atas atau sama dengan energi keadaan dasar yang sesungguhnya. Oleh karena itu, kualitas solusi VQE ditentukan oleh kemampuan Ansatz dalam menghasilkan keadaan kuantum yang cukup dekat dengan keadaan dasar sistem yang dicari.Meskipun prinsip variasi menjamin keberadaan prosedur optimasi untuk mendekati energi keadaan dasar, keberhasilan VQE sangat bergantung pada bentuk Ansatz yang digunakan untuk merepresentasikan keadaan kuantum. Oleh karena itu, langkah berikutnya adalah membangun suatu Ansatz yang cukup ekspresif untuk menangkap korelasi kuantum sistem, tetapi tetap dapat direalisasikan pada perangkat kuantum yang tersedia.

2.7.3 Ansatz UCC (*Unitary Coupled Cluster*)

Sebagaimana dijelaskan pada bagian sebelumnya, algoritma VQE mencari keadaan kuantum yang meminimalkan nilai ekspektasi Hamiltonian. Namun, proses optimasi tidak dapat dilakukan secara langsung terhadap seluruh ruang Hilbert karena dimensinya bertumbuh secara eksponensial terhadap jumlah partikel atau qubit. Oleh karena itu, diperlukan suatu bentuk keadaan percobaan yang direpresentasikan melalui sejumlah parameter yang dapat dioptimalkan. Dalam konteks VQE, keadaan percobaan tersebut dikenal sebagai Ansatz.

Salah satu Ansatz yang banyak digunakan dalam simulasi kimia kuantum adalah Klaster Tergandeng Uniter atau *Unitary Coupled Cluster* (UCC). Ansatz ini dikembangkan dari metode klaster tergandeng atau *Coupled Cluster* pada kimia komputasi dan dirancang untuk menangkap korelasi elektron secara sistematis. Keunggulan utama UCC adalah kemampuannya mempertahankan sifat uniter evolusi kuantum sehingga lebih sesuai untuk implementasi pada komputer kuantum dibandingkan formulasi Klaster Tergandeng konvensional. Keadaan kuantum pada Ansatz UCC dibangkitkan melalui operator uniter

$$|\psi_{\mathrm{UCC}}(\boldsymbol{\theta})\rangle = e^{\hat{T}(\boldsymbol{\theta})-\hat{T}^\dagger(\boldsymbol{\theta})}|\Phi_0\rangle. \tag{2.120}$$

Pada persamaan tersebut, $|\Phi_0\rangle$ umumnya dipilih sebagai keadaan referensi Hartree–Fock yang merepresentasikan pendekatan medan-rata (*mean-field*) terhadap sistem. Operator eksponensial kemudian menghasilkan superposisi berbagai konfigurasi tereksitasi yang memperbaiki deskripsi korelasi antarpartikel. Dengan demikian, keadaan akhir memiliki akurasi yang lebih baik dibandingkan keadaan referensi awal.

Operator cluster $\hat{T}$ biasanya dibangun dari kombinasi eksitasi tunggal dan eksitasi ganda yang dituliskan sebagai

$$\hat{T}_1 = \sum_{i,a} t_i^a \hat{a}_a^\dagger \hat{a}_i, \quad \hat{T}_2 = \sum_{i,j,a,b} t_{ij}^{ab} \hat{a}_a^\dagger \hat{a}_b^\dagger \hat{a}_j \hat{a}_i. \tag{2.121}$$

Indeks i, j menyatakan orbital yang mula-mula terisi, sedangkan a, b menyatakan orbital yang tidak terisi. Operator $\hat{T}_1$ merepresentasikan perpindahan satu partikel dari orbital terisi ke orbital kosong, sedangkan $\hat{T}_2$ merepresentasikan eksitasi dua partikel secara simultan. Koefisien t_a^i dan t_{ab}^{ij} berperan sebagai parameter variasional yang akan dioptimalkan selama proses VQE.

Karena generator $\hat{T} - \hat{T}^\dagger$ bersifat anti-Hermitian, operator eksponensial yang dihasilkan bersifat uniter (Dixit & Kumari, 2025). Sifat ini menjamin bahwa probabilitas total sistem tetap terjaga selama evolusi kuantum berlangsung (Tilly dkk., 2022). Selain itu, unitaritas memungkinkan implementasi Ansatz dalam bentuk rangkaian gerbang kuantum yang sesuai dengan prinsip dasar komputasi kuantum.Meskipun memiliki dasar fisika yang kuat, implementasi UCC sering kali memerlukan jumlah gerbang kuantum yang besar (A. Wang dkk., 2025). Kompleksitas tersebut menyebabkan kedalaman sirkuit meningkat secara signifikan ketika ukuran sistem bertambah (Wei dkk., 2025). Pada perangkat NISQ yang masih rentan terhadap noise dan dekoherensi, kebutuhan sumber daya yang tinggi ini menjadi salah satu hambatan utama penggunaan UCC secara langsung.

2.7.4 Arsitektur Ansatz EfficientSU2

Keterbatasan perangkat NISQ mendorong pengembangan berbagai Ansatz yang Efisien terhadap Perangkat Keras atau *hardware-efficient Ansatz* (HEA) yang lebih mudah diimplementasikan secara eksperimental. Jika UCC dibangun berdasarkan struktur fisika sistem, maka HEA dirancang dengan mempertimbangkan keterbatasan perangkat keras kuantum yang tersedia. Tujuannya adalah menghasilkan Ansatz yang cukup ekspresif untuk proses optimasi sekaligus memiliki kedalaman sirkuit yang relatif rendah.

Salah satu HEA yang banyak digunakan dalam implementasi VQE adalah EfficientSU2. Ansatz ini dibangun dari kombinasi gerbang rotasi satu-qubit dan lapisan keterbelitan yang disusun secara berulang. Struktur tersebut memungkinkan eksplorasi ruang keadaan kuantum yang luas tanpa memerlukan jumlah gerbang yang terlalu besar. Untuk sistem yang terdiri atas N qubit dan kedalaman sirkuit sebesar L, keadaan kuantum dapat dituliskan sebagai

$$|\psi(\boldsymbol{\theta})\rangle = \left[\prod_{l=1}^{L} \hat{U}_{\text{ent}} \left(\bigotimes_{j=1}^{N} \hat{R}_y(\theta_{l,j,y})\hat{R}_z(\theta_{l,j,z})\right)\right] \left(\bigotimes_{j=1}^{N} \hat{R}_y(\theta_{0,j,y})\hat{R}_z(\theta_{0,j,z})\right) |0\rangle^{\otimes N}, \quad (2.122)$$

Operator gerbang rotasi $\hat{R}_y$ dan $\hat{R}_z$ memungkinkan setiap qubit bergerak pada berbagai arah di bola Bloch. Parameter sudut yang terkait dengan gerbang-gerbang tersebut menjadi variabel bebas yang akan dioptimalkan oleh algoritma klasik. Semakin banyak parameter yang tersedia, semakin besar pula ruang solusi yang dapat dieksplorasi selama proses optimasi. Selain rotasi satu-qubit, EfficientSU2 juga memanfaatkan lapisan keterbelitan yang umumnya dibangun menggunakan gerbang CNOT. Lapisan ini bertujuan agar korelasi kuantum antar qubit dapat terbentuk. Dibandingkan dengan UCC, EfficientSU2 umumnya menghasilkan sirkuit yang lebih dangkal dan lebih sesuai untuk perangkat kuantum saat ini.

2.8 Validasi *Barren Plateau*

Keberhasilan algoritma hibrida kuantum-klasik pada era NISQ sangat bergantung pada efektivitas proses optimasi parameter. Salah satu tantangan utama dalam pelatihan sirkuit variasional adalah fenomena Barren Plateau, yaitu kondisi ketika informasi gradien menghilang sehingga proses optimasi mengalami stagnasi (McClean dkk., 2018). Untuk mendeteksi kondisi tersebut, penelitian ini memanfaatkan Entropi Von Neumann sebagai indikator keterbelitan sistem kuantum. Pada Bab ini, akan dibahas landasan teoretis fenomena *Barren Plateau* dan formulasi Entropi Von Neumann yang digunakan sebagai metrik validasi keterbelitan sistem kuantum.

2.8.1 *Barren Plateau*

Ketika algoritma kuantum variasional semakin berkembang, muncul pertanyaan mengenai kemampuan algoritma tersebut dalam mengatasi sistem ketika kuantum semakin besar saat algoritma dilatih. Pada tahap awal pengembangannya, banyak penelitian berfokus pada peningkatan ekspresivitas Ansatz melalui penambahan jumlah qubit dan kedalaman sirkuit. Namun, pendekatan tersebut ternyata memunculkan tantangan baru berupa hilangnya informasi gradien yang dibutuhkan selama proses optimasi. Fenomena ini pertama kali dianalisis oleh (McClean dkk., 2018), yang menunjukkan bahwa lanskap Hamiltonian pada sirkuit kuantum acak dapat menjadi sangat datar sehingga pelatihan parameter menjadi semakin sulit. Fenomena tersebut dikenal sebagai barren plateau, yaitu kondisi ketika lanskap Hamiltonian pada algoritma kuantum variasional menjadi semakin datar seiring bertambahnya jumlah qubit (N) dan kedalaman sirkuit (L). Pada kondisi ini, perubahan kecil pada parameter klasik hanya menghasilkan perubahan yang sangat kecil pada nilai fungsi energi.

Secara etimologi, *barren plateau* berasal dari kata barren yang berarti tandus atau tidak produktif dan plateau yang berarti dataran tinggi yang relatif datar. Karakteristik utama dari *barren plateau* dapat diamati melalui perilaku varians gradien terhadap ukuran sistem kuantum. Ketika jumlah qubit dan kedalaman sirkuit meningkat, varians gradien cenderung menurun secara eksponensial sehingga nilai gradien yang terukur semakin sulit dibedakan dari fluktuasi statistik dan derau pengukuran (Serafini dkk., 2003; Anshu dkk., 2020). Secara umum, hubungan tersebut dapat dituliskan sebagai

$$\mathrm{Var}[\partial_\theta E] \approx \frac{C}{2^{N+L}} \tag{2.123}$$

dengan C merupakan konstanta yang bergantung pada Hamiltonian sistem dan struktur Hamiltonian yang digunakan. Persamaan tersebut menunjukkan bahwa peningkatan jumlah qubit maupun kedalaman sirkuit menyebabkan sinyal gradien menyusut secara eksponensial (Ortiz Marrero dkk., 2021). Dalam batas tertentu, nilai gradien menjadi sangat kecil sehingga hampir tidak memberikan informasi yang berguna bagi proses optimasi (Barron & Cover, 1991).

2.8.2 Entropi Von Neumann

Entropi Von Neumann digunakan untuk mengukur tingkat ketidakpastian dan keterbelitan dalam sistem kuantum. Konsep ini merupakan generalisasi dari Entropi Shannon

pada teori informasi klasik (Shannon, 1948). Untuk suatu distribusi probabilitas diskrit p_i, Entropi Shannon ($H(p_i)$) didefinisikan sebagai

$$H(p_i) = -\sum_i p_i \log_2(p_i). \tag{2.124}$$

Logaritma berbasis 2 digunakan karena satuan informasi yang dihasilkan dinyatakan dalam bit sebagai satuan dasar informasi pada sistem komputasi biner. Entropi Shannon merepresentasikan jumlah rata-rata informasi yang diperoleh ketika hasil suatu peristiwa acak diketahui (Page, 1993). Semakin besar nilai entropi, semakin tinggi pula tingkat ketidakpastian distribusi probabilitas yang mendeskripsikan sistem tersebut (Ohya, 1998). Dalam mekanika kuantum, distribusi probabilitas tersebut direpresentasikan menggunakan operator densitas ρ, yang mana untuk suatu ansambel keadaan dapat dituliskan sebagai

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|. \tag{2.125}$$

Operator densitas memiliki beberapa sifat penting, yaitu bersifat Hermitian, memiliki jejak bernilai satu, dan bersifat semidefinit positif (Shin dkk., 2024). Sifat Hermitian tersebut dapat ditunjukkan melalui hubungan

$$\rho^\dagger = \left(\sum_i p_i |\psi_i\rangle\langle\psi_i|\right)^\dagger = \sum_i p_i |\psi_i\rangle\langle\psi_i| = \rho. \tag{2.126}$$

Karena ρ merupakan operator Hermitian, maka berdasarkan Teorema Spektral operator tersebut memiliki himpunan nilai eigen real dan vektor-vektor eigen yang saling ortogonal. Dengan demikian, ρ dapat didiagonalisasi pada suatu basis ortonormal yang dibentuk oleh vektor-vektor eigennya (Kumar, 2025). Melalui dekomposisi spektral, operator densitas dapat dinyatakan dalam basis ortonormal $|e_j\rangle$ dengan nilai eigen λ_j sebagai

$$\rho = \sum_j \lambda_j |e_j\rangle\langle e_j|. \tag{2.127}$$

Karena fungsi suatu operator dapat dievaluasi melalui nilai eigennya, logaritma berbasis dua dari operator densitas dapat dituliskan sebagai

$$\log_2(\rho) = \sum_j \log_2(\lambda_j) |e_j\rangle\langle e_j| \tag{2.128}$$

Menggunakan proses penurunan lengkap yang ada pada Lampiran D. Dengan sifat ortonormalitas basis

$$\langle e_j | e_k\rangle = \delta_{jk} \tag{2.129}$$

yang akan bernilai 0 jika $j \neq k$ dan bernilai 1 jika $j = k$. Maka perkalian antara operator densitas dan logaritmanya dapat disederhanakan menjadi

$$\begin{aligned} \rho \log_2(\rho) &= \left(\sum_j \lambda_j |e_j\rangle\langle e_j|\right)\left(\sum_k \log_2(\lambda_k) |e_k\rangle\langle e_k|\right) \\ &= \sum_j \sum_k \lambda_j \log_2(\lambda_k) |e_j\rangle\langle e_j|e_k\rangle\langle e_k| \\ &= \sum_j \sum_k \lambda_j \log_2(\lambda_k) |e_j\rangle\delta_{jk}\langle e_k|. \end{aligned} \tag{2.130}$$

Karena $\delta_{jk} = 0$ ketika $j \neq k$, maka suku yang tersisa hanya ketika $j = k$, maka

$$\rho \log_2(\rho) = \sum_j \lambda_j \log_2(\lambda_j) |e_j\rangle\langle e_j|. \tag{2.131}$$

Langkah berikutnya adalah menerapkan operasi *trace* untuk memperoleh besaran skalar yang merepresentasikan total informasi statistik sistem

$$\begin{aligned}
\mathrm{Tr}(\rho \log_2(\rho)) &= \mathrm{Tr}\left(\sum_j \lambda_j \log_2(\lambda_j) |e_j\rangle\langle e_j|\right) \\
&= \sum_k \langle e_k| \left(\sum_j \lambda_j \log_2(\lambda_j) |e_j\rangle\langle e_j|\right) |e_k\rangle \\
&= \sum_k \sum_j \lambda_j \log_2(\lambda_j) \langle e_k|e_j\rangle\langle e_j|e_k\rangle \\
&= \sum_k \lambda_k \log_2(\lambda_k) \delta_{kk}\delta_{kk} \\
&= \sum_k \lambda_k \log_2(\lambda_k).
\end{aligned} \tag{2.132}$$

Namun karena $j = k$, maka dapat dinotasikan juga dengan bentuk

$$\mathrm{Tr}(\rho \log_2(\rho)) = \sum_j \lambda_j \log_2(\lambda_j). \tag{2.133}$$

Berdasarkan penjabaran struktur operasi tersebut, perumusan persamaan Entropi Von Neumann $S(\rho)$ diperoleh dari

$$S(\rho) = -\mathrm{Tr}(\rho \log_2(\rho)) = -\sum_j \lambda_j \log_2(\lambda_j). \tag{2.134}$$

Dalam implementasi sirkuit kuantum berparameter, Entropi Von Neumann juga dapat digunakan sebagai indikator untuk menganalisis fenomena Barren Plateau. Peningkatan nilai entropi yang mendekati batas maksimumnya mengindikasikan bahwa keadaan kuantum menjadi semakin terdistribusi secara merata di dalam ruang Hilbert (Crosato & Morris, 2023). Secara matematis, jika keadaan sistem memiliki entropi yang mendekati maksimum

$$S(\rho_A) \approx \log(d_A), \tag{2.135}$$

maka gradien dari ekspektasi Hamiltonian $\nabla\langle\psi(\theta)|\hat{H}|\psi(\theta)\rangle$ cenderung mengalami penyusutan amplitudo akibat distribusi probabilitas yang semakin homogen dalam ruang keadaan. Hal ini mengakibatkan sensitivitas Hamiltonian terhadap perubahan parameter menjadi sangat kecil, yang secara empiris tercermin sebagai penurunan varians gradien pada proses optimasi.

Bab 3

Metodologi

Tugas akhir ini menggunakan tiga pilar metodologis utama yaitu analisis ekonofisika, teori permainan strategis, dan komputasi kuantum variasional diintegrasikan untuk mengoptimalkan pemilihan portofolio biner secara hibrida. Untuk menghindari kebingungan simbolis, diberikan notasi matematis yang digunakan pada modul-modul metodologi sebagaimana pada Tabel 3.1.

Tabel 3.1: Daftar Simbol dan Notasi Matematis dalam Metodologi.

Simbol	**Definisi**
μ	Vektor ekspektasi imbal hasil (*expected return*) aset.
Σ	Matriks kovariansi yang merepresentasikan risiko sistemik antar aset.
γ	Parameter penghindaran risiko (*risk aversion*) endogen.
λ	Koefisien pinalti untuk penegakan kendala kardinalitas aset.
K	Jumlah target aset yang akan dipilih ke dalam portofolio (*budget*).
N	Jumlah total aset yang tersedia dalam semesta pemilihan.
x_i	Variabel keputusan biner untuk aset ke-i ($x_i \in \{0, 1\}$).
Φ	Fungsi potensial global dalam kerangka *Exact Potential Game*.
$\hat{\mathcal{H}}$	Operator Hamiltonian yang merepresentasikan energi total sistem kuantum.
θ	Vektor parameter sudut rotasi pada sirkuit kuantum variasional.
a_k, c_k	Hiperparameter laju pembelajaran dan perturbasi pada optimasi SPSA.
s.t.	*subject to* (dengan kendala atau batasan matematis).

3.1 Alur Penelitian

Penelitian ini diawali dengan merumuskan alur metodologi yang komprehensif untuk memecahkan masalah optimasi portofolio biner menggunakan pendekatan hibrida kuantum. Struktur metodologis ini dipetakan secara sistematis untuk menghubungkan tahap ekstraksi data finansial historis dengan perumusan model Hamiltonian Ising yang merepresentasikan lanskap energi sistem. Lanskap energi tersebut selanjutnya dikalibrasi menggunakan prinsip EPG guna menemukan titik ekuilibrium strategis sebelum diproses secara intensif oleh perangkat keras kuantum. Titik ekuilibrium strategis ini bertindak sebagai basis inisialisasi awal bagi algoritma Pencari Nilai Eigen Kuantum Variasional yang

efisien terhadap perangkaat keras atau *Hardware-Efficient Variational Quantum Eigensolver* (HE-VQE) dalam menemukan konfigurasi portofolio dengan energi paling minimum. Keseluruhan alur kerja komputasi kuantum tersebut kemudian dievaluasi menggunakan simulasi pengujian historis (*backtesting*) berkelanjutan untuk mengukur ketahanan strategi investasi pada kondisi pasar riil dengan penyimbangan kembali (*rebalancing*) portofolio tiap bulan.

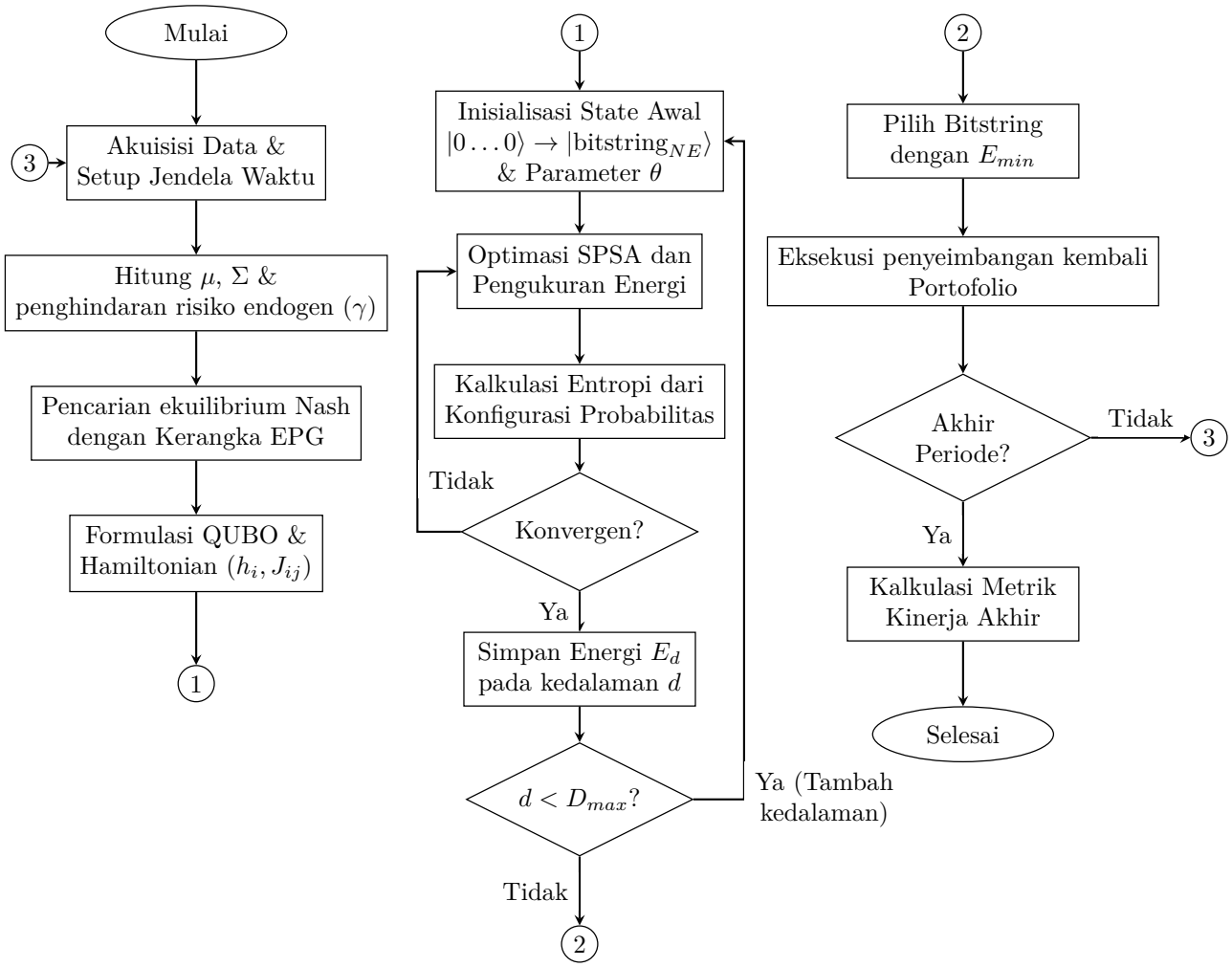
Diagram alir metodologi yang divisualisasikan pada Gambar 3.1 menjabarkan keterkaitan fungsional antar setiap modul komputasi secara sekuensial dari awal hingga evaluasi akhir. Modul pra-pemrosesan data klasik berperan krusial dalam menghasilkan parameter fundamental seperti vektor imbal hasil dan matriks kovariansi yang mutlak dibutuhkan untuk membentuk operator kuantum. Proses optimasi kuantum yang divisualisasikan secara eksplisit mencakup siklus iteratif penyesuaian sudut gerbang rotasi kuantum serta mekanisme adaptasi kedalaman sirkuit secara bertahap sesuai kebutuhan. Parameter pinalti dinamis turut diintegrasikan ke dalam rangkaian siklus optimasi untuk membatasi ruang pencarian sehingga konfigurasi portofolio yang dihasilkan selalu terjamin memenuhi kendala kardinalitas. Keterpaduan struktural antara ekstraksi parameter klasik, inisialisasi teori permainan, dan optimasi kuantum adaptif ini dikonstruksi secara logis untuk menjamin keandalan operasional metodologi secara keseluruhan.

3.2 Akuisisi dan Pra-pemrosesan Data

Prosedur akuisisi data finansial dirancang untuk memperoleh representasi harga penutupan aset secara historis yang kemudian dikonversi menjadi variabel yang dibutuhkan untuk membentuk model Hamiltonian. Parameter penghindaran risiko diekstraksi secara otomatis menggunakan skor-Z agregat, yang mencerminkan tingkat volatilitas serta ekspektasi keuntungan pasar pada jendela observasi tertentu. Rangkaian prosedur pra-pemrosesan data beserta ekstraksi metrik statistika tersebut dideskripsikan ke dalam Algoritma 2 untuk memudahkan pembacaan kode. Algoritma ini menerima input berupa daftar simbol aset serta batasan periode waktu untuk menghasilkan matriks kovariansi, vektor imbal hasil, dan nilai penghindaran risiko. Modul pembersihan data terintegrasi di dalam prosedur algoritma untuk menghilangkan nilai kosong atau anomali yang berpotensi merusak integritas matriks kovariansi dan operator kuantum. Penggunaan formula sigmoid pada perhitungan variabel penghindaran risiko memberikan batas atas dan batas bawah yang terukur sehingga koefisien pinalti tidak mendominasi fungsi objektif utama. Konfigurasi numerik yang dihasilkan oleh algoritma awal ini langsung didistribusikan ke dalam modul konstruksi Hamiltonian untuk dipetakan menjadi arsitektur sirkuit kuantum yang setara.

3.3 Pemodelan Hamiltonian Portofolio dan Pemetaan Qubit

Transformasi masalah optimasi portofolio menjadi kerangka kerja QUBO bertujuan untuk merepresentasikan objektif minimalisasi risiko dan maksimalisasi imbal hasil finansial. Variabel keputusan biner dalam matriks QUBO dipetakan ke dalam variabel spin fisis untuk membentuk model Hamiltonian Ising agar menghasilkan vektor bias serta matriks interaksi. Prosedur transformasi QUBO menuju representasi operator Pauli



Gambar 3.1: Diagram Alir Metodologi Optimasi Portofolio Berbasis GT-VQE (Nash SBR dan Adaptive VQE dengan Penalty Annealing).

disajikan secara algoritmik melalui tahapan komputasi berurutan yang diformalkan pada Algoritma 3. Algoritma tersebut menerima input matriks statistik finansial klasik untuk kemudian mengonstruksi nilai diagonal serta elemen luar diagonal dari matriks kuadratik bobot pinalti. Transformasi variabel Ising diaplikasikan pada baris-baris algoritma untuk mendapatkan koefisien kopling magnetik dan nilai bias lokal yang akan disuntikkan ke dalam gerbang rotasi Kuantum.

3.4 Algoritma Pencarian Ekuilibrium Nash dengan *Sequential Best Response* (SBR)

Interaksi strategis antar instrumen finansial dimodelkan sebagai sebuah EPG untuk mengekstraksi informasi mengenai konfigurasi aset yang paling rasional secara ekonomi. Fungsi potensial global dalam permainan strategis ini dikonstruksi dengan fungsi objektif portofolio agar setiap titik ekuilibrium merepresentasikan kombinasi aset yang optimal. Mekanisme komputasi untuk menelusuri titik ekuilibrium strategis tersebut dideskripsikan melalui skema iteratif yang terangkum dalam Algoritma 4. Algoritma ini mengadopsi

Algorithm 2 Akuisisi dan Pra-pemrosesan Data Finansial

procedure DATAPREPROCESSING(tickers, t_{start}, t_{end}, K)
  $\mathbf{P} \leftarrow$ DownloadClosingPrice($tickers, t_{start}, t_{end}$)
  $\mathbf{P}_{clean} \leftarrow$ DropNaN($\mathbf{P}$)
  $r_{log} \leftarrow \ln(\mathbf{P}_t/\mathbf{P}_{t-1})$
  $\bar{\mu} \leftarrow$ Mean($|r_{log}|$)
  $\bar{\sigma} \leftarrow$ Mean(Variance(r_{log}))
  $Z \leftarrow \bar{\mu}/\bar{\sigma}$ ▷ Skor-Z Agregat
  $\gamma \leftarrow 1/(1 + \exp(Z))$ ▷ Risk Aversion Endogen
  $\mu_{log}, \Sigma \leftarrow$ CalculateStats(r_{log})
  return $\mu_{log}, \Sigma, \gamma$
end procedure

Algorithm 3 Konstruksi Hamiltonian Portofolio Ising

procedure BUILDHAMILTONIAN($\mu, \Sigma, \gamma, K, \lambda$)
  $Q_{ii} \leftarrow \frac{\gamma\Sigma_{ii}}{2K^2} - \frac{\mu_i}{K} + \lambda(1 - 2K)$ ▷ Diagonal QUBO
  $Q_{ij} \leftarrow \frac{\gamma\Sigma_{ij}}{2K^2} + \lambda$ ▷ Off-diagonal QUBO
  $h_i \leftarrow -\frac{1}{2}(Q_{ii} + \sum_{j\neq i} Q_{ij})$ ▷ Transformasi ke Bias Ising
  $J_{ij} \leftarrow \frac{1}{4}Q_{ij}$ ▷ Transformasi ke Interaksi Ising
  $C \leftarrow \frac{1}{4}\sum_i Q_{ii} + \frac{1}{8}\sum_{i\neq j} Q_{ij} + \lambda K^2$ ▷ Konstanta Energi
  $H \leftarrow \sum_i h_i Z_i + \sum_{i<j} J_{ij} Z_i Z_j + C \cdot I$ ▷ Konstruksi Matriks Hamiltonian
  return H
end procedure

mekanisme respons terbaik berurutan atau *Sequential Best Response* (SBR) yang bekerja dengan mengevaluasi peningkatan fungsi potensial melalui pertukaran posisi tunggal antar kandidat aset yang tersedia. Pada setiap iterasi komputasi, algoritma melakukan inspeksi terhadap semua kemungkinan kombinasi pertukaran aset untuk menemukan langkah menuju maksimalisasi utilitas sistem. Proses pertukaran ini terus dilanjutkan tanpa henti hingga algoritma mencapai status keadaan tetap (*steady-state*) di mana tidak ada lagi modifikasi tunggal yang mampu meningkatkan utilitas portofolio tersebut. Bitstring final yang diproduksi oleh prosedur klasik ini kemudian diserahkan sepenuhnya ke modul komputasi kuantum sebagai parameter awal bagi gerbang rotasi Ansatz.

3.5 Optimasi Variasional GT-VQE Adaptif

Proses penyelesaian operator Hamiltonian portofolio dieksekusi menggunakan kerangka kerja VQE. Arsitektur sirkuit Ansatz dikonfigurasi dengan memanfaatkan gerbang rotasi tunggal serta gerbang keterbelitan untuk meminimalkan kedalaman sirkuit namun tetap mempertahankan tingkat ekspresivitas model yang tinggi. Alur instruksi komputasi kuantum variasional yang mencakup kalibrasi hiperparameter hingga evaluasi nilai ekspektasi energi dispesifikasikan secara menyeluruh pada Algoritma 5. Algoritma hibrida ini menerima konfigurasi bitstring inisial dari keluaran pada algoritma 4 untuk menyiapkan parameter rotasi Ansatz awal sebelum memulai siklus optimasi stokastik berbasis SPSA. Di dalam blok iterasi utama, algoritma akan menghitung gradien fungsi energi. Apabila

Algorithm 4 Pencarian Nash Equilibrium menggunakan Sequential Best Response (SBR)

```
procedure NashSBR(μ, Σ, γ, N, K, max_iter)
    x ← InitialSelection(K)                          ▷ Inisialisasi K aset pertama
    Φ ← (1/K) Σ μ_i x_i − (γ/2K²) Σ Σ_ij x_i x_j     ▷ Fungsi Potensial
    for iter = 1 to max_iter do
        improved ← False
        S_in ← {i | x_i = 1}, S_out ← {j | x_j = 0}
        for i ∈ S_in and j ∈ S_out do
            x' ← Swap(x, i, j)
            Φ' ← CalculatePotential(x', μ, Σ, γ)
            if Φ' > Φ then
                x ← x', Φ ← Φ', improved ← True
                break                                ▷ Pembaruan Sekuensial
            end if
        end for
        if not improved then break
        end if
    end for
    return x                                         ▷ Bitstring Ekuilibrium Nash
end procedure
```

konvergensi energi tidak tercapai pada batas iterasi tertentu, modul ekspansi sirkuit akan diaktifkan untuk menambah kedalaman Ansatz beserta parameter rotasi tambahannya. Output dari algoritma kuantum ini adalah distribusi probabilitas keadaan basis untuk merepresentasikan alokasi aset final.

3.6 Simulasi Backtesting dan Metrik Evaluasi

Verifikasi empiris terhadap performa portofolio kuantum dijalankan melalui mekanisme simulasi *backtesting* untuk mereplikasi eksekusi investasi riil secara langsung pada data pasar historis. Jendela observasi data digeser secara periodik seiring berjalannya waktu investasi untuk memastikan bahwa model kuantum selalu dilatih ulang menggunakan distribusi harga saham paling mutakhir. Seluruh rangkaian prosedur validasi investasi dan perhitungan performa metrik evaluasi portofolio diformulasikan ke dalam struktur iteratif yang dijabarkan pada Algoritma 6. Algoritma ini menyatukan proses eksekusi seluruh modul metodologi sebelumnya mulai dari akuisisi data awal, pemetaan matriks Ising, hingga optimasi kuantum variasional secara berurutan. Pencatatan riwayat keputusan alokasi aset beserta fluktuasi nilai energi sirkuit dieksekusi secara otomatis pada setiap akhir periode untuk keperluan visualisasi diagnostik komputasi akhir.

Algorithm 5 Adaptive HE-VQE dengan Penalty Annealing dan Nash Warm-Start

procedure ADAPTIVEVQE($H_{obj}, H_{pen}, x_{NE}, D_{max}, \lambda_{target}$)
  $\theta \leftarrow$ InitialParameters(x_{NE}) ▷ Inisialisasi NE-VQE
  for $d = 1$ to D_{max} **do**
    $a, c \leftarrow$ CalibrateSPSA($H_{obj}, H_{pen}, \lambda_{target}$) ▷ Kalibrasi *Learning Rate*
    for $k = 1$ to N_{iter} **do**
      $\lambda_k \leftarrow \lambda_{target} \cdot (0.1 + 0.9 \cdot \frac{k}{0.7N_{iter}})$ ▷ Penalty Annealing
      $\mathcal{L}(\theta) \leftarrow \langle\psi(\theta)|H_{obj} + \lambda_k H_{pen}|\psi(\theta)\rangle$
      $g_k \leftarrow$ SPSA_Gradient($\mathcal{L}, \theta, c_k$)
      $\theta \leftarrow \theta - a_k g_k$ ▷ Update Parameter
    end for
    $\theta \leftarrow$ LayerExpansion(θ) ▷ Tambah Kedalaman (Adaptif)
    if Converged(θ) **then break**
    end if
  end for
  $probs \leftarrow |\langle x|\psi(\theta_{final})\rangle|^2$ ▷ Pengukuran Probabilitas
  $x^* \leftarrow$ Argmax($probs$) s.t. $\sum x_i = K$
  return x^*
end procedure

3.7 Optimasi Klasik SLSQP (*Sequential Least Squares Programming*)

Penyelesaian masalah portofolio Markowitz secara tradisional menggunakan algoritma pemrograman kuadrat terkecil berurutan atau *Sequential Least Squares Programming* (SLSQP) diposisikan sebagai standar komparasi utama bagi keandalan model komputasi kuantum. Algoritma optimasi non-linear ini beroperasi dengan menyelesaikan serangkaian sub-masalah pemrograman kuadratik yang diaproksimasi melalui matriks Hessian serta vektor gradien dari fungsi utilitas finansial objektif. Alur komputasi untuk optimasi bobot portofolio dilakukan secara iteratif dan berkelanjutan dengan memanfaatkan matriks Hessian hampiran yang disajikan pada Algoritma 7. Instruksi algoritma ini dimulai dengan penetapan bobot alokasi yang setara pada seluruh kandidat aset sebagai titik inisialisasi yang netral bagi pergerakan ruang solusi berkelanjutan. Pada setiap siklus komputasi, algoritma menghitung arah pencarian ideal berdasarkan gradien analitis dan melakukan pembaruan matriks menggunakan teknik kalibrasi kuasi-Newton secara sistematis dan efisien. Eksekusi program terus diulang hingga parameter bobot portofolio mencapai kriteria toleransi konvergensi yang ditetapkan guna menghasilkan vektor investasi kontinu terbaik sebagai perbandingan evaluasi.

3.8 Validasi Solusi Eksak *Brute Force*

Metode pencarian solusi eksak secara menyeluruh (*brute force*) diterapkan sebagai instrumen verifikasi absolut untuk menentukan kebenaran mutlak dari titik minimum global pada model energi Ising. Prosedur penelusuran solusi mutlak melalui perhitungan fungsi biaya Hamiltonian pada seluruh elemen ruang keadaan biner diformalkan ke

Algorithm 6 Prosedur Backtesting Global Portofolio GT-VQE

```
procedure BacktestLoop(Data, K, window = 126)
    Timeline ← GenerateInvestmentDates(Data)
    for t ∈ Timeline do
        Data_train ← Data[t − window : t]
        μ, Σ, γ ← DataPreprocessing(Data_train, K)                ▷ Algoritma 2
        H_obj, H_pen ← BuildHamiltonian(μ, Σ, γ, K, λ)            ▷ Algoritma 3
        x_NE ← NashSBR(μ, Σ, γ, K)                                ▷ Algoritma 4
        x*_t, θ_t ← AdaptiveVQE(H_obj, H_pen, x_NE, D_max)        ▷ Optimasi VQE Adaptif
        R_t ← ExecuteRebalance(x*_t, Data[t : t + 1])            ▷ Eksekusi Rebalancing
        LogPortfolioHistory(x*_t, R_t, θ_t)
    end for
    Metrics ← ComputePerformance(History)                         ▷ Kalkulasi Metrik Performa
    return EquityCurve, Metrics
end procedure
```

Algorithm 7 Optimasi Klasik Markowitz menggunakan SLSQP

```
procedure SLSQPOptimizer(μ, Σ, γ)
    w_0 ← InitialWeights(1/N)                                     ▷ Inisialisasi Seragam
    B_0 ← I                                                       ▷ Inisialisasi Hessian Identitas
    for k = 0, 1, 2, ... until converged do
        g_k ← ∇f(w_k) = γΣw_k − μ                                 ▷ Hitung Gradien
        Solve Sub-QP: min_d(g_k^T d + (1/2) d^T B_k d) s.t. 1^T d = 0, bounds
        α_k ← LineSearch(w_k, d)                                  ▷ Cari ukuran langkah
        w_{k+1} ← w_k + α_k d                                     ▷ Update Bobot
        s_k ← w_{k+1} − w_k, y_k ← g_{k+1} − g_k
        B_{k+1} ← UpdateBFGS(B_k, s_k, y_k)                       ▷ Update Aproksimasi Hessian
    end for
    return w_final
end procedure
```

dalam kerangka Algoritma 8. Algoritma komputasi eksak ini menerima parameter bias linier dan parameter kopling kuadratik untuk mengeksekusi perhitungan utilitas energi secara langsung pada setiap iterasi kombinasi aset. Mekanisme pencatatan nilai minimum dijalankan dengan memperbarui variabel energi terbaik setiap kali algoritma menemukan konfigurasi aset dengan ukuran energi yang lebih rendah secara numerik. Hasil keluaran berupa bitstring optimal dan energi terendah mutlak dari prosedur eksak ini dicatat secara permanen untuk memvalidasi tingkat akurasi konvergensi model hibrida kuantum.

Algorithm 8 Pencarian Solusi Eksak secara Menyeluruh

```
procedure BruteForceValidation(h, J, C, N, K)
    best_energy ← ∞
    best_x ← None
    all_combinations ← GenerateCombinations(N, K)
    for each x ∈ all_combinations do
        E ← HitungEnergiIsing(x, h, J, C)          ▷ Kalkulasi Energi Hamilton
        if E < best_energy then
            best_energy ← E
            best_x ← x
        end if
    end for
    return best_x, best_energy
end procedure
```

Bab 4

Hasil dan Pembahasan

Investasi pada instrumen ekuitas merupakan instrumen krusial bagi investor dalam melakukan lindung nilai terhadap inflasi serta mengupayakan pertumbuhan modal jangka panjang melalui alokasi sumber daya finansial pada sektor-sektor yang produktif dan kompetitif. Akan tetapi, proses penentuan kombinasi aset yang paling efisien sering kali terbentur pada kompleksitas komputasi klasik yang harus memproses miliaran kemungkinan konfigurasi portofolio secara sekuensial, yang mengakibatkan timbulnya risiko keterlambatan pengambilan keputusan strategis. Oleh Karena itu, penggunaan algoritma kuantum hadir sebagai solusi dengan memanfaatkan ruang Hilbert sebagai eksplorasi seluruh kandidat solusi secara paralel, sehingga unggul eksponensial walaupun dengan jumlah qubitnya sangat besar karena dapat menemukan titik optimal global pada lanskap energi Hamiltonian yang sangat luas. Penelitian ini mengusulkan sebuah metodologi hibrida dengan integrasi VQE dan teori permainan agar dapat menghasilkan keputusan investasi sekaligus meningkatkan akurasi serta efisiensi daya komputasi mengingat kompleksitas optimasi portofolio yang begitu luas.

Penlitian ini menggunakan landasan teoretis yang terbagi ke dalam tiga fase utama, yakni formulasi ekonomi-strategis, pemetaan fisis, dan optimasi kuantum. Fase pertama melibatkan transformasi model Markowitz tradisional ke dalam struktur EPG agar dapat menemukan ekuilibrium Nash sebagai solusi awal klasik. Fase kedua mencakup pemetaan fungsi potensial tersebut ke dalam formulasi QUBO yang kemudian ditransformasikan menjadi operator Hamiltonian Ising. Terakhir, fase ketiga menerapkan algoritma VQE berbasis HEA dengan optimasi stokastik SPSA untuk menavigasi lanskap energi untuk mencapai keadaan dasar yang merepresentasikan portofolio optimal.

Metodologi penelitian ini mengadopsi pendekatan hibrida kuantum-klasik yang mensimulasikan interaksi multi-agen di pasar saham menggunakan pustaka PennyLane untuk simulasi perangkat keras secara virtual. Analisis dilakukan dengan membandingkan secara empiris antara skenario dengan integrasi teori permaian (GT) dan skenario tanpa interaksi strategis dari teori permainan ($NoGT$) untuk melihat dampak stabilitas pada keputusan investasi. Proses evaluasi mencakup pemantauan metrik internal sirkuit, seperti dinamika nilai ekspektasi energi dan evolusi entropi Von Neumann selama proses iterasi berlangsung. Melalui pendekatan ini, penelitian mampu mengidentifikasi korelasi antara fenomena fisis dalam sirkuit kuantum dengan perilaku ekonomi yang muncul pada hasil optimasi portofolio.

Data yang dianalisis dalam penelitian ini bersumber dari runtun waktu harga saham harian untuk sistem dengan $N = 2$ aset (BBCA dan ADRO) serta sistem dengan $N = 4$

aset (BBCA, ADRO, SMGR, dan TLKM) dalam rentang waktu periode 2021 hingga 2023. Akuisisi data menghasilkan kumpulan parameter fisis yang terdokumentasi pada Lampiran F dan Lampiran H, serta riwayat konvergensi energi yang disajikan melalui perbandingan numerik pada bagian lampiran tersebut. Metrik performa finansial seperti imbal hasil kumulatif (*Cumulative Return*) dan rasio Sharpe (*Sharpe Ratio*) divisualisasikan melalui kurva performa yang disajikan langsung pada Subbab 4.8. Seluruh dataset tersebut menjadi basis bukti empiris yang krusial untuk memvalidasi ketangguhan algoritma dalam merespons volatilitas pasar melalui mekanisme penyeimbangan kembali pada jendela waktu yang ekstrem sebagaimana ditampilkan pada Lampiran G dan Lampiran I.

Berdasarkan seluruh hasil simulasi dan analisis teknis yang telah dilakukan, bab ini akan menjabarkan pembahasan ke dalam sepuluh subbab. Pembahasan dimulai dari proses akuisisi data dan prapemrosesan runtun waktu saham di dalam Subbab 4.1, yang dilanjutkan dengan formulasi EPG dan pencarian ekuilibrium Nash di dalam Subbab 4.2. Tahapan berikutnya mencakup pemetaan Hamiltonian melalui formulasi QUBO dan inisialisasi kuantum dengan strategi inisialisasi awal di dalam Subbab 4.3–4.4. Proses kalibrasi parameter melalui algoritma pencari laju pembelajaran (*Learning Rate*) serta analisis konvergensi energi optimasi SPSA akan dibahas secara detail di dalam Subbab 4.5–4.6. Dinamika internal sirkuit dievaluasi melalui evolusi entropi Von Neumann di dalam Subbab 4.7, yang diikuti oleh pembahasan pengaruh kedalaman sirkuit di dalam Subbab 4.8. Terakhir, validasi solusi menggunakan algoritma pembanding klasik, analisis performa portofolio akhir, hingga dinamika penyeimbangan kembali pada jendela waktu ekstrem akan dibedah secara mendalam di dalam Subbab 4.9–4.10.

4.1 Akuisisi dan Prapemrosesan Data Runtun Waktu Saham

Proses akuisisi data difokuskan pada saham-saham unggulan yang termasuk dalam indeks IHSG dan disusun ke dalam dua konfigurasi sistem. Konfigurasi pertama terdiri atas dua aset ($N = 2$), yaitu BBCA dan TLKM, sedangkan konfigurasi kedua terdiri atas empat aset ($N = 4$), yaitu BBCA, ADRO, SMGR, dan TLKM. Pemilihan aset dilakukan berdasarkan prinsip diversifikasi lintas sektor perusahaan yang mencakup sektor perbankan, energi, industri dasar, dan telekomunikasi. Diversifikasi ini bertujuan untuk menghasilkan portofolio dengan karakteristik pertumbuhan dan tingkat risiko yang beragam sehingga dapat mengurangi ketergantungan terhadap kinerja satu aset tertentu. Data yang digunakan berupa harga penutupan harian pada periode Januari 2021 hingga Desember 2023. Rentang waktu tersebut dipilih untuk merepresentasikan dinamika pasar modal Indonesia pada fase pemulihan pasca-pandemi. Seluruh data harga historis yang digunakan sebagai masukan utama algoritma optimasi disajikan secara lengkap pada Lampiran E.

Secara komputasional, data historis diperoleh melalui antarmuka pemrograman aplikasi atau *Application Programming Interface* (API) Yahoo Finance menggunakan pustaka `yfinance` pada bahasa pemrograman Python. Pengunduhan data dilakukan secara otomatis untuk memperoleh harga penutupan harian setiap saham serta indeks pembanding secara konsisten. Setelah proses ekstraksi selesai, data dibersihkan dari nilai yang hilang (*missing values*) dan diurutkan berdasarkan indeks waktu guna menjaga konsistensi de-

ret waktu (*time series*) sebelum digunakan pada tahap analisis berikutnya. Implementasi proses pengambilan dan pembersihan data tersebut disajikan pada Daftar Kode 4.1.

```
import yfinance as yf
import pandas as pd

def download_market_data(tickers, benchmark_ticker, start_date, end_date, start_bt_date):
    """
    Mengunduh data saham dan benchmark, melakukan pembersihan dasar,
    dan mengekspor data harga harian.
    """
    print("Mengunduh data saham...")
    data = yf.download(tickers, start=start_date, end=end_date, progress=False)['Close']
    data = data.dropna()
    data_clean = data.sort_index()

    print(f"Mengunduh data indeks pembanding ({benchmark_ticker})...")
    # Mengunduh benchmark mulai dari tanggal backtest agar rets bisa dihitung dengan benar
    benchmark_data = yf.download(benchmark_ticker, start=start_bt_date, end=end_date, progress
    =False)['Close']
    benchmark_data = benchmark_data.dropna()
    benchmark_rets = benchmark_data.pct_change().dropna()

    print(f"Data Berhasil Diunduh. Total hari observasi: {len(data_clean)}")

    return data_clean, benchmark_data, benchmark_rets
```

Kode 4.1: Skrip ekstraksi data harga historis saham menggunakan API `yfinance`

4.1.1 Imbal Hasil dan Kovariansi

Tahap prapemrosesan diawali dengan mentransformasikan data harga saham menjadi data imbal hasil sederhana dan imbal hasil logaritmik. Imbal hasil sederhana harian untuk aset ke-i pada waktu t didefinisikan sebagai rasio perubahan harga terhadap harga pada periode sebelumnya, sedangkan imbal hasil logaritmik diperoleh melalui logaritma natural dari rasio harga berturut-turut. Berdasarkan kedua bentuk imbal hasil tersebut, selanjutnya dihitung nilai ekspektasi imbal hasil (μ_i), varians (σ_i^2), dan kovariansi antar aset sebagai parameter utama dalam model optimasi portofolio. Nilai ekspektasi imbal hasil diperoleh melalui rata-rata data historis pada setiap jendela pelatihan yang terdiri atas $n = 126$ hari perdagangan. Sementara itu, ukuran risiko dihitung menggunakan varians karena menunjukkan tingkat penyebaran imbal hasil terhadap nilai rata-ratanya. Semakin besar nilai varians, semakin tinggi tingkat ketidakpastian pergerakan harga aset tersebut. Selain risiko individual, hubungan antar aset juga dievaluasi melalui kovariansi. Kovariansi bernilai positif menunjukkan kecenderungan dua aset bergerak searah, sedangkan kovariansi bernilai negatif menunjukkan kecenderungan pergerakan yang berlawanan.

Dalam tugas akhir ini, imbal hasil sederhana digunakan untuk menghitung ekspektasi imbal hasil portofolio karena memiliki sifat linear terhadap bobot aset. Sebaliknya, penggunaan imbal hasil logaritmik untuk menghitung ekspektasi imbal hasil portofolio dapat menghasilkan bias akibat sifat nonlinier fungsi logaritma. Bias tersebut dapat dianalisis melalui aproksimasi deret Taylor orde kedua sehingga hubungan antara imbal hasil logaritmik portofolio yang sebenarnya (r_{true}) dan aproksimasi logaritmik berbobot ($\hat{r}_p$) dapat dievaluasi secara kuantitatif. Proses derivasi diawali dengan mendefinisikan dua bentuk representasi imbal hasil logaritmik portofolio. Bentuk pertama adalah aproksimasi berbobot yang diperoleh dari penjumlahan imbal hasil logaritmik masing-masing aset, sedangkan bentuk kedua merupakan imbal hasil logaritmik portofolio yang sebenarnya

berdasarkan total imbal hasil sederhana portofolio. Dengan menggunakan aproksimasi deret Taylor orde kedua,

$$\ln(1+x) \approx x - \frac{x^2}{2}, \tag{4.1}$$

yang mana kedua bentuk tersebut dapat diekspansikan menjadi

$$\hat{r}_p = \sum_{i=1}^{n} w_i \ln(1+R_i) \approx \sum_{i=1}^{n} w_i R_i - \frac{1}{2}\sum_{i=1}^{n} w_i R_i^2, \tag{4.2}$$

$$r_{true} = \ln(1+R_p) \approx \left(\sum_{i=1}^{n} w_i R_i\right) - \frac{1}{2}\left(\sum_{i=1}^{n} w_i R_i\right)^2. \tag{4.3}$$

Selisih antara kedua ekspansi tersebut merepresentasikan galat aproksimasi yang muncul akibat penggunaan imbal hasil logaritmik berbobot sebagai pengganti imbal hasil logaritmik portofolio yang sebenarnya. Galat tersebut dapat dituliskan sebagai

$$\text{Error} = r_{true} - \hat{r}_p,$$

$$\text{Error} = \left(\sum_{i=1}^{n} w_i R_i - \frac{1}{2}\left(\sum_{i=1}^{n} w_i R_i\right)^2\right) - \left(\sum_{i=1}^{n} w_i R_i - \frac{1}{2}\sum_{i=1}^{n} w_i R_i^2\right),$$

$$\text{Error} = \frac{1}{2}\left[\sum_{i=1}^{n} w_i R_i^2 - \left(\sum_{i=1}^{n} w_i R_i\right)^2\right].$$

Bentuk dalam tanda kurung tersebut identik dengan varians portofolio sehingga diperoleh

$$\text{Error} = \frac{1}{2}\text{Var}(R). \tag{4.4}$$

Sehingga, hubungan antara imbal hasil logaritmik portofolio yang sebenarnya dan aproksimasi berbobotnya dapat dinyatakan sebagai

$$r_{true} \approx \hat{r}_p + \frac{1}{2}\text{Var}(R), \tag{4.5}$$

yang menunjukkan bahwa aproksimasi berbasis imbal hasil logaritmik mengandung koreksi sebesar setengah varians portofolio. Oleh karena itu, ekspektasi imbal hasil dalam model optimasi portofolio dihitung menggunakan imbal hasil sederhana agar tetap konsisten dengan sifat linear formulasi Markowitz.

Meskipun ekspektasi imbal hasil dihitung menggunakan imbal hasil sederhana, estimasi matriks kovariansi dalam penelitian ini dilakukan menggunakan imbal hasil logaritmik. Pemilihan tersebut didasarkan pada sifat imbal hasil logaritmik yang lebih stabil secara statistik serta memiliki distribusi yang cenderung lebih mendekati asumsi normalitas dibandingkan imbal hasil sederhana. Selain itu, untuk nilai imbal hasil harian yang relatif kecil, perbedaan antara kedua representasi tersebut sangat kecil sehingga tidak menghasilkan perubahan yang signifikan terhadap estimasi risiko portofolio. Berdasarkan Persamaan 4.5, hubungan antara imbal hasil sederhana dan imbal hasil logaritmik dapat dituliskan secara aproksimatif sebagai

$$r_i \approx R_i - \frac{1}{2}R_i^2. \tag{4.6}$$

Karena nilai imbal hasil harian umumnya berada pada orde 10^{-2}, maka suku kuadrat R_i^2 hanya berada pada orde 10^{-4} dan jauh lebih kecil dibandingkan suku utamanya. Dengan demikian dapat digunakan pendekatan

$$r_i \approx R_i. \tag{4.7}$$

Akibatnya, kovariansi yang dihitung menggunakan imbal hasil logaritmik akan mendekati kovariansi yang dihitung menggunakan imbal hasil sederhana. Secara eksplisit,

$$\mathrm{Cov}(r_A, r_B) \approx \mathrm{Cov}\left(R_A - \frac{R_A^2}{2}, R_B - \frac{R_B^2}{2}\right).$$

Dengan memanfaatkan sifat bilinear kovariansi diperoleh

$$\mathrm{Cov}(r_A, r_B) \approx \mathrm{Cov}(R_A, R_B) - \frac{1}{2}\mathrm{Cov}(R_A, R_B^2) - \frac{1}{2}\mathrm{Cov}(R_A^2, R_B) + \frac{1}{4}\mathrm{Cov}(R_A^2, R_B^2). \tag{4.8}$$

Suku utama pada persamaan tersebut adalah $\mathrm{Cov}(R_A, R_B)$ yang berada pada orde 10^{-4}, sedangkan suku-suku yang melibatkan pangkat dua imbal hasil berada pada orde yang lebih kecil, yaitu sekitar 10^{-6} hingga 10^{-8}. Oleh karena itu kontribusi suku-suku koreksi dapat diabaikan sehingga diperoleh pendekatan

$$\mathrm{Cov}(r_A, r_B) \approx \mathrm{Cov}(R_A, R_B). \tag{4.9}$$

Hasil ini menunjukkan bahwa penggunaan imbal hasil logaritmik untuk estimasi matriks kovariansi tetap memberikan representasi risiko yang konsisten terhadap penggunaan imbal hasil sederhana, terutama pada data saham harian dengan tingkat volatilitas yang relatif rendah.

Transformasi data harga saham menjadi imbal hasil sederhana dan imbal hasil logaritmik diimplementasikan menggunakan pustaka `NumPy` dan `Pandas`. Fungsi yang dikembangkan menerima masukan berupa deret harga historis dan menghasilkan deret imbal hasil sesuai dengan formulasi matematis yang telah dijelaskan sebelumnya. Selain itu, dilakukan pula perhitungan nilai ekspektasi imbal hasil sederhana sebagai parameter pertumbuhan aset pada setiap jendela pelatihan. Untuk keperluan analisis risiko, imbal hasil logaritmik dihitung menggunakan transformasi logaritma natural terhadap rasio harga berturut-turut. Sementara itu, estimasi ekspektasi imbal hasil logaritmik diperoleh melalui pendekatan koreksi volatilitas (*volatility drag*) yang dinyatakan sebagai selisih antara rata-rata imbal hasil sederhana dan setengah variansnya. Pendekatan ini digunakan untuk menjaga konsistensi antara estimasi pertumbuhan geometrik dan karakteristik fluktuasi harga yang diamati pada data historis. Implementasi lengkap dari proses transformasi imbal hasil dan perhitungan parameter statistik tersebut ditunjukkan pada Daftar Kode 4.2.

```
import numpy as np
import pandas as pd

def calculate_simple_return(prices):
    """
    Menghitung simple return secara manual berdasarkan rumus:
    R_t = (P_t - P_{t-1}) / P_{t-1}
    """
    prev_prices = prices.shift(1)
    simple_returns = (prices - prev_prices) / prev_prices
    return simple_returns.dropna()
```

```python
def calculate_log_return(prices):
    """
    Menghitung log return secara manual berdasarkan rumus:
    r_t = ln(P_t / P_{t-1})
    """
    prev_prices = prices.shift(1)
    log_returns = np.log(prices / prev_prices)
    return log_returns.dropna()

def calculate_expected_simple_return(simple_rets):
    """
    Menghitung expected simple return: mu_R = sum(R_t) / T
    """
    return simple_rets.mean()

def calculate_expected_log_return_with_drag(mu_simple, var_simple):
    """
    Menghitung expected log return dengan volatility drag:
    mu_r = mu_R - 0.5 * sigma_R^2
    """
    return mu_simple - 0.5 * var_simple
```

Kode 4.2: Fungsi prapemrosesan imbal hasil dan ekspektasi pertumbuhan portofolio

4.1.2 Implementasi Numerik pada Sampel Data

Sebagai ilustrasi penerapan formulasi yang telah dijelaskan sebelumnya, dilakukan perhitungan numerik menggunakan data historis saham BBCA pada periode observasi pertama. Berdasarkan Tabel E.1, harga penutupan meningkat dari 4515,50 menjadi 4869,32 sehingga diperoleh imbal hasil sederhana dan imbal hasil logaritmik harian sebagai berikut

$$R_{BBCA,1} = \frac{4869,32 - 4515,50}{4515,50} \approx 0,0783,$$
$$r_{BBCA,1} = \ln(1 + 0,0783) \approx 0,0754.$$

Nilai varians dan kovariansi selanjutnya dihitung menggunakan data imbal hasil logaritmik pada jendela pelatihan yang terdiri atas 126 hari perdagangan. Untuk dua aset i dan j, kovariansi sampel didefinisikan sebagai

$$\begin{aligned} \sigma_{ij} &= \frac{1}{n-1} \sum_{t=1}^{n} (r_{i,t} - \bar{r}_i)(r_{j,t} - \bar{r}_j) \\ &\approx \frac{1}{n-1} \sum_{t=1}^{n} (R_{i,t} - \bar{R}_i)(R_{j,t} - \bar{R}_j), \end{aligned} \tag{4.10}$$

dengan pendekatan pada baris kedua mengikuti hasil analisis pada Subsection 4.1.1 yang menunjukkan bahwa kovariansi imbal hasil logaritmik dan imbal hasil sederhana bernilai hampir identik untuk data imbal hasil harian.

Untuk konfigurasi sistem dengan $N = 2$ aset, matriks kovariansi pada jendela pelatihan pertama diperoleh dalam bentuk

$$\Sigma_{N2} = \begin{pmatrix} \mathrm{Var}(r_1) & \mathrm{Cov}(r_1, r_2) \\ \mathrm{Cov}(r_2, r_1) & \mathrm{Var}(r_2) \end{pmatrix}, \tag{4.11}$$

yang secara numerik menghasilkan

$$\Sigma_{N2} = \begin{pmatrix} 0,0915 & 0,0202 \\ 0,0202 & 0,0367 \end{pmatrix}. \tag{4.12}$$

Sementara itu, untuk konfigurasi sistem dengan $N = 4$ aset, matriks kovariansi pada jendela pelatihan pertama diperoleh sebagai

$$\Sigma_{N4} = \begin{pmatrix} 0,0171 & 0,0012 & 0,0009 & 0,0022 \\ 0,0012 & 0,0198 & 0,0015 & 0,0018 \\ 0,0009 & 0,0015 & 0,0241 & 0,0011 \\ 0,0022 & 0,0018 & 0,0011 & 0,0225 \end{pmatrix}. \tag{4.13}$$

Matriks kovariansi tersebut selanjutnya digunakan sebagai masukan utama dalam pembentukan fungsi objektif Markowitz dan konstruksi Hamiltonian portofolio pada tahap optimasi kuantum.

4.1.3 Parameter penghindaran risiko Endogen

Dalam model optimasi portofolio Markowitz, parameter penghindaran risiko (γ) umumnya diperlakukan sebagai konstanta eksogen yang nilainya ditentukan sebelum proses optimasi dilakukan. Berbeda dengan pendekatan tersebut, penelitian ini menggunakan parameter penghindaran risiko yang dihitung secara endogen berdasarkan karakteristik data pasar pada setiap jendela pelatihan. Dengan demikian, tingkat penalti risiko dapat berubah secara adaptif mengikuti kondisi pasar yang diamati. Nilai γ ditentukan melalui fungsi sigmoid yang dibangun dari perbandingan antara rata-rata ekspektasi imbal hasil logaritmik ($\bar{\mu}_l$) dan rata-rata volatilitas logaritmik ($\bar{\sigma}_l$), yaitu

$$\gamma = \frac{1}{1 + \exp\left(\bar{\mu}_l / \bar{\sigma}_l\right)}. \tag{4.14}$$

Rasio $\bar{\mu}_l/\bar{\sigma}_l$ berperan sebagai indikator keseimbangan antara potensi keuntungan dan risiko pasar. Ketika ekspektasi imbal hasil relatif lebih tinggi dibandingkan volatilitas, nilai rasio akan meningkat sehingga menghasilkan nilai γ yang lebih kecil. Sebaliknya, ketika volatilitas mendominasi, nilai γ akan meningkat dan memberikan penalti risiko yang lebih besar pada fungsi objektif optimasi. Sebagai ilustrasi, pada jendela pelatihan pertama diperoleh parameter logaritmik BBCA sebesar $\mu_{l,1} = 0,3164$ dan $\sigma_{l,1} = 0,3024$, serta parameter ADRO sebesar $\mu_{l,2} = 0,1756$ dan $\sigma_{l,2} = 0,1916$. Dari data tersebut diperoleh rata-rata lintas aset

$$\bar{\mu}_l = \frac{0,3164 + 0,1756}{2} = 0,2460, \tag{4.15}$$

$$\bar{\sigma}_l = \frac{0,3024 + 0,1916}{2} = 0,2470. \tag{4.16}$$

Rasio keuntungan terhadap risiko kemudian dihitung sebagai

$$Z = \frac{\bar{\mu}_l}{\bar{\sigma}_l} = \frac{0,2460}{0,2470} \approx 0,9960. \tag{4.17}$$

Dengan mensubstitusikan nilai tersebut ke dalam fungsi sigmoid diperoleh

$$\gamma = \frac{1}{1 + e^{0,9960}} = 0,2697. \tag{4.18}$$

Nilai γ yang relatif rendah menunjukkan bahwa kondisi pasar pada periode tersebut memiliki rasio keuntungan terhadap risiko yang cukup baik. Akibatnya, penalti risiko yang

diberikan pada fungsi objektif menjadi lebih kecil sehingga algoritma optimasi cenderung memberikan toleransi yang lebih besar terhadap pengambilan risiko untuk memperoleh potensi imbal hasil yang lebih tinggi. Riwayat perubahan parameter γ untuk seluruh jendela pengamatan disajikan pada Tabel F.1 dan Tabel H.1.

Perhitungan parameter penghindaran risiko endogen diimplementasikan menggunakan pustaka `NumPy`. Fungsi yang dikembangkan menerima masukan berupa vektor ekspektasi imbal hasil dan volatilitas dari seluruh aset pada suatu jendela pelatihan, kemudian menghitung rata-rata lintas aset untuk memperoleh rasio keuntungan terhadap risiko. Nilai rasio tersebut selanjutnya ditransformasikan menggunakan fungsi sigmoid sehingga menghasilkan parameter γ yang berada pada rentang 0 hingga 1. Untuk menjaga stabilitas numerik, fungsi juga melakukan pemeriksaan terhadap nilai yang hilang maupun kondisi volatilitas bernilai nol. Apabila salah satu kondisi tersebut terdeteksi, fungsi akan mengembalikan nilai $\gamma = 0,5$ sebagai nilai netral. Implementasi lengkap prosedur perhitungan parameter penghindaran risiko endogen ditunjukkan pada Daftar Kode 4.3.

```
import numpy as np

def compute_endogenous_lambda(mu_period, sigma_period):
    """
    Menghitung parameter risk-aversion (gamma) secara endogen berdasarkan
    Sharpe Ratio rata-rata lintas aset menggunakan variabel yang sudah
    dihitung (termasuk volatility drag).

    Z = mu_avg / sigma_avg (Sharpe Ratio agregat)
    gamma = 1 / (1 + exp(Z))
    """
    mu_avg    = np.abs(mu_period).mean()
    sigma_avg = sigma_period.mean()

    if np.isnan(mu_avg) or np.isnan(sigma_avg) or sigma_avg == 0:
        return 0.5

    Z = mu_avg / sigma_avg   # Sharpe Ratio agregat
    gamma = 1.0 / (1.0 + np.exp(Z))

    return gamma
```

Kode 4.3: Algoritma kalkulasi parameter penghindaran risiko dinamis berbasis Rasio Sharpe agregat

4.2 Formulasi EPG dan Pencarian Ekuilibrium Nash

Tahap berikutnya adalah memetakan permasalahan optimasi portofolio Markowitz ke dalam kerangka EPG. Formulasi yang digunakan mengacu pada model varians-rerata biner sebagaimana telah dijabarkan pada Bab 2.3. Dalam representasi ini, setiap aset diperlakukan sebagai agen strategis yang memilih status biner ($x_i \in 0, 1$), dengan nilai ($x_i = 1$) menyatakan bahwa aset dipilih ke dalam portofolio dan ($x_i = 0$) menyatakan sebaliknya.Untuk memenuhi kendala anggaran, jumlah aset yang dipilih ditetapkan sebanyak ($k = N/2$). Dengan demikian, bobot portofolio dapat dituliskan sebagai ($\omega_i = x_i/k$), sehingga berlaku

$$\sum_i \frac{x_i}{k} = 1. \tag{4.19}$$

Konstruksi tersebut dilakukan agar total alokasi modal selalu memenuhi kendala investasi yang digunakan dalam formulasi optimasi portofolio dasar dari model Markowitz.

4.2.1 Formulasi EPG

Berdasarkan kerangka teorema Monderer dan Shapley yang mendasari teori permainan kooperatif, suatu permainan finansial dikategorikan secara formal sebagai EPG apabila terdapat sebuah fungsi potensial skalar $\Phi(\vec{x})$ sedemikian rupa sehingga ekuivalensi utilitas terpenuhi. Ekuivalensi tersebut mensyaratkan bahwa selisih utilitas individu Δu_i akibat dinamika perubahan strategi satu agen harus secara eksak sama nilainya dengan selisih evaluasi nilai fungsi potensial sistem global tersebut. Fungsi potensial skalar ini secara matematis diperoleh dari penjabaran ekspansi fungsi biaya portofolio kuadratik yang memetakan hubungan kovariansi multivariat antar instrumen. Untuk mengidentifikasi kontribusi marginal spesifik dari agen ke-i, proses derivasi melakukan dekomposisi jumlahan polinomial tersebut dengan mengisolasi suku-suku yang mengandung variabel x_i serta mendayagunakan sifat komutatif variabel biner $x_i^2 = x_i$. Derivasi fisis fungsi potensial tersebut pada akhirnya menghasilkan kuantitas selisih potensial $\Delta\Phi$ yang bertindak langsung sebagai insentif marginal ketika agen i memutuskan untuk mengubah status transisi dari kondisi 0 menjadi 1. Implementasi prosedur perhitungan tersebut menggunakan pustaka `Pandas` untuk melakukan pengelompokan data dan evaluasi probabilitas setiap keadaan mikro (*microstate*). Detail implementasinya ditunjukkan pada Daftar Kode 4.4.

```
import numpy as np

def compute_strategic_returns(log_rets, binary_st, tickers):
    """
    [Tugas 2: Perhitungan Imbal Hasil Strategis]
    Menghitung imbal hasil strategis (mu_tilde_i) sebagai jumlahan terbobot
    dari ekspektasi bersyarat pada 16 microstates sistem 4 aset.
    """
    n_assets = len(tickers)
    total_days = len(log_rets)
    mu_tilde = np.zeros(n_assets)

    # Kelompokkan return berdasarkan binary states (microstates) dari semua aset
    grouped = log_rets.groupby([binary_st[t] for t in tickers])

    for state, group in grouped:
        P_s = len(group) / total_days           # Probabilitas microstate P(s)
        R_bar_s = group[tickers].mean().values  # Ekspektasi return bersyarat R_bar_i(s)
        mu_tilde += P_s * R_bar_s               # Jumlahan terbobot

    return mu_tilde
```

Kode 4.4: Algoritma kalkulasi ekspektasi imbal hasil strategis berbasis observasi probabilitas keadaan mikro

4.2.2 Pencarian Ekuilibrium Nash

Setelah fungsi potensial sistem diperoleh, langkah berikutnya adalah menentukan konfigurasi strategi yang memenuhi kondisi Ekuilibrium Nash. Pada kerangka EPG, titik Ekuilibrium Nash ekuivalen dengan maksimum lokal dari fungsi potensial $\Phi(\mathbf{x})$. Oleh karena itu, pencarian kesetimbangan dapat dilakukan dengan mencari konfigurasi biner yang tidak lagi memberikan peningkatan nilai fungsi potensial ketika salah satu agen mengubah strateginya. Pada tugas akhir ini, pencarian Ekuilibrium Nash dilakukan menggunakan metode SBR. Algoritma ini bekerja melalui mekanisme pertukaran (*swap move*), yaitu menukar satu aset yang sedang berada di dalam portofolio dengan satu aset yang berada di luar portofolio. Pada setiap iterasi, seluruh kandidat pertukaran dievaluasi dan dipilih

perubahan yang memberikan kenaikan fungsi potensial terbesar. Proses iteratif tersebut dihentikan ketika tidak ditemukan lagi pertukaran yang mampu meningkatkan nilai fungsi potensial. Sebagai ilustrasi pada sistem dengan $N = 2$, interaksi strategis antar aset dapat direpresentasikan melalui matriks imbal hasil pada Tabel 4.1. Setiap elemen matriks menunjukkan utilitas yang diperoleh kedua agen untuk kombinasi status biner tertentu. Berdasarkan evaluasi seluruh konfigurasi yang memungkinkan, keadaan $|10\rangle$ menghasilkan utilitas sistem tertinggi sehingga diidentifikasi sebagai titik Ekuilibrium Nash pada periode observasi tersebut.

Tabel 4.1: Matriks Imbal Hasil Strategis Sistem N=2 (Januari 2021).

		Aset ADRO (x_2)	
		Status 0	Status 1
Aset BBCA (x_1)	Status 0	(0, 0)	(0, 0,1890)
	Status 1	(0,3498, 0)	(0,3444, 0,1836)

Pada sistem dengan $N = 4$, jumlah konfigurasi yang memenuhi kendala $k = 2$ meningkat secara signifikan sehingga evaluasi seluruh kemungkinan strategi menjadi lebih kompleks. Oleh karena itu, pencarian kesetimbangan harus dilakukan menggunakan prosedur SBR yang menelusuri ruang solusi melalui serangkaian pertukaran aset sampai pada titik konfigurasi yang tidak mengalami peningkatan fungsi potensial lagi. Riwayat iterasi dan konfigurasi yang dihasilkan pada setiap periode pengamatan disajikan pada Lampiran H. Konfigurasi ekuilibrium Nash yang diperoleh memiliki peran penting dalam penelitian ini karena digunakan sebagai titik inisialisasi bagi algoritma VQE. Pemanfaatan solusi hasil teori permainan sebagai tebakan awal memungkinkan proses optimasi kuantum dimulai dari wilayah solusi yang telah memiliki kualitas utilitas tinggi sehingga mempercepat konvergensi menuju keadaan energi minimum.

Implementasi pencarian ekuilibrium Nash dilakukan menggunakan Python melalui algoritma SBR. Fungsi yang dikembangkan menerima masukan berupa vektor ekspektasi imbal hasil, matriks kovariansi, dan parameter penghindaran risiko, kemudian mengevaluasi fungsi potensial untuk setiap kandidat pertukaran aset yang memenuhi kendala portofolio. Selama masih terdapat pertukaran yang meningkatkan fungsi potensial, konfigurasi portofolio akan diperbarui secara iteratif hingga tercapai kondisi konvergen. Selain menghasilkan konfigurasi kesetimbangan akhir, algoritma juga merekam seluruh riwayat iterasi sehingga proses konvergensi dapat dianalisis lebih lanjut. Implementasi lengkap prosedur tersebut ditunjukkan pada Daftar Kode 4.5.

```
import numpy as np
import pandas as pd
import os

def find_nash_sbr(mu, sigma_matrix, gamma, curr_date, N=4, K=2, max_iters=100, history_file='
    riwayat_nash_sbr.csv'):
    """
    Pencarian Nash Equilibrium menggunakan Sequential Best Response (SBR)
    dalam Exact Potential Game.

    Potensial (Utilitas Portofolio):
    Phi = (1/K) * mu^T * x - (gamma / (2 * K^2)) * x^T * sigma * x

    Dalam Potential Game, Nash Equilibrium adalah local maximum dari fungsi potensial.
    Karena ada batasan sum(x) = K, kita menggunakan swap moves untuk mencari NE.
    """
    # Inisialisasi: pilih K aset pertama
    current_selection = set(range(K))
```

```python
history = []

def calculate_potential(selection):
    x = np.zeros(N)
    for idx in selection: x[idx] = 1

    # 1. Komponen Return: (1/K) * sum(mu_i * x_i)
    ret = np.sum(mu[list(selection)]) / K

    # 2. Komponen Risiko: (gamma / (2 * K^2)) * sum(sigma_ij * x_i * x_j)
    # x^T * Sigma * x untuk subset selection
    if len(selection) > 0:
        sub_sigma = sigma_matrix[np.ix_(list(selection), list(selection))]
        risk = np.sum(sub_sigma) * (gamma / (2.0 * K**2))
    else:
        risk = 0.0

    return ret - risk

current_phi = calculate_potential(current_selection)

history.append({
    'Date': curr_date.date(),
    'Iteration': 0,
    'Bitstring': "".join(str(int(i in current_selection)) for i in range(N)),
    'Utility': current_phi,
    'Swap': 'Initial'
})

for iteration in range(1, max_iters + 1):
    improved = False
    out_portfolio = set(range(N)) - current_selection

    best_swap = None
    max_phi = current_phi

    # SBR/Local Search: Coba tukar satu aset di dalam dengan satu aset di luar
    # Mencari swap yang meningkatkan fungsi potensial paling besar
    for i in current_selection:
        for j in out_portfolio:
            new_selection = (current_selection - {i}) | {j}
            new_phi = calculate_potential(new_selection)

            if new_phi > max_phi:
                max_phi = new_phi
                best_swap = (i, j)

    if best_swap:
        i, j = best_swap
        current_selection = (current_selection - {i}) | {j}
        current_phi = max_phi
        improved = True
        history.append({
            'Date': curr_date.date(),
            'Iteration': iteration,
            'Bitstring': "".join(str(int(idx in current_selection)) for idx in range(N)),
            'Utility': current_phi,
            'Swap': f"{i}<->{j}"
        })

    if not improved:
        break

# --- EKSPOR RIWAYAT Nash SBR ---
df_history = pd.DataFrame(history)
if not os.path.exists(history_file):
    df_history.to_csv(history_file, index=False)
else:
    df_history.to_csv(history_file, mode='a', header=False, index=False)

final_x = np.zeros(N, dtype=int)
for idx in current_selection: final_x[idx] = 1
```

```
# Kembalikan bitstring dan utilitas akhir (sebagai potensial)
return "".join(str(bit) for bit in final_x), current_phi
```

Kode 4.5: Skrip implementasi SBR untuk pencarian konvergensi ekuilibrium Nash

4.3 Pemetaan Hamiltonian Ising

Transformasi permasalahan optimasi portofolio ke dalam domain komputasi kuantum dilakukan melalui formulasi QUBO. Pada tahap ini, fungsi potensial pada teori permainan ditransformasikan menjadi fungsi energi sehingga sesuai dengan prinsip minimisasi energi yang digunakan oleh algoritma VQE. Hubungan tersebut dinyatakan sebagai

$$\begin{aligned} E(\vec{x}) &= -\Phi(\vec{x}) \\ &= -\left(\frac{1}{k}\vec{\mu}'^{T}\vec{x} - \frac{\gamma}{2k^2}\vec{x}'^{T}\Sigma\vec{x}\right). \end{aligned} \tag{4.20}$$

Selanjutnya, fungsi energi dalam bentuk QUBO dipetakan ke Hamiltonian Ising menggunakan transformasi variabel biner ke variabel spin sebagaimana telah dijelaskan pada Bab 2. Proses ini menghasilkan operator Hamiltonian yang dapat direpresentasikan dalam bentuk

$$\hat{\mathcal{H}} = \sum_i h_i \hat{Z}_i + \sum_{i<j} J_{ij}\hat{Z}_i\hat{Z}_j + C, \tag{4.21}$$

dengan h_i menyatakan koefisien bias linear, J_{ij} menyatakan koefisien interaksi antar-spin, dan C merupakan konstanta pergeseran energi. Operator Hamiltonian inilah yang kemudian digunakan sebagai fungsi objektif kuantum pada proses optimasi VQE.

4.3.1 Implementasi Numerik Hamiltonian Ising

Setelah parameter QUBO dipetakan ke model Ising, Hamiltonian kuantum dapat dinyatakan dalam bentuk operator Pauli-Z. Untuk sistem dengan dua aset ($N = 2$) dan kendala kardinalitas ($k = 1$), Hamiltonian memiliki bentuk

$$\hat{\mathcal{H}}_{N2} = h_1\hat{Z}_1 + h_2\hat{Z}_2 + J_{12}\hat{Z}_1\hat{Z}_2 + C. \tag{4.22}$$

Pada sistem dengan empat aset ((N=4)) dan kendala kardinalitas (k=2), jumlah interaksi pasangan meningkat sehingga Hamiltonian berkembang menjadi

$$\begin{aligned} \hat{\mathcal{H}}_{N4} =, h_1\hat{Z}_1 + h_2\hat{Z}_2 + h_3\hat{Z}_3 + h_4\hat{Z}_4 \\ +J_{12}\hat{Z}_1\hat{Z}_2 + J_{13}\hat{Z}_1\hat{Z}_3 + J_{14}\hat{Z}_1\hat{Z}_4 \ + J_{23}\hat{Z}_2\hat{Z}_3 + J_{24}\hat{Z}_2\hat{Z}_4 + J_{34}\hat{Z}_3\hat{Z}_4 + C. \end{aligned} \tag{4.23}$$

Persamaan di atas menunjukkan bahwa setiap aset direpresentasikan oleh sebuah qubit, sedangkan hubungan risiko antar aset direpresentasikan melalui koefisien interaksi (J_{ij}). Dengan demikian, struktur korelasi yang semula tersimpan dalam matriks kovariansi ditransformasikan menjadi jaringan interaksi spin pada Hamiltonian Ising. Sebagai ilustrasi numerik pada jendela pelatihan pertama untuk sistem dua aset, proses pemetaan menghasilkan Hamiltonian Ising

$$\hat{\mathcal{H}}_{N2} = 0,1735\,\hat{Z}_1 + 0,0931\,\hat{Z}_2 + 0,5014\,\hat{Z}_1\hat{Z}_2 + 0,2320. \tag{4.24}$$

Koefisien linear $h_1 = 0,1735$ dan $h_2 = 0,0931$ merepresentasikan bias energi yang terkait dengan masing-masing aset, sedangkan koefisien interaksi $J_{12} = 0,5014$ menunjukkan adanya kopling yang relatif lebih dominan dibandingkan kontribusi linear individual. Sementara itu, konstanta $C = 0,2320$ berfungsi sebagai pergeseran energi global dan tidak memengaruhi posisi keadaan dasar dari sistem. Besarnya nilai koefisien interaksi menunjukkan bahwa struktur Hamiltonian tidak hanya dipengaruhi oleh ekspektasi imbal hasil dan kovariansi aset, tetapi juga oleh komponen penalti yang digunakan untuk menegakkan kendala kardinalitas portofolio. Dengan demikian, konfigurasi yang melanggar kendala pemilihan aset akan memperoleh energi yang lebih tinggi dibandingkan konfigurasi yang memenuhi batasan investasi yang telah ditetapkan. Hamiltonian hasil pemetaan inilah yang selanjutnya digunakan sebagai fungsi objektif kuantum pada algoritma VQE. Hamiltonian inilah yang selanjutnya digunakan sebagai masukan utama pada algoritma VQE untuk mencari konfigurasi portofolio berenergi minimum.

Hamiltonian Ising diimplementasikakn ke dalam simulator kuantum menggunakan pustaka *PennyLane*. Fungsi yang dikembangkan menerima masukan berupa himpunan koefisien bias linear (h_i), koefisien interaksi (J_{ij}), jumlah aset, serta konstanta pergeseran energi C. Selanjutnya, setiap koefisien dipetakan ke operator Pauli-Z yang sesuai sehingga terbentuk Hamiltonian Ising dalam format yang dapat dievaluasi secara langsung oleh algoritma VQE. Koefisien yang bernilai sangat kecil ($< 10^{-12}$) dapat diabaikan untuk menghindari penambahan operator yang tidak memberikan kontribusi signifikan terhadap energi sistem. Implementasi lengkap prosedur perakitan Hamiltonian tersebut ditunjukkan pada Daftar Kode 4.6.

```
import pennylane as qml

def build_hamiltonian_total(h_total, J_total, n_assets, offset=0.0):
    """
    Membangun operator Hamiltonian Ising dari parameter h_i dan J_ij,
    termasuk konstanta pergeseran energi C_Ising.
    H = sum(h_i * Z_i) + sum(J_ij * Z_i * Z_j) + offset * I
    """
    coeffs = []
    obs    = []

    # Suku Bias (h_i * Z_i)
    for i in range(n_assets):
        if abs(h_total[i]) > 1e-12:
            coeffs.append(float(h_total[i]))
            obs.append(qml.PauliZ(i))

    # Suku Interaksi (J_ij * Z_i * Z_j)
    for i in range(n_assets):
        for j in range(i + 1, n_assets):
            if abs(J_total[i, j]) > 1e-12:
                coeffs.append(float(J_total[i, j]))
                obs.append(qml.PauliZ(i) @ qml.PauliZ(j))

    # Suku Konstanta (offset * Identity)
    if abs(offset) > 1e-12:
        coeffs.append(float(offset))
        obs.append(qml.Identity(0))

    if len(coeffs) == 0:
        coeffs.append(0.0)
        obs.append(qml.Identity(0))

    return qml.Hamiltonian(coeffs, obs)
```

Kode 4.6: Skrip perakitan operator Hamiltonian Ising dinamis menggunakan pustaka `PennyLane`

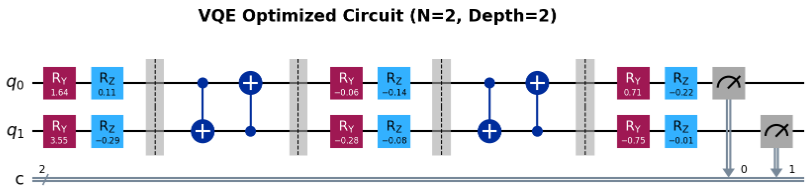
4.4 Arsitektur Ansatz dalam VQE

4.4.1 Konstruksi *Hardware-Efficient Ansatz* (HEA)

Setelah Hamiltonian Ising berhasil dibangun, langkah berikutnya adalah menyiapkan keadaan kuantum variasional yang digunakan untuk menghitung energi ekspektasi sistem. Dalam algoritma VQE, keadaan kuantum tersebut dinyatakan sebagai fungsi dari parameter bebas ($\boldsymbol{\theta}$) yang dioptimasi secara iteratif untuk mendekati keadaan dasar Hamiltonian. Ansatz yang digunakan pada tugas akhir ini adalah arsitektur HEA, yaitu rangkaian kuantum yang tersusun atas lapisan gerbang rotasi satu-qubit dan lapisan keterbelitan antar-qubit. Struktur ini dipilih karena memiliki jumlah parameter yang relatif sedikit, mudah diimplementasikan pada perangkat kuantum, dan mampu menghasilkan ruang keadaan yang cukup fleksibel untuk merepresentasikan solusi optimasi portofolio.Secara umum, keadaan variasional yang dibangkitkan oleh HEA dapat dituliskan sebagai

$$|\psi(\boldsymbol{\theta})\rangle = \left(U_{\text{ent}} U_{\text{rot}}(\boldsymbol{\theta})\right)^{L} |0\rangle^{\otimes N} , \tag{4.25}$$

dengan L menyatakan kedalaman (*depth*) sirkuit, U_{rot} menyatakan lapisan rotasi parametrik, dan U_{ent} menyatakan lapisan keterbelitan antar-qubit. Pada penelitian ini digunakan konfigurasi sebagaimana ditunjukkan pada Gambar 4.1, yaitu HEA dengan satu lapisan keterbelitan ($L = 1$). Melalui variasi parameter ($\boldsymbol{\theta}$), sirkuit menghasilkan himpunan keadaan kuantum kandidat yang berbeda-beda. Algoritma optimasi klasik kemudian memperbarui parameter tersebut secara iteratif untuk memperoleh keadaan dengan energi ekspektasi minimum terhadap Hamiltonian yang telah dibangun pada tahap sebelumnya.



Gambar 4.1: Rangkaian Sirkuit Kuantum HEA EfficientSU2 dengan Kedalaman (d) = 1.

4.4.2 Implementasi Sirkuit Kuantum Variasional

Pada arsitektur HEA yang digunakan, setiap lapisan rotasi dibangun dari kombinasi gerbang $\hat{R}_Y$ dan $\hat{R}_Z$ yang diterapkan pada seluruh qubit. Untuk lapisan ke-l, operator rotasi dapat dinyatakan sebagai

$$\hat{U}_{\text{rot}}^{(l)}(\boldsymbol{\theta}) = \bigotimes_{j=0}^{N-1} \left[\hat{R}_Y(\theta_{l,j,y}) \hat{R}_Z(\theta_{l,j,z}) \right] , \tag{4.26}$$

dengan N menyatakan jumlah qubit yang merepresentasikan aset portofolio, sedangkan $\theta_{l,j,y}$ dan $\theta_{l,j,z}$ merupakan parameter variasional yang dioptimasi selama proses VQE. Lapisan rotasi ini berfungsi untuk mengatur amplitudo dan fase keadaan kuantum sehingga ruang solusi dapat dieksplorasi secara fleksibel. Berdasarkan definisi gerbang rotasi $R_y(\theta)$

dan $R_z(\theta)$:

$$\begin{aligned} U_{1q}(\theta) &= R_z(\theta_{1z})R_y(\theta_{1y}) \\ &= \begin{pmatrix} e^{-i\theta_{1z}/2} & 0 \\ 0 & e^{i\theta_{1z}/2} \end{pmatrix} \begin{pmatrix} \cos(\theta_{1y}/2) & -\sin(\theta_{1y}/2) \\ \sin(\theta_{1y}/2) & \cos(\theta_{1y}/2) \end{pmatrix} \\ &= \begin{pmatrix} e^{-i\theta_{1z}/2}\cos(\theta_{1y}/2) & -e^{-i\theta_{1z}/2}\sin(\theta_{1y}/2) \\ e^{i\theta_{1z}/2}\sin(\theta_{1y}/2) & e^{i\theta_{1z}/2}\cos(\theta_{1y}/2) \end{pmatrix} \end{aligned} \tag{4.27}$$

Maka untuk sistem 2 qubit bentuknya menjadi

$$\begin{aligned} U_q(\theta) &= [R_z(\theta_{1z})R_y(\theta_{1y})] \otimes [R_z(\theta_{2z})R_y(\theta_{2y})] \\ &= \begin{pmatrix} e^{-i\theta_{1z}/2}\cos(\theta_{1y}/2) & -e^{-i\theta_{1z}/2}\sin(\theta_{1y}/2) \\ e^{i\theta_{1z}/2}\sin(\theta_{1y}/2) & e^{i\theta_{1z}/2}\cos(\theta_{1y}/2) \end{pmatrix} \otimes \begin{pmatrix} e^{-i\theta_{2z}/2}\cos(\theta_{2y}/2) & -e^{-i\theta_{2z}/2}\sin(\theta_{2y}/2) \\ e^{i\theta_{2z}/2}\sin(\theta_{2y}/2) & e^{i\theta_{2z}/2}\cos(\theta_{2y}/2) \end{pmatrix} \\ &= \begin{pmatrix} e^{-i(\theta_{1z}+\theta_{2z})/2}c_1c_2 & e^{-i(\theta_{1z}+\theta_{2z})/2}c_1s_2 & e^{-i(\theta_{1z}+\theta_{2z})/2}s_1c_2 & e^{-i(\theta_{1z}+\theta_{2z})/2}s_1s_2 \\ e^{-i\theta_{1z}/2}e^{i\theta_{2z}/2}c_1s_2 & e^{-i\theta_{1z}/2}e^{i\theta_{2z}/2}c_1c_2 & e^{-i\theta_{1z}/2}e^{i\theta_{2z}/2}s_1s_2 & e^{-i\theta_{1z}/2}e^{i\theta_{2z}/2}s_1c_2 \\ e^{i\theta_{1z}/2}e^{-i\theta_{2z}/2}s_1c_2 & e^{i\theta_{1z}/2}e^{-i\theta_{2z}/2}s_1s_2 & e^{i\theta_{1z}/2}e^{-i\theta_{2z}/2}c_1c_2 & e^{i\theta_{1z}/2}e^{-i\theta_{2z}/2}c_1s_2 \\ e^{i(\theta_{1z}+\theta_{2z})/2}s_1s_2 & e^{i(\theta_{1z}+\theta_{2z})/2}s_1c_2 & e^{i(\theta_{1z}+\theta_{2z})/2}c_1s_2 & e^{i(\theta_{1z}+\theta_{2z})/2}c_1c_2 \end{pmatrix} \end{aligned} \tag{4.28}$$

dengan

$$c_i = \cos\frac{\theta_{iy}}{2}, \qquad s_i = \sin\frac{\theta_{iy}}{2}.$$

Setelah seluruh qubit mengalami rotasi parametrik, diterapkan lapisan keterbelitan menggunakan gerbang CNOT. Secara umum, lapisan keterbelitan didefinisikan sebagai

$$U_{ent} = CNOT_{(N-1,0)} \prod_{q=0}^{N-2} CNOT_{(q,q+1)} \tag{4.29}$$

menggunakan konvensi

$$\prod_{q=0}^{N-2} A_{(q)} = A_{N-2}A_{N-3}\cdots A_1A_0 \tag{4.30}$$

karena operator yang paling kanan bekerja terlebih dulu pada keadaan kuantum. Untuk sistem dua aset ($N = 2$), keterikatan dibangun melalui hubungan langsung antara kedua qubit sehingga bentuknya menjadi

$$U_{ent} = CNOT_{(1,0)} \cdot CNOT_{(0,1)} \tag{4.31}$$

dengan matriks untuk masing-masing gerbang adalah:

$$CNOT_{(1,0)} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad CNOT_{(0,1)} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{4.32}$$

Sehingga matriks total keterbelitannya pada sistem 2 aset berbentuk

$$U_{ent} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix} \tag{4.33}$$

Sementara itu, pada sistem empat qubit ($N = 4$), operator keterikatan dibentuk oleh rangkaian gerbang CNOT yang disusun secara melingkar sehingga diperoleh

$$U_{\text{ent}} = CNOT_{(3,0)}\, CNOT_{(2,3)}\, CNOT_{(1,2)}\, CNOT_{(0,1)}. \tag{4.34}$$

Karena sistem terdiri atas empat qubit, maka ruang Hilbert yang dihasilkan memiliki dimensi 2^4. Artinya, setiap operator $CNOT_{(i,j)}$ direpresentasikan oleh matriks uniter berukuran 16×16, sehingga operator keseluruhan U_{ent} juga merupakan matriks uniter berukuran 16×16. Kombinasi lapisan rotasi dan lapisan keterikatan menghasilkan keadaan variasional ($|\psi(\boldsymbol{\theta})\rangle$) yang selanjutnya digunakan untuk menghitung nilai energi ekspektasi Hamiltonian Ising. Melalui pembaruan parameter ($\boldsymbol{\theta}$) secara iteratif, algoritma VQE menelusuri ruang keadaan kuantum untuk menemukan konfigurasi dengan energi minimum yang merepresentasikan solusi optimasi portofolio.

HEA juga diimplementasikan menggunakan pustaka `PennyLane`. Fungsi Ansatz menerima vektor parameter variasional yang kemudian diorganisasikan ke dalam beberapa lapisan rotasi sesuai kedalaman sirkuit yang ditentukan. Pada setiap lapisan, seluruh qubit dikenai gerbang RY dan RZ, kemudian dihubungkan melalui rangkaian gerbang CNOT dengan pola keterikatan melingkar. Struktur ini menghasilkan rangkaian kuantum yang bersifat fleksibel namun tetap efisien dari sisi jumlah parameter. Selain mampu merepresentasikan berbagai konfigurasi keadaan kuantum kandidat, arsitektur tersebut juga memungkinkan pembentukan keterbelitan antar-qubit yang diperlukan untuk memodelkan korelasi antar aset pada proses optimasi portofolio. Implementasi lengkap konstruksi Ansatz ditunjukkan pada Daftar Kode 4.7.

```
def Ansatz(params):
    """
    Membangun lapisan-lapisan sirkuit kuantum HEA berdasarkan
    parameter rotasi (theta) yang diberikan.
    """
    w = params.reshape((depth + 1, n_qubits, 2))

    # Iterasi penyusunan lapisan (layer)
    for layer in range(depth + 1):
        # 1. Lapisan rotasi lokal (RY dan RZ)
        for q in range(n_qubits):
            qml.RY(w[layer, q, 0], wires=q)
            qml.RZ(w[layer, q, 1], wires=q)

        # 2. Lapisan keterikatan / entanglement (CNOT)
        if layer < depth:
            for q in range(n_qubits - 1):
                qml.CNOT(wires=[q, q + 1])
            qml.CNOT(wires=[n_qubits - 1, 0])
```

Kode 4.7: Fungsi penyusunan arsitektur HEA pada iterasi sirkuit kuantum VQE

4.5 Optimasi *Simultaneous Perturbation Stochastic Approximation* (SPSA)

4.5.1 Dinamika Pembaruan Parameter SPSA

Setelah struktur Ansatz variasional berhasil dibangun, tahap berikutnya dalam algoritma VQE adalah menentukan himpunan parameter rotasi $\boldsymbol{\theta}$ yang meminimalkan nilai

ekspektasi energi Hamiltonian. Permasalahan optimasi ini secara matematis dapat dituliskan sebagai pencarian solusi

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \langle \psi(\boldsymbol{\theta}) | \hat{\mathcal{H}} | \psi(\boldsymbol{\theta}) \rangle , \tag{4.35}$$

dengan fungsi biaya yang didefinisikan sebagai energi ekspektasi keadaan kuantum variasional

$$E(\boldsymbol{\theta}) = \langle \psi(\boldsymbol{\theta}) | \hat{\mathcal{H}} | \psi(\boldsymbol{\theta}) \rangle . \tag{4.36}$$

Pada sistem berdimensi tinggi, jumlah parameter rotasi yang harus dioptimasi meningkat seiring bertambahnya jumlah qubit dan kedalaman sirkuit. Kondisi tersebut menyebabkan evaluasi gradien eksak menjadi semakin mahal karena memerlukan banyak pengukuran sirkuit kuantum pada setiap iterasi optimasi. Untuk mengatasi kendala tersebut, penelitian ini menggunakan algoritma Aproksimasi Stokastik dengan Perturbasi Simultan atau *Simultaneous Perturbation Stochastic Approximation* (SPSA) sebagai prosedur pembaruan parameter utama (Spall, 1998).

Prinsip dasar SPSA adalah memperkirakan seluruh komponen gradien menggunakan hanya dua evaluasi fungsi biaya pada setiap iterasi, terlepas dari banyaknya parameter yang dimiliki sirkuit. Pada iterasi ke-k, dibangkitkan sebuah vektor gangguan acak

$$\Delta_k = (\Delta_{k,1}, \Delta_{k,2}, \ldots, \Delta_{k,p})^T , \tag{4.37}$$

dengan setiap komponen $\Delta_{k,i}$ mengikuti distribusi Rademacher

$$\Delta_{k,i} = \begin{cases} +1, & P = \frac{1}{2}, \\ -1, & P = \frac{1}{2}. \end{cases} \tag{4.38}$$

Vektor gangguan ini digunakan untuk membentuk dua titik evaluasi parameter yang terletak simetris terhadap posisi parameter saat ini,

$$\boldsymbol{\theta}_k^{(+)} = \boldsymbol{\theta}_k + c_k \Delta_k, \tag{4.39}$$

dan

$$\boldsymbol{\theta}_k^{(-)} = \boldsymbol{\theta}_k - c_k \Delta_k, \tag{4.40}$$

dengan c_k menyatakan amplitudo gangguan pada iterasi ke-k. Kedua konfigurasi parameter tersebut kemudian digunakan untuk menghitung energi ekspektasi

$$E_k^{(+)} = E\left(\boldsymbol{\theta}_k^{(+)}\right), \qquad E_k^{(-)} = E\left(\boldsymbol{\theta}_k^{(-)}\right). \tag{4.41}$$

Berdasarkan dua hasil evaluasi energi tersebut, estimator gradien SPSA diperoleh melalui hubungan

$$\hat{\mathbf{g}}_k = \frac{E_k^{(+)} - E_k^{(-)}}{2c_k} \Delta_k^{-1}, \tag{4.42}$$

dengan operasi invers dilakukan secara elemen-per-elemen sehingga

$$\hat{g}_{k,i} = \frac{E_k^{(+)} - E_k^{(-)}}{2c_k \Delta_{k,i}}. \tag{4.43}$$

Karena setiap komponen $\Delta_{k,i}$ hanya bernilai ± 1, proses perhitungan estimator gradien menjadi sangat sederhana dan tidak memerlukan evaluasi parsial terhadap masing-masing

parameter secara terpisah. Karakteristik inilah yang membuat SPSA jauh lebih efisien dibandingkan metode diferensiasi numerik konvensional yang memerlukan setidaknya $2p$ evaluasi fungsi biaya untuk sistem dengan p parameter.

Setelah estimator gradien diperoleh, parameter sirkuit diperbarui menggunakan aturan iteratif

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - a_k \hat{\mathbf{g}}_k, \tag{4.44}$$

dengan a_k menyatakan laju pembelajaran yang nilainya diperkecil secara bertahap sepanjang iterasi. Dalam implementasi SPSA, koefisien pembaruan dan amplitudo gangguan umumnya mengikuti bentuk

$$a_k = \frac{a}{(A+k+1)^{\alpha}}, \qquad c_k = \frac{c}{(k+1)^{\gamma}}, \tag{4.45}$$

dengan parameter a, c, A, α, dan γ dipilih untuk mengendalikan stabilitas proses konvergensi. Melalui mekanisme tersebut, SPSA mampu mengarahkan parameter variasional menuju daerah energi yang semakin rendah meskipun informasi gradien yang digunakan hanya berupa estimasi stokastik. Efisiensi evaluasi fungsi biaya inilah yang menjadikan SPSA sangat sesuai untuk optimasi VQE pada perangkat kuantum NISQ, khususnya ketika jumlah parameter sirkuit meningkat dan pengukuran kuantum menjadi sumber biaya komputasi yang dominan.

4.5.2 Implementasi Numerik dan Konvergensi Energi Optimal

Untuk memberikan gambaran yang lebih konkret mengenai mekanisme optimasi SPSA, pada subsection ini ditunjukkan satu siklus pembaruan parameter menggunakan Hamiltonian Ising dua aset yang telah diperoleh pada pembahasan sebelumnya. Sebagai ilustrasi, digunakan Hamiltonian

$$\hat{\mathcal{H}}_{N2} = 0,1735\hat{Z}_1 + 0,0931\hat{Z}_2 + 0,5014\hat{Z}_1\hat{Z}_2 + 0,2320. \tag{4.46}$$

Hamiltonian tersebut mendefinisikan lanskap energi yang harus diminimalkan oleh algoritma VQE melalui penyesuaian parameter rotasi sirkuit kuantum. Dalam praktiknya, energi ekspektasi dihitung melalui pengukuran keadaan variasional $|\psi(\boldsymbol{\theta})\rangle$ yang dibangkitkan oleh HEA. Namun, untuk mempermudah ilustrasi matematis, energi ekspektasi pada contoh ini direpresentasikan menggunakan hubungan sederhana $\langle \hat{Z}_i \rangle = \cos(\theta_i)$ sehingga fungsi biaya dapat dituliskan sebagai

$$E(\boldsymbol{\theta}) = 0,1735\cos(\theta_1) + 0,0931\cos(\theta_2) + 0,5014\cos(\theta_1)\cos(\theta_2) + 0,2320. \tag{4.47}$$

Berdasarkan fungsi biaya tersebut, prosedur SPSA dimulai dengan menentukan parameter awal, amplitudo gangguan, serta vektor perturbasi Rademacher yang akan digunakan untuk mengestimasi gradien. Misalkan parameter awal dipilih sebagai

$$\boldsymbol{\theta}^{(0)} = \begin{pmatrix} \pi/2 \\ \pi/2 \end{pmatrix} \approx \begin{pmatrix} 1,5708 \\ 1,5708 \end{pmatrix}, \tag{4.48}$$

dengan koefisien perturbasi $c_k = 0,1$ dan vektor gangguan

$$\Delta_k = \begin{pmatrix} 1 \\ -1 \end{pmatrix}. \tag{4.49}$$

Dari konfigurasi tersebut diperoleh dua titik evaluasi parameter

$$\boldsymbol{\theta}_+ = \boldsymbol{\theta}^{(0)} + c_k \Delta_k = \begin{pmatrix} 1,6708 \\ 1,4708 \end{pmatrix}, \tag{4.50}$$

dan

$$\boldsymbol{\theta}_- = \boldsymbol{\theta}^{(0)} - c_k \Delta_k = \begin{pmatrix} 1,4708 \\ 1,6708 \end{pmatrix}. \tag{4.51}$$

Nilai energi pada kedua titik tersebut kemudian dievaluasi menggunakan fungsi biaya yang sama sehingga diperoleh energi E_+ dan E_-. Selisih kedua energi ini digunakan untuk membangun estimator gradien SPSA

$$\hat{\mathbf{g}} = \frac{E_+ - E_-}{2c_k} \Delta_k^{-1}. \tag{4.52}$$

Sebagai ilustrasi numerik, misalkan hasil evaluasi menghasilkan $E_+ = 0,2066$ dan $E_- = 0,2227$, sehingga

$$E_+ - E_- = -0,0161. \tag{4.53}$$

Estimator gradien yang diperoleh menjadi

$$\hat{\mathbf{g}} = \begin{pmatrix} -0,0805 \\ 0,0805 \end{pmatrix}. \tag{4.54}$$

Dengan menggunakan laju pembelajaran $\eta = 0,1$, parameter kemudian diperbarui menurut aturan SPSA

$$\boldsymbol{\theta}^{(1)} = \boldsymbol{\theta}^{(0)} - \eta \hat{\mathbf{g}} = \begin{pmatrix} 1,5789 \\ 1,5627 \end{pmatrix}. \tag{4.55}$$

Siklus perturbasi, estimasi gradien, dan pembaruan parameter tersebut diulang secara terus-menerus hingga perubahan energi antar iterasi menjadi sangat kecil. Pada implementasi penelitian ini, proses optimasi dilakukan secara penuh terhadap energi ekspektasi yang dihitung langsung dari simulasi sirkuit kuantum VQE, sehingga seluruh parameter rotasi pada HEA diperbarui secara simultan menuju keadaan energi minimum. Konvergensi energi yang dicapai pada akhir proses optimasi kemudian diinterpretasikan sebagai pendekatan terhadap keadaan dasar Hamiltonian Ising yang merepresentasikan solusi portofolio optimal.

Untuk mengotomatisasi proses optimasi tersebut pada seluruh jendela data pengamatan, algoritma SPSA diimplementasikan menggunakan pustaka NumPy. Implementasi ini membangkitkan vektor perturbasi Rademacher secara acak pada setiap iterasi, mengevaluasi energi ekspektasi pada dua titik perturbasi yang simetris, lalu memperbarui parameter sirkuit menggunakan estimator gradien SPSA. Selain itu, koefisien pembelajaran a_k dan amplitudo perturbasi c_k dibuat menurun secara adaptif mengikuti formulasi standar SPSA sehingga proses optimasi tetap stabil ketika mendekati titik minimum energi. Struktur algoritma lengkap yang digunakan pada penelitian ini ditunjukkan pada Daftar Kode 4.8.

```
import numpy as np

def run_spsa_test(cost_circuit, n_params, init_params=None, a_base=0.1, c_base=0.1, max_iters
    =30, batch_size=25):
    """Versi ringan SPSA untuk pengetesan Learning Rate."""
    rng = np.random.default_rng(42)
    if init_params is not None:
```

```
        params = init_params.copy()
    else:
        params = rng.uniform(0, 2 * np.pi, n_params)

    A_coeff = max_iters * 0.1
    alpha_exp = 0.602
    gamma_exp = 0.101

    total_iters = 0
    while total_iters < max_iters:
        for _ in range(batch_size):
            k = total_iters
            a_k = a_base / (A_coeff + k + 1) ** alpha_exp
            c_k = c_base / (k + 1) ** gamma_exp

            # Formulasi parameter perturbasi serentak (Rademacher)
            delta = 2 * rng.integers(0, 2, size=n_params) - 1

            # Estimasi fungsi biaya energi (cost circuit)
            cost_plus = float(cost_circuit(params + c_k * delta))
            cost_minus = float(cost_circuit(params - c_k * delta))

            # Kalkulasi gradien
            grad = (cost_plus - cost_minus) / (2 * c_k * delta)
            params = params - a_k * grad

            total_iters += 1

    return float(cost_circuit(params))
```

Kode 4.8: Implementasi algoritma dasar SPSA

4.6 Evolusi Entropi Von Neumann dan Distribusi Probabilitas

Dalam kasus simulasi pencarian alokasi portofolio optimum menggunakan VQE, probabilitas keputusan strategis yang didapatkan senantiasa mengalami perubahan pada setiap siklus evaluasi parameter. Pada tahap awal pembaruan iterasi, sirkuit kuantum pada umumnya menghasilkan distribusi probabilitas yang tersebar secara merata pada seluruh kemungkinan formasi kombinasi basis. Pola persebaran semacam ini mengindikasikan bahwa arsitektur algoritma kuantum masih berada dalam pengaruh keadaan superposisi maksimal dengan nilai entropi global yang sangat tinggi. Namun, seiring dengan peluruhan pembaruan parameter rotasi oleh optimasi, sistem kuantum pada akhirnya akan mengalami peluruhan fungsi gelombang menuju satu basis keadaan. Fenomena tersebut dapat teramati pada hasil simulasi yang diperoleh. Pada Gambar I.5, nilai Entropi Von Neumann mengalami penurunan sepanjang proses optimasi SPSA hingga mencapai daerah konvergensi bernilai rendah. Penurunan entropi tersebut mengindikasikan bahwa parameter sirkuit mampu menghasilkan perubahan distribusi probabilitas signifikan selama proses optimasi berlangsung. Sebaliknya, pada Gambar I.2, nilai Entropi Von Neumann cenderung bertahan pada tingkat yang relatif tinggi meskipun proses optimasi telah berlangsung selama banyak iterasi. Kondisi tersebut mengindikasikan bahwa distribusi probabilitas sistem tetap tersebar pada banyak keadaan basis sehingga proses lokalisasi probabilitas menuju satu konfigurasi dominan tidak berlangsung secara efektif.

Untuk merealisasikan evaluasi tersebut pada lingkungan komputasi numerik, keadaan kuantum akhir yang dihasilkan oleh sirkuit VQE terlebih dahulu direpresentasikan sebagai vektor keadaan $|\psi\rangle$. Selanjutnya dilakukan proses trace tereduksi (*partial tra-*

ce) terhadap seluruh qubit kecuali satu subsistem pengamatan guna memperoleh matriks densitas tereduksi ρ_A. Nilai eigen dari matriks ρ_A kemudian digunakan untuk menghitung Entropi Von Neumann. Implementasi algoritmik dari prosedur tersebut ditunjukkan pada Daftar Kode 4.9.

```
import numpy as np

def calculate_entanglement_entropy(psi):
    """
    Menghitung Entanglement Entropy (Von Neumann) dari statevector N-qubit.
    Menggunakan partial trace terhadap semua qubit kecuali qubit 0.
    """
    # 1. Konversi ke numpy array complex
    psi = np.array(psi, dtype=complex)
    dim = len(psi)
    n_qubits = int(np.round(np.log2(dim)))

    # 2. Partial Trace untuk mendapatkan Reduced Density Matrix rho_A (Qubit 0)
    psi_reshaped = psi.reshape(2, 2**(n_qubits - 1))
    rho_A = np.dot(psi_reshaped, psi_reshaped.conj().T)

    # 3. Hitung Eigenvalues dari rho_A (matriks 2x2)
    eigvals = np.linalg.eigvalsh(rho_A)

    # 4. Bersihkan nilai negatif sangat kecil akibat floating point error
    eigvals = eigvals[eigvals > 1e-12]

    # 5. Von Neumann Entropy: S = -sum(p * log2(p))
    if len(eigvals) == 0:
        return 0.0

    return -np.sum(eigvals * np.log2(eigvals))
```

Kode 4.9: Skrip kalkulasi fungsional matriks evaluasi observabel algoritma Entropi Von Neumann komputasional menggunakan matriks trace tereduksi arsitektur operasional pada ruang fase deterministik fungsi ekuilibrium

4.7 Validasi Solusi melalui Algoritma *Brute Force*

4.7.1 Kalkulasi Energi Kombinatorial

Validasi hasil optimasi yang diperoleh melalui algoritma VQE dilakukan menggunakan metode *brute force*, yaitu dengan mengevaluasi seluruh konfigurasi rangkaian bit yang mungkin pada ruang solusi. Berbeda dengan VQE yang menelusuri lanskap energi secara variasional melalui pembaruan parameter sirkuit kuantum, metode *brute force* menghitung energi setiap keadaan basis secara eksplisit sehingga solusi energi minimum global dapat diperoleh secara pasti. Oleh karena itu, metode ini digunakan sebagai acuan untuk menilai akurasi hasil optimasi kuantum yang diperoleh pada setiap jendela data pengamatan. Secara matematis, energi setiap konfigurasi portofolio dihitung menggunakan Hamiltonian Ising yang telah diperoleh pada bagian sebelumnya. Untuk suatu konfigurasi biner $\mathbf{x}$, energi sistem diberikan oleh

$$E(\mathbf{x}) = \sum_i h_i z_i + \sum_{i<j} J_{ij} z_i z_j + C, \tag{4.56}$$

dengan variabel spin didefinisikan sebagai

$$z_i = 1 - 2x_i, \tag{4.57}$$

sehingga setiap konfigurasi rangkaian bit portofolio dapat langsung dipetakan ke keadaan spin yang bersesuaian. Pada sistem yang terdiri atas N aset, metode *brute force* harus mengevaluasi sebanyak 2^N konfigurasi yang berbeda untuk menemukan energi minimum global.

Sebagai ilustrasi numerik, untuk sistem dua aset ($N = 2$) dengan Hamiltonian

$$\hat{\mathcal{H}}_{N2} = 0,1735\hat{Z}_1 + 0,0931\hat{Z}_2 + 0,5014\hat{Z}_1\hat{Z}_2 + 0,2320, \tag{4.58}$$

terdapat empat konfigurasi basis yang harus dievaluasi, yaitu $|00\rangle$, $|01\rangle$, $|10\rangle$, dan $|11\rangle$. Dengan mensubstitusikan nilai spin yang bersesuaian ke dalam fungsi energi Ising, diperoleh

$$E(|00\rangle) \approx 1,0000, \tag{4.59}$$
$$E(|01\rangle) \approx -0,1890, \tag{4.60}$$
$$E(|10\rangle) \approx -0,3498, \tag{4.61}$$
$$E(|11\rangle) \approx 0,4667. \tag{4.62}$$

Hasil evaluasi tersebut menunjukkan bahwa konfigurasi $|10\rangle$ memiliki energi terendah sehingga ditetapkan sebagai keadaan dasar sistem. Solusi ini identik dengan hasil yang diperoleh melalui pencarian ekuilibrium Nash maupun optimasi VQE pada jendela data yang sama. Kesesuaian ketiga pendekatan tersebut menunjukkan bahwa proses transformasi dari model portofolio klasik menuju Hamiltonian Ising serta prosedur optimasi kuantum yang diterapkan telah menghasilkan solusi yang konsisten.

4.7.2 Implementasi *Brute Force* Berbasis Python

Proses validasi solusi optimal dilakukan melalui eksplorasi menyeluruh terhadap seluruh konfigurasi rangkaian bit yang mungkin pada Hamiltonian Ising hasil transformasi portofolio. Berbeda dengan algoritma VQE yang menelusuri lanskap energi secara variasional, metode *brute force* mengevaluasi energi setiap keadaan basis secara eksplisit sehingga solusi energi minimum global dapat diperoleh tanpa aproksimasi. Oleh karena itu, hasil yang diperoleh dari prosedur ini digunakan sebagai acuan pembanding untuk mengukur tingkat akurasi solusi yang ditemukan oleh algoritma kuantum.

Implementasi numeriknya diawali dengan pendefinisian fungsi calculate_ising_energy(), yang bertugas menghitung energi suatu konfigurasi biner berdasarkan parameter Hamiltonian Ising (h_i, J_{ij}, C). Pada fungsi ini, setiap variabel biner portofolio x_i terlebih dahulu dikonversi ke representasi spin Ising melalui transformasi $z_i = 1-2x_i$. Kemudian dihitung energi total yang terdiri dari penjumlahan kontribusi bias linear, interaksi pasangan spin, serta konstanta pergeseran energi. Selanjutnya, fungsi brute_force_portfolio() memanfaatkan pustaka `itertools.product` untuk membangkitkan seluruh konfigurasi rangkaian bit sebanyak 2^N keadaan. Untuk setiap konfigurasi yang dihasilkan, program menghitung energi Ising yang bersesuaian dan membandingkannya dengan nilai minimum yang telah ditemukan sebelumnya. Mekanisme ini memungkinkan proses identifikasi keadaan dasar global dilakukan secara deterministik tanpa memerlukan prosedur optimasi iteratif. Selain mencari energi minimum global, algoritma juga mengevaluasi solusi yang memenuhi kendala kardinalitas portofolio $\sum_i x_i = K$. Hal tersebut berguna supaya hasil validasi juga memastikan solusi tetap konsisten dengan variabel batasan. Keseluruhan prosedur validasi numerik tersebut diimplementasikan menggunakan bahasa pemrograman Python.

Struktur algoritma yang digunakan untuk menghitung energi setiap konfigurasi, mengeksplorasi seluruh ruang solusi, serta mengidentifikasi keadaan dasar portofolio ditunjukkan pada Daftar Kode 4.10.

```
import numpy as np
import pandas as pd
from itertools import product

def calculate_ising_energy(x, h, J, C_Ising):
    n_assets = len(x)
    Z = 1 - 2 * np.array(x)

    # Suku Bias: sum(h_i * Z_i)
    energy_bias = np.sum(h * Z)

    # Suku Interaksi: sum(J_ij * Z_i * Z_j)
    energy_interaction = 0
    for i in range(n_assets):
        for j in range(i + 1, n_assets):
            energy_interaction += J[i, j] * Z[i] * Z[j]

    return energy_bias + energy_interaction + C_Ising

def brute_force_portfolio(n_assets, h, J, C_Ising, K=2):
    best_global_e = float('inf')
    best_global_bs = None
    best_constrained_e = float('inf')
    best_constrained_bs = None

    # Eksplorasi 2^N
    for bit_tuple in product([0, 1], repeat=n_assets):
        x = np.array(bit_tuple)
        energy = calculate_ising_energy(x, h, J, C_Ising)
        bit_str = "".join(map(str, bit_tuple))

        if energy < best_global_e:
            best_global_e = energy
            best_global_bs = bit_str

        if sum(x) == K:
            if energy < best_constrained_e:
                best_constrained_e = energy
                best_constrained_bs = bit_str

    return best_constrained_bs, best_constrained_e
```

Kode 4.10: Skrip komputasi algoritma *brute force* untuk validasi lanskap energi portofolio pada arsitektur kuantum

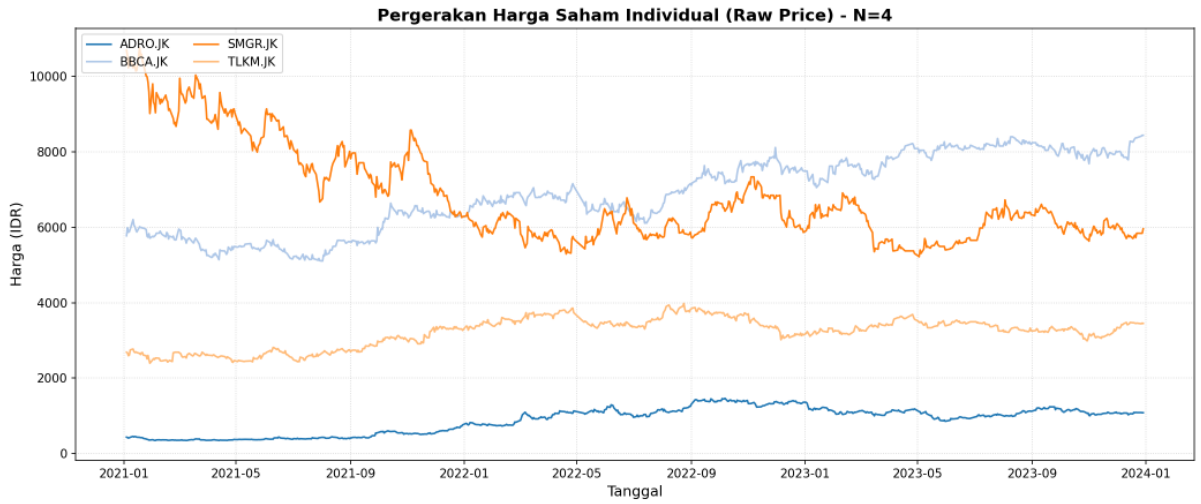
4.8 Analisis Performa Portofolio dan Dinamika penyeimbangan kembali

Evaluasi performa dilakukan melalui simulasi pengujian historis pada rentang waktu Januari 2021 hingga Desember 2023 dengan menerapkan mekanisme penyeimbangan kembali bulanan. Pada setiap awal jendela observasi, parameter statistik pasar dihitung kembali menggunakan data historis pada periode tinjauan, kemudian dipetakan ke dalam fungsi Hamiltonian portofolio untuk menentukan konfigurasi aset optimal pada bulan berikutnya. Pendekatan ini memungkinkan sistem menyesuaikan komposisi portofolio secara dinamis terhadap perubahan karakteristik pasar yang terus berkembang. Untuk menilai efektivitas metode yang diusulkan, performa strategi GT-VQE dibandingkan dengan tiga pendekatan pembanding, yaitu VQE murni tanpa integrasi teori permainan,

optimasi klasik berbasis SLSQP berdasarkan model Markowitz, dan strategi pasif bobot disetarakan (*Equal Weight*) yang membeli dan menahan seluruh aset porporsi yang sama. Seluruh pengujian dilakukan menggunakan kombinasi saham BBCA, ADRO, TLKM, dan SMGR yang pergerakan harganya ditampilkan pada Gambar 4.2. Ringkasan hasil evaluasi kuantitatif untuk sistem $N = 2$ dan $N = 4$ kemudian disajikan pada Tabel 4.2. Berdasarkan hasil pengujian tersebut, strategi GT-VQE menunjukkan performa terbaik pada hampir seluruh metrik evaluasi yang digunakan. Pada sistem $N = 2$, strategi ini menghasilkan imbal hasil kumulatif sebesar $145,07\%$, melampaui VQE murni ($121,30\%$), bobot disetarakan ($97,88\%$), maupun optimasi klasik SLSQP ($51,34\%$). Hal ini juga didukung dengan metrik rasio Sharpe dengan rumus

$$Sharpe\ Ratio = \frac{R_p - R_f}{\sigma_p}, \tag{4.63}$$

dimana R_p adalah imbal hasil portofolio, R_f adalah imbal hasil bebas risiko (diasumsikan sebesar 4% per tahun), dan σ_p adalah deviasi standar imbal hasil portofolio. Nilai rasio Sharpe yang dicapai strategi GT-VQE mencapai $1,0185$, yang mengindikasikan bahwa setiap unit risiko yang ditanggung mampu dikonversi menjadi tingkat imbal hasil yang lebih tinggi dibandingkan seluruh metode pembanding. Walaupun nilai penurunan maksimum yang dicatat mencapai $30,33\%$, tingkat imbal hasil yang diperoleh tetap mampu mengompensasi risiko tersebut sehingga menghasilkan profil kinerja keseluruhan yang paling kompetitif.



Gambar 4.2: Pergerakan Harga Saham Individual (*Raw Price*) pada Sistem N=4 (2021–2023).

Keunggulan metode hibrida semakin terlihat ketika dimensi masalah diperbesar menjadi sistem $N = 4$. Pada kondisi ini, performa optimasi klasik mengalami degradasi yang cukup signifikan dengan imbal hasil hanya sebesar $12,91\%$, sedangkan VQE murni menghasilkan imbal hasil sebesar $16,89\%$. Sebaliknya, GT-VQE masih mampu mempertahankan imbal hasil sebesar $73,15\%$ dengan rasio Sharpe mencapai $1,0107$. Hasil ini menunjukkan bahwa integrasi informasi strategis yang diperoleh dari teori permainan berperan penting dalam menjaga kualitas solusi ketika ukuran ruang pencarian meningkat secara eksponensial. Selain menghasilkan imbal hasil yang lebih tinggi, strategi GT-VQE

juga mampu mempertahankan nilai penurunan maksimum sebesar $16,68\%$, lebih rendah dibandingkan VQE murni maupun strategi bobot disetarakan. Dengan demikian, peningkatan performa yang dicapai tidak hanya berasal dari peningkatan potensi keuntungan, tetapi juga dari kemampuan sistem dalam mengendalikan risiko selama proses investasi berlangsung. Jika dibandingkan dengan kinerja pasar secara keseluruhan yang direpresentasikan oleh IHSG dengan imbal hasil kumulatif sebesar $19,13\%$ dan rasio Sharpe $0,5557$, seluruh pendekatan berbasis GT-VQE mampu menghasilkan keunggulan yang sangat signifikan. Temuan ini mengindikasikan bahwa kombinasi antara formulasi Hamiltonian portofolio, optimasi kuantum variasional, serta inisialisasi berbasis ekuilibrium Nash berhasil mengeksploitasi informasi struktural pasar yang tidak sepenuhnya dapat ditangkap oleh pendekatan optimasi konvensional. Oleh karena itu, hasil pada Tabel 4.2 menjadi indikasi awal bahwa integrasi teori permainan dan komputasi kuantum memberikan kontribusi nyata terhadap peningkatan kualitas keputusan alokasi aset pada berbagai skala kompleksitas sistem.

Tabel 4.2: Ringkasan Performa Finansial Strategi Kuantum, SLSQP, dan *Equal Weight* ($N = 2, 4$).

Strategi	**Sistem N=2**			**Sistem N=4**		
	Return	**Sharpe**	**MDD**	**Return**	**Sharpe**	**MDD**
GT-VQE (Hibrida)	**145,07%**	**1,0185**	30,33%	**73,15%**	**1,0107**	**16,68%**
VQE (Tanpa GT)	121,30%	0,9331	30,33%	16,89%	0,3592	20,89%
SLSQP (Klasik)	51,34%	0,6736	**15,91%**	12,91%	0,2810	21,12%
Equal Weight	97,88%	0,9764	27,49%	35,57%	0,6790	18,42%

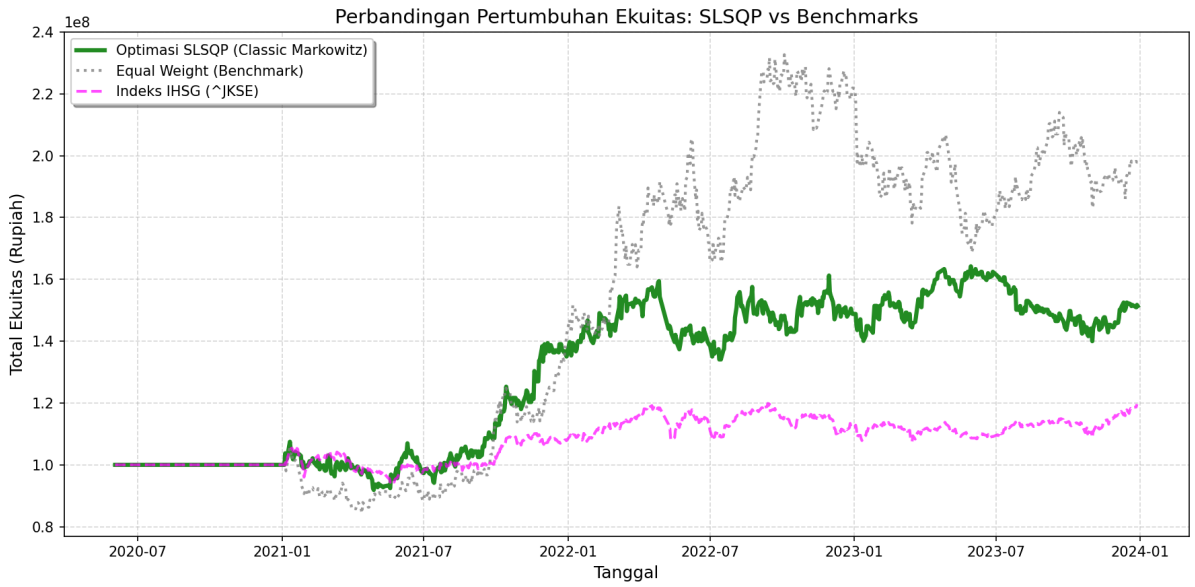
Keunggulan performa GT-VQE yang ditunjukkan pada Tabel 4.2 sesuai karakteristik dinamika kuantum yang telah diamati pada tahapan optimasi sebelumnya. Pada setiap jendela penyeimbangan kembali, proses pencarian portofolio dilakukan dengan memetakan fungsi objektif investasi ke dalam Hamiltonian Ising, kemudian menavigasi lanskap energinya menggunakan algoritma VQE berbasis SPSA. Dalam kerangka tersebut, kualitas keputusan investasi sangat bergantung pada kemampuan sirkuit variasional untuk menemukan keadaan dasar Hamiltonian yang merepresentasikan konfigurasi aset dengan utilitas optimal. Semakin dekat energi ekspektasi yang dicapai terhadap energi minimum global hasil validasi *brute force*, semakin besar probabilitas bahwa portofolio yang dipilih merepresentasikan solusi optimum dari permasalahan optimasi yang sedang dihadapi. Hubungan antara kualitas solusi tersebut dan dinamika internal sirkuit kuantum dapat diamati melalui evolusi entropi Von Neumann yang telah dibahas pada Section 4.6. Pada jendela-jendela simulasi yang berhasil mencapai solusi optimal, distribusi probabilitas portofolio secara bertahap mengalami pemusatan pada satu rangkaian bit dominan sehingga nilai entropi menurun menuju kondisi yang relatif rendah. Fenomena ini menunjukkan bahwa parameter rotasi $\boldsymbol{\theta}$ berhasil mengarahkan sistem keluar dari superposisi yang terlalu menyebar menuju keadaan kuantum yang lebih terlokalisasi. Sebaliknya, ketika distribusi probabilitas tetap tersebar pada banyak konfigurasi portofolio sekaligus, entropi cenderung bertahan pada nilai yang tinggi dan kualitas solusi yang diperoleh menjadi kurang optimal.

Interpretasi tersebut juga sejalan dengan analisis fenomena dataran tandus. Pada beberapa jendela observasi ditemukan kondisi ketika nilai entropi gagal mengalami penurunan yang berarti dan bertahan pada tingkat yang relatif tinggi sepanjang proses

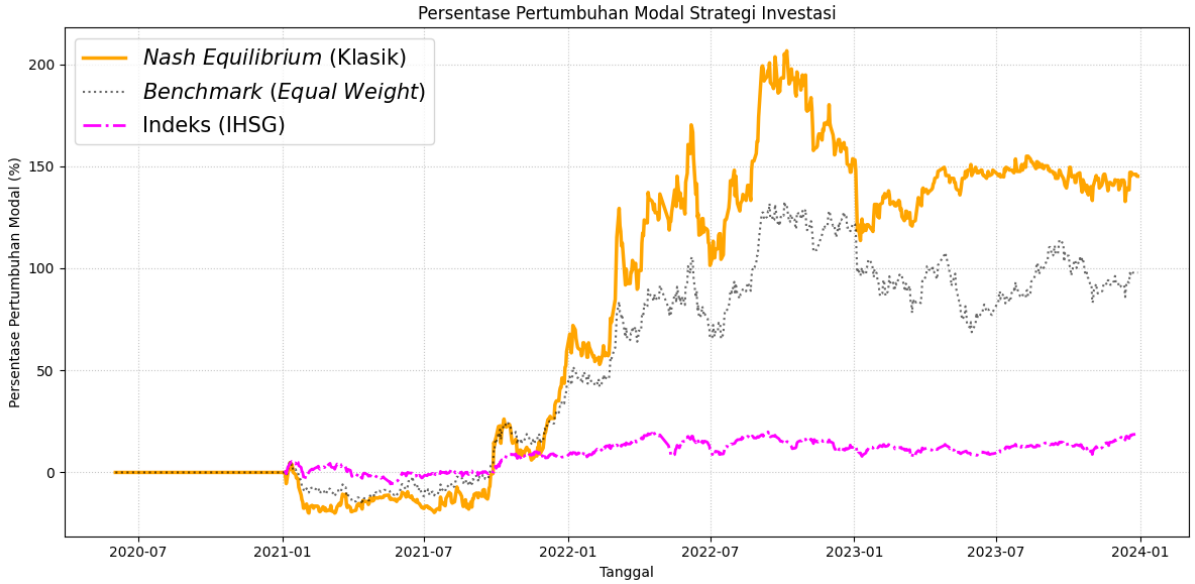
optimasi. Hal ini mengindikasikan bahwa lanskap energi yang dieksplorasi memiliki gradien yang sangat kecil sehingga pembaruan parameter SPSA tidak mampu menghasilkan pergeseran amplitudo probabilitas yang signifikan. Akibatnya, sirkuit kesulitan membedakan konfigurasi portofolio yang benar-benar optimal dari konfigurasi lain yang memiliki energi serupa. Sebaliknya, jendela-jendela yang menunjukkan penurunan entropi secara konsisten umumnya juga menghasilkan konvergensi energi yang lebih cepat serta probabilitas solusi optimal yang lebih tinggi. Hubungan ini memperlihatkan bahwa entropi Von Neumann tidak hanya berfungsi sebagai ukuran keterbelitan dan ketidakpastian kuantum, tetapi juga dapat digunakan sebagai indikator tambahan untuk mendeteksi keberhasilan proses optimasi VQE dalam menghindari hambatan dataran tandus. Dalam konteks tersebut, keberhasilan GT-VQE mempertahankan performa pada sistem $N = 4$ mengindikasikan bahwa mekanisme inisialisasi awal berbasis ekuilibrium Nash mampu memberikan titik awal parameter yang lebih dekat terhadap wilayah solusi berkualitas tinggi. Dampaknya, ruang eksplorasi yang harus ditelusuri oleh SPSA menjadi lebih sempit sehingga probabilitas terjebak pada daerah gradien rendah dapat dikurangi. Efek ini menjelaskan mengapa GT-VQE tetap mampu menghasilkan rasio Sharpe yang tinggi dan imbal hasil yang stabil ketika ukuran ruang solusi meningkat secara eksponensial, sedangkan VQE murni dan optimasi klasik mulai mengalami penurunan performa yang cukup signifikan. Dengan kata lain, peningkatan performa finansial yang diamati pada tahap pengujian historis merupakan konsekuensi langsung dari peningkatan kualitas proses optimasi kuantum yang terjadi pada tingkat fundamental sirkuit variasional.

Perbandingan performa antara strategi kuantum hibrida, kuantum murni, dan klasik untuk sistem $N = 2$ dapat ditinjau pada Gambar 4.3 hingga 4.6, sedangkan untuk perbandingan ekstensif pada sistem $N = 4$ disajikan pada Gambar 4.7 hingga 4.10. Analisis visual pada Gambar 4.6 dan Gambar 4.10 memperlihatkan bahwa strategi GT-VQE menghasilkan kurva pertumbuhan modal yang lebih stabil dibandingkan metode pembanding. Pada sistem $N = 2$, baik VQE murni maupun GT-VQE mampu menghasilkan pertumbuhan portofolio yang melampaui strategi klasik. Namun, ketika jumlah aset diperbesar menjadi $N = 4$, perbedaan karakteristik kedua pendekatan tersebut mulai terlihat secara jelas. VQE murni mengalami penurunan performa yang cukup drastis, sedangkan GT-VQE tetap mempertahankan laju pertumbuhan modal yang konsisten hingga akhir periode pengujian. Fenomena ini menunjukkan bahwa penambahan dimensi ruang solusi memberikan tantangan optimasi yang semakin besar bagi algoritma variasional, sehingga kualitas inisialisasi parameter menjadi faktor yang semakin menentukan keberhasilan pencarian solusi.

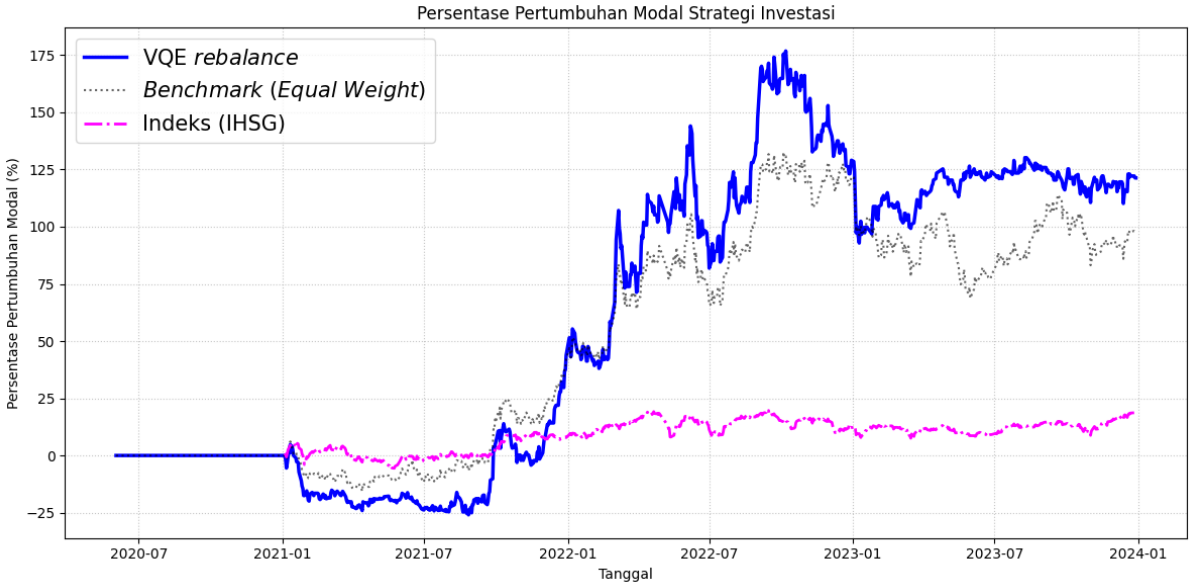
Efektivitas pendekatan inisialisasi awal juga dapat diamati melalui dinamika adaptasi kedalaman sirkuit pada Gambar 4.11. Untuk sistem tanpa integrasi teori permainan, algoritma VQE cenderung membutuhkan kedalaman sirkuit yang lebih tinggi dan lebih fluktuatif dari satu jendela observasi ke jendela berikutnya. Kondisi ini menunjukkan bahwa sistem harus mengeksplorasi ruang parameter yang lebih luas untuk mencapai energi minimum yang memadai. Sebaliknya, GT-VQE secara dominan mampu beroperasi pada kedalaman rendah, terutama pada kedalaman 2, baik untuk sistem $N = 2$ maupun $N = 4$. Temuan ini memiliki implikasi penting karena peningkatan kedalaman sirkuit umumnya memperbesar risiko munculnya dataran tandus, meningkatkan akumulasi galat numerik, serta memperpanjang waktu komputasi. Oleh sebab itu, kemampuan GT-VQE mencapai solusi berkualitas tinggi pada kedalaman yang relatif dangkal menunjukkan adanya peningkatan efisiensi optimasi yang signifikan.



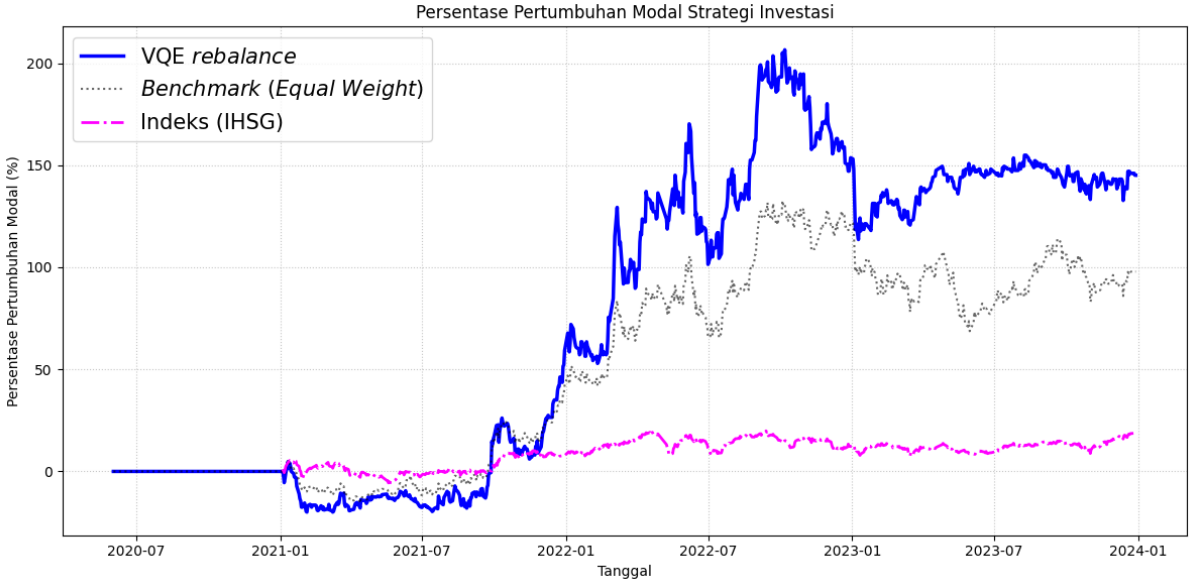
Gambar 4.3: Visualisasi Pertumbuhan Modal Optimasi Klasik SLSQP pada Sistem N=2.



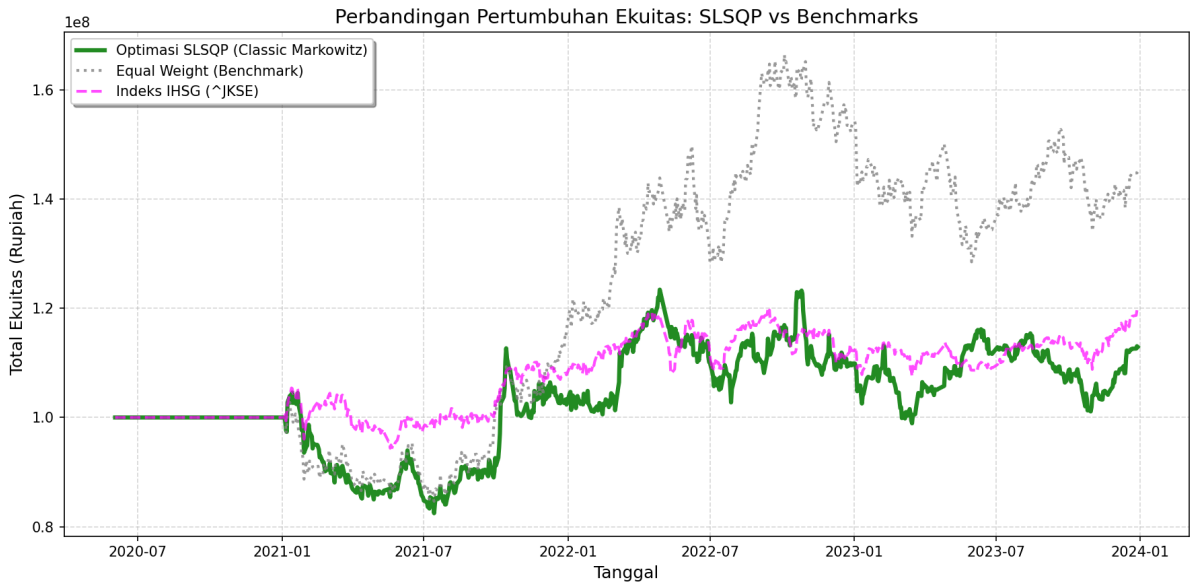
Gambar 4.4: Visualisasi Pertumbuhan Modal Strategi Nash (Klasik) pada Sistem N=2.



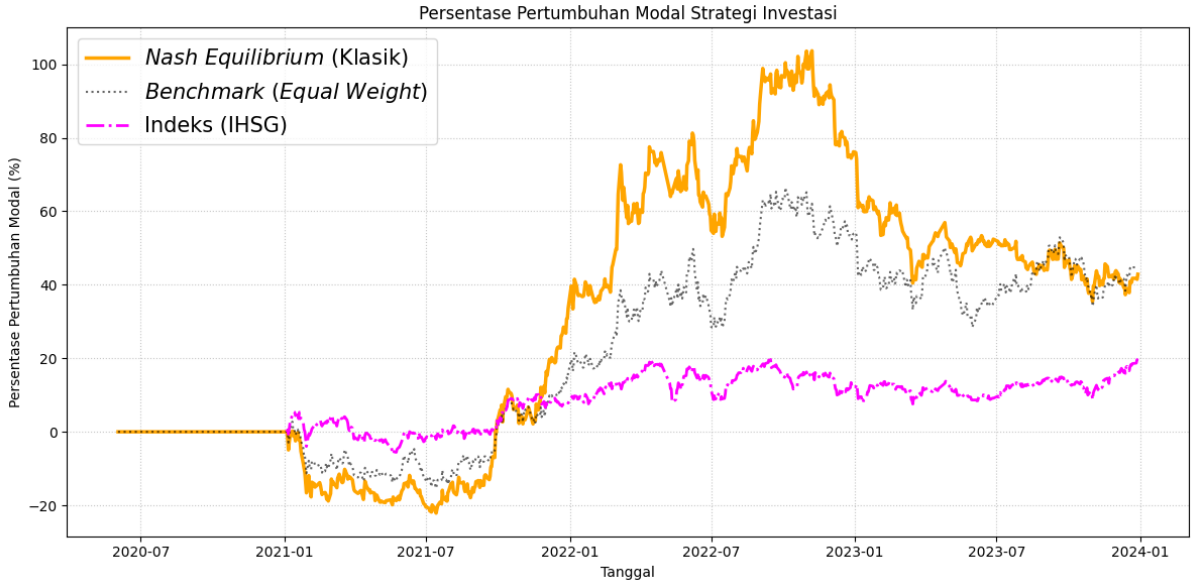
Gambar 4.5: Visualisasi Pertumbuhan Modal VQE Murni (Tanpa GT) pada Sistem N=2.



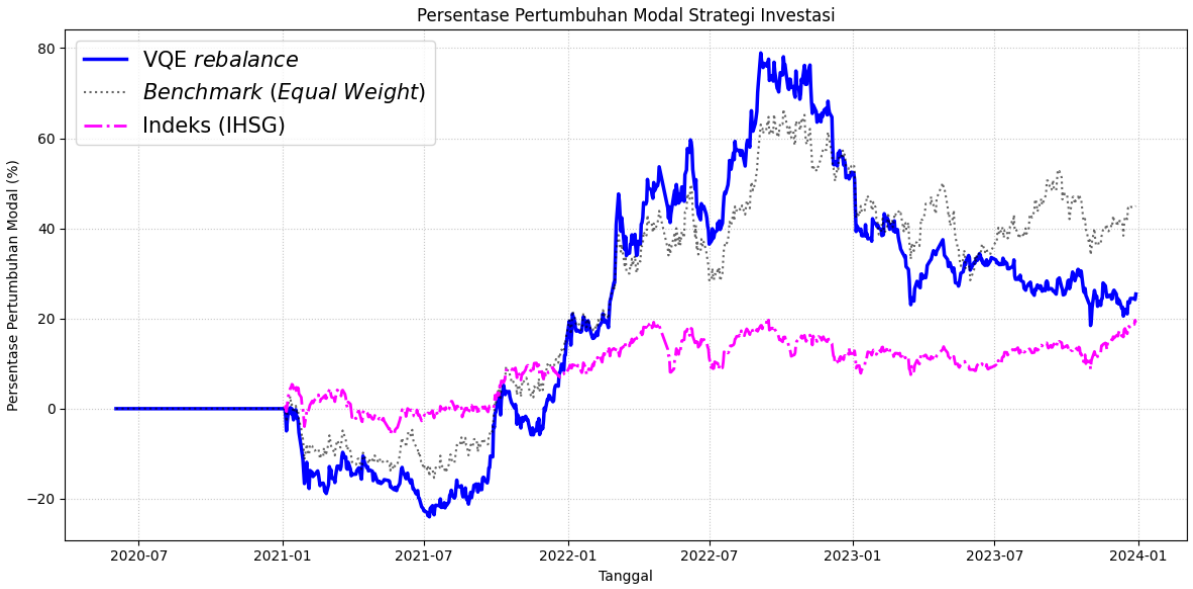
Gambar 4.6: Visualisasi Pertumbuhan Modal Strategi GT-VQE (Hibrida) pada Sistem N=2.



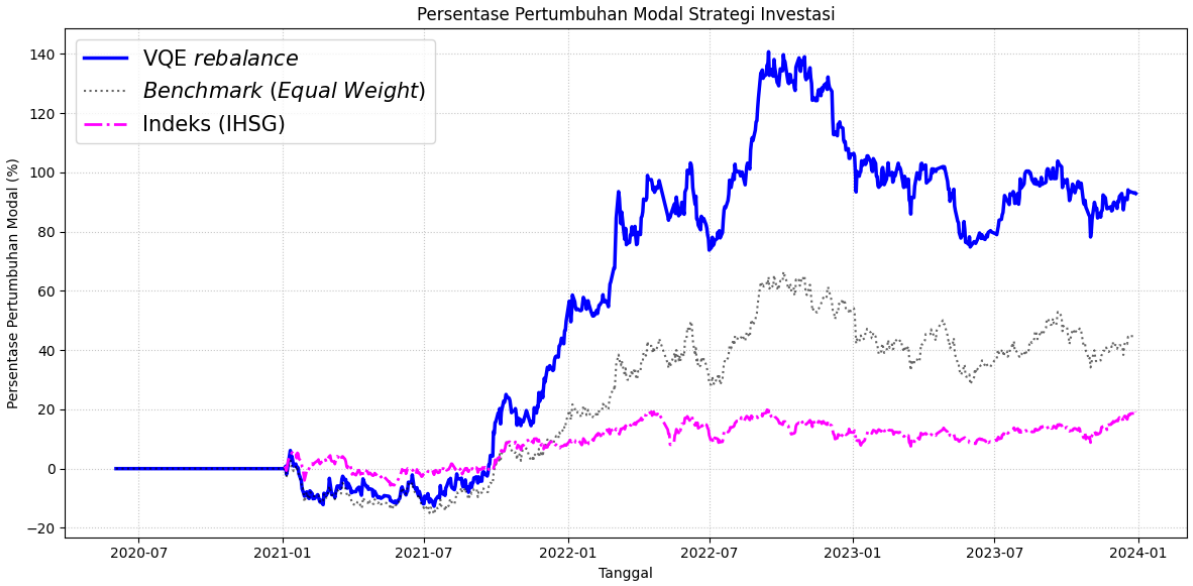
Gambar 4.7: Visualisasi Pertumbuhan Modal Optimasi Klasik SLSQP pada Sistem N=4.



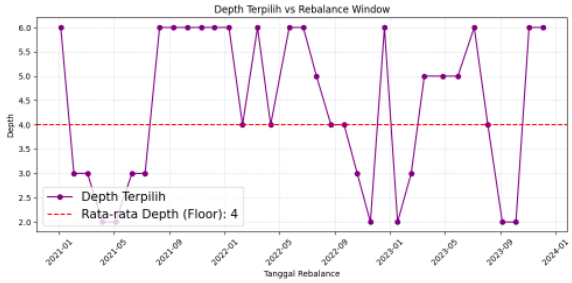
Gambar 4.8: Visualisasi Pertumbuhan Modal Strategi Nash (Klasik) pada Sistem N=4.



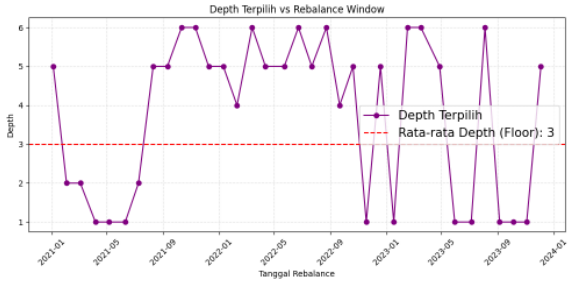
Gambar 4.9: Visualisasi Pertumbuhan Modal VQE Murni (Tanpa GT) pada Sistem N=4.



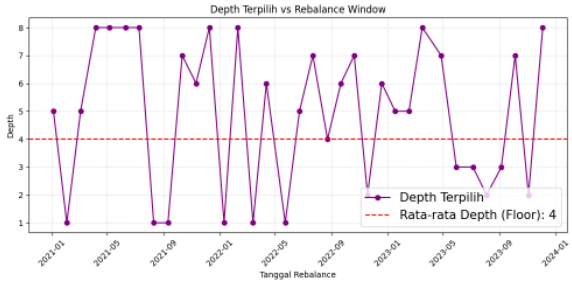
Gambar 4.10: Visualisasi Pertumbuhan Modal Strategi GT-VQE (Hibrida) pada Sistem N=4.



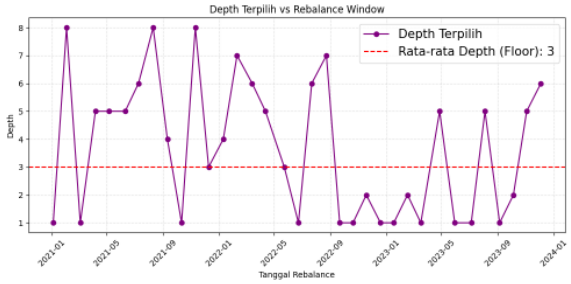
(a) VQE Murni ($N = 2$)



(b) GT-VQE Hibrida ($N = 2$)



(c) VQE Murni ($N = 4$)



(d) GT-VQE Hibrida ($N = 4$)

Gambar 4.11: Perbandingan Dinamika Adaptasi Kedalaman Sirkuit (kedalaman) per Jendela Simulasi antara VQE Murni dan GT-VQE untuk Berbagai Konfigurasi Aset.

Secara keseluruhan, hasil pengujian historis memperlihatkan bahwa integrasi teori permainan, optimasi variasional kuantum, dan mekanisme penyeimbangan kembali adaptif membentuk suatu kerangka pengambilan keputusan investasi yang saling melengkapi. Informasi strategis yang diperoleh dari ekuilibrium Nash berperan mempersempit ruang pencarian awal, SPSA bertugas mengarahkan parameter menuju energi minimum secara efisien, sedangkan evaluasi entropi memberikan indikator tambahan terhadap kualitas konvergensi yang dicapai. Kombinasi ketiga komponen tersebut menjelaskan mengapa GT-VQE mampu mempertahankan performa yang lebih baik dibandingkan pendekatan kuantum murni maupun optimasi klasik ketika kompleksitas sistem terus meningkat. Proses evaluasi performa dan simulasi pengujian historis ini dikendalikan oleh skrip utama yang melakukan iterasi jendela waktu dan mengeksekusi penyesuaian bobot portofolio. Skrip operasional pengujian historis dapat dilihat pada Daftar Kode 4.11, sedangkan mekanisme penyesuaian bobot setiap periode diimplementasikan pada Daftar Kode 4.12.

```
import numpy as np
import pandas as pd
from run_strategy_step import run_strategy_step
from rebalance_portfolio import rebalance_portfolio

def run_backtest(data_clean, tickers, config):
    """
    Menjalankan simulasi backtest berdasarkan data historis dan konfigurasi.
    """
    N = len(tickers)
    K = config['K']
    penalty_A = config['penalty_A']
    max_depth = config['max_depth']
    maxiter = config['maxiter']
    max_total_iter = config.get('max_total_iter', 500)
    batch_size = config.get('batch_size', 10)
    conv_window = config.get('conv_window', 4)
```

```
conv_tol = config.get('conv_tol', 1e-3)
use_warm_start = config.get('use_warm_start', True)
initial_capital = config['initial_capital']
lookback_days = config['lookback_days']
rebalance_days = config['rebalance_days']
start_bt_date = pd.to_datetime(config['start_bt_date'])

start_idx = np.searchsorted(data_clean.index, start_bt_date)
rebalance_indices = range(start_idx, len(data_clean), rebalance_days)

value_vqe = [initial_capital] * start_idx
value_bench = [initial_capital] * start_idx
value_nash = [initial_capital] * start_idx
value_assets = {t: [initial_capital] * start_idx for t in tickers}

holdings_vqe, holdings_bench, holdings_nash = np.zeros(N), np.zeros(N), np.zeros(N)
cash_vqe, cash_bench, cash_nash = initial_capital, initial_capital, initial_capital
cash_assets = {t: initial_capital for t in tickers}
holdings_assets = {t: 0.0 for t in tickers}

detail_logs = []
depths_history = []
rebalance_dates = []
window_analysis_history = [] # Menyimpan semua data untuk plotting per window

print(f"\n--- Memulai Backtest dari {data_clean.index[start_idx].date()} hingga {
data_clean.index[-1].date()} ---")

for i, curr_idx in enumerate(rebalance_indices):
    curr_date = data_clean.index[curr_idx]
    train_start = max(0, curr_idx - lookback_days)
    train_data = data_clean.iloc[train_start:curr_idx]
    next_idx = (rebalance_indices[i + 1] if i + 1 < len(rebalance_indices) else len(
data_clean))

    # Strategi Step
    selected_indices, depth_used, energy_final, best_history, best_ent_hist,
best_gvar_hist, depth_energies, lr_data, ne_bs, ne_utility, winning_probs, best_params =
run_strategy_step(
        train_data, tickers, curr_date, K=K, penalty_A=penalty_A, max_depth=max_depth,
maxiter=maxiter,
        max_total_iter=max_total_iter, batch_size=batch_size,
        conv_window=conv_window, conv_tol=conv_tol,
        use_warm_start=use_warm_start,
        file_config=config.get('files')
    )

    depths_history.append(depth_used)
    rebalance_dates.append(curr_date.date())

    # Simpan semua detail untuk plotting nanti
    window_analysis_history.append({
        'date': curr_date.date(),
        'best_history': best_history,
        'best_ent_hist': best_ent_hist,
        'best_gvar_hist': best_gvar_hist,
        'depth_energies': depth_energies,
        'lr_data': lr_data,
        'winning_probs': winning_probs,
        'ne_bs': ne_bs,
        'best_params': best_params,
        'best_depth': depth_used,
        'K': K,
        'use_warm_start': use_warm_start
    })

    selected_names = [tickers[idx] for idx in selected_indices]
    vqe_details = "".join([f"    [Depth {d}] Konvergen dalam {iters} iterasi | E = {en:.6f
}\n" for d, en, iters in depth_energies])

    print(f"[{curr_date.date()}] VQE Terpilih: {selected_names} | Depth: {depth_used} |
E_min: {energy_final:.6f}")
```

```
    log_entry = (f"[{curr_date.date()}]\n"
                 f"  - Nash Eq: {ne_bs} | Utility: {ne_utility:.6f}\n"
                 f"  - LR: {lr_data[2]:.4f}\n"
                 f"  - Detail:\n{vqe_details}"
                 f"  - Terpilih: {selected_names} | Depth: {depth_used} | E_min: {
energy_final:.6f}\n" + "-"*40)
    detail_logs.append(log_entry)

    # Rebalancing VQE
    target_w_vqe = np.zeros(N)
    if len(selected_indices) > 0:
        w = 1.0 / len(selected_indices)
        for idx in selected_indices: target_w_vqe[idx] = w

    # Rebalancing Nash
    target_w_nash = np.zeros(N)
    nash_indices = [j for j, bit in enumerate(ne_bs) if bit == '1']
    if len(nash_indices) > 0:
        w_n = 1.0 / len(nash_indices)
        for idx in nash_indices: target_w_nash[idx] = w_n

    target_w_bench = np.full(N, 1.0 / N)
    current_prices = data_clean.iloc[curr_idx].values

    cash_vqe, holdings_vqe = rebalance_portfolio(cash_vqe, holdings_vqe, target_w_vqe,
current_prices, N)
    cash_nash, holdings_nash = rebalance_portfolio(cash_nash, holdings_nash, target_w_nash
, current_prices, N)

    if i == 0:
        cash_bench, holdings_bench = rebalance_portfolio(cash_bench, holdings_bench,
target_w_bench, current_prices, N)
        for j, t in enumerate(tickers):
            target_w_indiv = np.zeros(N)
            target_w_indiv[j] = 1.0
            c_t, h_t = rebalance_portfolio(cash_assets[t], np.zeros(N), target_w_indiv,
current_prices, N)
            cash_assets[t], holdings_assets[t] = c_t, h_t[j]

    for d in range(curr_idx, next_idx):
        prices = data_clean.iloc[d].values
        value_vqe.append(cash_vqe + np.sum(holdings_vqe * prices))
        value_nash.append(cash_nash + np.sum(holdings_nash * prices))
        value_bench.append(cash_bench + np.sum(holdings_bench * prices))
        for j, t in enumerate(tickers):
            value_assets[t].append(cash_assets[t] + holdings_assets[t] * prices[j])

# --- EKSPOR HASIL EKUITAS KE CSV (Untuk Perbandingan) ---
suffix = f"_N{N}"
equity_df = pd.DataFrame({
    'Date': data_clean.index[:len(value_vqe)],
    'VQE': value_vqe,
    'Nash': value_nash,
    'Benchmark': value_bench
})
# Gunakan folder files config jika ada, atau default suffix
equity_csv_path = config.get('files', {}).get('equity_history', f'hasil_ekuitas_backtest{
suffix}.csv')
equity_df.to_csv(equity_csv_path, index=False)

return {
    'value_vqe': value_vqe,
    'value_nash': value_nash,
    'value_bench': value_bench,
    'value_assets': value_assets,
    'depths_history': depths_history,
    'rebalance_dates': rebalance_dates,
    'detail_logs': detail_logs,
    'window_analysis_history': window_analysis_history,
    'start_idx': start_idx
```

```
    }
```

Kode 4.11: Skrip komputasional untuk eksekusi simulasi pengujian historis dengan iterasi jendela bergulir (`backtest_runner.py`)

```
import numpy as np

def rebalance_portfolio(current_cash, current_holdings, target_weights, prices, N):
    """
    Menyeimbangkan portofolio berdasarkan bobot target.
    N: Jumlah aset
    """
    total_value    = current_cash + np.sum(current_holdings * prices)
    target_values  = total_value * target_weights
    new_holdings   = current_holdings.copy()
    new_cash       = current_cash

    for j in range(N):
        c_val = new_holdings[j] * prices[j]
        if c_val > target_values[j]:
            sell_val        = c_val - target_values[j]
            new_cash       += sell_val
            new_holdings[j] -= sell_val / prices[j]

    for j in range(N):
        c_val = new_holdings[j] * prices[j]
        if c_val < target_values[j]:
            buy_val       = min(target_values[j] - c_val, new_cash)
            new_cash     -= buy_val
            new_holdings[j] += buy_val / prices[j]

    return new_cash, new_holdings
```

Kode 4.12: Fungsi operasional untuk penyesuaian bobot portofolio pada momen evaluasi (`rebalance_portfolio.py`)

Bab 5

Penutup

5.1 Kesimpulan

Berdasarkan analisis hasil simulasi dan evaluasi secara komprehensif yang telah dilakukan pada bab sebelumnya, penelitian ini menghasilkan tiga kesimpulan utama yang menjawab rumusan masalah sebagai berikut:

1. Formulasi model Markowitz ke dalam kerangka EPG telah berhasil memetakan utilitas ekonomi menjadi sistem Hamiltonian fisis secara akurat dan konsisten. Penggunaan imbal hasi logaritmik untuk pembentukan matriks kovarians terbukti efektif menjaga simetri interaksi risiko (J_{ij}) tanpa distorsi, sementara imbal hasil sederhana mempertahankan linieritas bias ekspektasi imbal hasil (h_i). Implementasi suku penalti anggaran (A) dapat mendominasi koefisien kopling sehingga menciptakan batasan energi untuk mencegah sirkuit kuantum terjebak pada status portofolio yang melanggar aturan diversifikasi investasi.

2. Penerapan ekuilibrium Nash sebagai inisialisasi awal mampu meningkatkan efisiensi dan stabilitas konvergensi algoritma VQE. Injeksi solusi ekuilibrium Nash pada rotasi parameter awal memandu sirkuit melewati area dataran tandus yang dicirikan oleh kelenyapan varians gradien. Keunggulan strategis ini memungkinkan algoritma mencapai keadaan dasar pada kedalaman sirkuit yang dangkal ($L = 2$ hingga $L = 3$). Hal tersebut dapat mereduksi beban komputasi dan memicu keruntuhan entropi Von Neumann menuju konfigurasi probabilitas tunggal yang paling optimal.

3. Arsitektur HEA yang dikombinasikan dengan algoritma SPSA menunjukkan performa komputasi yang efisien dalam mengeksekusi optimasi portofolio. Estimasi gradien menggunakan distribusi Rademacher memungkinkan sistem merespons volatilitas pasar melalui mekanisme penyeimbangan kembali. Hal ini divalidasi oleh hasil pengujian historis yang mana algoritma GT-VQE menghasilkan pertumbuhan modal lebih superior dibandingkan strategi klasik maupun VQE murni dengan nilai rasio Sharpe sebesar 1,0107 pada sistem $N = 4$.

5.2 Saran

Berdasarkan keterbatasan dan temuan dalam pengerjaan tugas akhir ini, terdapat beberapa saran untuk penelitian lanjutan di masa mendatang

1. Mempertimbangkan parameter dearu sirkuit, seperti dekoherensi qubit dan fidelitas gerbang dalam mengeksekusi algoritma ini agar dapat diaplikasikan pada infrastruktur perangkat keras kuantum sesungguhnya.

2. Menggunakan variasi struktur Ansatz lain serta membandingkan performa berbagai algoritma optimasi klasik lainnya untuk menemukan kombinasi hibrida yang memiliki laju konvergensi paling efisien dalam menangkap kompleksitas interaksi agen di pasar modal.

3. Menggunakan rentang periode data historis yang lebih lama. Selain itu, pengujian pada periode yang sedang mengalami fenomena angsa hitam seperti dengan krisis ekonomi supaya dapat mengevaluasi agilitas dan reliabilitas algoritma dalam mempertahankan stabilitas ekuilibrium portofolio di tengah volatilitas pasar yang ekstrem.

Daftar Pustaka

Abdul Hali, N., & Yuliati, A. (2020, October). Markowitz Model Investment Portfolio Optimization: a Review Theory. *International Journal of Research in Community Services*, *1*(3), 14–18. Retrieved 2026-03-11, from https://journal.rescollacomm.com/index.php/ijrcs/article/view/104 doi: 10.46336/ijrcs.v1i3.104

Ahmadi, A., Roy, S., Mehrabbeik, M., Ghosh, D., Jafari, S., & Perc, M. (2023, April). The dynamics of a duopoly Stackelberg game with marginal costs among heterogeneous players. *PLOS ONE*, *18*(4), e0283757. Retrieved 2026-03-11, from https://dx.plos.org/10.1371/journal.pone.0283757 doi: 10.1371/journal.pone.0283757

Ahmed, S., Coulibaly, B., & Zlate, A. (2017, September). International financial spillovers to emerging market economies: How important are economic fundamentals? *Journal of International Money and Finance*, *76*, 133–152. Retrieved 2026-06-17, from https://linkinghub.elsevier.com/retrieve/pii/S026156061730092X doi: 10.1016/j.jimonfin.2017.05.001

Anshu, A., Arunachalam, S., Kuwahara, T., & Soleimanifar, M. (2020). Sample-efficient learning of quantum many-body systems. Retrieved 2026-03-11, from https://arxiv.org/abs/2004.07266 (Version Number: 1) doi: 10.48550/ARXIV.2004.07266

Baaquie, B. E., Corianò, C., & Srikant, M. (2003, April). QUANTUM MECHANICS, PATH INTEGRALS AND OPTION PRICING: REDUCING THE COMPLEXITY OF FINANCE. , 333–339. Retrieved 2026-06-16, from http://www.worldscientific.com/doi/abs/10.1142/9789812704467_0046 doi: 10.1142/9789812704467_0046

Barron, A., & Cover, T. (1991, July). Minimum complexity density estimation. *IEEE Transactions on Information Theory*, *37*(4), 1034–1054. Retrieved 2026-03-26, from http://ieeexplore.ieee.org/document/86996/ doi: 10.1109/18.86996

Benedetti, M., Lloyd, E., Sack, S., & Fiorentini, M. (2019, November). Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, *4*(4), 043001. Retrieved 2026-05-20, from https://iopscience.iop.org/article/10.1088/2058-9565/ab4eb5 doi: 10.1088/2058-9565/ab4eb5

C, A., M, V., & P, P. (2025). A Scoping Survey of Quantum Machine Learning and Deep Learning for Real-World Applications. *Procedia Computer Science*, *258*, 633–646. Retrieved 2026-06-18, from https://linkinghub.elsevier.com/retrieve/pii/S1877050925013997 doi: 10.1016/j.procs.2025.04.297

Chamberlin, R. V., & Lindsay, S. M. (2024, November). Nanothermodynamics: There's Plenty of Room on the Inside. *Nanomaterials*, *14*(22), 1828. Retrieved 2026-05-20, from https://www.mdpi.com/2079-4991/14/22/1828 doi: 10.3390/nano14221828

Chen, Y., Hou, Y., Wang, Z., Wang, T., Wu, Z., Li, Z., & Peng, X. (2025, Febru-

ary). Enhanced natural parameterized quantum circuit. *Physical Review Research*, *7*(1), 013221. Retrieved 2026-05-20, from https://link.aps.org/doi/10.1103/PhysRevResearch.7.013221 doi: 10.1103/PhysRevResearch.7.013221

Cohen, J., Khan, A., & Alexander, C. (2020). Portfolio Optimization of 40 Stocks Using the DWave Quantum Annealer. Retrieved 2026-04-20, from https://arxiv.org/abs/2007.01430 (Version Number: 1) doi: 10.48550/ARXIV.2007.01430

Coyle, B., Henderson, M., Chan Jin Le, J., Kumar, N., Paini, M., & Kashefi, E. (2021, April). Quantum versus classical generative modelling in finance. *Quantum Science and Technology*, *6*(2), 024013. Retrieved 2026-03-26, from https://iopscience.iop.org/article/10.1088/2058-9565/abd3db doi: 10.1088/2058-9565/abd3db

Crosato, E., & Morris, R. G. (2023, September). Shannon entropy and mutual information.

Dao, H. L. (2025, June). Exploring new variational quantum circuit ansatzes for solving SU(2) matrix models. *The European Physical Journal C*, *85*(6), 705. Retrieved 2026-03-11, from https://link.springer.com/10.1140/epjc/s10052-025-14322-7 doi: 10.1140/epjc/s10052-025-14322-7

Datta, S. N., & Hansda, S. (2015, February). Relationship between coupling constants in Heisenberg exchange Hamiltonian and Ising model. *Chemical Physics Letters*, *621*, 102–108. Retrieved 2026-05-20, from https://linkinghub.elsevier.com/retrieve/pii/S0009261415000044 doi: 10.1016/j.cplett.2015.01.001

DeMiguel, V., Garlappi, L., & Uppal, R. (2009, May). Optimal Versus Naive Diversification: How Inefficient is the 1/ *N* Portfolio Strategy? *Review of Financial Studies*, *22*(5), 1915–1953. Retrieved 2026-03-26, from https://academic.oup.com/rfs/article-lookup/doi/10.1093/rfs/hhm075 doi: 10.1093/rfs/hhm075

Dixit, A., & Kumari, A. (2025, September). Variational Principle 2.0: The Variational Quantum Eigensolver.

Fedorov, D. A., Peng, B., Govind, N., & Alexeev, Y. (2022, January). VQE method: a short survey and recent developments. *Materials Theory*, *6*(1), 2. Retrieved 2026-03-11, from https://materialstheory.springeropen.com/articles/10.1186/s41313-021-00032-6 doi: 10.1186/s41313-021-00032-6

Feynman, R. P. (1981, May). Simulating Physics with Computers. *International Journal of Theoretical Physics*, *Vol. 21*.

Fontana, E., Fitzpatrick, N., Ramo, D. M., Duncan, R., & Rungger, I. (2021, August). Evaluating the noise resilience of variational quantum algorithms. *Physical Review A*, *104*(2), 022403. Retrieved 2026-06-23, from https://link.aps.org/doi/10.1103/PhysRevA.104.022403 doi: 10.1103/PhysRevA.104.022403

Freitas, J. C., Pinto, A. A., & Felgueiras, . (2024, August). Game Theory for Predicting Stocks' Closing Prices. *Mathematics*, *12*(17), 2676. Retrieved 2026-03-11, from https://www.mdpi.com/2227-7390/12/17/2676 doi: 10.3390/math12172676

Gabaix, X. (2020, August). A Behavioral New Keynesian Model. *American Economic Review*, *110*(8), 2271–2327. Retrieved 2026-06-16, from https://pubs.aeaweb.org/doi/10.1257/aer.20162005 doi: 10.1257/aer.20162005

Galam, S., & Walliser, B. (2010, February). Ising model versus normal form game. *Physica A: Statistical Mechanics and its Applications*, *389*(3), 481–489. Retrieved 2026-03-11, from https://linkinghub.elsevier.com/retrieve/pii/S0378437109007821 doi: 10.1016/j.physa.2009.09.029

Gong, H., Sedai, A., & Medda, F. (2025). The Quantum Network of Assets: A Non-

Classical Framework for Market Correlation and Structural Risk. Retrieved 2026-03-11, from https://arxiv.org/abs/2511.21515 (Version Number: 2) doi: 10.48550/ARXIV.2511.21515

Hidalgo, E. G. (2006). Quantum Econophysics. Retrieved 2026-03-11, from https://arxiv.org/abs/physics/0609245 (Version Number: 3) doi: 10.48550/ARXIV.PHYSICS/0609245

Ibrahim, A., Kashef, R., Li, M., Valencia, E., & Huang, E. (2020, August). Bitcoin Network Mechanics: Forecasting the BTC Closing Price Using Vector Auto-Regression Models Based on Endogenous and Exogenous Feature Variables. *Journal of Risk and Financial Management*, *13*(9), 189. Retrieved 2026-03-13, from https://www.mdpi.com/1911-8074/13/9/189 doi: 10.3390/jrfm13090189

Innan, N., Saleem, A., Marchisio, A., & Shafique, M. (2025). Quantum Portfolio Optimization with Expert Analysis Evaluation. Retrieved 2026-03-11, from https://arxiv.org/abs/2507.20532 (Version Number: 1) doi: 10.48550/ARXIV.2507.20532

Jiang, Y.-C., Chang, M.-H., Chang, Y.-Y., Wu, K.-M., Chen, P.-C., Tong, Y. F., … Chou, Y.-H. (2023, July). Trend Ratio-Based Portfolio Optimization Model Adopting Entanglement-enhanced Quantum-Inspired Evolutionary Computation in the Global Financial Markets. , 1–9. Retrieved 2026-05-20, from https://ieeexplore.ieee.org/document/10254025/ doi: 10.1109/CEC53210.2023.10254025

John Von Neumann, & Oskar Morgenster. (1953). Theory of Game and Economic Behavior.

Kabelac, A. (2021, November). One- and two-dimensional Ising model.

Kahneman, D., & Tversky, A. (1979, March). Prospect Theory: An Analysis of Decision under Risk. *Econometrica*, *47*(2), 263. Retrieved 2026-03-26, from https://www.jstor.org/stable/1914185?origin=crossref doi: 10.2307/1914185

Kolmogorov, A. N. (1968, January). Three approaches to the quantitative definition of information*. *International Journal of Computer Mathematics*, *2*(1-4), 157–168. Retrieved 2026-03-26, from http://www.tandfonline.com/doi/abs/10.1080/00207166808803030 doi: 10.1080/00207166808803030

Kumar, A. (2025, January). Family of quantum mutual information in multiparty quantum systems. *Physics Letters A*, *529*, 130091. Retrieved 2026-03-26, from https://linkinghub.elsevier.com/retrieve/pii/S0375960124007850 doi: 10.1016/j.physleta.2024.130091

La Torre, D., & Maggistro, R. (2026, January). Multi-agent dynamic financial portfolio management: a differential game approach. *Annals of Operations Research*, *356*(1), 559–580. Retrieved 2026-03-11, from https://link.springer.com/10.1007/s10479-024-06070-w doi: 10.1007/s10479-024-06070-w

Levy, H., & Levy, A. (1991, November). Arrow-Pratt Measures of Risk Aversion: The Multivariate Case. *International Economic Review*, *32*(4), 891. Retrieved 2026-04-20, from https://www.jstor.org/stable/2527041?origin=crossref doi: 10.2307/2527041

Li, H., Liang, J., Wang, Z., & Fei, S. (2023, September). Ising Hamiltonians for Constrained Combinatorial Optimization Problems and the Metropolis-Hastings Warm-Starting Algorithm. *Advanced Quantum Technologies*, *6*(9), 2300101. Retrieved 2026-05-19, from https://advanced.onlinelibrary.wiley.com/doi/10.1002/qute.202300101 doi: 10.1002/qute.202300101

Li, W., Zhang, S.-X., Sheng, Z., Gong, C., Chen, J., & Shuai, Z. (2025). Quantum machine learning of molecular energies with hybrid quantum-neural wavefunction. *Digital Discovery*, *4*(10), 2697–2710. Retrieved 2026-06-18, from https://xlink.rsc.org/?DOI=D5DD00222B doi: 10.1039/D5DD00222B

Lucas, A. (2014). Ising formulations of many NP problems. *Frontiers in Physics*, *2*. Retrieved 2026-03-18, from http://journal.frontiersin.org/article/10.3389/fphy.2014.00005/abstract doi: 10.3389/fphy.2014.00005

Mantegna, R. N., Stanley, H. E., & Chriss, N. A. (2000, December). *An Introduction to Econophysics: Correlations and Complexity in Finance. Physics Today*, *53*(12), 70–70. Retrieved 2026-03-12, from https://pubs.aip.org/physicstoday/article/53/12/70/411205/An-Introduction-to-Econophysics-Correlations-and doi: 10.1063/1.1341926

Markowitz, H. (1952, March). PORTFOLIO SELECTION*. *The Journal of Finance*, *7*(1), 77–91. Retrieved 2026-03-12, from https://onlinelibrary.wiley.com/doi/10.1111/j.1540-6261.1952.tb01525.x doi: 10.1111/j.1540-6261.1952.tb01525.x

McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R., & Neven, H. (2018, November). Barren plateaus in quantum neural network training landscapes. *Nature Communications*, *9*(1), 4812. Retrieved 2026-05-08, from https://www.nature.com/articles/s41467-018-07090-4 doi: 10.1038/s41467-018-07090-4

Menezes, F. M., & Quiggin, J. (2025, April). Ex ante and ex post equilibrium supply curves. *Economic Theory Bulletin*, *13*(1), 79–98. Retrieved 2026-03-12, from https://link.springer.com/10.1007/s40505-024-00282-w doi: 10.1007/s40505-024-00282-w

Mitarai, K., Negoro, M., Kitagawa, M., & Fujii, K. (2018, September). Quantum circuit learning. *Physical Review A*, *98*(3), 032309. Retrieved 2026-03-12, from https://link.aps.org/doi/10.1103/PhysRevA.98.032309 doi: 10.1103/PhysRevA.98.032309

Monderer, D., & Shapley, L. S. (1996, May). Potential Games. *Games and Economic Behavior*, *14*(1), 124–143. Retrieved 2026-03-26, from https://linkinghub.elsevier.com/retrieve/pii/S0899825696900445 doi: 10.1006/game.1996.0044

Mukhopadhyay, A., Saha, T., Sur, S., & Chakraborty, S. (2025, December). Analogies between phase transitions in potential games and quantum phase transitions. *New Journal of Physics*, *27*(12), 123901. Retrieved 2026-05-19, from https://iopscience.iop.org/article/10.1088/1367-2630/ae2356 doi: 10.1088/1367-2630/ae2356

Nakahara, M. (2008, March). Quantum Computing: From Linear Algebra to Physical Realizations. Retrieved 2026-03-15, from https://www.taylorfrancis.com/articles/9781420012293 doi: 10.1201/9781420012293

Navarrete, Y., & Davis, S. (2022, November). Quantum Mutual Information, Fragile Systems and Emergence. *Entropy*, *24*(11), 1676. Retrieved 2026-03-26, from https://www.mdpi.com/1099-4300/24/11/1676 doi: 10.3390/e24111676

Nielsen, M. A., & Chuan, I. L. (2010). Quantum Computation and Quantum Information.

Niss, M. (2005, March). History of the Lenz-Ising Model 1920?1950: From Ferromagnetic to Cooperative Phenomena. *Archive for History of Exact Sciences*, *59*(3), 267–318. Retrieved 2026-05-20, from http://link.springer.com/10.1007/s00407-004-0088-3 doi: 10.1007/s00407-004-0088-3

Ohya, M. (1998). Fundamentals of quantum mutual entropy and capacity. Retrieved 2026-03-11, from https://arxiv.org/abs/quant-ph/9806042 (Version Number: 2) doi: 10.48550/ARXIV.QUANT-PH/9806042

Onnela, J.-P., Chakraborti, A., Kaski, K., Kertész, J., & Kanto, A. (2003, November). Dynamics of market correlations: Taxonomy and portfolio analysis. *Physical Review E*, *68*(5), 056110. Retrieved 2026-03-26, from https://link.aps.org/doi/10.1103/PhysRevE.68.056110 doi: 10.1103/PhysRevE.68.056110

Ortiz Marrero, C., Kieferová, M., & Wiebe, N. (2021, October). Entanglement-Induced Barren Plateaus. *PRX Quantum*, *2*(4), 040316. Retrieved 2026-06-23, from https://link.aps.org/doi/10.1103/PRXQuantum.2.040316 doi: 10.1103/PRXQuantum.2.040316

Padilla, G. (2025, September). Quantum Approximate Optimization Algorithm for Portfolio Selection: A Comparative Study with Classical Markowitz Optimization.

Page, D. N. (1993, December). Information in black hole radiation. *Physical Review Letters*, *71*(23), 3743–3746. Retrieved 2026-06-23, from https://link.aps.org/doi/10.1103/PhysRevLett.71.3743 doi: 10.1103/PhysRevLett.71.3743

Pagni, V., Giachetti, G., Trombettoni, A., & Defenu, N. (2026, January). One-dimensional long-range Ising model: Two almost equivalent approximations. *Physical Review B*, *113*(1), 014406. Retrieved 2026-05-20, from https://link.aps.org/doi/10.1103/2y4k-rvcf doi: 10.1103/2y4k-rvcf

Peruzzo, A., McClean, J., Shadbolt, P., Yung, M.-H., Zhou, X.-Q., Love, P. J., … O'Brien, J. L. (2014, July). A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, *5*(1), 4213. Retrieved 2026-05-20, from https://www.nature.com/articles/ncomms5213 doi: 10.1038/ncomms5213

Preskill, J. (2018, August). Quantum Computing in the NISQ era and beyond. *Quantum*, *2*, 79. Retrieved 2026-03-12, from http://arxiv.org/abs/1801.00862 (arXiv:1801.00862 [quant-ph]) doi: 10.22331/q-2018-08-06-79

Purwanto, A. (2016). Fisika Kuantum.

Rajak, A., Suzuki, S., Dutta, A., & Chakrabarti, B. K. (2023, January). Quantum annealing: an overview. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *381*(2241), 20210417. Retrieved 2026-04-21, from https://royalsocietypublishing.org/doi/10.1098/rsta.2021.0417 doi: 10.1098/rsta.2021.0417

Regadío, A. (2025, June). Exoplanet discovery with variational quantum circuits. *Quantum Machine Intelligence*, *7*(1), 11. Retrieved 2026-06-17, from https://link.springer.com/10.1007/s42484-024-00229-1 doi: 10.1007/s42484-024-00229-1

Reifenstein, S., Kako, S., Khoyratee, F., Leleu, T., & Yamamoto, Y. (2021, November). Coherent Ising Machines with Optical Error Correction Circuits. *Advanced Quantum Technologies*, *4*(11), 2100077. Retrieved 2026-03-12, from https://advanced.onlinelibrary.wiley.com/doi/10.1002/qute.202100077 doi: 10.1002/qute.202100077

Revythi, M., & Koukiou, G. (2025, August). Quantum Machine Learning and Deep Learning: Fundamentals, Algorithms, Techniques, and Real-World Applications. *Machine Learning and Knowledge Extraction*, *7*(3), 75. Retrieved 2026-06-18, from https://www.mdpi.com/2504-4990/7/3/75 doi: 10.3390/make7030075

Sarkar, S., & Benjamin, C. (2018). Quantum Nash equilibrium in the thermodynamic limit. Retrieved 2026-03-11, from https://arxiv.org/abs/1806.07343 (Version

Number: 3) doi: 10.48550/ARXIV.1806.07343

Saslow, W. M. (1999, December). An economic analogy to thermodynamics. *American Journal of Physics*, *67*(12), 1239–1247. Retrieved 2026-03-13, from https://pubs.aip.org/ajp/article/67/12/1239/1044925/An-economic-analogy-to-thermodynamics doi: 10.1119/1.19110

Schultz, T. D., Mattis, D. C., & Lieb, E. H. (1964, July). Two-Dimensional Ising Model as a Soluble Problem of Many Fermions. *Reviews of Modern Physics*, *36*(3), 856–871. Retrieved 2026-05-19, from https://link.aps.org/doi/10.1103/RevModPhys.36.856 doi: 10.1103/RevModPhys.36.856

Schwert, G. W. (2003). Chapter 15 Anomalies and market efficiency. , *1*, 939–974. Retrieved 2026-03-11, from https://linkinghub.elsevier.com/retrieve/pii/S1574010203010240 doi: 10.1016/S1574-0102(03)01024-0

Scursulim, J. V. S., Langeloh, G. M., Beltran, V. L., & Brito, S. (2026, February). Multiclass portfolio optimization via variational quantum Eigensolver with Dicke state ansatz. *Scientific Reports*, *16*(1), 6208. Retrieved 2026-04-20, from https://www.nature.com/articles/s41598-026-36333-4 doi: 10.1038/s41598-026-36333-4

Serafini, A., Illuminati, F., & De Siena, S. (2003). Symplectic invariants, entropic measures and correlations of Gaussian states. Retrieved 2026-03-11, from https://arxiv.org/abs/quant-ph/0307073 (Version Number: 4) doi: 10.48550/ARXIV.QUANT-PH/0307073

Shaikh, M. T., Hamza, M., Ali, S. B., Rafi, M., & Zahid, S. (2025). Feature Selection Using Quantum Annealing: A Mutual Information Based QUBO Approach. Retrieved from https://api.semanticscholar.org/CorpusID:282825737

Shannon, C. E. (1948, October). A Mathematical Theory of Communication. *The Bell System Technical Journal*, *Vol. 27*, pp. 379–423, 623–656.

Sharma, K., Khatri, S., Cerezo, M., & Coles, P. J. (2020, April). Noise resilience of variational quantum compiling. *New Journal of Physics*, *22*(4), 043006. Retrieved 2026-06-23, from https://iopscience.iop.org/article/10.1088/1367-2630/ab784c doi: 10.1088/1367-2630/ab784c

Shin, M., Lee, J., & Jeong, K. (2024, February). Estimating quantum mutual information through a quantum neural network. *Quantum Information Processing*, *23*(2), 57. Retrieved 2026-03-11, from https://link.springer.com/10.1007/s11128-023-04253-1 doi: 10.1007/s11128-023-04253-1

Sikalo, M., Arnaut-Berilo, A., & Zaimovic, A. (2022, March). Efficient Asset Allocation: Application of Game Theory-Based Model for Superior Performance. *International Journal of Financial Studies*, *10*(1), 20. Retrieved 2026-03-11, from https://www.mdpi.com/2227-7072/10/1/20 doi: 10.3390/ijfs10010020

Sim, S., Johnson, P. D., & Aspuru-Guzik, A. (2019, December). Expressibility and Entangling Capability of Parameterized Quantum Circuits for Hybrid Quantum-Classical Algorithms. *Advanced Quantum Technologies*, *2*(12), 1900070. Retrieved 2026-06-04, from https://advanced.onlinelibrary.wiley.com/doi/10.1002/qute.201900070 doi: 10.1002/qute.201900070

Spall, J. C. (1998, July). Implementation of the Simulateous Perturbation Algorithm for Stochastic Optimization. , *34*(3).

Suzuki, K. (2025, February). Spin- S Ising models with multispin interactions on the one-dimensional chain and two-dimensional square lattice. *Physical Review E*,

111(2), 024132. Retrieved 2026-05-19, from https://link.aps.org/doi/10.1103/PhysRevE.111.024132 doi: 10.1103/PhysRevE.111.024132

Szabó, G., & Borsos, I. (2016, April). Evolutionary potential games on lattices. *Physics Reports*, *624*, 1–60. Retrieved 2026-03-26, from https://linkinghub.elsevier.com/retrieve/pii/S0370157316000843 doi: 10.1016/j.physrep.2016.02.006

Tilly, J., Chen, H., Cao, S., Picozzi, D., Setia, K., Li, Y., … Tennyson, J. (2022, November). The Variational Quantum Eigensolver: A review of methods and best practices. *Physics Reports*, *986*, 1–128. Retrieved 2026-06-18, from https://linkinghub.elsevier.com/retrieve/pii/S0370157322003118 doi: 10.1016/j.physrep.2022.08.003

Tomaz, A. A., Mattos, R. S., & Barbatti, M. (2025, December). The quantum measurement problem: a review of recent trends. *Philosophical Magazine*, 1–38. Retrieved 2026-06-21, from https://www.tandfonline.com/doi/full/10.1080/14786435.2025.2601922 doi: 10.1080/14786435.2025.2601922

Wang, A., Lv, Z., Ma, Z., Cai, D., & Zhang, Z. (2025, August). Achieving High-Quality Portfolio Optimization with the Variational Quantum Eigensolver. Retrieved 2026-03-11, from http://arxiv.org/abs/2508.18625 (arXiv:2508.18625 [quant-ph]) doi: 10.48550/arXiv.2508.18625

Wang, S., Wang, P., Li, G., Zhao, S., Zhao, D., Wang, J., … Guo, G.-P. (2025, August). Variational quantum eigensolver with linear depth problem-inspired ansatz for solving portfolio optimization in finance. *Science China Information Sciences*, *68*(8), 180504. Retrieved 2026-03-11, from https://link.springer.com/10.1007/s11432-024-4185-1 doi: 10.1007/s11432-024-4185-1

Wang, Y., Wang, Y., Chen, C., Jiang, R., & Huang, W. (2022, August). Development of variational quantum deep neural networks for image recognition. *Neurocomputing*, *501*, 566–582. Retrieved 2026-06-18, from https://linkinghub.elsevier.com/retrieve/pii/S0925231222007275 doi: 10.1016/j.neucom.2022.06.010

Wei, H., Wang, Y. J., Yang, H., Yang, X., Cao, M., Xu, Q., … Zhao, W. (2025, July). Solving Multiple Discretization Portfolio Optimization Problem with Quantum-Classical Hybrid Algorithms. *Computational Economics*. Retrieved 2026-03-11, from https://link.springer.com/10.1007/s10614-025-11061-5 doi: 10.1007/s10614-025-11061-5

Wierichs, D., Izaac, J., Wang, C., & Lin, C. Y.-Y. (2022, March). General parameter-shift rules for quantum gradients. *Quantum*, *6*, 677. Retrieved 2026-05-20, from https://quantum-journal.org/papers/q-2022-03-30-677/ doi: 10.22331/q-2022-03-30-677

Yang, X., Jiang, A., Jiang, W., Zhao, Y., Tang, E., & Chang, S. (2024, May). Abnormal Detection and Fault Diagnosis of Adjustment Hydraulic Servomotor Based on Genetic Algorithm to Optimize Support Vector Data Description with Negative Samples and One-Dimensional Convolutional Neural Network. *Machines*, *12*(6), 368. Retrieved 2026-03-13, from https://www.mdpi.com/2075-1702/12/6/368 doi: 10.3390/machines12060368

Yang, Y., Rubio, F., Scutari, G., & Palomar, D. (2012). Multi-portfolio Optimization: A Potential Game Approach. , *75*, 182–189. Retrieved 2026-03-26, from http://link.springer.com/10.1007/978-3-642-30373-9_13 (Series Title: Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering) doi: 10.1007/978-3-642-30373-9_13

Halaman ini sengaja dikosongkan

Lampiran A

Operasi Aljabar Linier pada Ruang Hilbert

A.1 Notasi Bra dan Ket

Pada mekanika kuantum, keadaan suatu sistem direpresentasikan sebagai vektor pada ruang Hilbert. Notasi yang umum digunakan untuk menyatakan vektor keadaan adalah notasi bra dan ket yang diperkenalkan oleh Paul Dirac. Sebuah vektor keadaan dinyatakan dalam bentuk ket dan direpresentasikan sebagai vektor kolom

$$|\psi\rangle = \begin{pmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_N \end{pmatrix}, \tag{A.1}$$

dengan $\psi_i \in \mathbb{C}$ menyatakan komponen-komponen kompleks dari vektor keadaan pada ruang Hilbert berdimensi N. Pasangan dual dari ket disebut bra, yang diperoleh melalui operasi sekawan Hermit (Hermitian Adjoint) terhadap ket. Secara matematis, bra didefinisikan sebagai

$$\langle\psi| = \begin{pmatrix} \psi_1^* & \psi_2^* & \cdots & \psi_N^* \end{pmatrix}, \tag{A.2}$$

dengan simbol * menyatakan operasi konjugasi kompleks.

Hubungan antara bra dan ket dapat dituliskan sebagai

$$\langle\psi| = (|\psi\rangle)^\dagger, \tag{A.3}$$

di mana simbol $\dagger$ menyatakan operasi sekawan Hermit. Sebagai contoh, dua keadaan dasar sebuah qubit dapat direpresentasikan dalam notasi bra dan ket sebagai

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{A.4}$$

Representasi bra untuk kedua keadaan tersebut diperoleh dengan mengambil sekawan Hermitnya, yaitu

$$\langle 0| = \begin{pmatrix} 1 & 0 \end{pmatrix}, \qquad \langle 1| = \begin{pmatrix} 0 & 1 \end{pmatrix}. \tag{A.5}$$

Notasi bra dan ket menjadi dasar bagi berbagai operasi aljabar linier pada ruang Hilbert, seperti perkalian dalam, perkalian luar, representasi operator, serta pembentukan matriks densitas yang digunakan dalam komputasi dan informasi kuantum.

A.2 Operasi Perkalian Dalam (*Inner Product*)

Perkalian dalam (*inner product*) merupakan operasi yang memetakan dua vektor pada ruang Hilbert menjadi sebuah skalar kompleks. Operasi ini digunakan untuk menentukan hubungan antara dua keadaan kuantum, termasuk ortogonalitas dan normalisasi suatu vektor keadaan. Untuk dua buah vektor keadaan $|\psi\rangle$ dan $|\phi\rangle$ yang didefinisikan pada ruang Hilbert berdimensi N, perkalian dalam dinyatakan sebagai

$$\langle\psi|\phi\rangle = \begin{pmatrix}\psi_1^* & \psi_2^* & \cdots & \psi_N^*\end{pmatrix}\begin{pmatrix}\phi_1\\ \phi_2\\ \vdots\\ \phi_N\end{pmatrix} = \sum_{i=1}^{N}\psi_i^*\phi_i. \tag{A.6}$$

Salah satu penggunaan penting perkalian dalam adalah untuk menentukan ortogonalitas dua keadaan kuantum. Dua vektor dikatakan ortogonal apabila hasil perkalian dalamnya bernilai nol. Sebagai contoh,

$$\langle 0|1\rangle = \begin{pmatrix}1 & 0\end{pmatrix}\begin{pmatrix}0\\ 1\end{pmatrix} = (1\times 0) + (0\times 1) = 0. \tag{A.7}$$

Selain itu, perkalian dalam juga digunakan untuk memeriksa normalisasi suatu keadaan kuantum. Sebuah vektor keadaan dikatakan ternormalisasi apabila memenuhi

$$\langle\psi|\psi\rangle = 1. \tag{A.8}$$

Sebagai contoh, keadaan dasar *qubit* $|0\rangle$ memenuhi

$$\langle 0|0\rangle = \begin{pmatrix}1 & 0\end{pmatrix}\begin{pmatrix}1\\ 0\end{pmatrix} = (1\times 1) + (0\times 0) = 1. \tag{A.9}$$

Sehingga keadaan $|0\rangle$ merupakan vektor yang telah ternormalisasi.

A.3 Operasi Perkalian Luar (*Outer Product*)

Perkalian luar (*outer product*) bertujuan menghasilkan sebuah matriks atau operator linier yang mana diperoleh melalui perkalian antara sebuah ket dan sebuah bra. Operasi ini banyak digunakan dalam pembentukan operator proyeksi, matriks densitas, dan representasi operator kuantum. Untuk dua buah vektor keadaan $|\psi\rangle$ dan $|\phi\rangle$, perkalian luar didefinisikan sebagai

$$|\psi\rangle\langle\phi| = \begin{pmatrix}\psi_1\\ \psi_2\\ \vdots\\ \psi_N\end{pmatrix}\begin{pmatrix}\phi_1^* & \phi_2^* & \cdots & \phi_N^*\end{pmatrix} = \begin{pmatrix}\psi_1\phi_1^* & \psi_1\phi_2^* & \cdots & \psi_1\phi_N^*\\ \psi_2\phi_1^* & \psi_2\phi_2^* & \cdots & \psi_2\phi_N^*\\ \vdots & \vdots & \ddots & \vdots\\ \psi_N\phi_1^* & \psi_N\phi_2^* & \cdots & \psi_N\phi_N^*\end{pmatrix}. \tag{A.10}$$

Sebagai contoh, perkalian luar antara keadaan $|0\rangle$ dan $\langle 1|$ menghasilkan

$$|0\rangle\langle 1| = \begin{pmatrix}1\\ 0\end{pmatrix}\begin{pmatrix}0 & 1\end{pmatrix} = \begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}. \tag{A.11}$$

Operator tersebut dapat dipandang sebagai operator transisi yang memetakan keadaan $|1\rangle$ menuju $|0\rangle$. Selain itu, terdapat operator proyeksi yang diperoleh dari perkalian suatu keadaan dengan dirinya sendiri. Untuk keadaan $|1\rangle$, diperoleh

$$|1\rangle\langle 1| = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}. \tag{A.12}$$

Operator ini disebut operator proyeksi pada keadaan $|1\rangle$ karena hanya mempertahankan komponen yang sejajar dengan keadaan tersebut.

A.4 Operasi Konjugasi Kompleks dan Adjoin Hermitian

A.4.1 Konjugasi Kompleks

Konjugasi kompleks merupakan operasi yang mengubah tanda bagian imajiner suatu bilangan kompleks. Jika suatu bilangan kompleks dinyatakan sebagai

$$c = a + ib, \tag{A.13}$$

dengan $a, b \in \mathbb{R}$ dan $i = \sqrt{-1}$, maka konjugasi kompleks dari c didefinisikan sebagai

$$c^* = (a + ib)^* = a - ib. \tag{A.14}$$

Sebagai contoh,

$$(3 + 2i)^* = 3 - 2i, \tag{A.15}$$

dan

$$(-1 - 4i)^* = -1 + 4i. \tag{A.16}$$

Pada vektor atau matriks, operasi konjugasi kompleks diterapkan pada setiap elemen secara independen. Sebagai contoh,

$$\begin{pmatrix} 1 + i \\ 2 - 3i \end{pmatrix}^* = \begin{pmatrix} 1 - i \\ 2 + 3i \end{pmatrix}. \tag{A.17}$$

A.4.2 Transpose dan Sekawan Hermit

Untuk suatu matriks A, transpose didefinisikan sebagai pertukaran baris menjadi kolom sehingga diperoleh A^T.Sebagai contoh,

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \quad \Rightarrow \quad A^T = \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}. \tag{A.18}$$

Adjoin Hermitian merupakan kombinasi antara operasi transpose dan konjugasi kompleks yang didefinisikan sebagai

$$A^\dagger = (A^*)^T. \tag{A.19}$$

Sebagai contoh,

$$A = \begin{pmatrix} 1 + i & 2 \\ 3i & 4 - i \end{pmatrix}, \tag{A.20}$$

maka

$$A^* = \begin{pmatrix} 1-i & 2 \\ -3i & 4+i \end{pmatrix}, \tag{A.21}$$

dan

$$A^\dagger = \begin{pmatrix} 1-i & -3i \\ 2 & 4+i \end{pmatrix}. \tag{A.22}$$

A.5 Operasi *Trace* Tereduksi

Misalkan sistem kuantum terdiri atas dua subsistem A dan B dengan matriks densitas gabungan ρ_{AB}. Matriks densitas tereduksi untuk subsistem A diperoleh melalui operasi

$$\rho_A = \mathrm{Tr}_B(\rho_{AB}), \tag{A.23}$$

sedangkan matriks densitas tereduksi untuk subsistem B diberikan oleh

$$\rho_B = \mathrm{Tr}_A(\rho_{AB}). \tag{A.24}$$

Secara umum, *trace* tereduksi terhadap subsistem B didefinisikan sebagai

$$\rho_A = \sum_j \left(I_A \otimes \langle j_B|\right) \rho_{AB} \left(I_A \otimes |j_B\rangle\right), \tag{A.25}$$

dengan $\{|j_B\rangle\}$ merupakan basis ortonormal pada subsistem B dan I_A adalah operator identitas pada subsistem A.

A.5.1 Langkah Perhitungan *Trace* Tereduksi

Untuk sistem dua qubit, basis komputasional yang umum digunakan adalah

$$\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}. \tag{A.26}$$

Apabila akan dihitung matriks densitas tereduksi ρ_A, maka langkah perhitungannya dapat diringkas sebagai berikut:

1. Bentuk matriks densitas gabungan ρ_{AB}.
2. Pilih basis ortonormal pada subsistem yang akan ditelusuri.
3. Hitung kontribusi setiap basis menggunakan Persamaan (A.25).
4. Jumlahkan seluruh kontribusi untuk memperoleh matriks densitas tereduksi.

A.5.2 Contoh pada Keadaan Bell

Kita tinjau keadaan Bell

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}} \left(|00\rangle + |11\rangle\right). \tag{A.27}$$

Matriks densitasnya diperoleh dari perkalian luar

$$\rho_{AB} = |\Phi^+\rangle\langle\Phi^+|, \tag{A.28}$$

sehingga

$$\rho_{AB} = \frac{1}{2}\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}. \tag{A.29}$$

Untuk memperoleh keadaan subsistem A, dilakukan *partial trace* terhadap subsistem B

$$\rho_A = \mathrm{Tr}_B(\rho_{AB}). \tag{A.30}$$

Dengan menggunakan basis $\{|0\rangle_B, |1\rangle_B\}$ diperoleh

$$\rho_A = (I_A \otimes \langle 0|)\rho_{AB}(I_A \otimes |0\rangle) + (I_A \otimes \langle 1|)\rho_{AB}(I_A \otimes |1\rangle), \tag{A.31}$$

yang menghasilkan

$$\rho_A = \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{A.32}$$

Karena Bell state bersifat simetris terhadap kedua qubit, maka matriks densitas tereduksi untuk subsistem B memiliki bentuk yang sama

$$\rho_B = \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{A.33}$$

Hasil ini menunjukkan bahwa meskipun keadaan gabungan $|\Phi^+\rangle$ merupakan keadaan murni (*pure state*), masing-masing subsistem berada pada keadaan campuran maksimum (*maximally mixed state*).

A.6 Sifat Dasar Operator Linier

A.6.1 Linearitas dan Sifat Distributif

Suatu operator A dikatakan linier apabila memenuhi sifat penjumlahan vektor dan perkalian skalar. Untuk dua buah keadaan $|\psi_1\rangle$ dan $|\psi_2\rangle$ serta skalar kompleks c_1 dan c_2, berlaku

$$A\left(c_1|\psi_1\rangle + c_2|\psi_2\rangle\right) = c_1 A|\psi_1\rangle + c_2 A|\psi_2\rangle. \tag{A.34}$$

Sebagai contoh, apabila

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{A.35}$$

maka aksi operator A terhadap keadaan tersebut diberikan oleh

$$A|\psi\rangle = \alpha A|0\rangle + \beta A|1\rangle. \tag{A.36}$$

Selain linearitas, terdapat sifat distributif. Untuk operator A dan dua buah vektor keadaan $|\psi\rangle$ serta $|\phi\rangle$, berlaku

$$A\left(|\psi\rangle + |\phi\rangle\right) = A|\psi\rangle + A|\phi\rangle. \tag{A.37}$$

A.6.2 Komutator dan Non-Komutativitas

Pada umumnya, dua operator kuantum tidak selalu dapat dipertukarkan urutan operasinya. Untuk memeriksa hal tersebut digunakan komutator yang didefinisikan sebagai

$$[A, B] = AB - BA. \tag{A.38}$$

Jika

$$[A, B] = 0, \tag{A.39}$$

maka operator A dan B dikatakan komutatif. Sebaliknya, apabila

$$[A, B] \neq 0, \tag{A.40}$$

maka kedua operator bersifat non-komutatif. Sebagai contoh, operator Pauli-X dan Pauli-Z diberikan oleh

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{A.41}$$

Perkalian kedua operator tersebut menghasilkan

$$XZ = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \tag{A.42}$$

sedangkan

$$ZX = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}. \tag{A.43}$$

Karena

$$XZ \neq ZX, \tag{A.44}$$

maka

$$[X, Z] = XZ - ZX \neq 0. \tag{A.45}$$

Hasil ini menunjukkan bahwa keberurutan penerapan operator kuantum dapat memengaruhi keadaan akhir sistem.

Lampiran B

Penurunan Matriks Umum Gerbang Uniter Satu Qubit

Operator uniter U didefinisikan sebagai matriks kompleks berukuran dua kali dua yang dinotasikan dengan persamaan

$$U = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix}. \tag{B.1}$$

Karena operator tersebut merupakan representasi dari elemen grup khusus uniter $SU(2)$, maka syarat keuniteran matriks dan nilai determinan satu harus dipenuhi. Syarat uniter tersebut menyatakan bahwa perkalian antara konjugat Hermitian dari matriks tersebut dengan matriks itu sendiri menghasilkan matriks identitas kuantum dua dimensi dengan bentuk matematis

$$U^{\dagger}U = I, \tag{B.2}$$

$$\det(U) = 1. \tag{B.3}$$

Sekawan Hermit dari matriks kompleks uniter tersebut yang disimbolkan dengan $U^{\dagger}$ dan dinyatakan melalui bentuk persamaan

$$U^{\dagger} = \begin{pmatrix} u_{11}^{*} & u_{21}^{*} \\ u_{12}^{*} & u_{22}^{*} \end{pmatrix}. \tag{B.4}$$

Dengan menyubstitusikan persamaan (B.1) dan (B.4) ke dalam syarat keuniteran (B.2), maka perkalian matriks tersebut akan menghasilkan beberapa hubungan elemen internal. Hubungan tersebut diperoleh melalui perkalian baris terhadap kolom yang menghasilkan persamaan kuadrat mutlak dari masing-masing elemen matriks pada baris pertama dan kedua. Hubungan tersebut dapat dinyatakan dalam bentuk matematis

$$|u_{11}|^2 + |u_{21}|^2 = 1, \tag{B.5}$$

$$|u_{12}|^2 + |u_{22}|^2 = 1, \tag{B.6}$$

$$u_{11}^{*}u_{12} + u_{21}^{*}u_{22} = 0. \tag{B.7}$$

Persamaan (B.7) menyatakan bahwa kolom pertama dan kolom kedua dari matriks uniter tersebut harus saling ortogonal satu sama lain.

Berdasarkan syarat kekekalan probabilitas pada persamaan (B.5), elemen-elemen pada kolom pertama dapat diparameterisasikan menggunakan fungsi trigonometri dengan

amplitudo polar θ dan fase kompleks. Parameterisasi ini dilakukan dengan mendefinisikan elemen matriks u_{11} dan u_{21} menggunakan koordinat lingkaran yang dimodifikasi oleh parameter fase independen γ_1 dan γ_2 sebagai berikut:

$$u_{11} = e^{i\gamma_1} \cos\theta, \tag{B.8}$$

$$u_{21} = e^{i\gamma_2} \sin\theta. \tag{B.9}$$

Secara geometris, representasi kolom pertama tersebut bertujuan untuk memetakan posisi keadaan qubit pada proyeksi lingkaran satuan di bawah koordinat bola Bloch. Parameter θ menentukan orientasi sudut terhadap kutub bola, sedangkan fase γ_1 dan γ_2 mengatur pergeseran fase kompleks. Dengan demikian, kolom pertama dari matriks uniter U dapat dinyatakan secara formal sebagai vektor kolom

$$\begin{pmatrix} u_{11} \\ u_{21} \end{pmatrix} = \begin{pmatrix} e^{i\gamma_1} \cos\theta \\ e^{i\gamma_2} \sin\theta \end{pmatrix}. \tag{B.10}$$

Untuk menentukan elemen-elemen pada kolom kedua dari matriks U, dapat dimisalkan suatu representasi variabel bebas berupa komponen a dan b.

$$\begin{pmatrix} u_{12} \\ u_{22} \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}. \tag{B.11}$$

Dengan menggunakan syarat ortogonalitas kolom yang dinyatakan pada persamaan (B.7), substitusi dari parameterisasi sebelumnya menghasilkan hubungan linear

$$e^{-i\gamma_1} \cos\theta \cdot a + e^{-i\gamma_2} \sin\theta \cdot b = 0. \tag{B.12}$$

Salah satu solusi umum yang dapat memenuhi persamaan (B.12) dapat dinyatakan dengan menyertakan parameter fase tambahan δ pada komponen tersebut adalah

$$a = -e^{i\delta} e^{i\gamma_1} \sin\theta \tag{B.13}$$

dan

$$b = e^{i\delta} e^{i\gamma_2} \cos\theta. \tag{B.14}$$

Hal ini dapat dibuktikan dengan menyubstitusikan persamaan (B.13) dan (B.14) ke dalam persamaan (B.12) yang menghasilkan

$$e^{-i\gamma_1} \cos\theta \left(-e^{i\delta} e^{i\gamma_1} \sin\theta\right) + e^{-i\gamma_2} \sin\theta \left(e^{i\delta} e^{i\gamma_2} \cos\theta\right) = 0. \tag{B.15}$$

Dengan menyubstitusikan seluruh parameter solusi kolom pertama dan kedua, diperoleh matriks uniter dengan bentuk parametrisasi fase kompleks dengan bentuk umum

$$U = \begin{pmatrix} e^{i\gamma_1} \cos\theta & -e^{i\delta} e^{i\gamma_1} \sin\theta \\ e^{i\gamma_2} \sin\theta & e^{i\delta} e^{i\gamma_2} \cos\theta \end{pmatrix}. \tag{B.16}$$

Untuk menyederhanakan bentuk representasi matriks tersebut, faktor fase global sebesar $e^{i(\gamma_1+\gamma_2)/2}$ dapat dikeluarkan dari setiap elemen di dalam matriks tersebut. Sehingga dapat didefinisikan dua buah variabel fase baru α dan β yang mewakili kombinasi penjumlahan dan pengurangan fase awal

$$\alpha = \frac{\gamma_1 + \gamma_2}{2}, \tag{B.17}$$

$$\beta = \frac{\gamma_1 - \gamma_2}{2}. \tag{B.18}$$

Berdasarkan definisi fase baru tersebut, bentuk eksponensial fase γ_1 dan γ_2 dapat ditulis ulang ke dalam kombinasi parameter fase sehingga berbentuk

$$e^{i\gamma_1} = e^{i\alpha}e^{i\beta}, \tag{B.19}$$

$$e^{i\gamma_2} = e^{i\alpha}e^{-i\beta}. \tag{B.20}$$

Hubungan eksponensial fase tersebut dapat disubstitusikan ke dalam struktur matriks U sehingga menghasilkan formulasi dengan representasi fase global terpisah berbentuk

$$U = e^{i\alpha}\begin{pmatrix} e^{i\beta}\cos\theta & -e^{i(\delta+\beta)}\sin\theta \\ e^{-i\beta}\sin\theta & e^{i(\delta-\beta)}\cos\theta \end{pmatrix}. \tag{B.21}$$

Untuk menyempurnakan bentuk uniter, langkah selanjutnya yaitu menerapkan syarat nilai determinan sama dengan satu dari grup khusus uniter $SU(2)$. Nilai determinan dapat dihitung dengan mengalikan elemen diagonal utama lalu dikurangi hasil kali elemen-elemen pada diagonal yang dalam bentuk matematis menjadi

$$\det(U) = e^{i(2\alpha+\delta)}. \tag{B.22}$$

Karena nilai determinan dari elemen grup khusus ini harus bernilai satu, maka diperoleh hubungan fase kuantum baru sebesar $2\alpha + \delta = 2\pi n$, dengan n sebagai bilangan bulat ($n \in \mathbb{Z}$). Dari penyederhanaan tersebut, hubungan antara parameter fase δ dan α dapat didefinisikan melalui hubungan matematis $\delta = -2\alpha$. Setelah memasukkan nilai fase δ dan mendefinisikan ulang variabel fase $\gamma = \delta + \beta$ ke dalam persamaan (B.21), diperoleh representasi matriks uniter paling umum untuk gerbang satu *qubit* dengan bentuk

$$U = e^{i\alpha}\begin{pmatrix} e^{i\beta}\cos\theta & e^{i\gamma}\sin\theta \\ -e^{-i\gamma}\sin\theta & e^{-i\beta}\cos\theta \end{pmatrix}. \tag{B.23}$$

Halaman ini sengaja dikosongkan

Lampiran C

Penurunan Operator Rotasi SU(2) dari Ekspansi Taylor

Operator rotasi pada ruang Hilbert dua dimensi didefinisikan sebagai

$$U(\theta, \hat{n}) = e^{-i\frac{\theta}{2}(\hat{n}\cdot\vec{\sigma})}, \tag{C.1}$$

dengan

$$\hat{n}\cdot\vec{\sigma} = n_x X + n_y Y + n_z Z, \tag{C.2}$$

di mana $\hat{n} = (n_x, n_y, n_z)$ merupakan vektor satuan yang memenuhi

$$n_x^2 + n_y^2 + n_z^2 = 1, \tag{C.3}$$

untuk normalisasi radius bernilai 1 dan $\vec{\sigma} = (X, Y, Z)$ menyatakan vektor matriks Pauli. Untuk memperoleh bentuk tertutup dari operator tersebut, maka perlu dihitung kuadrat dari operator $\hat{n}\cdot\vec{\sigma}$ terlebih dahulu menggunakan sifat distributif karena n_x, n_y, dan n_z merupakan skalar bernilai real. Ketiga koefisien tersebut akan bersifat komutatif terhadap sesama skalar maupun terhadap operator matriks Pauli sehingga berlaku

$$n_i\sigma_j = \sigma_j n_i, \qquad n_i n_j = n_j n_i, \tag{C.4}$$

untuk setiap $i, j \in \{x, y, z\}$. Namun sifat komutatif tersebut tidak berlaku pada matriks Pauli sehingga secara umum

$$\sigma_i\sigma_j \neq \sigma_j\sigma_i, \qquad i \neq j. \tag{C.5}$$

Oleh karena itu, hanya koefisien skalar yang dipertukarkan, sedangkan urutan perkalian matriks Pauli tetap dipertahankan. Dengan menggunakan sifat distributif diperoleh

$$\begin{aligned}(\hat{n}\cdot\vec{\sigma})^2 &= (n_x X + n_y Y + n_z Z)(n_x X + n_y Y + n_z Z)\\ &= n_x X\, n_x X + n_x X\, n_y Y + n_x X\, n_z Z\\ &\quad + n_y Y\, n_x X + n_y Y\, n_y Y + n_y Y\, n_z Z\\ &\quad + n_z Z\, n_x X + n_z Z\, n_y Y + n_z Z\, n_z Z.\end{aligned} \tag{C.6}$$

Karena koefisien n_x, n_y, dan n_z merupakan skalar, maka seluruh koefisien tersebut dapat dipindahkan ke bagian depan tanpa mengubah urutan perkalian matriks Pauli, sehingga

$$\begin{aligned}(\hat{n}\cdot\vec{\sigma})^2 &= n_x^2 XX + n_x n_y XY + n_x n_z XZ\\ &\quad + n_x n_y YX + n_y^2 YY + n_y n_z YZ\\ &\quad + n_x n_z ZX + n_y n_z ZY + n_z^2 ZZ.\end{aligned} \tag{C.7}$$

Selanjutnya suku-suku yang memiliki koefisien sama dapat dikelompokkan sehingga diperoleh

$$\begin{aligned}(\hat{n}\cdot\vec{\sigma})^2 = n_x^2X^2 + n_y^2Y^2 + n_z^2Z^2 \\ + n_xn_y(XY+YX) \\ + n_xn_z(XZ+ZX) \\ + n_yn_z(YZ+ZY).\end{aligned} \tag{C.8}$$

Lalu dengan sifat-sifat dasar matriks Pauli, Relasi antikomutator tersebut diperoleh dari hasil perkalian masing-masing matriks Pauli. Sebagai contoh,

$$\begin{aligned}XY &= \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}\begin{pmatrix}0 & -i\\ i & 0\end{pmatrix} \\ &= \begin{pmatrix}i & 0\\ 0 & -i\end{pmatrix} = iZ,\end{aligned} \tag{C.9}$$

sedangkan

$$\begin{aligned}YX &= \begin{pmatrix}0 & -i\\ i & 0\end{pmatrix}\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix} \\ &= \begin{pmatrix}-i & 0\\ 0 & i\end{pmatrix} = -iZ.\end{aligned} \tag{C.10}$$

Sehingga diperoleh

$$XY + YX = iZ - iZ = 0. \tag{C.11}$$

Melalui cara yang sama diperoleh pula

$$XZ = -iY, \qquad ZX = iY, \tag{C.12}$$

sehingga

$$XZ + ZX = 0, \tag{C.13}$$

serta

$$YZ = iX, \qquad ZY = -iX, \tag{C.14}$$

yang menghasilkan

$$YZ + ZY = 0. \tag{C.15}$$

Seluruh suku silang pada persamaan sebelumnya akan saling meniadakan akibat sifat antikomutasi matriks Pauli.

$$X^2 = Y^2 = Z^2 = I, \tag{C.16}$$

dan relasi antikomutator

$$XY + YX = 0, \tag{C.17}$$

$$XZ + ZX = 0, \tag{C.18}$$

$$YZ + ZY = 0, \tag{C.19}$$

maka diperoleh

$$\begin{aligned}(\hat{n}\cdot\vec{\sigma})^2 &= n_x^2I + n_y^2I + n_z^2I \\ &= (n_x^2 + n_y^2 + n_z^2)I.\end{aligned} \tag{C.20}$$

Karena $\hat{n}$ merupakan vektor satuan yang mana memenuhi

$$n_x^2 + n_y^2 + n_z^2 = 1, \tag{C.21}$$

maka

$$(\hat{n}\cdot\vec{\sigma})^2 = I. \tag{C.22}$$

Seluruh pangkat genap operator tersebut akan selalu bernilai I, sedangkan seluruh pangkat ganjil akan selalu kembali menjadi $\hat{n}\cdot\vec{\sigma}$. Dari sifat tersebut, deret Taylor dari operator eksponensial dapat dipisahkan menjadi deret fungsi kosinus dan sinus. Sehingga seluruh pangkat genap dan ganjil dari operator tersebut dapat dituliskan sebagai

$$(\hat{n}\cdot\vec{\sigma})^{2k} = I, \tag{C.23}$$

dan

$$(\hat{n}\cdot\vec{\sigma})^{2k+1} = \hat{n}\cdot\vec{\sigma}, \tag{C.24}$$

dengan $k = 0, 1, 2, \ldots$.

Operator eksponensial yang digunakan dalam rumus rotasi diekspansi menggunakan deret Taylor,

$$e^{-i\frac{\theta}{2}(\hat{n}\cdot\vec{\sigma})} = \sum_{m=0}^{\infty} \frac{\left(-i\frac{\theta}{2}(\hat{n}\cdot\vec{\sigma})\right)^m}{m!}. \tag{C.25}$$

Deret tersebut kemudian dipisahkan menjadi suku berpangkat genap dan ganjil,

$$\begin{aligned} e^{-i\frac{\theta}{2}(\hat{n}\cdot\vec{\sigma})} = &\sum_{k=0}^{\infty} \frac{\left(-i\frac{\theta}{2}\right)^{2k}(\hat{n}\cdot\vec{\sigma})^{2k}}{(2k)!} \\ &+ \sum_{k=0}^{\infty} \frac{\left(-i\frac{\theta}{2}\right)^{2k+1}(\hat{n}\cdot\vec{\sigma})^{2k+1}}{(2k+1)!}. \end{aligned} \tag{C.26}$$

Jika disubstitusikan kembali identitas pangkat genap dan ganjil, maka akan diperoleh

$$e^{-i\frac{\theta}{2}(\hat{n}\cdot\vec{\sigma})} = I\sum_{k=0}^{\infty} \frac{\left(-i\frac{\theta}{2}\right)^{2k}}{(2k)!} + (\hat{n}\cdot\vec{\sigma})\sum_{k=0}^{\infty} \frac{\left(-i\frac{\theta}{2}\right)^{2k+1}}{(2k+1)!}. \tag{C.27}$$

Lalu, karena

$$(-i)^{2k} = (-1)^k, \tag{C.28}$$

dan

$$(-i)^{2k+1} = -i(-1)^k, \tag{C.29}$$

maka

$$\begin{aligned} e^{-i\frac{\theta}{2}(\hat{n}\cdot\vec{\sigma})} = &I\sum_{k=0}^{\infty} \frac{(-1)^k(\theta/2)^{2k}}{(2k)!} \\ &- i(\hat{n}\cdot\vec{\sigma})\sum_{k=0}^{\infty} \frac{(-1)^k(\theta/2)^{2k+1}}{(2k+1)!}. \end{aligned} \tag{C.30}$$

Bentuk tersebut dapat dikenali bahwa kedua deret tersebut merupakan definisi dari fungsi kosinus dan sinus,

$$\cos x = \sum_{k=0}^{\infty} \frac{(-1)^k x^{2k}}{(2k)!}, \tag{C.31}$$

$$\sin x = \sum_{k=0}^{\infty} \frac{(-1)^k x^{2k+1}}{(2k+1)!}. \tag{C.32}$$

Dengan mengambil $x = \theta/2$, maka diperoleh

$$e^{-i\frac{\theta}{2}(\hat{n}\cdot\vec{\sigma})} = I\cos\left(\frac{\theta}{2}\right) - i(\hat{n}\cdot\vec{\sigma})\sin\left(\frac{\theta}{2}\right). \tag{C.33}$$

Sehingga dengan mensubstitusikan kembali persamaan (C.2), maka bentuk eksplisit operator rotasi menjadi

$$U(\theta,\hat{n}) = \cos\left(\frac{\theta}{2}\right) I - i\sin\left(\frac{\theta}{2}\right)(n_x X + n_y Y + n_z Z). \tag{C.34}$$

Lampiran D

Penurunan Diagonalisasi Operator Densitas dan Fungsi Logaritma Operator

Operator densitas ρ merupakan operator Hermitian sehingga memenuhi

$$\rho^\dagger = \rho. \tag{D.1}$$

Berdasarkan Teorema Spektral, setiap operator Hermitian memiliki himpunan nilai eigen real beserta vektor-vektor eigen ortonormal yang memenuhi persamaan eigen

$$\rho|e_j\rangle = \lambda_j|e_j\rangle, \tag{D.2}$$

dengan λ_j merupakan nilai eigen dan $|e_j\rangle$ merupakan vektor eigen ke-j. Ortonormalitas seluruh vektor eigen dinyatakan sebagai

$$\langle e_i|e_j\rangle = \delta_{ij}. \tag{D.3}$$

Seluruh vektor eigen kemudian disusun sebagai kolom-kolom suatu matriks

$$U = \begin{bmatrix} |e_1\rangle & |e_2\rangle & \cdots & |e_n\rangle \end{bmatrix}. \tag{D.4}$$

Karena seluruh vektor eigen saling ortonormal, maka berlaku

$$U^\dagger U = I. \tag{D.5}$$

Dengan mengalikan operator densitas terhadap matriks U, maka diperoleh

$$\begin{aligned} \rho U &= \rho \begin{bmatrix} |e_1\rangle & |e_2\rangle & \cdots & |e_n\rangle \end{bmatrix} \\ &= \begin{bmatrix} \rho|e_1\rangle & \rho|e_2\rangle & \cdots & \rho|e_n\rangle \end{bmatrix}. \end{aligned} \tag{D.6}$$

Sesuai dengan persamaan (D.2), diperoleh

$$\begin{aligned} \rho U &= \begin{bmatrix} \lambda_1|e_1\rangle & \lambda_2|e_2\rangle & \cdots & \lambda_n|e_n\rangle \end{bmatrix} \\ &= \begin{bmatrix} |e_1\rangle & |e_2\rangle & \cdots & |e_n\rangle \end{bmatrix} \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}. \end{aligned} \tag{D.7}$$

Matriks diagonal tersebut dapat dinotasikan sebagai

$$D = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n), \tag{D.8}$$

agar diperoleh hubungan

$$\rho U = UD. \tag{D.9}$$

Kemudian kedua ruas dikalikan dengan $U^\dagger$ sehingga diperoleh

$$\rho U U^\dagger = U D U^\dagger. \tag{D.10}$$

Karena $UU^\dagger = I$, maka diperoleh bentuk diagonalisasi operator

$$\rho = U D U^\dagger. \tag{D.11}$$

Untuk memperoleh bentuk logaritma operator densitas, terlebih dahulu ditinjau deret Maclaurin dari fungsi logaritma

$$\log_2(1+x) = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n \ln 2} x^n, \tag{D.12}$$

$$\log_2(x) = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n \ln 2} (x-1)^n. \tag{D.13}$$

Karena operator densitas merupakan operator Hermitian yang telah didiagonalisasi sebagaimana ditunjukkan pada persamaan (D.11), maka deret Maclaurin fungsi logaritma dapat diterapkan pada operator tersebut melalui *functional calculus*. Maka dari itu, diperoleh representasi deret logaritma operator

$$\log_2(\rho) = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n \ln 2} (\rho - I)^n. \tag{D.14}$$

Karena operator densitas telah didiagonalisasi sebagaimana disajikan pada persamaan (D.11),maka

$$\begin{aligned} (\rho - I)^n &= (UDU^\dagger - I)^n \\ &= \left(U(D-I)U^\dagger\right)^n. \end{aligned} \tag{D.15}$$

Dengan menggunakan sifat $U^\dagger U = I$, diperoleh

$$\begin{aligned} (\rho - I)^n &= U(D-I)\underbrace{U^\dagger U}_{I}(D-I)\cdots\underbrace{U^\dagger U}_{I}(D-I)U^\dagger \\ &= U(D-I)^n U^\dagger. \end{aligned} \tag{D.16}$$

Karena

$$D = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}, \tag{D.17}$$

maka

$$(D-I)^n = \begin{pmatrix} (\lambda_1-1)^n & 0 & \cdots & 0 \\ 0 & (\lambda_2-1)^n & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & (\lambda_n-1)^n \end{pmatrix}. \tag{D.18}$$

Substitusi hasil tersebut ke persamaan (D.14) sehingga menghasilkan

$$\begin{aligned} \log_2(\rho) &= \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n\ln 2} U(D-I)^n U^\dagger \\ &= U\left[\sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n\ln 2}(D-I)^n\right] U^\dagger. \end{aligned} \tag{D.19}$$

Karena matriks U dan $U^\dagger$ tidak bergantung pada indeks penjumlahan n, kedua matriks tersebut dapat dikeluarkan dari operator penjumlahan. Sementara itu, matriks di dalam tanda kurung tetap berbentuk diagonal sehingga fungsi logaritma cukup diterapkan pada setiap elemen diagonalnya.

$$\log_2(\rho) = U \begin{pmatrix} \log_2(\lambda_1) & 0 & \cdots & 0 \\ 0 & \log_2(\lambda_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \log_2(\lambda_n) \end{pmatrix} U^\dagger. \tag{D.20}$$

Untuk memperoleh bentuk eksplisit dari hasil perkalian pada persamaan di atas, ditinjau terlebih dahulu suatu matriks diagonal umum

$$A = \mathrm{diag}(a_1, a_2, \ldots, a_n), \tag{D.21}$$

dengan a_j merupakan suatu konstanta. Untuk memperoleh bentuk eksplisit dari hasil perkalian pada persamaan (D.20), ditinjau terlebih dahulu suatu matriks diagonal umum

$$\begin{aligned} UA &= \begin{bmatrix} |e_1\rangle & |e_2\rangle & \cdots & |e_n\rangle \end{bmatrix} \begin{pmatrix} a_1 & 0 & \cdots & 0 \\ 0 & a_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_n \end{pmatrix} \\ &= \begin{bmatrix} a_1|e_1\rangle & a_2|e_2\rangle & \cdots & a_n|e_n\rangle \end{bmatrix}. \end{aligned} \tag{D.22}$$

Sementara itu, konjugat Hermitian dari matriks U diberikan oleh

$$U^\dagger = \begin{bmatrix} \langle e_1| \\ \langle e_2| \\ \vdots \\ \langle e_n| \end{bmatrix}. \tag{D.23}$$

Sehingga bentuk penuh dari perkalian matriks diagonal tersebut adalah

$$\begin{aligned} UAU^\dagger &= \begin{bmatrix} a_1|e_1\rangle & a_2|e_2\rangle & \cdots & a_n|e_n\rangle \end{bmatrix} \begin{bmatrix} \langle e_1| \\ \langle e_2| \\ \vdots \\ \langle e_n| \end{bmatrix} \\ &= a_1|e_1\rangle\langle e_1| + a_2|e_2\rangle\langle e_2| + \cdots + a_n|e_n\rangle\langle e_n| \\ &= \sum_{j=1}^{n} a_j|e_j\rangle\langle e_j|. \end{aligned} \tag{D.24}$$

Dengan mengambil

$$a_j = \log_2(\lambda_j),$$

maka berdasarkan persamaan (D.20) diperoleh

$$\log_2(\rho) = \sum_{j=1}^{n} \log_2(\lambda_j)|e_j\rangle\langle e_j|. \tag{D.25}$$

Karena operator densitas bersifat semidefinit positif, seluruh nilai eigennya memenuhi

$$0 \leq \lambda_j \leq 1,$$

dengan

$$\sum_{j=1}^{n} \lambda_j = 1.$$

Apabila terdapat nilai eigen $\lambda_j = 0$, fungsi $\log_2(\lambda_j)$ memang tidak terdefinisi pada titik tersebut. Akan tetapi, dalam Entropi Von Neumann yang muncul adalah hasil kali $\lambda_j \log_2(\lambda_j)$ sehingga nilainya didefinisikan melalui limit. Akan tetapi, pada Entropi Von Neumann suku yang muncul adalah hasil kali $\lambda_j \log_2(\lambda_j)$ sehingga nilainya didefinisikan melalui limit

$$\lim_{x \to 0^+} x \log_2 x = 0, \tag{D.26}$$

sehingga keberadaan nilai eigen nol tidak menimbulkan singularitas pada perhitungan Entropi Von Neumann.

Lampiran E

Data Harga Harian Saham

Lampiran ini menyajikan dataset *adjusted closing price* atau harga penutupan aset yang disesuaikan untuk empat aset utama (BBCA, ADRO, SMGR, TLKM) yang digunakan selama periode simulasi.

Tabel E.1: Data Harga Harian Saham Gabungan (BBCA, ADRO, SMGR, TLKM)

Tanggal	BBCA	ADRO	SMGR	TLKM
2020-06-02	4515.4961	328.1817	8238.6943	2381.0249
2020-06-03	4869.3228	333.9141	8217.5156	2410.3298
2020-06-04	4877.7471	333.6549	8132.7979	2417.6560
2020-06-05	4822.9883	327.6968	7984.5444	2366.3728
2020-06-08	4970.4165	345.5712	8598.7402	2366.3728
2020-06-09	4890.3833	351.5293	8005.7236	2300.4365
2020-06-10	4886.1714	327.6968	7815.1108	2278.4578
2020-06-11	4785.0791	318.7596	8005.7236	2256.4792
2020-06-12	4776.6538	315.7805	7963.3652	2219.8479
2020-06-15	4633.4395	303.8643	7984.5444	2263.8054
2020-06-16	4852.4736	320.2490	8238.6943	2344.3936
2020-06-17	4818.7764	318.7596	8153.9780	2351.7202
2020-06-18	4705.0469	306.8434	7899.8276	2403.0034
2020-06-19	4696.6221	309.8224	8048.0815	2403.0034
2020-06-22	4667.1362	306.8434	8111.6201	2344.3936
2020-06-23	4734.5317	299.3957	8090.4404	2293.1104
2020-06-24	4806.1396	311.3120	8026.9023	2329.7412
2020-06-25	4822.9883	299.3957	8005.7236	2315.0886
2020-06-26	4755.5933	302.3748	8048.0815	2337.0676
2020-06-29	4780.8662	299.3957	8026.9023	2329.7412
2020-06-30	4797.7148	296.4166	8188.8350	2348.2717
2020-07-01	4886.1714	311.3120	8210.1035	2340.5728
2020-07-02	4945.1431	312.8015	8188.8350	2417.5654
2020-07-03	4945.1431	309.8224	8316.4521	2402.1670
2020-07-06	4999.9019	317.2701	8465.3389	2348.2717
2020-07-07	5046.2363	312.8015	8231.3740	2402.1670
2020-07-08	5223.1489	317.2701	8273.9121	2425.2644
2020-07-09	5138.9053	326.2072	8082.4849	2394.4678
2020-07-10	5223.1489	323.2281	8018.6758	2394.4678
2020-07-13	5202.0879	344.0816	7954.8672	2379.0691
2020-07-14	5223.1489	344.0816	8210.1035	2371.3699

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2020-07-15	5181.0273	336.6339	8039.9453	2379.0691
2020-07-16	5206.2998	330.6758	7997.4062	2386.7686
2020-07-17	5155.7534	344.0816	7954.8672	2355.9712
2020-07-20	5172.6021	336.6339	8039.9453	2355.9712
2020-07-21	5223.1489	338.1235	8061.2144	2355.9712
2020-07-22	5206.2998	335.1444	8167.5640	2363.6704
2020-07-23	5223.1489	336.6339	8039.9453	2363.6704
2020-07-24	5138.9053	324.7177	7891.0566	2325.1746
2020-07-27	5138.9053	330.6758	7976.1357	2340.5728
2020-07-28	5210.5122	324.7177	7997.4062	2325.1746
2020-07-29	5168.3892	323.2281	8061.2144	2309.7759
2020-07-30	5256.8472	323.2281	7848.5171	2348.2717
2020-08-03	5164.1777	306.8434	7763.4380	2248.1819
2020-08-04	5231.5728	308.3328	7827.2476	2271.2795
2020-08-05	5227.3608	312.8015	8039.9453	2325.1746
2020-08-06	5273.6958	332.1653	8167.5640	2325.1746
2020-08-07	5206.2998	326.2072	8231.3740	2294.3772
2020-08-10	5155.7534	339.6130	8167.5640	2286.6782
2020-08-11	5202.0879	342.5921	8167.5640	2255.8809
2020-08-12	5273.6958	333.6549	8380.2598	2271.2795
2020-08-13	5307.3931	332.1653	8273.9121	2317.4753
2020-08-14	5395.8506	332.1653	8082.4849	2332.8735
2020-08-18	5357.9399	332.1653	8295.1807	2348.2717
2020-08-19	5332.6665	332.1653	8125.0239	2309.7759
2020-08-24	5320.0298	326.2072	8252.6426	2309.7759
2020-08-25	5362.1523	327.6968	8763.1152	2294.3772
2020-08-26	5341.0908	330.6758	8890.7344	2325.1746
2020-08-27	5560.1265	335.1444	9082.1602	2302.0767
2020-08-28	5471.6699	341.1026	9060.8896	2278.9788
2020-08-31	5286.3320	323.2281	8975.8115	2201.9863
2020-09-01	5492.7310	339.6130	9358.6660	2232.7832
2020-09-02	5421.1240	361.9560	9103.4297	2271.2795
2020-09-03	5475.8823	358.9769	8954.5420	2232.7832
2020-09-04	5374.7881	358.9769	8805.6553	2201.9863
2020-09-07	5294.7568	372.3827	8997.0820	2232.7832
2020-09-08	5336.8784	369.4037	8678.0352	2201.9863
2020-09-09	5261.0586	357.4874	8592.9570	2155.7905
2020-09-10	4894.5952	333.6549	7997.4062	2078.7981
2020-09-11	4974.6289	351.5293	7997.4062	2163.4897
2020-09-14	5096.7827	357.4874	8380.2598	2225.0840
2020-09-15	4936.7188	354.5083	8188.8350	2178.8887
2020-09-16	4844.0503	341.1026	7997.4062	2148.0916
2020-09-17	4848.2617	336.6339	7912.3262	2171.1892
2020-09-18	4742.9561	341.1026	7933.5977	2225.0840
2020-09-21	4721.8955	341.1026	7827.2476	2163.4897
2020-09-22	4591.3159	345.5712	7678.3599	2140.3921
2020-09-23	4637.6509	345.5712	7572.0107	2155.7905
2020-09-24	4587.1045	336.6339	7486.9321	2101.8960
2020-09-25	4726.1074	339.6130	7486.9321	2071.0989
2020-09-28	4646.0747	336.6339	7763.4380	2048.0012
2020-09-29	4637.6509	347.0607	7742.1680	2024.9034
2020-09-30	4566.0430	338.1235	7805.9775	1971.0087

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2020-10-01	4692.4097	351.5293	8146.2939	2117.2944
2020-10-02	4637.6509	339.6130	8039.9453	2063.3997
2020-10-05	4650.2871	335.1444	8039.9453	2040.3019
2020-10-06	4801.9272	335.1444	8018.6758	2040.3019
2020-10-07	4848.2617	336.6339	8061.2144	2048.0012
2020-10-08	4869.3228	336.6339	8210.1035	2078.7981
2020-10-09	4865.1099	332.1653	7976.1357	2101.8960
2020-10-12	4932.5059	330.6758	7997.4062	2101.8960
2020-10-13	4932.5059	342.5921	7954.8672	2101.8960
2020-10-14	4970.4165	353.0188	7912.3262	2163.4897
2020-10-15	4873.5356	344.0816	7742.1680	2140.3921
2020-10-16	4852.4736	363.4455	7805.9775	2117.2944
2020-10-19	4970.4165	363.4455	7805.9775	2086.4973
2020-10-20	4890.3833	360.4664	8061.2144	2055.7007
2020-10-21	4869.3228	357.4874	8082.4849	2063.3997
2020-10-22	4886.1714	345.5712	7933.5977	2078.7981
2020-10-23	4860.8975	341.1026	8018.6758	2024.9034
2020-10-26	4898.8081	341.1026	8167.5640	2040.3019
2020-10-27	4877.7471	335.1444	8146.2939	2017.2042
2020-11-02	4903.0205	336.6339	7763.4380	1971.0087
2020-11-03	4961.9917	341.1026	7593.2808	1986.4075
2020-11-04	4903.0205	330.6758	7763.4380	1986.4075
2020-11-05	5181.0273	339.6130	8465.3389	2132.6931
2020-11-06	5307.3931	339.6130	8805.6553	2178.8887
2020-11-09	5294.7568	336.6339	8699.3057	2217.3848
2020-11-10	5459.0342	339.6130	9018.3516	2201.9863
2020-11-11	5509.5791	358.9769	9528.8242	2371.3699
2020-11-12	5408.4863	350.0397	9635.1719	2340.5728
2020-11-13	5383.2129	347.0607	9528.8242	2302.0767
2020-11-16	5459.0342	355.9978	9316.1270	2363.6704
2020-11-17	5518.0044	351.5293	9337.3955	2479.1594
2020-11-18	5534.8525	354.5083	9486.2842	2448.3623
2020-11-19	5572.7632	369.4037	9486.2842	2440.6631
2020-11-20	5560.1265	361.9560	9656.4414	2479.1594
2020-11-23	5560.1265	384.2989	10294.5342	2556.1519
2020-11-24	5530.6406	387.2779	9975.4873	2579.2498
2020-11-25	5400.0625	387.2779	10124.3750	2525.3547
2020-11-26	5459.0342	405.1523	10145.6445	2671.6406
2020-11-27	5379.0000	414.0896	10145.6445	2663.9417
2020-11-30	5227.3608	414.0896	9954.2168	2486.8586
2020-12-01	5387.4268	414.0896	10209.4551	2494.5581
2020-12-02	5433.7598	414.0896	10166.9160	2563.8513
2020-12-03	5442.1841	409.6210	10209.4551	2540.7532
2020-12-04	5383.2129	427.4954	9932.9473	2502.2568
2020-12-07	5492.7310	442.3906	9996.7568	2563.8513
2020-12-08	5467.4570	460.2651	9847.8691	2540.7532
2020-12-10	5555.8438	460.2651	9954.2168	2509.9563
2020-12-11	5691.0435	457.2860	9762.7910	2525.3547
2020-12-14	5762.8682	467.7126	10209.4551	2556.1519
2020-12-15	5737.5181	455.7964	10422.1514	2656.2422
2020-12-16	5872.7183	463.2440	10719.9268	2779.4302
2020-12-17	5860.0435	443.8802	10996.4336	2725.5356

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2020-12-18	5745.9683	440.9011	10677.3877	2702.4377
2020-12-21	5771.3179	463.2440	10634.8477	2710.1367
2020-12-22	5674.1436	437.9221	10592.3086	2586.9490
2020-12-23	5682.5928	443.8802	10507.2305	2556.1519
2020-12-28	5729.0679	448.3488	10677.3877	2640.8435
2020-12-29	5716.3936	443.8802	10698.6572	2633.1443
2020-12-30	5720.6187	426.0058	10571.0400	2548.4526
2021-01-04	5775.5435	433.4535	10698.6572	2687.0393
2021-01-05	5991.0166	424.5163	10549.7705	2671.6406
2021-01-06	5868.4922	409.6210	10230.7246	2594.6482
2021-01-07	5885.3926	418.5582	10464.6914	2610.0466
2021-01-08	5957.2168	430.4744	10485.9609	2748.6331
2021-01-11	6206.4907	454.3069	10230.7246	2771.7310
2021-01-12	6050.1670	439.4116	10315.8037	2702.4377
2021-01-13	6016.3672	451.3278	10251.9951	2679.3398
2021-01-14	5931.8677	445.3697	10145.6445	2694.7383
2021-01-15	5876.9434	433.4535	10166.9160	2679.3398
2021-01-18	6016.3672	428.9849	10762.4668	2656.2422
2021-01-19	5995.2422	421.5372	10549.7705	2625.4451
2021-01-20	5995.2422	421.5372	10592.3086	2671.6406
2021-01-21	5978.3423	420.0476	10422.1514	2679.3398
2021-01-22	5982.5669	402.1733	10337.0732	2610.0466
2021-01-25	5944.5425	385.7885	10018.0264	2594.6482
2021-01-26	5762.8682	375.3618	9996.7568	2509.9563
2021-01-27	5737.5181	370.8931	9847.8691	2602.3474
2021-01-28	5830.4683	357.4874	9592.6328	2494.5581
2021-01-29	5712.1685	357.4874	9018.3516	2394.4678
2021-02-01	5762.8682	366.4245	9805.3301	2486.8586
2021-02-02	5745.9683	357.4874	9316.1270	2517.6558
2021-02-03	5767.0933	348.5503	9252.3174	2494.5581
2021-02-04	5792.4434	347.0607	9039.6211	2533.0542
2021-02-05	5843.1426	360.4664	9571.3633	2533.0542
2021-02-08	5847.3672	363.4455	9401.2051	2525.3547
2021-02-09	5898.0674	360.4664	9273.5869	2463.7607
2021-02-10	5847.3672	357.4874	9401.2051	2456.0615
2021-02-11	5813.5674	361.9560	9316.1270	2456.0615
2021-02-15	5745.9683	361.9560	9507.5537	2479.1594
2021-02-16	5864.2671	360.4664	9379.9355	2471.4602
2021-02-17	5830.4683	351.5293	9337.3955	2425.2644
2021-02-18	5691.0435	350.0397	9103.4297	2448.3623
2021-02-19	5767.0933	351.5293	9145.9697	2471.4602
2021-02-22	5737.5181	360.4664	8954.5420	2440.6631
2021-02-23	5767.0933	358.9769	8890.7344	2671.6406
2021-02-24	5682.5928	353.0188	8741.8447	2679.3398
2021-02-25	5665.6938	357.4874	8784.3848	2687.0393
2021-02-26	5669.9189	351.5293	8678.0352	2687.0393
2021-03-01	5952.9917	353.0188	9082.1602	2687.0393
2021-03-02	5927.6426	353.0188	9954.2168	2663.9417
2021-03-03	5914.9668	353.0188	9635.1719	2648.5430
2021-03-04	5678.3687	363.4455	9528.8242	2586.9490
2021-03-05	5745.9683	351.5293	9528.8242	2556.1519
2021-03-08	5678.3687	350.0397	9294.8564	2571.5505

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2021-03-09	5581.1943	347.0607	9358.6660	2533.0542
2021-03-10	5665.6938	348.5503	9613.9033	2617.7458
2021-03-12	5716.3936	360.4664	9720.2510	2656.2422
2021-03-15	5631.8936	367.9141	9465.0146	2602.3474
2021-03-16	5598.0938	363.4455	9465.0146	2586.9490
2021-03-17	5585.4189	366.4245	9422.4736	2610.0466
2021-03-18	5665.6938	378.3408	9677.7109	2656.2422
2021-03-19	5712.1685	387.2779	10039.2959	2648.5430
2021-03-22	5593.8687	382.8094	9869.1377	2602.3474
2021-03-23	5547.3945	387.2779	9677.7109	2586.9490
2021-03-24	5441.7695	364.9351	9805.3301	2594.6482
2021-03-25	5382.6196	358.9769	9592.6328	2625.4451
2021-03-26	5420.6450	363.4455	9422.4736	2687.0393
2021-03-29	5374.1694	358.9769	9486.2842	2625.4451
2021-03-30	5403.7456	350.0397	9252.3174	2602.3474
2021-03-31	5251.6460	350.0397	8869.4639	2633.1443
2021-04-01	5260.0957	351.5293	8869.4639	2610.0466
2021-04-05	5200.9463	353.0188	8763.1152	2602.3474
2021-04-06	5209.3960	360.4664	8826.9238	2594.6482
2021-04-07	5281.2212	363.4455	8805.6553	2617.7458
2021-04-08	5256.7148	366.4245	8882.1719	2610.0466
2021-04-09	5312.4092	358.9769	8990.4912	2586.9490
2021-04-12	5196.7368	351.5293	8600.5410	2548.4526
2021-04-13	5141.0415	345.5712	9098.8096	2563.8513
2021-04-14	5402.3784	355.9978	9575.4131	2571.5505
2021-04-15	5380.9565	354.5083	9380.4385	2586.9490
2021-04-16	5376.6724	353.0188	9228.7910	2586.9490
2021-04-19	5372.3882	353.0188	9033.8174	2571.5505
2021-04-20	5342.3989	353.0188	9142.1367	2556.1519
2021-04-21	5282.4209	351.5293	9055.4824	2525.3547
2021-04-22	5325.2622	348.5503	8947.1621	2563.8513
2021-04-23	5475.2090	357.4874	9012.1543	2548.4526
2021-04-26	5385.2412	360.4664	9120.4746	2502.2568
2021-04-27	5488.0615	360.4664	8925.4980	2432.9636
2021-04-28	5415.2310	361.9560	9120.4746	2417.5654
2021-04-29	5492.3452	373.8723	9142.1367	2463.7607
2021-04-30	5488.0615	370.8931	9033.8174	2463.7607
2021-05-03	5475.2090	369.4037	8817.1797	2440.6631
2021-05-04	5483.7778	372.3827	8708.8594	2471.4602
2021-05-05	5505.1992	375.9312	8795.5156	2463.7607
2021-05-06	5505.1992	372.7853	8687.1963	2456.0615
2021-05-07	5483.7778	372.7853	8492.2217	2456.0615
2021-05-10	5500.9146	375.9312	8622.2051	2440.6631
2021-05-11	5552.3242	375.9312	8600.5410	2448.3623
2021-05-17	5569.4614	372.7853	8427.2295	2456.0615
2021-05-18	5475.2090	377.5041	8253.9199	2448.3623
2021-05-19	5436.6509	369.6394	8037.2808	2432.9636
2021-05-20	5466.6411	368.0666	8253.9199	2548.4526
2021-05-21	5449.5034	366.4937	8167.2651	2517.6558
2021-05-24	5419.5137	364.9207	7993.9531	2509.9563
2021-05-25	5445.2207	368.0666	8145.6016	2540.7532
2021-05-27	5372.3882	366.4937	8232.2568	2602.3474

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2021-05-28	5432.3667	368.0666	8362.2402	2517.6558
2021-05-31	5462.3564	374.3583	8405.5684	2648.5430
2021-06-02	5539.4727	396.3794	9033.8174	2656.2422
2021-06-03	5655.1455	393.2335	9142.1367	2687.0393
2021-06-04	5638.0088	386.9417	8990.4912	2694.7383
2021-06-07	5595.1675	379.0771	9012.1543	2733.2346
2021-06-08	5509.4829	380.6501	8817.1797	2694.7383
2021-06-09	5595.1675	382.2229	9012.1543	2741.6538
2021-06-10	5672.2822	380.6501	8947.1621	2814.4409
2021-06-11	5543.7563	413.6816	8947.1621	2806.3535
2021-06-14	5492.3452	416.8275	8817.1797	2757.8286
2021-06-15	5543.7563	412.1087	8838.8438	2782.0913
2021-06-16	5470.9248	438.8486	8838.8438	2757.8286
2021-06-17	5423.7988	426.2652	8578.8770	2709.3042
2021-06-18	5419.5137	412.1087	8578.8770	2709.3042
2021-06-21	5355.2524	404.2440	8665.5332	2668.8665
2021-06-22	5432.3667	405.8170	8405.5684	2668.8665
2021-06-23	5338.1152	396.3794	8405.5684	2725.4788
2021-06-24	5320.9785	391.6606	8427.2295	2628.4290
2021-06-25	5303.8413	404.2440	8427.2295	2628.4290
2021-06-28	5188.1670	388.5147	8080.6104	2563.7292
2021-06-29	5179.5991	382.2229	8297.2480	2555.6418
2021-06-30	5162.4629	379.0771	8232.2568	2547.5544
2021-07-01	5162.4629	379.0771	8145.6016	2515.2046
2021-07-02	5226.7261	396.3794	8058.9438	2482.8545
2021-07-05	5235.2944	388.5147	7972.2896	2458.5920
2021-07-06	5188.1670	405.8170	8058.9438	2434.3301
2021-07-07	5196.7368	393.2335	7755.6514	2450.5046
2021-07-08	5153.8945	383.7959	7668.9951	2434.3301
2021-07-09	5158.1792	393.2335	7712.3232	2555.6418
2021-07-12	5286.7046	388.5147	7625.6685	2531.3796
2021-07-13	5179.5991	386.9417	7452.3574	2482.8545
2021-07-14	5132.4727	383.7959	7387.3667	2474.7671
2021-07-15	5239.5776	385.3689	7344.0386	2531.3796
2021-07-16	5235.2944	390.0876	7668.9951	2563.7292
2021-07-19	5145.3257	393.2335	7452.3574	2604.1670
2021-07-21	5149.6099	390.0876	7668.9951	2596.0793
2021-07-22	5273.8521	419.9733	7690.6602	2636.5166
2021-07-23	5171.0303	410.5358	7539.0132	2563.7292
2021-07-26	5145.3257	402.6711	7409.0308	2579.9041
2021-07-27	5145.3257	401.0981	7257.3838	2644.6040
2021-07-28	5123.9053	399.5253	7105.7363	2579.9041
2021-07-29	5175.3154	416.8275	6954.0898	2612.2544
2021-07-30	5115.3364	419.9733	6672.4600	2620.3418
2021-08-02	5106.7686	430.9839	6759.1152	2676.9541
2021-08-03	5265.2832	424.6922	7192.3921	2693.1292
2021-08-04	5243.8628	427.8380	7322.3745	2725.4788
2021-08-05	5398.0938	408.9628	7279.0474	2701.2166
2021-08-06	5278.1357	405.8170	7322.3745	2685.0415
2021-08-09	5312.4092	408.9628	7235.7197	2628.4290
2021-08-10	5398.0938	421.5463	7344.0386	2612.2544
2021-08-12	5398.0938	443.5673	7604.0049	2709.3042

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2021-08-13	5492.3452	435.7028	7690.6602	2668.8665
2021-08-16	5500.9146	429.4109	7842.3062	2701.2166
2021-08-18	5655.1455	421.5463	8167.2651	2765.9163
2021-08-19	5655.1455	407.3900	7885.6343	2741.6538
2021-08-20	5655.1455	397.9522	8123.9375	2749.7412
2021-08-23	5646.5776	413.6816	8275.5840	2782.0913
2021-08-24	5655.1455	396.3794	8123.9375	2749.7412
2021-08-25	5655.1455	394.8064	8167.2651	2725.4788
2021-08-26	5620.8721	402.6711	7604.0049	2717.3914
2021-08-27	5578.0303	391.6606	7668.9951	2685.0415
2021-08-30	5625.1567	408.9628	7907.2979	2749.7412
2021-08-31	5612.3037	396.3794	8015.6177	2749.7412
2021-09-01	5625.1567	415.2546	7820.6426	2701.2166
2021-09-02	5603.7354	415.2546	7928.9614	2725.4788
2021-09-03	5655.1455	426.2652	7972.2896	2741.6538
2021-09-06	5633.7251	429.4109	7972.2896	2741.6538
2021-09-07	5629.4409	423.1193	7733.9868	2733.5664
2021-09-08	5518.0513	418.4005	7409.0308	2693.1292
2021-09-09	5629.4409	429.4109	7582.3408	2741.6538
2021-09-10	5586.5986	424.6922	7712.3232	2693.1292
2021-09-13	5616.5879	421.5463	7712.3232	2709.3042
2021-09-14	5608.0190	435.7028	7712.3232	2782.0913
2021-09-15	5565.1782	437.2757	7712.3232	2790.1787
2021-09-16	5569.4614	437.2757	7560.6772	2782.0913
2021-09-17	5586.5986	419.9733	7517.3496	2854.8784
2021-09-20	5642.2930	415.2546	7365.7026	2846.7910
2021-09-21	5560.8936	423.1193	7300.7109	2854.8784
2021-09-22	5616.5879	440.4215	7409.0308	2903.4033
2021-09-23	5638.0088	445.1404	7365.7026	2879.1409
2021-09-24	5642.2930	471.8802	7192.3921	2879.1409
2021-09-27	5638.0088	475.0261	7105.7363	2846.7910
2021-09-28	5586.5986	547.3810	6802.4434	2854.8784
2021-09-29	5638.0088	541.0893	7344.0386	2854.8784
2021-09-30	5997.8823	553.6728	7105.7363	2984.2781
2021-10-01	5792.2397	559.9645	7019.0811	2960.0156
2021-10-04	5963.6084	585.1315	7170.7285	2968.1033
2021-10-05	5950.7544	575.6938	6910.7622	2968.1033
2021-10-06	6152.1138	586.7043	6910.7622	3040.8906
2021-10-07	6134.9761	545.8081	6997.4175	3008.5405
2021-10-08	6246.3652	570.9750	6997.4175	3073.2402
2021-10-11	6216.3765	602.4337	6824.1064	3081.3276
2021-10-12	6272.0713	592.9961	6867.4351	3057.0652
2021-10-13	6447.7231	585.1315	7192.3921	3048.9778
2021-10-14	6640.5122	583.5585	7733.9868	3097.5029
2021-10-15	6554.8281	585.1315	7604.0049	3081.3276
2021-10-18	6447.7231	589.8503	7712.3232	3065.1528
2021-10-19	6426.3013	583.5585	7604.0049	3024.7156
2021-10-21	6340.6182	552.0999	7625.6685	3113.6777
2021-10-22	6447.7231	545.8081	7668.9951	3129.8525
2021-10-25	6447.7231	548.9540	7430.6938	3057.0652
2021-10-26	6447.7231	555.2457	7257.3838	3073.2402
2021-10-27	6383.4600	544.2352	7495.6860	3065.1528

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2021-10-28	6319.1968	512.7765	7647.3320	3032.8030
2021-10-29	6404.8813	528.5058	7885.6343	3073.2402
2021-11-01	6340.6182	530.0787	7993.9531	2992.3655
2021-11-02	6254.9336	517.4953	7777.3145	2960.0156
2021-11-03	6297.7754	526.9329	8210.5928	3008.5405
2021-11-04	6319.1968	526.9329	8578.8770	3048.9778
2021-11-05	6383.4600	517.4953	8578.8770	3048.9778
2021-11-08	6490.5649	533.2246	8297.2480	3032.8030
2021-11-09	6576.2490	536.3705	8362.2402	3000.4529
2021-11-10	6554.8281	531.6516	8297.2480	2992.3655
2021-11-11	6576.2490	530.0787	8167.2651	2951.9282
2021-11-12	6447.7231	530.0787	8167.2651	2911.4910
2021-11-15	6426.3013	506.4848	7972.2896	2911.4910
2021-11-16	6404.8813	508.0578	7950.6255	2968.1033
2021-11-17	6512.3457	515.9224	7842.3062	2935.7532
2021-11-18	6361.8945	511.2036	7777.3145	2935.7532
2021-11-19	6383.3882	517.4953	7755.6514	3154.1152
2021-11-22	6426.3740	515.9224	7733.9868	3073.2402
2021-11-23	6426.3740	536.3705	7409.0308	3073.2402
2021-11-24	6426.3740	533.2246	7430.6938	3073.2402
2021-11-25	6383.3882	548.9540	7387.3667	3234.9897
2021-11-26	6254.4302	520.6411	7170.7285	3243.0774
2021-11-29	6361.8945	531.6516	7105.7363	3348.2144
2021-11-30	6254.4302	534.7975	6932.4268	3226.9023
2021-12-01	6275.9238	552.0999	7019.0811	3299.6897
2021-12-02	6447.8667	569.4022	6997.4175	3364.3892
2021-12-03	6340.4019	570.9750	6845.7705	3291.6018
2021-12-06	6318.9092	600.8607	7040.7451	3372.4768
2021-12-07	6318.9092	599.2878	6997.4175	3356.3018
2021-12-08	6383.3882	605.5796	6954.0898	3307.7771
2021-12-09	6318.9092	608.7255	6954.0898	3356.3018
2021-12-10	6340.4019	604.0066	6824.1064	3356.3018
2021-12-13	6275.9238	604.0066	6845.7705	3299.6897
2021-12-14	6275.9238	604.0066	6715.7876	3315.8647
2021-12-15	6275.9238	635.4654	6672.4600	3299.6897
2021-12-16	6254.4302	641.7571	6564.1411	3315.8647
2021-12-17	6447.8667	644.9030	6434.1577	3315.8647
2021-12-20	6340.4019	644.9030	6347.5024	3299.6897
2021-12-21	6340.4019	666.9241	6434.1577	3364.3892
2021-12-22	6297.4160	676.3615	6282.5107	3323.9521
2021-12-23	6275.9238	676.3615	6325.8389	3364.3892
2021-12-24	6275.9238	698.3828	6304.1748	3323.9521
2021-12-27	6318.9092	685.7993	6412.4946	3299.6897
2021-12-28	6318.9092	723.5497	6369.1665	3307.7771
2021-12-29	6275.9238	726.6955	6260.8467	3299.6897
2021-12-30	6275.9238	760.5519	6282.5107	3267.3396
2022-01-03	6297.4160	801.1145	6304.1748	3380.5642
2022-01-04	6361.8945	777.4529	6390.8306	3372.4768
2022-01-05	6404.8813	757.1716	6260.8467	3275.4270
2022-01-06	6426.3740	777.4529	6217.5190	3332.0396
2022-01-07	6576.8242	821.3961	6195.8560	3372.4768
2022-01-10	6533.8384	811.2552	6152.5278	3315.8647

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2022-01-11	6619.8096	790.9740	6022.5449	3307.7771
2022-01-12	6619.8096	780.8333	6044.2090	3323.9521
2022-01-13	6619.8096	770.6925	6130.8638	3380.5642
2022-01-14	6748.7681	767.3123	6065.8726	3388.6519
2022-01-17	6662.7959	763.9321	5914.2256	3380.5642
2022-01-18	6598.3169	750.4112	5827.5703	3437.1765
2022-01-19	6598.3169	753.7914	5784.2432	3437.1765
2022-01-20	6684.2881	780.8333	5740.9155	3412.9143
2022-01-21	6834.7378	780.8333	6000.8804	3501.8765
2022-01-24	6705.7817	767.3123	6044.2090	3477.6138
2022-01-25	6684.2881	747.0311	5892.5615	3469.5264
2022-01-26	6619.8096	747.0311	5957.5537	3550.4011
2022-01-27	6705.7817	780.8333	5849.2344	3453.3516
2022-01-28	6684.2881	770.6925	5914.2256	3445.2642
2022-01-31	6555.3315	757.1716	5827.5703	3388.6519
2022-02-02	6705.7817	747.0311	5957.5537	3372.4768
2022-02-03	6641.3027	750.4112	5979.2178	3396.7393
2022-02-04	6641.3027	736.8903	6087.5371	3421.0017
2022-02-07	6705.7817	743.6508	6260.8467	3469.5264
2022-02-08	6641.3027	743.6508	6195.8560	3421.0017
2022-02-09	6834.7378	747.0311	6347.5024	3461.4392
2022-02-10	6662.7959	730.1298	6304.1748	3607.0137
2022-02-11	6727.2739	733.5101	6390.8306	3598.9258
2022-02-14	6619.8096	753.7914	6239.1831	3558.4888
2022-02-15	6770.2603	774.0727	6304.1748	3566.5764
2022-02-16	6856.2314	757.1716	6412.4946	3558.4888
2022-02-17	6791.7529	750.4112	6347.5024	3566.5764
2022-02-18	6813.2456	757.1716	6369.1665	3558.4888
2022-02-21	6834.7378	750.4112	6304.1748	3534.2263
2022-02-22	6791.7529	753.7914	6282.5107	3477.6138
2022-02-23	6920.7095	784.2134	6239.1831	3518.0515
2022-02-24	6877.7241	838.2972	5979.2178	3429.0891
2022-02-25	6920.7095	828.1564	6239.1831	3509.9641
2022-03-01	6920.7095	872.0995	6152.5278	3518.0515
2022-03-02	6856.2314	882.2402	5827.5703	3469.5264
2022-03-04	6791.7529	1027.5901	5740.9155	3558.4888
2022-03-07	6619.8096	1095.1947	5459.2852	3647.4512
2022-03-08	6576.8242	1054.6320	5524.2769	3736.4133
2022-03-09	6748.7681	1041.1111	5567.6045	3590.8386
2022-03-10	6813.2456	1007.3087	6022.5449	3566.5764
2022-03-11	6834.7378	1014.0691	5935.8896	3712.1509
2022-03-14	6942.2036	943.0844	5870.8984	3704.0635
2022-03-15	7006.6816	916.0426	5740.9155	3736.4133
2022-03-16	7049.6680	953.2251	5784.2432	3744.5005
2022-03-17	6877.7241	926.1831	5849.2344	3623.1887
2022-03-18	6791.7529	919.4228	5762.5791	3671.7131
2022-03-21	6791.7529	919.4228	5697.5874	3679.8008
2022-03-22	6813.2456	946.4646	5762.5791	3679.8008
2022-03-23	6791.7529	946.4646	5697.5874	3679.8008
2022-03-24	6813.2456	973.5063	5697.5874	3687.8879
2022-03-25	6834.7378	953.2251	5632.5967	3655.5383
2022-03-28	6895.8408	966.7459	5870.8984	3720.2380

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2022-03-29	6852.1973	936.3239	5784.2432	3687.8879
2022-03-30	6874.0190	905.9019	5849.2344	3704.0635
2022-03-31	6961.3081	909.2820	5762.5791	3704.0635
2022-04-01	6917.6631	949.8447	5870.8984	3704.0635
2022-04-04	6895.8408	949.8447	5935.8896	3671.7131
2022-04-05	6895.8408	1014.0691	5914.2256	3655.5383
2022-04-06	6764.9077	1037.7307	5805.9067	3582.7512
2022-04-07	6764.9077	1030.9703	5719.2520	3671.7131
2022-04-08	6852.1973	1068.1530	5784.2432	3704.0635
2022-04-11	6743.0854	1061.3926	5648.7095	3712.1509
2022-04-12	6808.5522	1061.3926	5604.2314	3744.5005
2022-04-13	6808.5522	1132.3773	5515.2759	3793.0256
2022-04-14	6721.2627	1118.8562	5359.6025	3784.9380
2022-04-18	6721.2627	1112.0957	5470.7979	3809.2007
2022-04-19	6655.7964	1105.3354	5381.8413	3784.9380
2022-04-20	6677.6182	1091.8145	5292.8848	3744.5005
2022-04-21	6917.6631	1105.3354	5404.0806	3736.4133
2022-04-22	6874.0190	1085.0542	5337.3633	3736.4133
2022-04-25	6983.1304	1085.0542	5315.1240	3809.2007
2022-04-26	7092.2417	1068.1530	5693.1875	3849.6377
2022-04-27	7157.7090	1071.5331	5759.9043	3857.7253
2022-04-28	7092.2417	1128.9971	5693.1875	3736.4133
2022-05-09	6633.9736	1068.1530	5448.5586	3485.7014
2022-05-10	6568.5068	1044.4913	5426.3198	3509.9641
2022-05-11	6677.6182	1064.7727	5759.9043	3501.8765
2022-05-12	6350.2837	1064.7727	5581.9927	3477.6138
2022-05-13	6393.9282	1085.0542	5604.2314	3445.2642
2022-05-17	6459.3950	1138.5851	5648.7095	3380.5642
2022-05-18	6612.1514	1127.9773	5759.9043	3437.1765
2022-05-19	6503.0405	1099.6896	5537.5146	3404.8269
2022-05-20	6459.3950	1170.4089	5648.7095	3372.4768
2022-05-23	6437.5728	1113.8334	5782.1436	3323.9521
2022-05-24	6415.7510	1131.5132	5960.0562	3356.3018
2022-05-25	6437.5728	1096.1534	5804.3828	3396.7393
2022-05-27	6612.1514	1117.3695	5960.0562	3477.6138
2022-05-30	6612.1514	1082.0096	6382.5967	3437.1765
2022-05-31	6764.9077	1156.2653	6493.7920	3485.7014
2022-06-02	6612.1514	1177.4812	6382.5967	3469.5264
2022-06-03	6633.9736	1244.6646	6315.8804	3485.7014
2022-06-06	6503.0405	1223.4487	6404.8354	3509.9641
2022-06-07	6437.5728	1290.6322	6360.3579	3437.1765
2022-06-08	6633.9736	1283.5604	6427.0747	3437.1477
2022-06-09	6546.6846	1272.9524	6226.9238	3395.2312
2022-06-10	6415.7510	1212.8407	6137.9673	3370.0815
2022-06-13	6415.7510	1127.9773	5826.6216	3420.3811
2022-06-14	6459.3950	1152.7290	6071.2505	3445.5310
2022-06-15	6393.9282	1074.9376	6137.9673	3403.6147
2022-06-16	6612.1514	1082.0096	6137.9673	3386.8481
2022-06-17	6546.6846	1032.5057	6071.2505	3453.9141
2022-06-20	6655.7964	1036.0419	6315.8804	3386.8481
2022-06-21	6677.6182	1071.4017	6360.3579	3445.5310
2022-06-22	6546.6846	1071.4017	6360.3579	3395.2312

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2022-06-23	6568.5068	1039.5780	6493.7920	3487.4475
2022-06-24	6524.8623	1050.1858	6782.8994	3479.0642
2022-06-27	6415.7510	1046.6498	6516.0312	3411.9978
2022-06-28	6372.1064	1043.1139	6315.8804	3378.4646
2022-06-29	6350.2837	1014.8260	6493.7920	3386.8481
2022-06-30	6328.4619	1011.2900	6338.1187	3353.3149
2022-07-01	6328.4619	961.7863	6115.7280	3370.0815
2022-07-04	6153.8833	979.4662	6049.0112	3344.9316
2022-07-05	6328.4619	1018.3619	5960.0562	3344.9316
2022-07-06	6372.1064	979.4662	5982.2939	3370.0815
2022-07-07	6197.5288	979.4662	5893.3384	3386.8481
2022-07-08	6241.1724	1000.6819	5937.8169	3361.6982
2022-07-11	6219.3501	1000.6819	5782.1436	3386.8481
2022-07-12	6262.9946	1032.5057	5693.1875	3361.6982
2022-07-13	6110.2393	1036.0419	5693.1875	3303.0151
2022-07-14	6132.0620	1036.0419	5804.3828	3370.0815
2022-07-15	6110.2393	975.9302	5670.9482	3479.0642
2022-07-18	6241.1724	986.5381	5782.1436	3512.5974
2022-07-19	6262.9946	1043.1139	5826.6216	3487.4475
2022-07-20	6459.3950	1067.8656	5782.1436	3529.3638
2022-07-21	6459.3950	1067.8656	5804.3828	3571.2803
2022-07-22	6393.9282	1071.4017	5804.3828	3546.1304
2022-07-25	6372.1064	1096.1534	5782.1436	3554.5137
2022-07-26	6372.1064	1117.3695	5848.8604	3562.8970
2022-07-27	6393.9282	1159.8010	5715.4268	3604.8135
2022-07-28	6415.7510	1159.8010	5715.4268	3554.5137
2022-07-29	6415.7510	1149.1934	5804.3828	3546.1304
2022-08-01	6546.6846	1184.5530	5737.6655	3604.8135
2022-08-02	6633.9736	1124.4413	5737.6655	3730.5627
2022-08-03	6655.7964	1170.4089	5759.9043	3772.4792
2022-08-04	6808.5522	1127.9773	6004.5332	3814.3960
2022-08-05	6874.0190	1106.7615	6049.0112	3898.2285
2022-08-08	6874.0190	1089.0817	6226.9238	3940.1453
2022-08-09	6895.8408	1106.7615	6182.4453	3889.8455
2022-08-10	6895.8408	1110.2974	6226.9238	3822.7786
2022-08-11	6939.4854	1113.8334	6160.2065	3831.1624
2022-08-12	6917.6631	1127.9773	6249.1631	3814.3960
2022-08-15	6939.4854	1117.3695	6160.2065	3730.5627
2022-08-16	6961.3081	1113.8334	6115.7280	3697.0298
2022-08-18	6983.1304	1149.1934	6071.2505	3814.3960
2022-08-19	6895.8408	1149.1934	5848.8604	3856.3118
2022-08-22	6983.1304	1113.8334	5871.0996	3898.2285
2022-08-23	6895.8408	1184.5530	5937.8169	3906.6121
2022-08-24	6939.4854	1205.7689	5848.8604	3982.0613
2022-08-25	7048.5977	1205.7689	5826.6216	3831.1624
2022-08-26	6983.1304	1205.7689	5826.6216	3764.0957
2022-08-29	7114.0635	1223.4487	5804.3828	3789.2458
2022-08-30	7135.8867	1251.7366	5826.6216	3755.7129
2022-08-31	7157.7090	1251.7366	5871.0996	3822.7786
2022-09-01	7114.0635	1308.3121	5737.6655	3839.5454
2022-09-02	7179.5317	1336.6002	5715.4268	3856.3118
2022-09-05	7223.1753	1424.9996	5804.3828	3873.0786

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2022-09-06	7223.1753	1428.5355	5804.3828	3780.8625
2022-09-07	7310.4639	1421.4636	5826.6216	3764.0957
2022-09-08	7288.6416	1393.1758	5826.6216	3822.7786
2022-09-09	7310.4639	1396.7118	5871.0996	3856.3118
2022-09-12	7310.4639	1403.7834	5893.3384	3806.0122
2022-09-13	7441.3979	1417.9276	5982.2939	3789.2458
2022-09-14	7419.5762	1417.9276	5915.5776	3755.7129
2022-09-15	7637.7988	1435.6074	5893.3384	3772.4792
2022-09-16	7375.9321	1389.6398	6115.7280	3697.0298
2022-09-19	7550.5098	1382.5679	6226.9238	3772.4792
2022-09-20	7463.2207	1375.4960	6404.8354	3755.7129
2022-09-21	7397.7534	1382.5679	6404.8354	3713.7964
2022-09-22	7397.7534	1449.7515	6471.5527	3713.7964
2022-09-23	7310.4639	1432.0715	6604.9873	3671.8796
2022-09-26	7354.1094	1364.8881	6604.9873	3738.9460
2022-09-27	7244.9971	1382.5679	6671.7041	3722.1797
2022-09-28	7266.8198	1368.4240	6582.7480	3747.3293
2022-09-29	7310.4639	1396.7118	6671.7041	3722.1797
2022-09-30	7463.2207	1400.2476	6649.4653	3738.9460
2022-10-03	7419.5762	1400.2476	6627.2261	3738.9460
2022-10-04	7463.2207	1456.8236	6627.2261	3722.1797
2022-10-05	7375.9321	1442.6796	6516.0312	3738.9460
2022-10-06	7354.1094	1449.7515	6516.0312	3713.7964
2022-10-07	7157.7090	1463.8951	6427.0747	3646.7297
2022-10-10	7244.9971	1386.1035	6538.2705	3705.4128
2022-10-11	7201.3525	1389.6398	6693.9429	3613.1968
2022-10-12	7266.8198	1414.3917	6671.7041	3638.3467
2022-10-13	7223.1753	1421.4636	6538.2705	3604.8135
2022-10-14	7201.3525	1396.7118	6404.8354	3596.4302
2022-10-17	7201.3525	1400.2476	6671.7041	3629.9634
2022-10-18	7244.9971	1364.8881	6693.9429	3562.8970
2022-10-19	7223.1753	1357.8159	6894.0942	3520.9807
2022-10-20	7419.5762	1414.3917	6938.5718	3646.7297
2022-10-21	7550.5098	1400.2476	6760.6606	3655.1133
2022-10-24	7768.7319	1386.1035	6738.4214	3697.0298
2022-10-25	7594.1548	1371.9597	6649.4653	3680.2629
2022-10-26	7463.2207	1389.6398	6649.4653	3663.4966
2022-10-27	7594.1548	1407.3196	6827.3770	3663.4966
2022-10-28	7637.7988	1382.5679	6894.0942	3730.5627
2022-10-31	7681.4438	1407.3196	7072.0063	3680.2629
2022-11-01	7681.4438	1325.9921	7138.7236	3705.4128
2022-11-02	7637.7988	1322.4562	7205.4399	3537.7471
2022-11-03	7681.4438	1325.9921	7138.7236	3462.2976
2022-11-04	7659.6211	1325.9921	7338.8745	3529.3638
2022-11-07	7725.0874	1354.2800	7338.8745	3588.0471
2022-11-08	7637.7988	1329.5281	7072.0063	3537.7471
2022-11-09	7746.9106	1287.0963	7005.2891	3512.5974
2022-11-10	7681.4438	1230.5206	7094.2456	3495.8308
2022-11-11	7725.0874	1234.0566	7227.6792	3479.0642
2022-11-14	7637.7988	1237.5927	7116.4849	3395.2312
2022-11-15	7681.4438	1241.1287	7072.0063	3395.2312
2022-11-16	7506.8647	1262.3445	6782.8994	3395.2312

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2022-11-17	7615.9775	1269.4164	6782.8994	3386.8481
2022-11-18	7703.2661	1269.4164	6871.8550	3361.6982
2022-11-21	7615.9775	1304.7762	6827.3770	3378.4646
2022-11-22	7768.7319	1308.3121	6849.6157	3353.3149
2022-11-23	7746.9106	1325.9921	6805.1377	3328.1650
2022-11-24	7856.0225	1325.9921	6760.6606	3378.4646
2022-11-25	7834.1982	1318.9203	6849.6157	3378.4646
2022-11-28	7877.8442	1322.4562	6849.6157	3336.5481
2022-11-29	7834.1982	1371.9597	6760.6606	3336.5481
2022-11-30	8117.8892	1368.4240	6760.6606	3386.8481
2022-12-01	7856.0225	1379.0319	6849.6157	3319.7815
2022-12-02	7799.0615	1361.3521	6827.3770	3353.3149
2022-12-05	7689.5244	1336.6002	6849.6157	3219.1821
2022-12-06	7601.8955	1347.2081	6382.5967	3017.9836
2022-12-07	7404.7280	1354.2800	6226.9238	3085.0496
2022-12-08	7448.5420	1322.4562	6182.4453	3135.3496
2022-12-09	7514.2646	1301.2402	6182.4453	3059.8999
2022-12-12	7623.8027	1343.6721	6004.5332	3101.8162
2022-12-13	7623.8027	1322.4562	6244.2480	3143.7327
2022-12-14	7558.0796	1371.9597	6244.2480	3143.7327
2022-12-15	7448.5420	1379.0319	6444.9561	3076.6665
2022-12-16	7536.1724	1375.4960	6266.5493	3085.0496
2022-12-19	7579.9878	1403.7834	6199.6465	3118.5830
2022-12-20	7514.2646	1386.1035	6043.5400	3118.5830
2022-12-21	7601.8955	1386.1035	5998.9385	3177.2659
2022-12-22	7514.2646	1389.6398	5998.9385	3143.7327
2022-12-23	7448.5420	1347.2081	6021.2397	3168.8826
2022-12-26	7514.2646	1343.6721	6065.8413	3143.7327
2022-12-27	7536.1724	1325.9921	5954.3364	3194.0325
2022-12-28	7579.9878	1325.9921	5954.3364	3126.9661
2022-12-29	7514.2646	1343.6721	5954.3364	3168.8826
2022-12-30	7492.3574	1361.3521	5865.1333	3143.7327
2023-01-02	7492.3574	1358.0535	5909.7349	3185.6489
2023-01-03	7492.3574	1331.5732	5976.6372	3235.9490
2023-01-04	7317.0972	1248.3500	6043.5400	3202.4155
2023-01-05	7229.4678	1172.6921	5976.6372	3160.4993
2023-01-06	7273.2822	1187.8239	6311.1504	3110.1995
2023-01-09	7404.7280	1146.2122	6400.3550	3177.2659
2023-01-10	7163.7456	1187.8239	6288.8501	3210.7991
2023-01-11	7119.9302	1202.9553	6244.2480	3244.3320
2023-01-12	7163.7456	1168.9093	6601.0625	3235.9490
2023-01-13	7054.2080	1187.8239	6422.6553	3168.8826
2023-01-16	7141.8379	1172.6921	6601.0625	3227.5657
2023-01-17	7295.1899	1187.8239	6444.9561	3311.3984
2023-01-18	7273.2822	1199.1724	6244.2480	3294.6318
2023-01-19	7295.1899	1221.8696	6355.7529	3261.0986
2023-01-20	7273.2822	1225.6523	6288.8501	3244.3320
2023-01-24	7207.5605	1210.5211	6333.4517	3227.5657
2023-01-25	7185.6533	1184.0409	6288.8501	3219.1821
2023-01-26	7426.6348	1138.6465	6534.1602	3328.1650
2023-01-27	7623.8027	1157.5608	6489.5581	3319.7815
2023-01-30	7623.8027	1138.6465	6444.9561	3319.7815

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2023-01-31	7426.6348	1119.7319	6601.0625	3227.5657
2023-02-01	7448.5420	1115.9491	6534.1602	3244.3320
2023-02-02	7404.7280	1085.6860	6534.1602	3261.0986
2023-02-03	7623.8027	1044.0745	6489.5581	3252.7156
2023-02-06	7645.7104	1051.6401	6467.2573	3252.7156
2023-02-07	7755.2476	1093.2520	6467.2573	3185.6489
2023-02-08	7733.3398	1104.6006	6444.9561	3219.1821
2023-02-09	7799.0615	1074.3374	6534.1602	3160.4993
2023-02-10	7733.3398	1036.5087	6913.2749	3185.6489
2023-02-13	7777.1558	1051.6401	6779.4697	3177.2659
2023-02-14	7842.8765	1070.5544	6824.0713	3185.6489
2023-02-15	7777.1558	1115.9491	6801.7710	3177.2659
2023-02-16	7623.8027	1104.6006	6712.5669	3160.4993
2023-02-17	7645.7104	1089.4690	6623.3633	3152.1162
2023-02-20	7667.6177	1097.0347	6712.5669	3202.4155
2023-02-21	7623.8027	1066.7717	6757.1689	3219.1821
2023-02-22	7601.8955	1078.1204	6623.3633	3235.9490
2023-02-23	7645.7104	1097.0347	6801.7710	3328.1650
2023-02-24	7601.8955	1093.2520	6601.0625	3386.8481
2023-02-27	7689.5244	1115.9491	6578.7612	3319.7815
2023-02-28	7667.6177	1131.0807	6444.9561	3252.7156
2023-03-01	7536.1724	1131.0807	6467.2573	3294.6318
2023-03-02	7558.0796	1131.0807	6378.0537	3252.7156
2023-03-03	7426.6348	1142.4292	6400.3550	3269.4822
2023-03-06	7360.9131	1100.8176	6355.7529	3252.7156
2023-03-07	7382.8203	1081.9031	6311.1504	3252.7156
2023-03-08	7514.2646	1104.6006	6155.0449	3303.0151
2023-03-09	7514.2646	1097.0347	6177.3452	3328.1650
2023-03-10	7404.7280	1081.9031	6043.5400	3361.6982
2023-03-13	7492.3574	1097.0347	5909.7349	3411.9978
2023-03-14	7295.1899	1047.8574	5865.1333	3370.0815
2023-03-15	7295.1899	1040.2915	5508.3188	3361.6982
2023-03-16	7273.2822	998.6799	5352.2129	3395.2312
2023-03-17	7339.0054	1051.6401	5441.4160	3395.2312
2023-03-20	7360.9131	1028.9429	5419.1152	3344.9316
2023-03-21	7448.5420	1032.7258	5575.2212	3453.9141
2023-03-24	7733.3398	1013.8113	5619.8232	3411.9978
2023-03-27	7623.8027	1013.8113	5575.2212	3395.2312
2023-03-28	7601.8955	1047.8574	5686.7261	3395.2312
2023-03-29	7865.5703	1089.4690	5686.7261	3437.1477
2023-03-30	7887.9150	1085.6860	5686.7261	3428.7644
2023-03-31	7820.8794	1097.0347	5619.8232	3403.6147
2023-04-03	7865.5703	1112.1661	5664.4248	3420.3811
2023-04-04	7843.2251	1153.7777	5686.7261	3403.6147
2023-04-05	7798.5347	1157.5608	5374.5137	3479.0642
2023-04-06	7820.8794	1142.4292	5374.5137	3562.8970
2023-04-10	7865.5703	1131.0807	5396.8149	3604.8135
2023-04-11	7887.9150	1153.7777	5486.0181	3604.8135
2023-04-12	7954.9507	1119.7319	5530.6201	3613.1968
2023-04-13	7977.2969	1119.7319	5441.4160	3638.3467
2023-04-14	8044.3335	1104.6006	5463.7173	3638.3467
2023-04-17	8066.6787	1081.9031	5396.8149	3579.6636

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2023-04-18	8156.0605	1134.8634	5374.5137	3579.6636
2023-04-26	8223.0967	1172.6921	5374.5137	3688.6462
2023-04-27	8178.4062	1165.1265	5374.5137	3621.5801
2023-04-28	8089.0234	1184.0409	5307.6104	3562.8970
2023-05-02	8089.0234	1127.2976	5240.7080	3512.5974
2023-05-03	7977.2969	1112.1661	5218.4072	3479.0642
2023-05-04	8044.3335	1104.6006	5400.1406	3479.0642
2023-05-05	8044.3335	1055.4230	5307.0352	3495.8308
2023-05-08	8044.3335	1085.6860	5423.4175	3403.6147
2023-05-09	7977.2969	1093.2520	5702.7349	3411.9978
2023-05-10	7977.2969	1112.1661	5632.9058	3462.2976
2023-05-11	7887.9150	1059.2059	5469.9697	3428.7644
2023-05-12	7865.5703	1044.0745	5493.2471	3336.5481
2023-05-15	7843.2251	1006.2456	5446.6938	3336.5481
2023-05-16	7776.1890	1010.0284	5493.2471	3336.5481
2023-05-17	7843.2251	960.8511	5516.5229	3344.9316
2023-05-19	8044.3335	911.6736	5516.5229	3370.0815
2023-05-22	8044.3335	930.5880	5632.9058	3386.8481
2023-05-23	8156.0605	960.2682	5539.7993	3386.8481
2023-05-24	8066.6787	943.4950	5516.5229	3487.4475
2023-05-25	8089.0234	888.9821	5493.2471	3453.9141
2023-05-26	8178.4062	876.4020	5493.2471	3462.2976
2023-05-29	8178.4062	868.0154	5493.2471	3520.9807
2023-05-30	8267.7861	884.7886	5493.2471	3453.9141
2023-05-31	8089.0234	855.4355	5400.1406	3386.8481
2023-06-05	8223.0967	880.5953	5423.4175	3420.3811
2023-06-06	8178.4062	930.9150	5446.6938	3428.7644
2023-06-07	8133.7153	909.9487	5446.6938	3479.0642
2023-06-08	8156.0605	922.5284	5469.9697	3495.8308
2023-06-09	8133.7153	918.3353	5516.5229	3479.0642
2023-06-12	8178.4062	926.7219	5632.9058	3538.1191
2023-06-13	8178.4062	926.7219	5586.3525	3520.6467
2023-06-14	8111.3691	943.4950	5563.0762	3494.4385
2023-06-15	8089.0234	964.4615	5656.1816	3511.9109
2023-06-16	8089.0234	972.8482	5586.3525	3468.2305
2023-06-19	8044.3335	947.6884	5586.3525	3459.4941
2023-06-20	8089.0234	951.8817	5609.6289	3485.7026
2023-06-21	8156.0605	947.6884	5609.6289	3476.9663
2023-06-22	8089.0234	947.6884	5609.6289	3442.0222
2023-06-23	8089.0234	935.1085	5702.7349	3450.7581
2023-06-26	8111.3691	926.7219	5726.0117	3485.7026
2023-06-27	8178.4062	935.1085	5656.1816	3494.4385
2023-07-03	8111.3691	977.0416	5656.1816	3494.4385
2023-07-04	8089.0234	981.2349	5656.1816	3468.2305
2023-07-05	8089.0234	1014.7813	5749.2876	3476.9663
2023-07-06	8111.3691	1027.3612	5819.1172	3459.4941
2023-07-07	8066.6787	998.0082	5982.0527	3459.4941
2023-07-10	8089.0234	1014.7813	5982.0527	3450.7581
2023-07-11	8066.6787	1018.9746	6098.4351	3476.9663
2023-07-12	8200.7510	993.8146	6051.8823	3476.9663
2023-07-13	8156.0605	972.8482	6261.3701	3424.5500
2023-07-14	8223.0967	989.6215	6401.0293	3433.2859

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2023-07-17	8200.7510	998.0082	6307.9229	3415.8137
2023-07-18	8178.4062	977.0416	6307.9229	3363.3972
2023-07-20	8178.4062	993.8146	6238.0938	3363.3972
2023-07-21	8178.4062	1014.7813	6470.8584	3372.1333
2023-07-24	8133.7153	1035.7479	6470.8584	3407.0776
2023-07-25	8178.4062	1018.9746	6517.4111	3389.6055
2023-07-26	8357.1680	1048.3279	6354.4761	3389.6055
2023-07-27	8245.4414	1027.3612	6214.8174	3249.8279
2023-07-28	8156.0605	1035.7479	6284.6470	3249.8279
2023-07-31	8156.0605	1010.5880	6494.1348	3249.8279
2023-08-01	8156.0605	1002.2014	6261.3701	3276.0361
2023-08-02	8223.0967	993.8146	6354.4761	3223.6196
2023-08-03	8267.7861	989.6215	6726.8994	3249.8279
2023-08-04	8178.4062	1006.3947	6587.2407	3223.6196
2023-08-07	8290.1318	998.0082	6470.8584	3214.8835
2023-08-08	8223.0967	993.8146	6377.7524	3258.5640
2023-08-09	8401.8584	1006.3947	6214.8174	3293.5083
2023-08-10	8401.8584	1010.5880	6191.5410	3319.7166
2023-08-11	8401.8584	985.4282	6261.3701	3328.4529
2023-08-14	8357.1680	1002.2014	6377.7524	3345.9250
2023-08-15	8312.4775	1052.5212	6377.7524	3337.1887
2023-08-16	8312.4775	1056.7145	6447.5815	3345.9250
2023-08-18	8267.7861	1065.1011	6261.3701	3284.7725
2023-08-21	8200.7510	1098.6476	6238.0938	3267.3003
2023-08-22	8312.4775	1132.1941	6284.6470	3276.0361
2023-08-23	8312.4775	1132.1941	6261.3701	3267.3003
2023-08-24	8223.0967	1119.6140	6401.0293	3249.8279
2023-08-25	8290.1318	1086.0676	6401.0293	3249.8279
2023-08-28	8223.0967	1123.8074	6377.7524	3232.3557
2023-08-29	8267.7861	1123.8074	6401.0293	3249.8279
2023-08-30	8223.0967	1128.0007	6401.0293	3328.4529
2023-08-31	8200.7510	1119.6140	6331.2002	3258.5640
2023-09-01	8245.4414	1123.8074	6331.2002	3249.8279
2023-09-04	8245.4414	1148.9673	6307.9229	3258.5640
2023-09-05	8245.4414	1148.9673	6494.1348	3267.3003
2023-09-06	8178.4062	1203.4803	6401.0293	3310.9807
2023-09-07	8200.7510	1207.6737	6401.0293	3284.7725
2023-09-08	8156.0605	1211.8671	6447.5815	3223.6196
2023-09-11	8156.0605	1174.1272	6610.5166	3232.3557
2023-09-12	8133.7153	1178.3203	6540.6880	3241.0918
2023-09-13	8111.3691	1199.2869	6494.1348	3241.0918
2023-09-14	8133.7153	1220.2537	6517.4111	3214.8835
2023-09-15	8044.3335	1190.9004	6517.4111	3249.8279
2023-09-18	8044.3335	1190.9004	6401.0293	3214.8835
2023-09-19	8111.3691	1216.0603	6377.7524	3258.5640
2023-09-20	8178.4062	1241.2201	6401.0293	3328.4529
2023-09-21	8156.0605	1237.0269	6424.3057	3328.4529
2023-09-22	8111.3691	1237.0269	6377.7524	3372.1333
2023-09-25	8044.3335	1237.0269	6191.5410	3293.5083
2023-09-26	7999.6421	1161.5474	6098.4351	3284.7725
2023-09-27	7932.6060	1203.4803	6098.4351	3249.8279
2023-09-29	7887.9150	1195.0938	5982.0527	3276.0361

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2023-10-02	8111.3691	1178.3203	6028.6055	3276.0361
2023-10-03	8223.0967	1128.0007	5958.7759	3293.5083
2023-10-04	8223.0967	1115.4208	5865.6704	3302.2444
2023-10-05	8111.3691	1094.4542	5819.1172	3249.8279
2023-10-06	8066.6787	1111.2275	5912.2236	3302.2444
2023-10-09	8089.0234	1161.5474	6144.9878	3284.7725
2023-10-10	7977.2969	1165.7406	6121.7114	3328.4529
2023-10-11	7977.2969	1157.3540	6075.1582	3302.2444
2023-10-12	8089.0234	1128.0007	6098.4351	3345.9250
2023-10-13	8111.3691	1136.3873	6121.7114	3363.3972
2023-10-16	8133.7153	1165.7406	6005.3291	3284.7725
2023-10-17	7999.6421	1161.5474	6121.7114	3310.9807
2023-10-18	7910.2612	1190.9004	5958.7759	3276.0361
2023-10-19	7820.8794	1182.5139	6028.6055	3258.5640
2023-10-20	8021.9868	1174.1272	5958.7759	3232.3557
2023-10-23	7910.2612	1115.4208	5842.3936	3153.7310
2023-10-24	7843.2251	1115.4208	5888.9463	3188.6753
2023-10-25	7932.6060	1119.6140	5958.7759	3144.9949
2023-10-26	7798.5347	1111.2275	5819.1172	3040.1616
2023-10-27	7776.1890	1107.0343	5749.2876	3057.6338
2023-10-30	7910.2612	1065.1011	5679.4585	2987.7451
2023-10-31	7820.8794	1073.4875	5679.4585	3048.8977
2023-11-01	7686.8071	1010.5880	5632.9058	3162.4668
2023-11-02	7910.2612	1002.2014	5819.1172	3162.4668
2023-11-03	7954.9507	1044.1345	5772.5645	3118.7864
2023-11-06	8089.0234	1060.9077	6028.6055	3188.6753
2023-11-07	8021.9868	1035.7479	6005.3291	3118.7864
2023-11-08	8044.3335	1023.1680	5912.2236	3075.1060
2023-11-09	8044.3335	1027.3612	5935.4995	3083.8420
2023-11-10	7887.9150	1039.9413	5842.3936	3101.3142
2023-11-13	7932.6060	1039.9413	5912.2236	3075.1060
2023-11-14	7977.2969	1039.9413	5795.8408	3057.6338
2023-11-15	8089.0234	1052.5212	5842.3936	3101.3142
2023-11-16	8111.3691	1060.9077	5842.3936	3092.5781
2023-11-17	8111.3691	1065.1011	6168.2642	3101.3142
2023-11-20	7932.6060	1069.2943	6238.0938	3136.2585
2023-11-21	7843.2251	1094.4542	6168.2642	3162.4668
2023-11-22	7932.6060	1086.0676	6005.3291	3153.7310
2023-11-23	7977.2969	1069.2943	6005.3291	3144.9949
2023-11-24	7977.2969	1065.1011	5982.0527	3162.4668
2023-11-27	7932.6060	1069.2943	6051.8823	3214.8835
2023-11-28	7932.6060	1081.8744	5958.7759	3293.5083
2023-11-29	7954.9507	1081.8744	5958.7759	3249.8279
2023-11-30	8021.9868	1098.6476	6051.8823	3284.7725
2023-12-01	7999.6421	1073.4875	6098.4351	3345.9250
2023-12-04	8015.3584	1073.4875	6005.3291	3328.4529
2023-12-05	7992.9062	1052.5212	5935.4995	3363.3972
2023-12-06	7903.0991	1060.9077	5982.0527	3424.5500
2023-12-07	7925.5503	1060.9077	5865.6704	3415.8137
2023-12-08	7858.1948	1073.4875	5865.6704	3442.0222
2023-12-11	7858.1948	1073.4875	5702.7349	3476.9663
2023-12-12	7813.2900	1069.2943	5749.2876	3415.8137

Bersambung...

Lanjutan Tabel E.1

Tanggal	BBCA	ADRO	SMGR	TLKM
2023-12-13	7790.8384	1027.3612	5795.8408	3424.5500
2023-12-14	8127.6182	1044.1345	5749.2876	3442.0222
2023-12-15	8284.7822	1056.7145	5772.5645	3476.9663
2023-12-18	8262.3301	1052.5212	5702.7349	3468.2305
2023-12-19	8307.2344	1090.2610	5749.2876	3468.2305
2023-12-20	8352.1377	1081.8744	5795.8408	3459.4941
2023-12-21	8374.5908	1090.2610	5726.0117	3450.7581
2023-12-22	8374.5908	1086.0676	5842.3936	3459.4941
2023-12-27	8419.4941	1086.0676	5842.3936	3442.0222
2023-12-28	8441.9463	1081.8744	5842.3936	3459.4941
2023-12-29	8441.9463	1081.8652	5958.7759	3450.7581

Tabel E.2: Data Simple Return dan Log Return Saham (BBCA, ADRO, SMGR, TLKM)

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2020-06-03	0.07836	0.07544	0.01747	0.01732	-0.00257	-0.00257	0.01231	0.01223
2020-06-04	0.00173	0.00173	-0.00078	-0.00078	-0.01031	-0.01036	0.00304	0.00303
2020-06-05	-0.01123	-0.01129	-0.01786	-0.01802	-0.01823	-0.01840	-0.02121	-0.02144
2020-06-08	0.03057	0.03011	0.05455	0.05311	0.07692	0.07411	0.00000	0.00000
2020-06-09	-0.01610	-0.01623	0.01724	0.01709	-0.06897	-0.07146	-0.02786	-0.02826
2020-06-10	-0.00086	-0.00086	-0.06780	-0.07020	-0.02381	-0.02410	-0.00955	-0.00960
2020-06-11	-0.02069	-0.02091	-0.02727	-0.02765	0.02439	0.02410	-0.00965	-0.00969
2020-06-12	-0.00176	-0.00176	-0.00935	-0.00939	-0.00529	-0.00531	-0.01623	-0.01637
2020-06-15	-0.02998	-0.03044	-0.03774	-0.03847	0.00266	0.00266	0.01980	0.01961
2020-06-16	0.04727	0.04619	0.05392	0.05252	0.03183	0.03133	0.03560	0.03498
2020-06-17	-0.00694	-0.00697	-0.00465	-0.00466	-0.01028	-0.01034	0.00313	0.00312
2020-06-18	-0.02360	-0.02388	-0.03738	-0.03810	-0.03117	-0.03166	0.02181	0.02157
2020-06-19	-0.00179	-0.00179	0.00971	0.00966	0.01877	0.01859	0.00000	0.00000
2020-06-22	-0.00628	-0.00630	-0.00962	-0.00966	0.00789	0.00786	-0.02439	-0.02469
2020-06-23	0.01444	0.01434	-0.02427	-0.02457	-0.00261	-0.00261	-0.02187	-0.02212
2020-06-24	0.01512	0.01501	0.03980	0.03903	-0.00785	-0.00788	0.01597	0.01585
2020-06-25	0.00351	0.00350	-0.03828	-0.03903	-0.00264	-0.00264	-0.00629	-0.00631
2020-06-26	-0.01397	-0.01407	0.00995	0.00990	0.00529	0.00528	0.00949	0.00945
2020-06-29	0.00531	0.00530	-0.00985	-0.00990	-0.00263	-0.00264	-0.00313	-0.00314
2020-06-30	0.00352	0.00352	-0.00995	-0.01000	0.02017	0.01997	0.00795	0.00792
2020-07-01	0.01844	0.01827	0.05025	0.04903	0.00260	0.00259	-0.00328	-0.00328
2020-07-02	0.01207	0.01200	0.00478	0.00477	-0.00259	-0.00259	0.03289	0.03237
2020-07-03	0.00000	0.00000	-0.00952	-0.00957	0.01558	0.01546	-0.00637	-0.00639
2020-07-06	0.01107	0.01101	0.02404	0.02375	0.01790	0.01774	-0.02244	-0.02269
2020-07-07	0.00927	0.00922	-0.01408	-0.01418	-0.02764	-0.02803	0.02295	0.02269
2020-07-08	0.03506	0.03446	0.01429	0.01418	0.00517	0.00515	0.00962	0.00957
2020-07-09	-0.01613	-0.01626	0.02817	0.02778	-0.02314	-0.02341	-0.01270	-0.01278
2020-07-10	0.01639	0.01626	-0.00913	-0.00917	-0.00789	-0.00793	0.00000	0.00000
2020-07-13	-0.00403	-0.00404	0.06452	0.06252	-0.00796	-0.00799	-0.00643	-0.00645
2020-07-14	0.00405	0.00404	0.00000	0.00000	0.03209	0.03158	-0.00324	-0.00324
2020-07-15	-0.00806	-0.00810	-0.02165	-0.02188	-0.02073	-0.02094	0.00325	0.00324
2020-07-16	0.00488	0.00487	-0.01770	-0.01786	-0.00529	-0.00531	0.00324	0.00323
2020-07-17	-0.00971	-0.00976	0.04054	0.03974	-0.00532	-0.00533	-0.01290	-0.01299

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2020-07-20	0.00327	0.00326	-0.02165	-0.02188	0.01070	0.01064	0.00000	0.00000
2020-07-21	0.00977	0.00972	0.00442	0.00442	0.00265	0.00264	0.00000	0.00000
2020-07-22	-0.00323	-0.00323	-0.00881	-0.00885	0.01319	0.01311	0.00327	0.00326
2020-07-23	0.00324	0.00323	0.00444	0.00443	-0.01563	-0.01575	0.00000	0.00000
2020-07-24	-0.01613	-0.01626	-0.03540	-0.03604	-0.01852	-0.01869	-0.01629	-0.01642
2020-07-27	0.00000	0.00000	0.01835	0.01818	0.01078	0.01072	0.00662	0.00660
2020-07-28	0.01393	0.01384	-0.01802	-0.01818	0.00267	0.00266	-0.00658	-0.00660
2020-07-29	-0.00808	-0.00812	-0.00459	-0.00460	0.00798	0.00795	-0.00662	-0.00664
2020-07-30	0.01712	0.01697	0.00000	0.00000	-0.02639	-0.02674	0.01667	0.01653
2020-08-03	-0.01763	-0.01779	-0.05069	-0.05202	-0.01084	-0.01090	-0.04262	-0.04356
2020-08-04	0.01305	0.01297	0.00485	0.00484	0.00822	0.00819	0.01027	0.01022
2020-08-05	-0.00081	-0.00081	0.01449	0.01439	0.02717	0.02681	0.02373	0.02345
2020-08-06	0.00886	0.00882	0.06190	0.06006	0.01587	0.01575	0.00000	0.00000
2020-08-07	-0.01278	-0.01286	-0.01794	-0.01810	0.00781	0.00778	-0.01325	-0.01333
2020-08-10	-0.00971	-0.00976	0.04110	0.04027	-0.00775	-0.00778	-0.00336	-0.00336
2020-08-11	0.00899	0.00895	0.00877	0.00873	0.00000	0.00000	-0.01347	-0.01356
2020-08-12	0.01377	0.01367	-0.02609	-0.02643	0.02604	0.02571	0.00683	0.00680
2020-08-13	0.00639	0.00637	-0.00446	-0.00447	-0.01269	-0.01277	0.02034	0.02014
2020-08-14	0.01667	0.01653	0.00000	0.00000	-0.02314	-0.02341	0.00664	0.00662
2020-08-18	-0.00703	-0.00705	0.00000	0.00000	0.02632	0.02598	0.00660	0.00658
2020-08-19	-0.00472	-0.00473	0.00000	0.00000	-0.02051	-0.02073	-0.01639	-0.01653
2020-08-24	-0.00237	-0.00237	-0.01794	-0.01810	0.01571	0.01558	0.00000	0.00000
2020-08-25	0.00792	0.00789	0.00457	0.00456	0.06186	0.06002	-0.00667	-0.00669
2020-08-26	-0.00393	-0.00394	0.00909	0.00905	0.01456	0.01446	0.01342	0.01333
2020-08-27	0.04101	0.04019	0.01351	0.01342	0.02153	0.02130	-0.00993	-0.00998
2020-08-28	-0.01591	-0.01604	0.01778	0.01762	-0.00234	-0.00234	-0.01003	-0.01008
2020-08-31	-0.03387	-0.03446	-0.05240	-0.05382	-0.00939	-0.00943	-0.03378	-0.03437
2020-09-01	0.03904	0.03830	0.05069	0.04945	0.04265	0.04177	0.01399	0.01389
2020-09-02	-0.01304	-0.01312	0.06579	0.06372	-0.02727	-0.02765	0.01724	0.01709
2020-09-03	0.01010	0.01005	-0.00823	-0.00826	-0.01636	-0.01649	-0.01695	-0.01709
2020-09-04	-0.01846	-0.01863	0.00000	0.00000	-0.01663	-0.01677	-0.01379	-0.01389
2020-09-07	-0.01489	-0.01500	0.03734	0.03666	0.02174	0.02151	0.01399	0.01389
2020-09-08	0.00796	0.00792	-0.00800	-0.00803	-0.03546	-0.03611	-0.01379	-0.01389
2020-09-09	-0.01421	-0.01431	-0.03226	-0.03279	-0.00980	-0.00985	-0.02098	-0.02120
2020-09-10	-0.06966	-0.07220	-0.06667	-0.06899	-0.06931	-0.07183	-0.03571	-0.03637
2020-09-11	0.01635	0.01622	0.05357	0.05219	0.00000	0.00000	0.04074	0.03993
2020-09-14	0.02456	0.02426	0.01695	0.01681	0.04787	0.04676	0.02847	0.02807
2020-09-15	-0.03140	-0.03191	-0.00833	-0.00837	-0.02284	-0.02311	-0.02076	-0.02098
2020-09-16	-0.01877	-0.01895	-0.03781	-0.03855	-0.02338	-0.02365	-0.01413	-0.01424
2020-09-17	0.00087	0.00087	-0.01310	-0.01319	-0.01064	-0.01070	0.01075	0.01070
2020-09-18	-0.02172	-0.02196	0.01327	0.01319	0.00269	0.00268	0.02482	0.02452
2020-09-21	-0.00444	-0.00445	0.00000	0.00000	-0.01341	-0.01350	-0.02768	-0.02807
2020-09-22	-0.02765	-0.02804	0.01310	0.01302	-0.01902	-0.01920	-0.01068	-0.01073
2020-09-23	0.01009	0.01004	0.00000	0.00000	-0.01385	-0.01395	0.00719	0.00717
2020-09-24	-0.01090	-0.01096	-0.02586	-0.02620	-0.01124	-0.01130	-0.02500	-0.02532
2020-09-25	0.03030	0.02985	0.00885	0.00881	0.00000	0.00000	-0.01465	-0.01476
2020-09-28	-0.01693	-0.01708	-0.00877	-0.00881	0.03693	0.03627	-0.01115	-0.01122
2020-09-29	-0.00181	-0.00181	0.03097	0.03050	-0.00274	-0.00274	-0.01128	-0.01134
2020-09-30	-0.01544	-0.01556	-0.02575	-0.02609	0.00824	0.00821	-0.02662	-0.02698
2020-10-01	0.02768	0.02730	0.03965	0.03888	0.04360	0.04267	0.07422	0.07159
2020-10-02	-0.01167	-0.01174	-0.03390	-0.03449	-0.01305	-0.01314	-0.02545	-0.02578

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2020-10-05	0.00272	0.00272	-0.01316	-0.01325	0.00000	0.00000	-0.01119	-0.01126
2020-10-06	0.03261	0.03209	0.00000	0.00000	-0.00265	-0.00265	0.00000	0.00000
2020-10-07	0.00965	0.00960	0.00444	0.00443	0.00530	0.00529	0.00377	0.00377
2020-10-08	0.00434	0.00433	0.00000	0.00000	0.01847	0.01830	0.01504	0.01493
2020-10-09	-0.00087	-0.00087	-0.01327	-0.01336	-0.02850	-0.02891	0.01111	0.01105
2020-10-12	0.01385	0.01376	-0.00448	-0.00449	0.00267	0.00266	0.00000	0.00000
2020-10-13	0.00000	0.00000	0.03604	0.03540	-0.00532	-0.00533	0.00000	0.00000
2020-10-14	0.00769	0.00766	0.03043	0.02998	-0.00535	-0.00536	0.02930	0.02888
2020-10-15	-0.01949	-0.01968	-0.02532	-0.02564	-0.02151	-0.02174	-0.01068	-0.01073
2020-10-16	-0.00432	-0.00433	0.05628	0.05475	0.00824	0.00821	-0.01079	-0.01085
2020-10-19	0.02431	0.02402	0.00000	0.00000	0.00000	0.00000	-0.01455	-0.01465
2020-10-20	-0.01610	-0.01623	-0.00820	-0.00823	0.03270	0.03217	-0.01476	-0.01487
2020-10-21	-0.00431	-0.00432	-0.00826	-0.00830	0.00264	0.00264	0.00375	0.00374
2020-10-22	0.00346	0.00345	-0.03333	-0.03390	-0.01842	-0.01859	0.00746	0.00743
2020-10-23	-0.00517	-0.00519	-0.01293	-0.01302	0.01072	0.01067	-0.02593	-0.02627
2020-10-26	0.00780	0.00777	0.00000	0.00000	0.01857	0.01840	0.00760	0.00758
2020-10-27	-0.00430	-0.00431	-0.01747	-0.01762	-0.00260	-0.00261	-0.01132	-0.01139
2020-11-02	0.00518	0.00517	0.00444	0.00443	-0.04700	-0.04814	-0.02290	-0.02317
2020-11-03	0.01203	0.01196	0.01327	0.01319	-0.02192	-0.02216	0.00781	0.00778
2020-11-04	-0.01188	-0.01196	-0.03057	-0.03104	0.02241	0.02216	0.00000	0.00000
2020-11-05	0.05670	0.05515	0.02703	0.02667	0.09041	0.08655	0.07364	0.07106
2020-11-06	0.02439	0.02410	0.00000	0.00000	0.04020	0.03941	0.02166	0.02143
2020-11-09	-0.00238	-0.00238	-0.00877	-0.00881	-0.01208	-0.01215	0.01767	0.01751
2020-11-10	0.03103	0.03055	0.00885	0.00881	0.03667	0.03602	-0.00694	-0.00697
2020-11-11	0.00926	0.00922	0.05702	0.05545	0.05660	0.05506	0.07692	0.07411
2020-11-12	-0.01835	-0.01852	-0.02490	-0.02521	0.01116	0.01110	-0.01299	-0.01307
2020-11-13	-0.00467	-0.00468	-0.00851	-0.00855	-0.01104	-0.01110	-0.01645	-0.01658
2020-11-16	0.01408	0.01399	0.02575	0.02543	-0.02232	-0.02257	0.02676	0.02640
2020-11-17	0.01080	0.01074	-0.01255	-0.01263	0.00228	0.00228	0.04886	0.04770
2020-11-18	0.00305	0.00305	0.00847	0.00844	0.01595	0.01582	-0.01242	-0.01250
2020-11-19	0.00685	0.00683	0.04202	0.04116	0.00000	0.00000	-0.00314	-0.00315
2020-11-20	-0.00227	-0.00227	-0.02016	-0.02037	0.01794	0.01778	0.01577	0.01565
2020-11-23	0.00000	0.00000	0.06173	0.05990	0.06608	0.06399	0.03106	0.03058
2020-11-24	-0.00530	-0.00532	0.00775	0.00772	-0.03099	-0.03148	0.00904	0.00900
2020-11-25	-0.02361	-0.02389	0.00000	0.00000	0.01493	0.01482	-0.02090	-0.02112
2020-11-26	0.01092	0.01086	0.04615	0.04512	0.00210	0.00210	0.05793	0.05631
2020-11-27	-0.01466	-0.01477	0.02206	0.02182	0.00000	0.00000	-0.00288	-0.00289
2020-11-30	-0.02819	-0.02860	0.00000	0.00000	-0.01887	-0.01905	-0.06647	-0.06879
2020-12-01	0.03062	0.03016	0.00000	0.00000	0.02564	0.02532	0.00310	0.00309
2020-12-02	0.00860	0.00856	0.00000	0.00000	-0.00417	-0.00418	0.02778	0.02740
2020-12-03	0.00155	0.00155	-0.01079	-0.01085	0.00418	0.00418	-0.00901	-0.00905
2020-12-04	-0.01084	-0.01090	0.04364	0.04271	-0.02708	-0.02746	-0.01515	-0.01527
2020-12-07	0.02034	0.02014	0.03484	0.03425	0.00642	0.00640	0.02462	0.02432
2020-12-08	-0.00460	-0.00461	0.04040	0.03961	-0.01489	-0.01501	-0.00901	-0.00905
2020-12-10	0.01617	0.01604	0.00000	0.00000	0.01080	0.01074	-0.01212	-0.01220
2020-12-11	0.02433	0.02404	-0.00647	-0.00649	-0.01923	-0.01942	0.00613	0.00612
2020-12-14	0.01262	0.01254	0.02280	0.02255	0.04575	0.04474	0.01220	0.01212
2020-12-15	-0.00440	-0.00441	-0.02548	-0.02581	0.02083	0.02062	0.03916	0.03841
2020-12-16	0.02356	0.02329	0.01634	0.01621	0.02857	0.02817	0.04638	0.04533
2020-12-17	-0.00216	-0.00216	-0.04180	-0.04270	0.02579	0.02547	-0.01939	-0.01958
2020-12-18	-0.01947	-0.01966	-0.00671	-0.00673	-0.02901	-0.02944	-0.00847	-0.00851

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2020-12-21	0.00441	0.00440	0.05068	0.04943	-0.00398	-0.00399	0.00285	0.00284
2020-12-22	-0.01684	-0.01698	-0.05466	-0.05621	-0.00400	-0.00401	-0.04545	-0.04652
2020-12-23	0.00149	0.00149	0.01361	0.01351	-0.00803	-0.00806	-0.01190	-0.01198
2020-12-28	0.00818	0.00815	0.01007	0.01002	0.01619	0.01606	0.03313	0.03260
2020-12-29	-0.00221	-0.00221	-0.00997	-0.01002	0.00199	0.00199	-0.00292	-0.00292
2020-12-30	0.00074	0.00074	-0.04027	-0.04110	-0.01193	-0.01200	-0.03216	-0.03269
2021-01-04	0.00960	0.00956	0.01748	0.01733	0.01207	0.01200	0.05438	0.05295
2021-01-05	0.03731	0.03663	-0.02062	-0.02083	-0.01392	-0.01401	-0.00573	-0.00575
2021-01-06	-0.02045	-0.02066	-0.03509	-0.03572	-0.03024	-0.03071	-0.02882	-0.02924
2021-01-07	0.00288	0.00288	0.02182	0.02158	0.02287	0.02261	0.00593	0.00592
2021-01-08	0.01220	0.01213	0.02847	0.02807	0.00203	0.00203	0.05310	0.05174
2021-01-11	0.04184	0.04099	0.05536	0.05389	-0.02434	-0.02464	0.00840	0.00837
2021-01-12	-0.02519	-0.02551	-0.03279	-0.03334	0.00832	0.00828	-0.02500	-0.02532
2021-01-13	-0.00559	-0.00560	0.02712	0.02676	-0.00619	-0.00620	-0.00855	-0.00858
2021-01-14	-0.01404	-0.01414	-0.01320	-0.01329	-0.01037	-0.01043	0.00575	0.00573
2021-01-15	-0.00926	-0.00930	-0.02676	-0.02712	0.00210	0.00209	-0.00571	-0.00573
2021-01-18	0.02372	0.02345	-0.01031	-0.01036	0.05858	0.05693	-0.00862	-0.00866
2021-01-19	-0.00351	-0.00352	-0.01736	-0.01751	-0.01976	-0.01996	-0.01159	-0.01166
2021-01-20	0.00000	0.00000	0.00000	0.00000	0.00403	0.00402	0.01760	0.01744
2021-01-21	-0.00282	-0.00282	-0.00353	-0.00354	-0.01606	-0.01619	0.00288	0.00288
2021-01-22	0.00071	0.00071	-0.04255	-0.04348	-0.00816	-0.00820	-0.02586	-0.02620
2021-01-25	-0.00636	-0.00638	-0.04074	-0.04159	-0.03086	-0.03135	-0.00590	-0.00592
2021-01-26	-0.03056	-0.03104	-0.02703	-0.02740	-0.00212	-0.00213	-0.03264	-0.03319
2021-01-27	-0.00440	-0.00441	-0.01190	-0.01198	-0.01489	-0.01501	0.03681	0.03615
2021-01-28	0.01620	0.01607	-0.03614	-0.03681	-0.02592	-0.02626	-0.04142	-0.04230
2021-01-29	-0.02029	-0.02050	0.00000	0.00000	-0.05987	-0.06173	-0.04012	-0.04095
2021-02-01	0.00888	0.00884	0.02500	0.02469	0.08726	0.08366	0.03859	0.03786
2021-02-02	-0.00293	-0.00294	-0.02439	-0.02469	-0.04989	-0.05118	0.01238	0.01231
2021-02-03	0.00368	0.00367	-0.02500	-0.02532	-0.00685	-0.00687	-0.00917	-0.00922
2021-02-04	0.00440	0.00439	-0.00427	-0.00428	-0.02299	-0.02326	0.01543	0.01531
2021-02-05	0.00875	0.00871	0.03863	0.03790	0.05882	0.05716	0.00000	0.00000
2021-02-08	0.00072	0.00072	0.00826	0.00823	-0.01778	-0.01794	-0.00304	-0.00304
2021-02-09	0.00867	0.00863	-0.00820	-0.00823	-0.01357	-0.01367	-0.02439	-0.02469
2021-02-10	-0.00860	-0.00863	-0.00826	-0.00830	0.01376	0.01367	-0.00312	-0.00313
2021-02-11	-0.00578	-0.00580	0.01250	0.01242	-0.00905	-0.00909	0.00000	0.00000
2021-02-15	-0.01163	-0.01170	0.00000	0.00000	0.02055	0.02034	0.00940	0.00936
2021-02-16	0.02059	0.02038	-0.00412	-0.00412	-0.01342	-0.01351	-0.00311	-0.00311
2021-02-17	-0.00576	-0.00578	-0.02479	-0.02511	-0.00454	-0.00455	-0.01869	-0.01887
2021-02-18	-0.02391	-0.02420	-0.00424	-0.00425	-0.02506	-0.02538	0.00952	0.00948
2021-02-19	0.01336	0.01327	0.00426	0.00425	0.00467	0.00466	0.00943	0.00939
2021-02-22	-0.00513	-0.00514	0.02542	0.02511	-0.02093	-0.02115	-0.01246	-0.01254
2021-02-23	0.00515	0.00514	-0.00413	-0.00414	-0.00713	-0.00715	0.09464	0.09042
2021-02-24	-0.01465	-0.01476	-0.01660	-0.01674	-0.01675	-0.01689	0.00288	0.00288
2021-02-25	-0.00297	-0.00298	0.01266	0.01258	0.00487	0.00485	0.00287	0.00287
2021-02-26	0.00075	0.00075	-0.01667	-0.01681	-0.01211	-0.01218	0.00000	0.00000
2021-03-01	0.04993	0.04872	0.00424	0.00423	0.04657	0.04552	0.00000	0.00000
2021-03-02	-0.00426	-0.00427	0.00000	0.00000	0.09602	0.09168	-0.00860	-0.00863
2021-03-03	-0.00214	-0.00214	0.00000	0.00000	-0.03205	-0.03258	-0.00578	-0.00580
2021-03-04	-0.04000	-0.04082	0.02954	0.02911	-0.01104	-0.01110	-0.02326	-0.02353
2021-03-05	0.01190	0.01183	-0.03279	-0.03334	0.00000	0.00000	-0.01190	-0.01198
2021-03-08	-0.01176	-0.01183	-0.00424	-0.00425	-0.02455	-0.02486	0.00602	0.00601

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2021-03-09	-0.01711	-0.01726	-0.00851	-0.00855	0.00687	0.00684	-0.01497	-0.01508
2021-03-10	0.01514	0.01503	0.00429	0.00428	0.02727	0.02691	0.03343	0.03289
2021-03-12	0.00895	0.00891	0.03419	0.03362	0.01106	0.01100	0.01471	0.01460
2021-03-15	-0.01478	-0.01489	0.02066	0.02045	-0.02626	-0.02661	-0.02029	-0.02050
2021-03-16	-0.00600	-0.00602	-0.01215	-0.01222	0.00000	0.00000	-0.00592	-0.00593
2021-03-17	-0.00226	-0.00227	0.00820	0.00816	-0.00449	-0.00450	0.00893	0.00889
2021-03-18	0.01437	0.01427	0.03252	0.03200	0.02709	0.02673	0.01770	0.01754
2021-03-19	0.00820	0.00817	0.02362	0.02335	0.03736	0.03668	-0.00290	-0.00290
2021-03-22	-0.02071	-0.02093	-0.01154	-0.01161	-0.01695	-0.01709	-0.01744	-0.01760
2021-03-23	-0.00831	-0.00834	0.01167	0.01161	-0.01940	-0.01959	-0.00592	-0.00593
2021-03-24	-0.01904	-0.01922	-0.05769	-0.05942	0.01319	0.01310	0.00298	0.00297
2021-03-25	-0.01087	-0.01093	-0.01633	-0.01646	-0.02169	-0.02193	0.01187	0.01180
2021-03-26	0.00706	0.00704	0.01245	0.01237	-0.01774	-0.01790	0.02346	0.02319
2021-03-29	-0.00857	-0.00861	-0.01230	-0.01237	0.00677	0.00675	-0.02292	-0.02319
2021-03-30	0.00550	0.00549	-0.02490	-0.02521	-0.02466	-0.02497	-0.00880	-0.00884
2021-03-31	-0.02815	-0.02855	0.00000	0.00000	-0.04138	-0.04226	0.01183	0.01176
2021-04-01	0.00161	0.00161	0.00426	0.00425	0.00000	0.00000	-0.00877	-0.00881
2021-04-05	-0.01124	-0.01131	0.00424	0.00423	-0.01199	-0.01206	-0.00295	-0.00295
2021-04-06	0.00162	0.00162	0.02110	0.02088	0.00728	0.00726	-0.00296	-0.00296
2021-04-07	0.01379	0.01369	0.00826	0.00823	-0.00241	-0.00241	0.00890	0.00886
2021-04-08	-0.00464	-0.00465	0.00820	0.00816	0.00869	0.00865	-0.00294	-0.00295
2021-04-09	0.01059	0.01054	-0.02033	-0.02053	0.01220	0.01212	-0.00885	-0.00889
2021-04-12	-0.02177	-0.02201	-0.02075	-0.02097	-0.04337	-0.04434	-0.01488	-0.01499
2021-04-13	-0.01072	-0.01078	-0.01695	-0.01709	0.05793	0.05632	0.00604	0.00602
2021-04-14	0.05083	0.04958	0.03017	0.02973	0.05238	0.05106	0.00300	0.00300
2021-04-15	-0.00397	-0.00397	-0.00418	-0.00419	-0.02036	-0.02057	0.00599	0.00597
2021-04-16	-0.00080	-0.00080	-0.00420	-0.00421	-0.01617	-0.01630	0.00000	0.00000
2021-04-19	-0.00080	-0.00080	0.00000	0.00000	-0.02113	-0.02135	-0.00595	-0.00597
2021-04-20	-0.00558	-0.00560	0.00000	0.00000	0.01199	0.01192	-0.00599	-0.00601
2021-04-21	-0.01123	-0.01129	-0.00422	-0.00423	-0.00948	-0.00952	-0.01205	-0.01212
2021-04-22	0.00811	0.00808	-0.00847	-0.00851	-0.01196	-0.01203	0.01524	0.01513
2021-04-23	0.02816	0.02777	0.02564	0.02532	0.00726	0.00724	-0.00601	-0.00602
2021-04-26	-0.01643	-0.01657	0.00833	0.00830	0.01202	0.01195	-0.01813	-0.01829
2021-04-27	0.01909	0.01891	0.00000	0.00000	-0.02138	-0.02161	-0.02769	-0.02808
2021-04-28	-0.01327	-0.01336	0.00413	0.00412	0.02184	0.02161	-0.00633	-0.00635
2021-04-29	0.01424	0.01414	0.03292	0.03239	0.00238	0.00237	0.01911	0.01893
2021-04-30	-0.00078	-0.00078	-0.00797	-0.00800	-0.01185	-0.01192	0.00000	0.00000
2021-05-03	-0.00234	-0.00234	-0.00402	-0.00402	-0.02398	-0.02427	-0.00937	-0.00942
2021-05-04	0.00157	0.00156	0.00806	0.00803	-0.01229	-0.01236	0.01262	0.01254
2021-05-05	0.00391	0.00390	0.00953	0.00948	0.00995	0.00990	-0.00312	-0.00312
2021-05-06	0.00000	0.00000	-0.00837	-0.00840	-0.01232	-0.01239	-0.00312	-0.00313
2021-05-07	-0.00389	-0.00390	0.00000	0.00000	-0.02244	-0.02270	0.00000	0.00000
2021-05-10	0.00312	0.00312	0.00844	0.00840	0.01531	0.01519	-0.00627	-0.00629
2021-05-11	0.00935	0.00930	0.00000	0.00000	-0.00251	-0.00252	0.00315	0.00315
2021-05-17	0.00309	0.00308	-0.00837	-0.00840	-0.02015	-0.02036	0.00314	0.00314
2021-05-18	-0.01692	-0.01707	0.01266	0.01258	-0.02057	-0.02078	-0.00313	-0.00314
2021-05-19	-0.00704	-0.00707	-0.02083	-0.02105	-0.02625	-0.02660	-0.00629	-0.00631
2021-05-20	0.00552	0.00550	-0.00425	-0.00426	0.02695	0.02660	0.04747	0.04638
2021-05-21	-0.00313	-0.00314	-0.00427	-0.00428	-0.01050	-0.01055	-0.01208	-0.01216
2021-05-24	-0.00550	-0.00552	-0.00429	-0.00430	-0.02122	-0.02145	-0.00306	-0.00306
2021-05-25	0.00474	0.00473	0.00862	0.00858	0.01897	0.01879	0.01227	0.01220

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2021-05-27	-0.01338	-0.01347	-0.00427	-0.00428	0.01064	0.01058	0.02424	0.02395
2021-05-28	0.01116	0.01110	0.00429	0.00428	0.01579	0.01567	-0.03254	-0.03309
2021-05-31	0.00552	0.00551	0.01709	0.01695	0.00518	0.00517	0.05199	0.05068
2021-06-02	0.01412	0.01402	0.05882	0.05716	0.07474	0.07208	0.00291	0.00290
2021-06-03	0.02088	0.02067	-0.00794	-0.00797	0.01199	0.01192	0.01159	0.01153
2021-06-04	-0.00303	-0.00303	-0.01600	-0.01613	-0.01659	-0.01673	0.00287	0.00286
2021-06-07	-0.00760	-0.00763	-0.02033	-0.02053	0.00241	0.00241	0.01429	0.01418
2021-06-08	-0.01531	-0.01543	0.00415	0.00414	-0.02163	-0.02187	-0.01408	-0.01418
2021-06-09	0.01555	0.01543	0.00413	0.00412	0.02211	0.02187	0.01741	0.01726
2021-06-10	0.01378	0.01369	-0.00412	-0.00412	-0.00721	-0.00724	0.02655	0.02620
2021-06-11	-0.02266	-0.02292	0.08678	0.08322	0.00000	0.00000	-0.00287	-0.00288
2021-06-14	-0.00927	-0.00932	0.00760	0.00758	-0.01453	-0.01463	-0.01729	-0.01744
2021-06-15	0.00936	0.00932	-0.01132	-0.01139	0.00246	0.00245	0.00880	0.00876
2021-06-16	-0.01314	-0.01322	0.06489	0.06287	0.00000	0.00000	-0.00872	-0.00876
2021-06-17	-0.00861	-0.00865	-0.02867	-0.02909	-0.02941	-0.02985	-0.01760	-0.01775
2021-06-18	-0.00079	-0.00079	-0.03321	-0.03377	0.00000	0.00000	0.00000	0.00000
2021-06-21	-0.01186	-0.01193	-0.01908	-0.01927	0.01010	0.01005	-0.01493	-0.01504
2021-06-22	0.01440	0.01430	0.00389	0.00388	-0.03000	-0.03046	0.00000	0.00000
2021-06-23	-0.01735	-0.01750	-0.02326	-0.02353	0.00000	0.00000	0.02121	0.02099
2021-06-24	-0.00321	-0.00322	-0.01190	-0.01198	0.00258	0.00257	-0.03561	-0.03626
2021-06-25	-0.00322	-0.00323	0.03213	0.03162	0.00000	0.00000	0.00000	0.00000
2021-06-28	-0.02181	-0.02205	-0.03891	-0.03969	-0.04113	-0.04200	-0.02462	-0.02492
2021-06-29	-0.00165	-0.00165	-0.01619	-0.01633	0.02681	0.02646	-0.00315	-0.00316
2021-06-30	-0.00331	-0.00331	-0.00823	-0.00826	-0.00783	-0.00786	-0.00316	-0.00317
2021-07-01	0.00000	0.00000	0.00000	0.00000	-0.01053	-0.01058	-0.01270	-0.01278
2021-07-02	0.01245	0.01237	0.04564	0.04463	-0.01064	-0.01070	-0.01286	-0.01295
2021-07-05	0.00164	0.00164	-0.01984	-0.02004	-0.01075	-0.01081	-0.00977	-0.00982
2021-07-06	-0.00900	-0.00904	0.04453	0.04357	0.01087	0.01081	-0.00987	-0.00992
2021-07-07	0.00165	0.00165	-0.03101	-0.03150	-0.03763	-0.03836	0.00664	0.00662
2021-07-08	-0.00824	-0.00828	-0.02400	-0.02429	-0.01117	-0.01124	-0.00660	-0.00662
2021-07-09	0.00083	0.00083	0.02459	0.02429	0.00565	0.00563	0.04983	0.04863
2021-07-12	0.02492	0.02461	-0.01200	-0.01207	-0.01124	-0.01130	-0.00949	-0.00954
2021-07-13	-0.02026	-0.02047	-0.00405	-0.00406	-0.02273	-0.02299	-0.01917	-0.01936
2021-07-14	-0.00910	-0.00914	-0.00813	-0.00816	-0.00872	-0.00876	-0.00326	-0.00326
2021-07-15	0.02087	0.02065	0.00410	0.00409	-0.00587	-0.00588	0.02288	0.02262
2021-07-16	-0.00082	-0.00082	0.01224	0.01217	0.04425	0.04330	0.01278	0.01270
2021-07-19	-0.01719	-0.01733	0.00806	0.00803	-0.02825	-0.02866	0.01577	0.01565
2021-07-21	0.00083	0.00083	-0.00800	-0.00803	0.02907	0.02866	-0.00311	-0.00311
2021-07-22	0.02413	0.02384	0.07661	0.07382	0.00283	0.00282	0.01558	0.01546
2021-07-23	-0.01950	-0.01969	-0.02247	-0.02273	-0.01972	-0.01992	-0.02761	-0.02800
2021-07-26	-0.00497	-0.00498	-0.01916	-0.01934	-0.01724	-0.01739	0.00631	0.00629
2021-07-27	0.00000	0.00000	-0.00391	-0.00391	-0.02047	-0.02068	0.02508	0.02477
2021-07-28	-0.00416	-0.00417	-0.00392	-0.00393	-0.02090	-0.02112	-0.02446	-0.02477
2021-07-29	0.01003	0.00998	0.04331	0.04240	-0.02134	-0.02157	0.01254	0.01246
2021-07-30	-0.01159	-0.01166	0.00755	0.00752	-0.04050	-0.04134	0.00310	0.00309
2021-08-02	-0.00167	-0.00168	0.02622	0.02588	0.01299	0.01290	0.02160	0.02137
2021-08-03	0.03104	0.03057	-0.01460	-0.01471	0.06410	0.06213	0.00604	0.00602
2021-08-04	-0.00407	-0.00408	0.00741	0.00738	0.01807	0.01791	0.01201	0.01194
2021-08-05	0.02941	0.02899	-0.04412	-0.04512	-0.00592	-0.00593	-0.00890	-0.00894
2021-08-06	-0.02222	-0.02247	-0.00769	-0.00772	0.00595	0.00593	-0.00599	-0.00601
2021-08-09	0.00649	0.00647	0.00775	0.00772	-0.01183	-0.01190	-0.02108	-0.02131

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2021-08-10	0.01613	0.01600	0.03077	0.03031	0.01497	0.01486	-0.00615	-0.00617
2021-08-12	0.00000	0.00000	0.05224	0.05092	0.03540	0.03479	0.03715	0.03648
2021-08-13	0.01746	0.01731	-0.01773	-0.01789	0.01140	0.01133	-0.01493	-0.01504
2021-08-16	0.00156	0.00156	-0.01444	-0.01455	0.01972	0.01953	0.01212	0.01205
2021-08-18	0.02804	0.02765	-0.01831	-0.01848	0.04144	0.04060	0.02395	0.02367
2021-08-19	0.00000	0.00000	-0.03358	-0.03416	-0.03448	-0.03509	-0.00877	-0.00881
2021-08-20	0.00000	0.00000	-0.02317	-0.02344	0.03022	0.02977	0.00295	0.00295
2021-08-23	-0.00152	-0.00152	0.03953	0.03876	0.01867	0.01849	0.01176	0.01170
2021-08-24	0.00152	0.00152	-0.04183	-0.04272	-0.01832	-0.01849	-0.01163	-0.01170
2021-08-25	0.00000	0.00000	-0.00397	-0.00398	0.00533	0.00532	-0.00882	-0.00886
2021-08-26	-0.00606	-0.00608	0.01992	0.01972	-0.06897	-0.07146	-0.00297	-0.00297
2021-08-27	-0.00762	-0.00765	-0.02734	-0.02772	0.00855	0.00851	-0.01190	-0.01198
2021-08-30	0.00845	0.00841	0.04418	0.04323	0.03107	0.03060	0.02410	0.02381
2021-08-31	-0.00228	-0.00229	-0.03077	-0.03125	0.01370	0.01361	0.00000	0.00000
2021-09-01	0.00229	0.00229	0.04762	0.04652	-0.02432	-0.02463	-0.01765	-0.01780
2021-09-02	-0.00381	-0.00382	0.00000	0.00000	0.01385	0.01376	0.00898	0.00894
2021-09-03	0.00917	0.00913	0.02652	0.02617	0.00546	0.00545	0.00593	0.00592
2021-09-06	-0.00379	-0.00379	0.00738	0.00735	0.00000	0.00000	0.00000	0.00000
2021-09-07	-0.00076	-0.00076	-0.01465	-0.01476	-0.02989	-0.03035	-0.00295	-0.00295
2021-09-08	-0.01979	-0.01999	-0.01115	-0.01122	-0.04202	-0.04292	-0.01479	-0.01490
2021-09-09	0.02019	0.01999	0.02632	0.02598	0.02339	0.02312	0.01802	0.01786
2021-09-10	-0.00761	-0.00764	-0.01099	-0.01105	0.01714	0.01700	-0.01770	-0.01786
2021-09-13	0.00537	0.00535	-0.00741	-0.00743	0.00000	0.00000	0.00601	0.00599
2021-09-14	-0.00153	-0.00153	0.03358	0.03303	0.00000	0.00000	0.02687	0.02651
2021-09-15	-0.00764	-0.00767	0.00361	0.00360	0.00000	0.00000	0.00291	0.00290
2021-09-16	0.00077	0.00077	0.00000	0.00000	-0.01966	-0.01986	-0.00290	-0.00290
2021-09-17	0.00308	0.00307	-0.03957	-0.04037	-0.00573	-0.00575	0.02616	0.02583
2021-09-20	0.00997	0.00992	-0.01124	-0.01130	-0.02017	-0.02038	-0.00283	-0.00284
2021-09-21	-0.01443	-0.01453	0.01894	0.01876	-0.00882	-0.00886	0.00284	0.00284
2021-09-22	0.01002	0.00997	0.04089	0.04008	0.01484	0.01473	0.01700	0.01685
2021-09-23	0.00381	0.00381	0.01071	0.01066	-0.00585	-0.00587	-0.00836	-0.00839
2021-09-24	0.00076	0.00076	0.06007	0.05834	-0.02353	-0.02381	0.00000	0.00000
2021-09-27	-0.00076	-0.00076	0.00667	0.00664	-0.01205	-0.01212	-0.01124	-0.01130
2021-09-28	-0.00912	-0.00916	0.15232	0.14178	-0.04268	-0.04362	0.00284	0.00284
2021-09-29	0.00920	0.00916	-0.01149	-0.01156	0.07962	0.07661	0.00000	0.00000
2021-09-30	0.06383	0.06188	0.02326	0.02299	-0.03245	-0.03299	0.04533	0.04433
2021-10-01	-0.03429	-0.03489	0.01136	0.01130	-0.01220	-0.01227	-0.00813	-0.00816
2021-10-04	0.02959	0.02916	0.04494	0.04396	0.02161	0.02137	0.00273	0.00273
2021-10-05	-0.00216	-0.00216	-0.01613	-0.01626	-0.03625	-0.03693	0.00000	0.00000
2021-10-06	0.03384	0.03328	0.01913	0.01895	0.00000	0.00000	0.02452	0.02423
2021-10-07	-0.00279	-0.00279	-0.06971	-0.07225	0.01254	0.01246	-0.01064	-0.01070
2021-10-08	0.01816	0.01799	0.04611	0.04508	0.00000	0.00000	0.02151	0.02128
2021-10-11	-0.00480	-0.00481	0.05510	0.05363	-0.02477	-0.02508	0.00263	0.00263
2021-10-12	0.00896	0.00892	-0.01567	-0.01579	0.00635	0.00633	-0.00787	-0.00791
2021-10-13	0.02801	0.02762	-0.01326	-0.01335	0.04732	0.04623	-0.00265	-0.00265
2021-10-14	0.02990	0.02946	-0.00269	-0.00269	0.07530	0.07260	0.01592	0.01579
2021-10-15	-0.01290	-0.01299	0.00270	0.00269	-0.01681	-0.01695	-0.00522	-0.00524
2021-10-18	-0.01634	-0.01647	0.00806	0.00803	0.01424	0.01414	-0.00525	-0.00526
2021-10-19	-0.00332	-0.00333	-0.01067	-0.01072	-0.01404	-0.01414	-0.01319	-0.01328
2021-10-21	-0.01333	-0.01342	-0.05391	-0.05542	0.00285	0.00284	0.02941	0.02899
2021-10-22	0.01689	0.01675	-0.01140	-0.01146	0.00568	0.00567	0.00519	0.00518

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2021-10-25	0.00000	0.00000	0.00576	0.00575	-0.03107	-0.03157	-0.02326	-0.02353
2021-10-26	0.00000	0.00000	0.01146	0.01140	-0.02332	-0.02360	0.00529	0.00528
2021-10-27	-0.00997	-0.01002	-0.01983	-0.02003	0.03284	0.03231	-0.00263	-0.00264
2021-10-28	-0.01007	-0.01012	-0.05780	-0.05954	0.02023	0.02003	-0.01055	-0.01061
2021-10-29	0.01356	0.01347	0.03067	0.03021	0.03116	0.03069	0.01333	0.01325
2021-11-01	-0.01003	-0.01008	0.00298	0.00297	0.01374	0.01364	-0.02632	-0.02667
2021-11-02	-0.01351	-0.01361	-0.02374	-0.02403	-0.02710	-0.02747	-0.01081	-0.01087
2021-11-03	0.00685	0.00683	0.01824	0.01807	0.05571	0.05421	0.01639	0.01626
2021-11-04	0.00340	0.00340	0.00000	0.00000	0.04485	0.04388	0.01344	0.01335
2021-11-05	0.01017	0.01012	-0.01791	-0.01807	0.00000	0.00000	0.00000	0.00000
2021-11-08	0.01678	0.01664	0.03040	0.02994	-0.03283	-0.03338	-0.00530	-0.00532
2021-11-09	0.01320	0.01311	0.00590	0.00588	0.00783	0.00780	-0.01067	-0.01072
2021-11-10	-0.00326	-0.00326	-0.00880	-0.00884	-0.00777	-0.00780	-0.00270	-0.00270
2021-11-11	0.00327	0.00326	-0.00296	-0.00296	-0.01567	-0.01579	-0.01351	-0.01361
2021-11-12	-0.01954	-0.01974	0.00000	0.00000	0.00000	0.00000	-0.01370	-0.01379
2021-11-15	-0.00332	-0.00333	-0.04451	-0.04553	-0.02387	-0.02416	0.00000	0.00000
2021-11-16	-0.00333	-0.00334	0.00311	0.00310	-0.00272	-0.00272	0.01944	0.01926
2021-11-17	0.01678	0.01664	0.01548	0.01536	-0.01362	-0.01372	-0.01090	-0.01096
2021-11-18	-0.02310	-0.02337	-0.00915	-0.00919	-0.00829	-0.00832	0.00000	0.00000
2021-11-19	0.00338	0.00337	0.01231	0.01223	-0.00279	-0.00279	0.07438	0.07174
2021-11-22	0.00673	0.00671	-0.00304	-0.00304	-0.00279	-0.00280	-0.02564	-0.02598
2021-11-23	0.00000	0.00000	0.03963	0.03887	-0.04202	-0.04292	0.00000	0.00000
2021-11-24	0.00000	0.00000	-0.00587	-0.00588	0.00292	0.00292	0.00000	0.00000
2021-11-25	-0.00669	-0.00671	0.02950	0.02907	-0.00583	-0.00585	0.05263	0.05129
2021-11-26	-0.02020	-0.02041	-0.05158	-0.05295	-0.02933	-0.02976	0.00250	0.00250
2021-11-29	0.01718	0.01704	0.02115	0.02093	-0.00906	-0.00910	0.03242	0.03190
2021-11-30	-0.01689	-0.01704	0.00592	0.00590	-0.02439	-0.02469	-0.03623	-0.03690
2021-12-01	0.00344	0.00343	0.03235	0.03184	0.01250	0.01242	0.02256	0.02231
2021-12-02	0.02740	0.02703	0.03134	0.03086	-0.00309	-0.00309	0.01961	0.01942
2021-12-03	-0.01667	-0.01681	0.00276	0.00276	-0.02167	-0.02191	-0.02163	-0.02187
2021-12-06	-0.00339	-0.00340	0.05234	0.05102	0.02848	0.02808	0.02457	0.02427
2021-12-07	0.00000	0.00000	-0.00262	-0.00262	-0.00615	-0.00617	-0.00480	-0.00481
2021-12-08	0.01020	0.01015	0.01050	0.01044	-0.00619	-0.00621	-0.01446	-0.01456
2021-12-09	-0.01010	-0.01015	0.00519	0.00518	0.00000	0.00000	0.01467	0.01456
2021-12-10	0.00340	0.00340	-0.00775	-0.00778	-0.01869	-0.01887	0.00000	0.00000
2021-12-13	-0.01017	-0.01022	0.00000	0.00000	0.00317	0.00317	-0.01687	-0.01701
2021-12-14	0.00000	0.00000	0.00000	0.00000	-0.01899	-0.01917	0.00490	0.00489
2021-12-15	0.00000	0.00000	0.05208	0.05077	-0.00645	-0.00647	-0.00488	-0.00489
2021-12-16	-0.00342	-0.00343	0.00990	0.00985	-0.01623	-0.01637	0.00490	0.00489
2021-12-17	0.03093	0.03046	0.00490	0.00489	-0.01980	-0.02000	0.00000	0.00000
2021-12-20	-0.01667	-0.01681	0.00000	0.00000	-0.01347	-0.01356	-0.00488	-0.00489
2021-12-21	0.00000	0.00000	0.03415	0.03358	0.01365	0.01356	0.01961	0.01942
2021-12-22	-0.00678	-0.00680	0.01415	0.01405	-0.02357	-0.02385	-0.01202	-0.01209
2021-12-23	-0.00341	-0.00342	0.00000	0.00000	0.00690	0.00687	0.01217	0.01209
2021-12-24	0.00000	0.00000	0.03256	0.03204	-0.00342	-0.00343	-0.01202	-0.01209
2021-12-27	0.00685	0.00683	-0.01802	-0.01818	0.01718	0.01704	-0.00730	-0.00733
2021-12-28	0.00000	0.00000	0.05505	0.05358	-0.00676	-0.00678	0.00245	0.00245
2021-12-29	-0.00680	-0.00683	0.00435	0.00434	-0.01701	-0.01715	-0.00244	-0.00245
2021-12-30	0.00000	0.00000	0.04659	0.04554	0.00346	0.00345	-0.00980	-0.00985
2022-01-03	0.00342	0.00342	0.05333	0.05196	0.00345	0.00344	0.03465	0.03407
2022-01-04	0.01024	0.01019	-0.02954	-0.02998	0.01375	0.01365	-0.00239	-0.00240

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2022-01-05	0.00676	0.00673	-0.02609	-0.02643	-0.02034	-0.02055	-0.02878	-0.02920
2022-01-06	0.00336	0.00335	0.02679	0.02643	-0.00692	-0.00694	0.01728	0.01714
2022-01-07	0.02341	0.02314	0.05652	0.05498	-0.00348	-0.00349	0.01214	0.01206
2022-01-10	-0.00654	-0.00656	-0.01235	-0.01242	-0.00699	-0.00702	-0.01679	-0.01693
2022-01-11	0.01316	0.01307	-0.02500	-0.02532	-0.02113	-0.02135	-0.00244	-0.00244
2022-01-12	0.00000	0.00000	-0.01282	-0.01290	0.00360	0.00359	0.00489	0.00488
2022-01-13	0.00000	0.00000	-0.01299	-0.01307	0.01434	0.01424	0.01703	0.01689
2022-01-14	0.01948	0.01929	-0.00439	-0.00440	-0.01060	-0.01066	0.00239	0.00239
2022-01-17	-0.01274	-0.01282	-0.00441	-0.00441	-0.02500	-0.02532	-0.00239	-0.00239
2022-01-18	-0.00968	-0.00972	-0.01770	-0.01786	-0.01465	-0.01476	0.01675	0.01661
2022-01-19	0.00000	0.00000	0.00450	0.00449	-0.00743	-0.00746	0.00000	0.00000
2022-01-20	0.01303	0.01295	0.03587	0.03525	-0.00749	-0.00752	-0.00706	-0.00708
2022-01-21	0.02251	0.02226	0.00000	0.00000	0.04528	0.04429	0.02607	0.02573
2022-01-24	-0.01887	-0.01905	-0.01732	-0.01747	0.00722	0.00719	-0.00693	-0.00695
2022-01-25	-0.00321	-0.00321	-0.02643	-0.02679	-0.02509	-0.02541	-0.00233	-0.00233
2022-01-26	-0.00965	-0.00969	0.00000	0.00000	0.01103	0.01097	0.02331	0.02304
2022-01-27	0.01299	0.01290	0.04525	0.04425	-0.01818	-0.01835	-0.02733	-0.02772
2022-01-28	-0.00321	-0.00321	-0.01299	-0.01307	0.01111	0.01105	-0.00234	-0.00234
2022-01-31	-0.01929	-0.01948	-0.01754	-0.01770	-0.01465	-0.01476	-0.01643	-0.01657
2022-02-02	0.02295	0.02269	-0.01339	-0.01348	0.02230	0.02206	-0.00477	-0.00478
2022-02-03	-0.00962	-0.00966	0.00452	0.00451	0.00364	0.00363	0.00719	0.00717
2022-02-04	0.00000	0.00000	-0.01802	-0.01818	0.01812	0.01795	0.00714	0.00712
2022-02-07	0.00971	0.00966	0.00917	0.00913	0.02847	0.02807	0.01418	0.01408
2022-02-08	-0.00962	-0.00966	0.00000	0.00000	-0.01038	-0.01043	-0.01399	-0.01408
2022-02-09	0.02913	0.02871	0.00455	0.00454	0.02448	0.02418	0.01182	0.01175
2022-02-10	-0.02516	-0.02548	-0.02262	-0.02288	-0.00683	-0.00685	0.04206	0.04120
2022-02-11	0.00968	0.00963	0.00463	0.00462	0.01375	0.01365	-0.00224	-0.00224
2022-02-14	-0.01597	-0.01610	0.02765	0.02727	-0.02373	-0.02401	-0.01124	-0.01130
2022-02-15	0.02273	0.02247	0.02691	0.02655	0.01042	0.01036	0.00227	0.00227
2022-02-16	0.01270	0.01262	-0.02183	-0.02208	0.01718	0.01704	-0.00227	-0.00227
2022-02-17	-0.00940	-0.00945	-0.00893	-0.00897	-0.01014	-0.01019	0.00227	0.00227
2022-02-18	0.00316	0.00316	0.00901	0.00897	0.00341	0.00341	-0.00227	-0.00227
2022-02-21	0.00315	0.00315	-0.00893	-0.00897	-0.01020	-0.01026	-0.00682	-0.00684
2022-02-22	-0.00629	-0.00631	0.00450	0.00449	-0.00344	-0.00344	-0.01602	-0.01615
2022-02-23	0.01899	0.01881	0.04036	0.03957	-0.00690	-0.00692	0.01163	0.01156
2022-02-24	-0.00621	-0.00623	0.06897	0.06669	-0.04167	-0.04256	-0.02529	-0.02561
2022-02-25	0.00625	0.00623	-0.01210	-0.01217	0.04348	0.04256	0.02358	0.02331
2022-03-01	0.00000	0.00000	0.05306	0.05170	-0.01389	-0.01399	0.00230	0.00230
2022-03-02	-0.00932	-0.00936	0.01163	0.01156	-0.05282	-0.05426	-0.01379	-0.01389
2022-03-04	-0.00940	-0.00945	0.16475	0.15251	-0.01487	-0.01498	0.02564	0.02532
2022-03-07	-0.02532	-0.02564	0.06579	0.06372	-0.04906	-0.05030	0.02500	0.02469
2022-03-08	-0.00649	-0.00651	-0.03704	-0.03774	0.01190	0.01183	0.02439	0.02410
2022-03-09	0.02614	0.02581	-0.01282	-0.01290	0.00784	0.00781	-0.03896	-0.03974
2022-03-10	0.00955	0.00951	-0.03247	-0.03301	0.08171	0.07855	-0.00676	-0.00678
2022-03-11	0.00315	0.00315	0.00671	0.00669	-0.01439	-0.01449	0.04082	0.04001
2022-03-14	0.01572	0.01560	-0.07000	-0.07257	-0.01095	-0.01101	-0.00218	-0.00218
2022-03-15	0.00929	0.00924	-0.02867	-0.02909	-0.02214	-0.02239	0.00873	0.00870
2022-03-16	0.00614	0.00612	0.04059	0.03979	0.00755	0.00752	0.00216	0.00216
2022-03-17	-0.02439	-0.02469	-0.02837	-0.02878	0.01124	0.01117	-0.03240	-0.03293
2022-03-18	-0.01250	-0.01258	-0.00730	-0.00733	-0.01481	-0.01493	0.01339	0.01330
2022-03-21	0.00000	0.00000	0.00000	0.00000	-0.01128	-0.01134	0.00220	0.00220

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2022-03-22	0.00316	0.00316	0.02941	0.02899	0.01141	0.01134	0.00000	0.00000
2022-03-23	-0.00315	-0.00316	0.00000	0.00000	-0.01128	-0.01134	0.00000	0.00000
2022-03-24	0.00316	0.00316	0.02857	0.02817	0.00000	0.00000	0.00220	0.00220
2022-03-25	0.00315	0.00315	-0.02083	-0.02105	-0.01141	-0.01147	-0.00877	-0.00881
2022-03-28	0.00894	0.00890	0.01418	0.01408	0.04231	0.04144	0.01770	0.01754
2022-03-29	-0.00633	-0.00635	-0.03147	-0.03197	-0.01476	-0.01487	-0.00870	-0.00873
2022-03-30	0.00318	0.00318	-0.03249	-0.03303	0.01124	0.01117	0.00439	0.00438
2022-03-31	0.01270	0.01262	0.00373	0.00372	-0.01481	-0.01493	0.00000	0.00000
2022-04-01	-0.00627	-0.00629	0.04461	0.04364	0.01880	0.01862	0.00000	0.00000
2022-04-04	-0.00315	-0.00316	0.00000	0.00000	0.01107	0.01101	-0.00873	-0.00877
2022-04-05	0.00000	0.00000	0.06762	0.06543	-0.00365	-0.00366	-0.00441	-0.00441
2022-04-06	-0.01899	-0.01917	0.02333	0.02307	-0.01831	-0.01848	-0.01991	-0.02011
2022-04-07	0.00000	0.00000	-0.00651	-0.00654	-0.01493	-0.01504	0.02483	0.02453
2022-04-08	0.01290	0.01282	0.03607	0.03543	0.01136	0.01130	0.00881	0.00877
2022-04-11	-0.01592	-0.01605	-0.00633	-0.00635	-0.02343	-0.02371	0.00218	0.00218
2022-04-12	0.00971	0.00966	0.00000	0.00000	-0.00787	-0.00791	0.00871	0.00868
2022-04-13	0.00000	0.00000	0.06688	0.06474	-0.01587	-0.01600	0.01296	0.01288
2022-04-14	-0.01282	-0.01290	-0.01194	-0.01201	-0.02823	-0.02863	-0.00213	-0.00213
2022-04-18	0.00000	0.00000	-0.00604	-0.00606	0.02075	0.02053	0.00641	0.00639
2022-04-19	-0.00974	-0.00979	-0.00608	-0.00610	-0.01626	-0.01639	-0.00637	-0.00639
2022-04-20	0.00328	0.00327	-0.01223	-0.01231	-0.01653	-0.01667	-0.01068	-0.01074
2022-04-21	0.03595	0.03532	0.01238	0.01231	0.02101	0.02079	-0.00216	-0.00216
2022-04-22	-0.00631	-0.00633	-0.01835	-0.01852	-0.01235	-0.01242	0.00000	0.00000
2022-04-25	0.01587	0.01575	0.00000	0.00000	-0.00417	-0.00418	0.01948	0.01929
2022-04-26	0.01562	0.01550	-0.01558	-0.01570	0.07113	0.06871	0.01062	0.01056
2022-04-27	0.00923	0.00919	0.00316	0.00316	0.01172	0.01165	0.00210	0.00210
2022-04-28	-0.00915	-0.00919	0.05363	0.05224	-0.01158	-0.01165	-0.03145	-0.03195
2022-05-09	-0.06462	-0.06680	-0.05389	-0.05540	-0.04297	-0.04392	-0.06710	-0.06946
2022-05-10	-0.00987	-0.00992	-0.02215	-0.02240	-0.00408	-0.00409	0.00696	0.00694
2022-05-11	0.01661	0.01647	0.01942	0.01923	0.06148	0.05966	-0.00230	-0.00231
2022-05-12	-0.04902	-0.05026	0.00000	0.00000	-0.03089	-0.03138	-0.00693	-0.00695
2022-05-13	0.00687	0.00685	0.01905	0.01887	0.00398	0.00398	-0.00930	-0.00935
2022-05-17	0.01024	0.01019	0.04933	0.04816	0.00794	0.00791	-0.01878	-0.01896
2022-05-18	0.02365	0.02337	-0.00932	-0.00936	0.01968	0.01949	0.01675	0.01661
2022-05-19	-0.01650	-0.01664	-0.02508	-0.02540	-0.03861	-0.03938	-0.00941	-0.00946
2022-05-20	-0.00671	-0.00673	0.06431	0.06233	0.02008	0.01988	-0.00950	-0.00955
2022-05-23	-0.00338	-0.00338	-0.04834	-0.04955	0.02362	0.02335	-0.01439	-0.01449
2022-05-24	-0.00339	-0.00340	0.01587	0.01575	0.03077	0.03031	0.00973	0.00969
2022-05-25	0.00340	0.00340	-0.03125	-0.03175	-0.02612	-0.02647	0.01205	0.01198
2022-05-27	0.02712	0.02676	0.01936	0.01917	0.02682	0.02647	0.02381	0.02353
2022-05-30	0.00000	0.00000	-0.03165	-0.03216	0.07090	0.06850	-0.01163	-0.01170
2022-05-31	0.02310	0.02284	0.06863	0.06638	0.01742	0.01727	0.01412	0.01402
2022-06-02	-0.02258	-0.02284	0.01835	0.01818	-0.01712	-0.01727	-0.00464	-0.00465
2022-06-03	0.00330	0.00329	0.05706	0.05549	-0.01045	-0.01051	0.00466	0.00465
2022-06-06	-0.01974	-0.01993	-0.01705	-0.01719	0.01408	0.01399	0.00696	0.00694
2022-06-07	-0.01007	-0.01012	0.05491	0.05346	-0.00694	-0.00697	-0.02074	-0.02096
2022-06-08	0.03051	0.03005	-0.00548	-0.00549	0.01049	0.01043	-0.00001	-0.00001
2022-06-09	-0.01316	-0.01325	-0.00826	-0.00830	-0.03114	-0.03164	-0.01220	-0.01227
2022-06-10	-0.02000	-0.02020	-0.04722	-0.04837	-0.01429	-0.01439	-0.00741	-0.00743
2022-06-13	0.00000	0.00000	-0.06997	-0.07254	-0.05072	-0.05206	0.01493	0.01482
2022-06-14	0.00680	0.00678	0.02194	0.02171	0.04198	0.04113	0.00735	0.00733

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2022-06-15	-0.01014	-0.01019	-0.06748	-0.06987	0.01099	0.01093	-0.01217	-0.01224
2022-06-16	0.03413	0.03356	0.00658	0.00656	0.00000	0.00000	-0.00493	-0.00494
2022-06-17	-0.00990	-0.00995	-0.04575	-0.04683	-0.01087	-0.01093	0.01980	0.01961
2022-06-20	0.01667	0.01653	0.00342	0.00342	0.04029	0.03950	-0.01942	-0.01961
2022-06-21	0.00328	0.00327	0.03413	0.03356	0.00704	0.00702	0.01733	0.01718
2022-06-22	-0.01961	-0.01980	0.00000	0.00000	0.00000	0.00000	-0.01460	-0.01471
2022-06-23	0.00333	0.00333	-0.02970	-0.03015	0.02098	0.02076	0.02716	0.02680
2022-06-24	-0.00664	-0.00667	0.01020	0.01015	0.04452	0.04356	-0.00240	-0.00241
2022-06-27	-0.01672	-0.01686	-0.00337	-0.00337	-0.03934	-0.04014	-0.01928	-0.01947
2022-06-28	-0.00680	-0.00683	-0.00338	-0.00338	-0.03072	-0.03120	-0.00983	-0.00988
2022-06-29	-0.00342	-0.00343	-0.02712	-0.02749	0.02817	0.02778	0.00248	0.00248
2022-06-30	-0.00344	-0.00344	-0.00348	-0.00349	-0.02397	-0.02426	-0.00990	-0.00995
2022-07-01	0.00000	0.00000	-0.04895	-0.05019	-0.03509	-0.03572	0.00500	0.00499
2022-07-04	-0.02759	-0.02797	0.01838	0.01822	-0.01091	-0.01097	-0.00746	-0.00749
2022-07-05	0.02837	0.02797	0.03971	0.03894	-0.01471	-0.01481	0.00000	0.00000
2022-07-06	0.00690	0.00687	-0.03819	-0.03894	0.00373	0.00372	0.00752	0.00749
2022-07-07	-0.02740	-0.02778	0.00000	0.00000	-0.01487	-0.01498	0.00498	0.00496
2022-07-08	0.00704	0.00702	0.02166	0.02143	0.00755	0.00752	-0.00743	-0.00745
2022-07-11	-0.00350	-0.00350	0.00000	0.00000	-0.02622	-0.02657	0.00748	0.00745
2022-07-12	0.00702	0.00699	0.03180	0.03131	-0.01538	-0.01550	-0.00743	-0.00745
2022-07-13	-0.02439	-0.02469	0.00342	0.00342	0.00000	0.00000	-0.01746	-0.01761
2022-07-14	0.00357	0.00357	0.00000	0.00000	0.01953	0.01934	0.02030	0.02010
2022-07-15	-0.00356	-0.00357	-0.05802	-0.05977	-0.02299	-0.02326	0.03234	0.03183
2022-07-18	0.02143	0.02120	0.01087	0.01081	0.01961	0.01942	0.00964	0.00959
2022-07-19	0.00350	0.00349	0.05735	0.05576	0.00769	0.00766	-0.00716	-0.00719
2022-07-20	0.03136	0.03088	0.02373	0.02345	-0.00763	-0.00766	0.01202	0.01195
2022-07-21	0.00000	0.00000	0.00000	0.00000	0.00385	0.00384	0.01188	0.01181
2022-07-22	-0.01014	-0.01019	0.00331	0.00331	0.00000	0.00000	-0.00704	-0.00707
2022-07-25	-0.00341	-0.00342	0.02310	0.02284	-0.00383	-0.00384	0.00236	0.00236
2022-07-26	0.00000	0.00000	0.01936	0.01917	0.01154	0.01147	0.00236	0.00236
2022-07-27	0.00342	0.00342	0.03797	0.03727	-0.02281	-0.02308	0.01176	0.01170
2022-07-28	0.00341	0.00341	0.00000	0.00000	0.00000	0.00000	-0.01395	-0.01405
2022-07-29	0.00000	0.00000	-0.00915	-0.00919	0.01556	0.01544	-0.00236	-0.00236
2022-08-01	0.02041	0.02020	0.03077	0.03031	-0.01149	-0.01156	0.01655	0.01641
2022-08-02	0.01333	0.01325	-0.05075	-0.05208	0.00000	0.00000	0.03488	0.03429
2022-08-03	0.00329	0.00328	0.04088	0.04007	0.00388	0.00387	0.01124	0.01117
2022-08-04	0.02295	0.02269	-0.03625	-0.03693	0.04247	0.04159	0.01111	0.01105
2022-08-05	0.00962	0.00957	-0.01881	-0.01899	0.00741	0.00738	0.02198	0.02174
2022-08-08	0.00000	0.00000	-0.01597	-0.01610	0.02941	0.02899	0.01075	0.01070
2022-08-09	0.00317	0.00317	0.01623	0.01610	-0.00714	-0.00717	-0.01277	-0.01285
2022-08-10	0.00000	0.00000	0.00319	0.00319	0.00719	0.00717	-0.01724	-0.01739
2022-08-11	0.00633	0.00631	0.00318	0.00318	-0.01071	-0.01077	0.00219	0.00219
2022-08-12	-0.00314	-0.00315	0.01270	0.01262	0.01444	0.01434	-0.00438	-0.00439
2022-08-15	0.00315	0.00315	-0.00940	-0.00945	-0.01423	-0.01434	-0.02198	-0.02222
2022-08-16	0.00314	0.00314	-0.00316	-0.00317	-0.00722	-0.00725	-0.00899	-0.00903
2022-08-18	0.00313	0.00313	0.03175	0.03125	-0.00727	-0.00730	0.03175	0.03125
2022-08-19	-0.01250	-0.01258	0.00000	0.00000	-0.03663	-0.03732	0.01099	0.01093
2022-08-22	0.01266	0.01258	-0.03077	-0.03125	0.00380	0.00380	0.01087	0.01081
2022-08-23	-0.01250	-0.01258	0.06349	0.06156	0.01136	0.01130	0.00215	0.00215
2022-08-24	0.00633	0.00631	0.01791	0.01775	-0.01498	-0.01509	0.01931	0.01913
2022-08-25	0.01572	0.01560	0.00000	0.00000	-0.00380	-0.00381	-0.03789	-0.03863

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2022-08-26	-0.00929	-0.00933	0.00000	0.00000	0.00000	0.00000	-0.01751	-0.01766
2022-08-29	0.01875	0.01858	0.01466	0.01456	-0.00382	-0.00382	0.00668	0.00666
2022-08-30	0.00307	0.00306	0.02312	0.02286	0.00383	0.00382	-0.00885	-0.00889
2022-08-31	0.00306	0.00305	0.00000	0.00000	0.00763	0.00760	0.01786	0.01770
2022-09-01	-0.00610	-0.00612	0.04520	0.04421	-0.02273	-0.02299	0.00439	0.00438
2022-09-02	0.00920	0.00916	0.02162	0.02139	-0.00388	-0.00388	0.00437	0.00436
2022-09-05	0.00608	0.00606	0.06614	0.06404	0.01556	0.01544	0.00435	0.00434
2022-09-06	0.00000	0.00000	0.00248	0.00248	0.00000	0.00000	-0.02381	-0.02410
2022-09-07	0.01208	0.01201	-0.00495	-0.00496	0.00383	0.00382	-0.00443	-0.00444
2022-09-08	-0.00299	-0.00299	-0.01990	-0.02010	0.00000	0.00000	0.01559	0.01547
2022-09-09	0.00299	0.00299	0.00254	0.00253	0.00763	0.00760	0.00877	0.00873
2022-09-12	0.00000	0.00000	0.00506	0.00505	0.00379	0.00378	-0.01304	-0.01313
2022-09-13	0.01791	0.01775	0.01008	0.01003	0.01509	0.01498	-0.00441	-0.00441
2022-09-14	-0.00293	-0.00294	0.00000	0.00000	-0.01115	-0.01121	-0.00885	-0.00889
2022-09-15	0.02941	0.02899	0.01247	0.01239	-0.00376	-0.00377	0.00446	0.00445
2022-09-16	-0.03429	-0.03489	-0.03202	-0.03254	0.03774	0.03704	-0.02000	-0.02020
2022-09-19	0.02367	0.02339	-0.00509	-0.00510	0.01818	0.01802	0.02041	0.02020
2022-09-20	-0.01156	-0.01163	-0.00512	-0.00513	0.02857	0.02817	-0.00444	-0.00445
2022-09-21	-0.00877	-0.00881	0.00514	0.00513	0.00000	0.00000	-0.01116	-0.01122
2022-09-22	0.00000	0.00000	0.04859	0.04745	0.01042	0.01036	0.00000	0.00000
2022-09-23	-0.01180	-0.01187	-0.01220	-0.01227	0.02062	0.02041	-0.01129	-0.01135
2022-09-26	0.00597	0.00595	-0.04691	-0.04805	0.00000	0.00000	0.01826	0.01810
2022-09-27	-0.01484	-0.01495	0.01295	0.01287	0.01010	0.01005	-0.00448	-0.00449
2022-09-28	0.00301	0.00301	-0.01023	-0.01028	-0.01333	-0.01342	0.00676	0.00673
2022-09-29	0.00601	0.00599	0.02067	0.02046	0.01351	0.01342	-0.00671	-0.00673
2022-09-30	0.02090	0.02068	0.00253	0.00253	-0.00333	-0.00334	0.00450	0.00449
2022-10-03	-0.00585	-0.00587	0.00000	0.00000	-0.00334	-0.00335	0.00000	0.00000
2022-10-04	0.00588	0.00587	0.04040	0.03961	0.00000	0.00000	-0.00448	-0.00449
2022-10-05	-0.01170	-0.01176	-0.00971	-0.00976	-0.01678	-0.01692	0.00450	0.00449
2022-10-06	-0.00296	-0.00296	0.00490	0.00489	0.00000	0.00000	-0.00673	-0.00675
2022-10-07	-0.02671	-0.02707	0.00976	0.00971	-0.01365	-0.01375	-0.01806	-0.01822
2022-10-10	0.01219	0.01212	-0.05314	-0.05460	0.01730	0.01715	0.01609	0.01596
2022-10-11	-0.00602	-0.00604	0.00255	0.00255	0.02381	0.02353	-0.02489	-0.02520
2022-10-12	0.00909	0.00905	0.01781	0.01766	-0.00332	-0.00333	0.00696	0.00694
2022-10-13	-0.00601	-0.00602	0.00500	0.00499	-0.02000	-0.02020	-0.00922	-0.00926
2022-10-14	-0.00302	-0.00303	-0.01741	-0.01757	-0.02041	-0.02062	-0.00233	-0.00233
2022-10-17	0.00000	0.00000	0.00253	0.00253	0.04167	0.04082	0.00932	0.00928
2022-10-18	0.00606	0.00604	-0.02525	-0.02558	0.00333	0.00333	-0.01848	-0.01865
2022-10-19	-0.00301	-0.00302	-0.00518	-0.00519	0.02990	0.02946	-0.01176	-0.01183
2022-10-20	0.02719	0.02683	0.04167	0.04082	0.00645	0.00643	0.03571	0.03509
2022-10-21	0.01765	0.01749	-0.01000	-0.01005	-0.02564	-0.02598	0.00230	0.00230
2022-10-24	0.02890	0.02849	-0.01010	-0.01015	-0.00329	-0.00329	0.01147	0.01140
2022-10-25	-0.02247	-0.02273	-0.01020	-0.01026	-0.01320	-0.01329	-0.00454	-0.00455
2022-10-26	-0.01724	-0.01739	0.01289	0.01280	0.00000	0.00000	-0.00456	-0.00457
2022-10-27	0.01754	0.01739	0.01272	0.01264	0.02676	0.02640	0.00000	0.00000
2022-10-28	0.00575	0.00573	-0.01759	-0.01774	0.00977	0.00972	0.01831	0.01814
2022-10-31	0.00571	0.00570	0.01790	0.01774	0.02581	0.02548	-0.01348	-0.01357
2022-11-01	0.00000	0.00000	-0.05779	-0.05953	0.00943	0.00939	0.00683	0.00681
2022-11-02	-0.00568	-0.00570	-0.00267	-0.00267	0.00935	0.00930	-0.04525	-0.04630
2022-11-03	0.00571	0.00570	0.00267	0.00267	-0.00926	-0.00930	-0.02133	-0.02156
2022-11-04	-0.00284	-0.00285	0.00000	0.00000	0.02804	0.02765	0.01937	0.01919

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2022-11-07	0.00855	0.00851	0.02133	0.02111	0.00000	0.00000	0.01663	0.01649
2022-11-08	-0.01130	-0.01136	-0.01828	-0.01845	-0.03636	-0.03704	-0.01402	-0.01412
2022-11-09	0.01429	0.01418	-0.03191	-0.03244	-0.00943	-0.00948	-0.00711	-0.00713
2022-11-10	-0.00845	-0.00849	-0.04396	-0.04495	0.01270	0.01262	-0.00477	-0.00478
2022-11-11	0.00568	0.00567	0.00287	0.00287	0.01881	0.01863	-0.00480	-0.00481
2022-11-14	-0.01130	-0.01136	0.00287	0.00286	-0.01538	-0.01550	-0.02410	-0.02439
2022-11-15	0.00571	0.00570	0.00286	0.00285	-0.00625	-0.00627	0.00000	0.00000
2022-11-16	-0.02273	-0.02299	0.01709	0.01695	-0.04088	-0.04174	0.00000	0.00000
2022-11-17	0.01454	0.01443	0.00560	0.00559	0.00000	0.00000	-0.00247	-0.00247
2022-11-18	0.01146	0.01140	0.00000	0.00000	0.01311	0.01303	-0.00743	-0.00745
2022-11-21	-0.01133	-0.01140	0.02786	0.02747	-0.00647	-0.00649	0.00499	0.00498
2022-11-22	0.02006	0.01986	0.00271	0.00271	0.00326	0.00325	-0.00744	-0.00747
2022-11-23	-0.00281	-0.00281	0.01351	0.01342	-0.00649	-0.00651	-0.00750	-0.00753
2022-11-24	0.01408	0.01399	0.00000	0.00000	-0.00654	-0.00656	0.01511	0.01500
2022-11-25	-0.00278	-0.00278	-0.00533	-0.00535	0.01316	0.01307	0.00000	0.00000
2022-11-28	0.00557	0.00556	0.00268	0.00268	0.00000	0.00000	-0.01241	-0.01248
2022-11-29	-0.00554	-0.00556	0.03743	0.03675	-0.01299	-0.01307	0.00000	0.00000
2022-11-30	0.03621	0.03557	-0.00258	-0.00258	0.00000	0.00000	0.01508	0.01496
2022-12-01	-0.03226	-0.03279	0.00775	0.00772	0.01316	0.01307	-0.01980	-0.02000
2022-12-02	-0.00725	-0.00728	-0.01282	-0.01290	-0.00325	-0.00325	0.01010	0.01005
2022-12-05	-0.01404	-0.01414	-0.01818	-0.01835	0.00326	0.00325	-0.04000	-0.04082
2022-12-06	-0.01140	-0.01146	0.00794	0.00791	-0.06818	-0.07062	-0.06250	-0.06454
2022-12-07	-0.02594	-0.02628	0.00525	0.00524	-0.02439	-0.02469	0.02222	0.02198
2022-12-08	0.00592	0.00590	-0.02350	-0.02378	-0.00714	-0.00717	0.01630	0.01617
2022-12-09	0.00882	0.00878	-0.01604	-0.01617	0.00000	0.00000	-0.02406	-0.02436
2022-12-12	0.01458	0.01447	0.03261	0.03209	-0.02878	-0.02920	0.01370	0.01361
2022-12-13	0.00000	0.00000	-0.01579	-0.01592	0.03992	0.03915	0.01351	0.01342
2022-12-14	-0.00862	-0.00866	0.03743	0.03675	0.00000	0.00000	0.00000	0.00000
2022-12-15	-0.01449	-0.01460	0.00515	0.00514	0.03214	0.03164	-0.02133	-0.02156
2022-12-16	0.01176	0.01170	-0.00256	-0.00257	-0.02768	-0.02807	0.00272	0.00272
2022-12-19	0.00581	0.00580	0.02057	0.02036	-0.01068	-0.01073	0.01087	0.01081
2022-12-20	-0.00867	-0.00871	-0.01259	-0.01267	-0.02518	-0.02550	0.00000	0.00000
2022-12-21	0.01166	0.01159	0.00000	0.00000	-0.00738	-0.00741	0.01882	0.01864
2022-12-22	-0.01153	-0.01159	0.00255	0.00255	0.00000	0.00000	-0.01055	-0.01061
2022-12-23	-0.00875	-0.00878	-0.03053	-0.03101	0.00372	0.00371	0.00800	0.00797
2022-12-26	0.00882	0.00878	-0.00262	-0.00263	0.00741	0.00738	-0.00794	-0.00797
2022-12-27	0.00292	0.00291	-0.01316	-0.01325	-0.01838	-0.01855	0.01600	0.01587
2022-12-28	0.00581	0.00580	0.00000	0.00000	0.00000	0.00000	-0.02100	-0.02122
2022-12-29	-0.00867	-0.00871	0.01333	0.01325	0.00000	0.00000	0.01340	0.01332
2022-12-30	-0.00292	-0.00292	0.01316	0.01307	-0.01498	-0.01509	-0.00794	-0.00797
2023-01-02	0.00000	0.00000	-0.00242	-0.00243	0.00760	0.00758	0.01333	0.01325
2023-01-03	0.00000	0.00000	-0.01950	-0.01969	0.01132	0.01126	0.01579	0.01567
2023-01-04	-0.02339	-0.02367	-0.06250	-0.06454	0.01119	0.01113	-0.01036	-0.01042
2023-01-05	-0.01198	-0.01205	-0.06061	-0.06252	-0.01107	-0.01113	-0.01309	-0.01318
2023-01-06	0.00606	0.00604	0.01290	0.01282	0.05597	0.05446	-0.01592	-0.01604
2023-01-09	0.01807	0.01791	-0.03503	-0.03566	0.01413	0.01404	0.02156	0.02133
2023-01-10	-0.03254	-0.03309	0.03630	0.03566	-0.01742	-0.01758	0.01055	0.01050
2023-01-11	-0.00612	-0.00614	0.01274	0.01266	-0.00709	-0.00712	0.01044	0.01039
2023-01-12	0.00615	0.00614	-0.02830	-0.02871	0.05714	0.05557	-0.00258	-0.00259
2023-01-13	-0.01529	-0.01541	0.01618	0.01605	-0.02703	-0.02740	-0.02073	-0.02094
2023-01-16	0.01242	0.01235	-0.01274	-0.01282	0.02778	0.02740	0.01852	0.01835

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2023-01-17	0.02147	0.02125	0.01290	0.01282	-0.02365	-0.02393	0.02597	0.02564
2023-01-18	-0.00300	-0.00301	0.00955	0.00951	-0.03114	-0.03164	-0.00506	-0.00508
2023-01-19	0.00301	0.00301	0.01893	0.01875	0.01786	0.01770	-0.01018	-0.01023
2023-01-20	-0.00300	-0.00301	0.00310	0.00309	-0.01053	-0.01058	-0.00514	-0.00515
2023-01-24	-0.00904	-0.00908	-0.01235	-0.01242	0.00709	0.00707	-0.00517	-0.00518
2023-01-25	-0.00304	-0.00304	-0.02188	-0.02212	-0.00704	-0.00707	-0.00260	-0.00260
2023-01-26	0.03354	0.03299	-0.03834	-0.03909	0.03901	0.03827	0.03385	0.03329
2023-01-27	0.02655	0.02620	0.01661	0.01647	-0.00683	-0.00685	-0.00252	-0.00252
2023-01-30	0.00000	0.00000	-0.01634	-0.01647	-0.00687	-0.00690	0.00000	0.00000
2023-01-31	-0.02586	-0.02620	-0.01661	-0.01675	0.02422	0.02393	-0.02778	-0.02817
2023-02-01	0.00295	0.00295	-0.00338	-0.00338	-0.01014	-0.01019	0.00519	0.00518
2023-02-02	-0.00588	-0.00590	-0.02712	-0.02749	0.00000	0.00000	0.00517	0.00515
2023-02-03	0.02959	0.02916	-0.03833	-0.03908	-0.00683	-0.00685	-0.00257	-0.00257
2023-02-06	0.00287	0.00287	0.00725	0.00722	-0.00344	-0.00344	0.00000	0.00000
2023-02-07	0.01433	0.01422	0.03957	0.03881	0.00000	0.00000	-0.02062	-0.02083
2023-02-08	-0.00282	-0.00283	0.01038	0.01033	-0.00345	-0.00345	0.01053	0.01047
2023-02-09	0.00850	0.00846	-0.02740	-0.02778	0.01384	0.01375	-0.01823	-0.01840
2023-02-10	-0.00843	-0.00846	-0.03521	-0.03585	0.05802	0.05640	0.00796	0.00793
2023-02-13	0.00567	0.00565	0.01460	0.01449	-0.01935	-0.01954	-0.00263	-0.00263
2023-02-14	0.00845	0.00841	0.01799	0.01783	0.00658	0.00656	0.00264	0.00263
2023-02-15	-0.00838	-0.00841	0.04240	0.04153	-0.00327	-0.00327	-0.00263	-0.00263
2023-02-16	-0.01972	-0.01992	-0.01017	-0.01022	-0.01311	-0.01320	-0.00528	-0.00529
2023-02-17	0.00287	0.00287	-0.01370	-0.01379	-0.01329	-0.01338	-0.00265	-0.00266
2023-02-20	0.00287	0.00286	0.00694	0.00692	0.01347	0.01338	0.01596	0.01583
2023-02-21	-0.00571	-0.00573	-0.02759	-0.02797	0.00664	0.00662	0.00524	0.00522
2023-02-22	-0.00287	-0.00288	0.01064	0.01058	-0.01980	-0.02000	0.00521	0.00519
2023-02-23	0.00576	0.00575	0.01754	0.01739	0.02694	0.02658	0.02850	0.02810
2023-02-24	-0.00573	-0.00575	-0.00345	-0.00345	-0.02951	-0.02995	0.01763	0.01748
2023-02-27	0.01153	0.01146	0.02076	0.02055	-0.00338	-0.00338	-0.01980	-0.02000
2023-02-28	-0.00285	-0.00285	0.01356	0.01347	-0.02034	-0.02055	-0.02020	-0.02041
2023-03-01	-0.01714	-0.01729	0.00000	0.00000	0.00346	0.00345	0.01289	0.01280
2023-03-02	0.00291	0.00290	0.00000	0.00000	-0.01379	-0.01389	-0.01272	-0.01280
2023-03-03	-0.01739	-0.01754	0.01003	0.00998	0.00350	0.00349	0.00515	0.00514
2023-03-06	-0.00885	-0.00889	-0.03642	-0.03710	-0.00697	-0.00699	-0.00513	-0.00514
2023-03-07	0.00298	0.00297	-0.01718	-0.01733	-0.00702	-0.00704	0.00000	0.00000
2023-03-08	0.01780	0.01765	0.02098	0.02076	-0.02473	-0.02505	0.01546	0.01535
2023-03-09	0.00000	0.00000	-0.00685	-0.00687	0.00362	0.00362	0.00761	0.00759
2023-03-10	-0.01458	-0.01468	-0.01379	-0.01389	-0.02166	-0.02190	0.01008	0.01003
2023-03-13	0.01183	0.01176	0.01399	0.01389	-0.02214	-0.02239	0.01496	0.01485
2023-03-14	-0.02632	-0.02667	-0.04483	-0.04586	-0.00755	-0.00758	-0.01228	-0.01236
2023-03-15	0.00000	0.00000	-0.00722	-0.00725	-0.06084	-0.06277	-0.00249	-0.00249
2023-03-16	-0.00300	-0.00301	-0.04000	-0.04082	-0.02834	-0.02875	0.00998	0.00993
2023-03-17	0.00904	0.00900	0.05303	0.05167	0.01667	0.01653	0.00000	0.00000
2023-03-20	0.00299	0.00298	-0.02158	-0.02182	-0.00410	-0.00411	-0.01481	-0.01493
2023-03-21	0.01190	0.01183	0.00368	0.00367	0.02881	0.02840	0.03258	0.03206
2023-03-24	0.03824	0.03752	-0.01832	-0.01848	0.00800	0.00797	-0.01214	-0.01221
2023-03-27	-0.01416	-0.01427	0.00000	0.00000	-0.00794	-0.00797	-0.00491	-0.00493
2023-03-28	-0.00287	-0.00288	0.03358	0.03303	0.02000	0.01980	0.00000	0.00000
2023-03-29	0.03469	0.03410	0.03971	0.03894	0.00000	0.00000	0.01235	0.01227
2023-03-30	0.00284	0.00284	-0.00347	-0.00348	0.00000	0.00000	-0.00244	-0.00244
2023-03-31	-0.00850	-0.00853	0.01045	0.01040	-0.01176	-0.01183	-0.00733	-0.00736

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2023-04-03	0.00571	0.00570	0.01379	0.01370	0.00794	0.00791	0.00493	0.00491
2023-04-04	-0.00284	-0.00284	0.03741	0.03673	0.00394	0.00393	-0.00490	-0.00491
2023-04-05	-0.00570	-0.00571	0.00328	0.00327	-0.05490	-0.05647	0.02217	0.02193
2023-04-06	0.00287	0.00286	-0.01307	-0.01316	0.00000	0.00000	0.02410	0.02381
2023-04-10	0.00571	0.00570	-0.00993	-0.00998	0.00415	0.00414	0.01176	0.01170
2023-04-11	0.00284	0.00284	0.02007	0.01987	0.01653	0.01639	0.00000	0.00000
2023-04-12	0.00850	0.00846	-0.02951	-0.02995	0.00813	0.00810	0.00233	0.00232
2023-04-13	0.00281	0.00281	0.00000	0.00000	-0.01613	-0.01626	0.00696	0.00694
2023-04-14	0.00840	0.00837	-0.01351	-0.01361	0.00410	0.00409	0.00000	0.00000
2023-04-17	0.00278	0.00277	-0.02055	-0.02076	-0.01224	-0.01232	-0.01613	-0.01626
2023-04-18	0.01108	0.01102	0.04895	0.04779	-0.00413	-0.00414	0.00000	0.00000
2023-04-26	0.00822	0.00819	0.03333	0.03279	0.00000	0.00000	0.03044	0.02999
2023-04-27	-0.00543	-0.00545	-0.00645	-0.00647	0.00000	0.00000	-0.01818	-0.01835
2023-04-28	-0.01093	-0.01099	0.01623	0.01610	-0.01245	-0.01253	-0.01620	-0.01634
2023-05-02	0.00000	0.00000	-0.04792	-0.04911	-0.01260	-0.01269	-0.01412	-0.01422
2023-05-03	-0.01381	-0.01391	-0.01342	-0.01351	-0.00426	-0.00426	-0.00955	-0.00959
2023-05-04	0.00840	0.00837	-0.00680	-0.00683	0.03483	0.03423	0.00000	0.00000
2023-05-05	0.00000	0.00000	-0.04452	-0.04554	-0.01724	-0.01739	0.00482	0.00481
2023-05-08	0.00000	0.00000	0.02867	0.02827	0.02193	0.02169	-0.02638	-0.02673
2023-05-09	-0.00833	-0.00837	0.00697	0.00694	0.05150	0.05022	0.00246	0.00246
2023-05-10	0.00000	0.00000	0.01730	0.01715	-0.01224	-0.01232	0.01474	0.01463
2023-05-11	-0.01120	-0.01127	-0.04762	-0.04879	-0.02893	-0.02935	-0.00969	-0.00973
2023-05-12	-0.00283	-0.00284	-0.01429	-0.01439	0.00426	0.00425	-0.02689	-0.02726
2023-05-15	-0.00284	-0.00284	-0.03623	-0.03690	-0.00847	-0.00851	0.00000	0.00000
2023-05-16	-0.00855	-0.00858	0.00376	0.00375	0.00855	0.00851	0.00000	0.00000
2023-05-17	0.00862	0.00858	-0.04869	-0.04991	0.00424	0.00423	0.00251	0.00251
2023-05-19	0.02564	0.02532	-0.05118	-0.05254	0.00000	0.00000	0.00752	0.00749
2023-05-22	0.00000	0.00000	0.02075	0.02053	0.02110	0.02088	0.00498	0.00496
2023-05-23	0.01389	0.01379	0.03189	0.03140	-0.01653	-0.01667	0.00000	0.00000
2023-05-24	-0.01096	-0.01102	-0.01747	-0.01762	-0.00420	-0.00421	0.02970	0.02927
2023-05-25	0.00277	0.00277	-0.05778	-0.05951	-0.00422	-0.00423	-0.00962	-0.00966
2023-05-26	0.01105	0.01099	-0.01415	-0.01425	0.00000	0.00000	0.00243	0.00242
2023-05-29	0.00000	0.00000	-0.00957	-0.00962	0.00000	0.00000	0.01695	0.01681
2023-05-30	0.01093	0.01087	0.01932	0.01914	0.00000	0.00000	-0.01905	-0.01923
2023-05-31	-0.02162	-0.02186	-0.03318	-0.03374	-0.01695	-0.01709	-0.01942	-0.01961
2023-06-05	0.01657	0.01644	0.02941	0.02899	0.00431	0.00430	0.00990	0.00985
2023-06-06	-0.00543	-0.00545	0.05714	0.05557	0.00429	0.00428	0.00245	0.00245
2023-06-07	-0.00546	-0.00548	-0.02252	-0.02278	0.00000	0.00000	0.01467	0.01456
2023-06-08	0.00275	0.00274	0.01382	0.01373	0.00427	0.00426	0.00482	0.00481
2023-06-09	-0.00274	-0.00274	-0.00455	-0.00456	0.00851	0.00847	-0.00480	-0.00481
2023-06-12	0.00549	0.00548	0.00913	0.00909	0.02110	0.02088	0.01697	0.01683
2023-06-13	0.00000	0.00000	0.00000	0.00000	-0.00826	-0.00830	-0.00494	-0.00495
2023-06-14	-0.00820	-0.00823	0.01810	0.01794	-0.00417	-0.00418	-0.00744	-0.00747
2023-06-15	-0.00275	-0.00276	0.02222	0.02198	0.01674	0.01660	0.00500	0.00499
2023-06-16	0.00000	0.00000	0.00870	0.00866	-0.01235	-0.01242	-0.01244	-0.01252
2023-06-19	-0.00552	-0.00554	-0.02586	-0.02620	0.00000	0.00000	-0.00252	-0.00252
2023-06-20	0.00556	0.00554	0.00442	0.00442	0.00417	0.00416	0.00758	0.00755
2023-06-21	0.00829	0.00825	-0.00441	-0.00442	0.00000	0.00000	-0.00251	-0.00251
2023-06-22	-0.00822	-0.00825	0.00000	0.00000	0.00000	0.00000	-0.01005	-0.01010
2023-06-23	0.00000	0.00000	-0.01327	-0.01336	0.01660	0.01646	0.00254	0.00253
2023-06-26	0.00276	0.00276	-0.00897	-0.00901	0.00408	0.00407	0.01013	0.01008

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2023-06-27	0.00826	0.00823	0.00905	0.00901	-0.01220	-0.01227	0.00251	0.00250
2023-07-03	-0.00820	-0.00823	0.04484	0.04387	0.00000	0.00000	0.00000	0.00000
2023-07-04	-0.00275	-0.00276	0.00429	0.00428	0.00000	0.00000	-0.00750	-0.00753
2023-07-05	0.00000	0.00000	0.03419	0.03362	0.01646	0.01633	0.00252	0.00252
2023-07-06	0.00276	0.00276	0.01240	0.01232	0.01215	0.01207	-0.00503	-0.00504
2023-07-07	-0.00551	-0.00552	-0.02857	-0.02899	0.02800	0.02762	0.00000	0.00000
2023-07-10	0.00277	0.00277	0.01681	0.01667	0.00000	0.00000	-0.00253	-0.00253
2023-07-11	-0.00276	-0.00277	0.00413	0.00412	0.01946	0.01927	0.00759	0.00757
2023-07-12	0.01662	0.01648	-0.02469	-0.02500	-0.00763	-0.00766	0.00000	0.00000
2023-07-13	-0.00545	-0.00546	-0.02110	-0.02132	0.03462	0.03403	-0.01508	-0.01519
2023-07-14	0.00822	0.00819	0.01724	0.01709	0.02230	0.02206	0.00255	0.00255
2023-07-17	-0.00272	-0.00272	0.00847	0.00844	-0.01455	-0.01465	-0.00509	-0.00510
2023-07-18	-0.00272	-0.00273	-0.02101	-0.02123	0.00000	0.00000	-0.01535	-0.01546
2023-07-20	0.00000	0.00000	0.01717	0.01702	-0.01107	-0.01113	0.00000	0.00000
2023-07-21	0.00000	0.00000	0.02110	0.02088	0.03731	0.03663	0.00260	0.00259
2023-07-24	-0.00546	-0.00548	0.02066	0.02045	0.00000	0.00000	0.01036	0.01031
2023-07-25	0.00549	0.00548	-0.01619	-0.01633	0.00719	0.00717	-0.00513	-0.00514
2023-07-26	0.02186	0.02162	0.02881	0.02840	-0.02500	-0.02532	0.00000	0.00000
2023-07-27	-0.01337	-0.01346	-0.02000	-0.02020	-0.02198	-0.02222	-0.04124	-0.04211
2023-07-28	-0.01084	-0.01090	0.00816	0.00813	0.01124	0.01117	0.00000	0.00000
2023-07-31	0.00000	0.00000	-0.02429	-0.02459	0.03333	0.03279	0.00000	0.00000
2023-08-01	0.00000	0.00000	-0.00830	-0.00833	-0.03584	-0.03650	0.00806	0.00803
2023-08-02	0.00822	0.00819	-0.00837	-0.00840	0.01487	0.01476	-0.01600	-0.01613
2023-08-03	0.00543	0.00542	-0.00422	-0.00423	0.05861	0.05695	0.00813	0.00810
2023-08-04	-0.01081	-0.01087	0.01695	0.01681	-0.02076	-0.02098	-0.00806	-0.00810
2023-08-07	0.01366	0.01357	-0.00833	-0.00837	-0.01767	-0.01783	-0.00271	-0.00271
2023-08-08	-0.00809	-0.00812	-0.00420	-0.00421	-0.01439	-0.01449	0.01359	0.01350
2023-08-09	0.02174	0.02151	0.01266	0.01258	-0.02555	-0.02588	0.01072	0.01067
2023-08-10	0.00000	0.00000	0.00417	0.00416	-0.00375	-0.00375	0.00796	0.00793
2023-08-11	0.00000	0.00000	-0.02490	-0.02521	0.01128	0.01122	0.00263	0.00263
2023-08-14	-0.00532	-0.00533	0.01702	0.01688	0.01859	0.01842	0.00525	0.00524
2023-08-15	-0.00535	-0.00536	0.05021	0.04899	0.00000	0.00000	-0.00261	-0.00261
2023-08-16	0.00000	0.00000	0.00398	0.00398	0.01095	0.01089	0.00262	0.00261
2023-08-18	-0.00538	-0.00539	0.00794	0.00791	-0.02888	-0.02931	-0.01828	-0.01845
2023-08-21	-0.00811	-0.00814	0.03150	0.03101	-0.00372	-0.00372	-0.00532	-0.00533
2023-08-22	0.01362	0.01353	0.03053	0.03008	0.00746	0.00744	0.00267	0.00267
2023-08-23	0.00000	0.00000	0.00000	0.00000	-0.00370	-0.00371	-0.00267	-0.00267
2023-08-24	-0.01075	-0.01081	-0.01111	-0.01117	0.02230	0.02206	-0.00535	-0.00536
2023-08-25	0.00815	0.00812	-0.02996	-0.03042	0.00000	0.00000	0.00000	0.00000
2023-08-28	-0.00809	-0.00812	0.03475	0.03416	-0.00364	-0.00364	-0.00538	-0.00539
2023-08-29	0.00543	0.00542	0.00000	0.00000	0.00365	0.00364	0.00541	0.00539
2023-08-30	-0.00541	-0.00542	0.00373	0.00372	0.00000	0.00000	0.02419	0.02391
2023-08-31	-0.00272	-0.00272	-0.00744	-0.00746	-0.01091	-0.01097	-0.02100	-0.02122
2023-09-01	0.00545	0.00543	0.00375	0.00374	0.00000	0.00000	-0.00268	-0.00268
2023-09-04	0.00000	0.00000	0.02239	0.02214	-0.00368	-0.00368	0.00269	0.00268
2023-09-05	0.00000	0.00000	0.00000	0.00000	0.02952	0.02909	0.00268	0.00268
2023-09-06	-0.00813	-0.00816	0.04745	0.04635	-0.01434	-0.01444	0.01337	0.01328
2023-09-07	0.00273	0.00273	0.00348	0.00348	0.00000	0.00000	-0.00792	-0.00795
2023-09-08	-0.00545	-0.00546	0.00347	0.00347	0.00727	0.00725	-0.01862	-0.01879
2023-09-11	0.00000	0.00000	-0.03114	-0.03164	0.02527	0.02496	0.00271	0.00271
2023-09-12	-0.00274	-0.00274	0.00357	0.00356	-0.01056	-0.01062	0.00270	0.00270

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2023-09-13	-0.00275	-0.00275	0.01779	0.01764	-0.00712	-0.00714	0.00000	0.00000
2023-09-14	0.00275	0.00275	0.01748	0.01733	0.00358	0.00358	-0.00809	-0.00812
2023-09-15	-0.01099	-0.01105	-0.02406	-0.02435	0.00000	0.00000	0.01087	0.01081
2023-09-18	0.00000	0.00000	0.00000	0.00000	-0.01786	-0.01802	-0.01075	-0.01081
2023-09-19	0.00833	0.00830	0.02113	0.02091	-0.00364	-0.00364	0.01359	0.01350
2023-09-20	0.00826	0.00823	0.02069	0.02048	0.00365	0.00364	0.02145	0.02122
2023-09-21	-0.00273	-0.00274	-0.00338	-0.00338	0.00364	0.00363	0.00000	0.00000
2023-09-22	-0.00548	-0.00549	0.00000	0.00000	-0.00725	-0.00727	0.01312	0.01304
2023-09-25	-0.00826	-0.00830	0.00000	0.00000	-0.02920	-0.02963	-0.02332	-0.02359
2023-09-26	-0.00556	-0.00557	-0.06102	-0.06296	-0.01504	-0.01515	-0.00265	-0.00266
2023-09-27	-0.00838	-0.00842	0.03610	0.03546	0.00000	0.00000	-0.01064	-0.01070
2023-09-29	-0.00563	-0.00565	-0.00697	-0.00699	-0.01908	-0.01927	0.00806	0.00803
2023-10-02	0.02833	0.02793	-0.01404	-0.01413	0.00778	0.00775	0.00000	0.00000
2023-10-03	0.01377	0.01368	-0.04270	-0.04364	-0.01158	-0.01165	0.00533	0.00532
2023-10-04	0.00000	0.00000	-0.01115	-0.01122	-0.01562	-0.01575	0.00265	0.00265
2023-10-05	-0.01359	-0.01368	-0.01880	-0.01898	-0.00794	-0.00797	-0.01587	-0.01600
2023-10-06	-0.00551	-0.00552	0.01533	0.01521	0.01600	0.01587	0.01613	0.01600
2023-10-09	0.00277	0.00277	0.04528	0.04429	0.03937	0.03861	-0.00529	-0.00530
2023-10-10	-0.01381	-0.01391	0.00361	0.00360	-0.00379	-0.00380	0.01330	0.01321
2023-10-11	0.00000	0.00000	-0.00719	-0.00722	-0.00760	-0.00763	-0.00787	-0.00791
2023-10-12	0.01401	0.01391	-0.02536	-0.02569	0.00383	0.00382	0.01323	0.01314
2023-10-13	0.00276	0.00276	0.00743	0.00741	0.00382	0.00381	0.00522	0.00521
2023-10-16	0.00275	0.00275	0.02583	0.02550	-0.01901	-0.01919	-0.02338	-0.02365
2023-10-17	-0.01648	-0.01662	-0.00360	-0.00360	0.01938	0.01919	0.00798	0.00795
2023-10-18	-0.01117	-0.01124	0.02527	0.02496	-0.02662	-0.02698	-0.01055	-0.01061
2023-10-19	-0.01130	-0.01136	-0.00704	-0.00707	0.01172	0.01165	-0.00533	-0.00535
2023-10-20	0.02571	0.02539	-0.00709	-0.00712	-0.01158	-0.01165	-0.00804	-0.00808
2023-10-23	-0.01393	-0.01403	-0.05000	-0.05129	-0.01953	-0.01972	-0.02432	-0.02463
2023-10-24	-0.00847	-0.00851	0.00000	0.00000	0.00797	0.00794	0.01108	0.01102
2023-10-25	0.01140	0.01133	0.00376	0.00375	0.01186	0.01179	-0.01370	-0.01379
2023-10-26	-0.01690	-0.01705	-0.00749	-0.00752	-0.02344	-0.02372	-0.03333	-0.03390
2023-10-27	-0.00287	-0.00287	-0.00377	-0.00378	-0.01200	-0.01207	0.00575	0.00573
2023-10-30	0.01724	0.01709	-0.03788	-0.03861	-0.01215	-0.01222	-0.02286	-0.02312
2023-10-31	-0.01130	-0.01136	0.00787	0.00784	0.00000	0.00000	0.02047	0.02026
2023-11-01	-0.01714	-0.01729	-0.05859	-0.06038	-0.00820	-0.00823	0.03725	0.03657
2023-11-02	0.02907	0.02866	-0.00830	-0.00833	0.03306	0.03252	0.00000	0.00000
2023-11-03	0.00565	0.00563	0.04184	0.04099	-0.00800	-0.00803	-0.01381	-0.01391
2023-11-06	0.01685	0.01671	0.01606	0.01594	0.04435	0.04340	0.02241	0.02216
2023-11-07	-0.00829	-0.00832	-0.02372	-0.02400	-0.00386	-0.00387	-0.02192	-0.02216
2023-11-08	0.00279	0.00278	-0.01215	-0.01222	-0.01550	-0.01563	-0.01401	-0.01410
2023-11-09	0.00000	0.00000	0.00410	0.00409	0.00394	0.00393	0.00284	0.00284
2023-11-10	-0.01944	-0.01964	0.01225	0.01217	-0.01569	-0.01581	0.00567	0.00565
2023-11-13	0.00567	0.00565	0.00000	0.00000	0.01195	0.01188	-0.00845	-0.00849
2023-11-14	0.00563	0.00562	0.00000	0.00000	-0.01969	-0.01988	-0.00568	-0.00570
2023-11-15	0.01401	0.01391	0.01210	0.01202	0.00803	0.00800	0.01429	0.01418
2023-11-16	0.00276	0.00276	0.00797	0.00794	0.00000	0.00000	-0.00282	-0.00282
2023-11-17	0.00000	0.00000	0.00395	0.00394	0.05578	0.05428	0.00282	0.00282
2023-11-20	-0.02204	-0.02229	0.00394	0.00393	0.01132	0.01126	0.01127	0.01120
2023-11-21	-0.01127	-0.01133	0.02353	0.02326	-0.01119	-0.01126	0.00836	0.00832
2023-11-22	0.01140	0.01133	-0.00766	-0.00769	-0.02642	-0.02677	-0.00276	-0.00277
2023-11-23	0.00563	0.00562	-0.01544	-0.01556	0.00000	0.00000	-0.00277	-0.00277

Bersambung...

Lanjutan Tabel E.2

Tanggal	BBCA		ADRO		SMGR		TLKM	
	Simple	Log	Simple	Log	Simple	Log	Simple	Log
2023-11-24	0.00000	0.00000	-0.00392	-0.00393	-0.00388	-0.00388	0.00556	0.00554
2023-11-27	-0.00560	-0.00562	0.00394	0.00393	0.01167	0.01161	0.01657	0.01644
2023-11-28	0.00000	0.00000	0.01176	0.01170	-0.01538	-0.01550	0.02446	0.02416
2023-11-29	0.00282	0.00281	0.00000	0.00000	0.00000	0.00000	-0.01326	-0.01335
2023-11-30	0.00843	0.00839	0.01550	0.01538	0.01563	0.01550	0.01075	0.01070
2023-12-01	-0.00279	-0.00279	-0.02290	-0.02317	0.00769	0.00766	0.01862	0.01845
2023-12-04	0.00196	0.00196	0.00000	0.00000	-0.01527	-0.01538	-0.00522	-0.00524
2023-12-05	-0.00280	-0.00281	-0.01953	-0.01972	-0.01163	-0.01170	0.01050	0.01044
2023-12-06	-0.01124	-0.01130	0.00797	0.00794	0.00784	0.00781	0.01818	0.01802
2023-12-07	0.00284	0.00284	0.00000	0.00000	-0.01946	-0.01965	-0.00255	-0.00255
2023-12-08	-0.00850	-0.00853	0.01186	0.01179	0.00000	0.00000	0.00767	0.00764
2023-12-11	0.00000	0.00000	0.00000	0.00000	-0.02778	-0.02817	0.01015	0.01010
2023-12-12	-0.00571	-0.00573	-0.00391	-0.00391	0.00816	0.00813	-0.01759	-0.01774
2023-12-13	-0.00287	-0.00288	-0.03922	-0.04001	0.00810	0.00806	0.00256	0.00255
2023-12-14	0.04323	0.04232	0.01633	0.01619	-0.00803	-0.00806	0.00510	0.00509
2023-12-15	0.01934	0.01915	0.01205	0.01198	0.00405	0.00404	0.01015	0.01010
2023-12-18	-0.00271	-0.00271	-0.00397	-0.00398	-0.01210	-0.01217	-0.00251	-0.00252
2023-12-19	0.00543	0.00542	0.03586	0.03523	0.00816	0.00813	0.00000	0.00000
2023-12-20	0.00541	0.00539	-0.00769	-0.00772	0.00810	0.00806	-0.00252	-0.00252
2023-12-21	0.00269	0.00268	0.00775	0.00772	-0.01205	-0.01212	-0.00253	-0.00253
2023-12-22	0.00000	0.00000	-0.00385	-0.00385	0.02033	0.02012	0.00253	0.00253
2023-12-27	0.00536	0.00535	0.00000	0.00000	0.00000	0.00000	-0.00505	-0.00506
2023-12-28	0.00267	0.00266	-0.00386	-0.00387	0.00000	0.00000	0.00508	0.00506
2023-12-29	0.00000	0.00000	-0.00001	-0.00001	0.01992	0.01972	-0.00253	-0.00253

Tabel E.3: Ekspektasi Return dan Varians Log Return 36 Periode - Skala 126 Hari (BBCA, ADRO, SMGR, TLKM)

P	BBCA			ADRO			SMGR			TLKM		
	μ_s	μ_l	σ_l^2	μ_s	μ_l	σ_l^2	μ_s	μ_l	σ_l^2	μ_s	μ_l	σ_l^2
1	0.362	0.316	0.091	0.122	0.090	0.064	0.311	0.278	0.067	0.194	0.176	0.037
2	0.133	0.085	0.095	0.090	0.051	0.077	0.241	0.200	0.083	0.119	0.099	0.041
3	0.096	0.052	0.088	0.183	0.141	0.083	0.107	0.060	0.095	0.083	0.062	0.041
4	0.077	0.039	0.076	0.269	0.230	0.077	0.203	0.159	0.089	0.113	0.096	0.032
5	0.122	0.088	0.068	0.229	0.193	0.070	0.111	0.065	0.093	0.131	0.114	0.034
6	-0.052	-0.081	0.058	0.058	0.029	0.058	-0.079	-0.119	0.081	0.054	0.040	0.029
7	-0.090	-0.123	0.065	-0.078	-0.100	0.044	-0.296	-0.335	0.079	-0.101	-0.115	0.027
8	0.166	0.136	0.061	0.062	0.043	0.038	-0.220	-0.255	0.070	-0.066	-0.079	0.025
9	0.205	0.168	0.073	0.074	0.058	0.032	-0.196	-0.230	0.068	0.002	-0.009	0.022
10	0.516	0.465	0.102	0.187	0.171	0.033	-0.198	-0.234	0.073	0.196	0.183	0.026
11	0.415	0.359	0.112	0.216	0.199	0.034	-0.010	-0.051	0.081	0.192	0.179	0.025
12	0.529	0.469	0.119	0.183	0.163	0.039	-0.216	-0.254	0.076	0.132	0.119	0.026
13	0.801	0.745	0.111	0.298	0.280	0.037	-0.184	-0.221	0.073	0.258	0.245	0.026
14	0.627	0.574	0.107	0.290	0.272	0.035	-0.138	-0.171	0.067	0.221	0.209	0.023
15	0.946	0.881	0.129	0.343	0.324	0.039	-0.230	-0.264	0.068	0.216	0.203	0.026
16	0.628	0.573	0.111	0.207	0.188	0.038	-0.160	-0.191	0.060	0.092	0.082	0.020
17	0.857	0.798	0.117	0.141	0.120	0.042	-0.359	-0.387	0.057	-0.001	-0.015	0.027
18	0.641	0.572	0.140	0.027	0.011	0.032	-0.056	-0.090	0.068	0.050	0.036	0.030

Bersambung...

Lanjutan Tabel E.3

P	BBCA			ADRO			SMGR			TLKM		
	μ_s	μ_l	σ_l^2	μ_s	μ_l	σ_l^2	μ_s	μ_l	σ_l^2	μ_s	μ_l	σ_l^2
19	0.347	0.278	0.139	0.090	0.075	0.031	-0.021	-0.059	0.075	0.004	-0.012	0.032
20	0.476	0.403	0.144	0.136	0.120	0.033	-0.042	-0.079	0.073	0.036	0.021	0.030
21	0.436	0.381	0.110	0.028	0.014	0.029	0.117	0.088	0.060	0.088	0.073	0.030
22	0.301	0.248	0.106	-0.048	-0.062	0.028	0.239	0.209	0.059	0.074	0.059	0.030
23	0.179	0.131	0.095	0.032	0.019	0.026	0.186	0.161	0.050	0.181	0.169	0.024
24	0.317	0.282	0.070	-0.108	-0.124	0.031	-0.014	-0.036	0.045	0.150	0.139	0.023
25	0.126	0.091	0.070	-0.079	-0.095	0.032	0.152	0.130	0.044	0.122	0.111	0.021
26	-0.028	-0.060	0.066	-0.193	-0.208	0.031	0.160	0.137	0.047	0.133	0.121	0.024
27	-0.296	-0.328	0.064	-0.076	-0.091	0.030	-0.155	-0.181	0.052	-0.006	-0.017	0.023
28	-0.157	-0.189	0.065	0.026	0.011	0.030	-0.231	-0.257	0.052	0.115	0.104	0.023
29	-0.345	-0.383	0.076	0.061	0.047	0.029	-0.200	-0.226	0.051	0.071	0.060	0.021
30	-0.321	-0.361	0.079	0.118	0.107	0.022	-0.071	-0.093	0.043	0.074	0.065	0.018
31	-0.148	-0.184	0.073	-0.009	-0.019	0.020	0.065	0.043	0.045	0.134	0.126	0.015
32	0.075	0.039	0.070	0.040	0.031	0.019	-0.043	-0.063	0.041	0.085	0.079	0.012
33	0.095	0.059	0.072	-0.019	-0.028	0.018	0.093	0.076	0.035	0.120	0.115	0.011
34	-0.115	-0.152	0.074	-0.127	-0.137	0.019	0.098	0.080	0.035	-0.028	-0.034	0.012
35	0.223	0.195	0.057	-0.023	-0.032	0.019	0.124	0.105	0.036	-0.027	-0.033	0.012

Tabel E.4: Kovarians Log Return 36 Periode - Skala 126 Hari

P	BB-BB	BB-SM	BB-TL	AD-SM	AD-TL	SM-TL
1	0.0365	0.0216	0.0249	0.0199	0.0338	0.0281
2	0.0406	0.0239	0.0275	0.0272	0.0442	0.0357
3	0.0427	0.0258	0.0274	0.0276	0.0404	0.0343
4	0.0322	0.0189	0.0222	0.0198	0.0309	0.0276
5	0.0334	0.0209	0.0193	0.0193	0.0267	0.0269
6	0.0291	0.0179	0.0134	0.0172	0.0168	0.0169
7	0.0266	0.0147	0.0087	0.0212	0.0118	0.0127
8	0.0254	0.0154	0.0068	0.0159	0.0070	0.0079
9	0.0217	0.0155	0.0070	0.0156	0.0118	0.0138
10	0.0260	0.0129	0.0098	0.0053	0.0135	0.0118
11	0.0245	0.0139	0.0099	0.0007	0.0127	0.0139
12	0.0260	0.0122	0.0104	-0.0018	0.0174	0.0135
13	0.0254	0.0118	0.0095	-0.0060	0.0191	0.0128
14	0.0239	0.0107	0.0090	-0.0070	0.0145	0.0142
15	0.0259	0.0121	0.0053	-0.0214	0.0124	0.0092
16	0.0202	0.0122	0.0028	-0.0101	0.0101	0.0102
17	0.0270	0.0173	0.0081	-0.0040	0.0124	0.0110
18	0.0293	0.0195	0.0081	-0.0035	0.0073	0.0087
19	0.0321	0.0209	0.0090	-0.0003	0.0006	0.0084
20	0.0298	0.0195	0.0095	-0.0029	0.0011	0.0055
21	0.0298	0.0142	0.0129	0.0083	-0.0005	0.0047
22	0.0306	0.0139	0.0136	0.0056	-0.0013	0.0044
23	0.0236	0.0064	0.0085	0.0027	-0.0034	0.0009
24	0.0229	0.0034	0.0096	-0.0042	-0.0024	0.0036
25	0.0215	0.0037	0.0094	-0.0101	0.0006	0.0024
26	0.0242	0.0031	0.0096	-0.0115	-0.0006	0.0033
27	0.0225	0.0044	0.0084	-0.0074	-0.0011	0.0038

Bersambung...

Lanjutan Tabel E.4

P	BB-BB	BB-SM	BB-TL	AD-SM	AD-TL	SM-TL
28	0.0231	0.0053	0.0071	-0.0070	0.0013	0.0040
29	0.0211	0.0037	0.0062	-0.0038	0.0013	0.0031
30	0.0174	0.0042	0.0047	0.0010	0.0021	0.0017
31	0.0151	0.0024	0.0040	0.0026	0.0033	0.0029
32	0.0121	0.0015	0.0029	0.0063	0.0049	0.0026
33	0.0114	0.0020	0.0034	0.0037	0.0049	0.0028
34	0.0118	0.0021	0.0015	0.0054	0.0015	0.0052
35	0.0118	0.0026	0.0014	0.0027	0.0036	0.0070

Halaman ini sengaja dikosongkan

Lampiran F

Data Numerik Hasil Simulasi N=2

Lampiran ini menyajikan dataset numerik yang menjadi dasar analisis pada bab-bab sebelumnya untuk sistem dengan $N = 2$ aset. Seluruh nilai numerik telah dibulatkan untuk kemudahan pembacaan tanpa mengurangi signifikansi teknis.

Data harga harian saham untuk periode simulasi ini dapat merujuk pada Tabel E.1 di Lampiran E.

F.1 Parameter Dinamis Hamiltonian N=2

Berikut adalah parameter h (bias), interaksi J, C_{obj} (konstanta), dan parameter γ yang dihitung berdasarkan volatilitas per jendela untuk N=2. Karena hanya terdapat dua aset, interaksi J digabungkan ke dalam satu tabel yang sama.

Tabel F.1: Parameter Dinamis Portofolio N=2 (γ, h_{obj}, J_{obj})

P	γ	h_{obj} (BBCA)	h_{obj} (ADRO)	J_{obj} (BC-AD)	C_{obj}
1	0.2698	0.1735	0.0931	0.0014	-0.2680
2	0.4108	0.0540	0.0530	0.0026	-0.1096
3	0.4431	0.0357	0.0342	0.0026	-0.0726
4	0.4262	0.0286	0.0513	0.0016	-0.0815
5	0.3883	0.0530	0.0605	0.0015	-0.1150
6	0.4273	-0.0335	0.0227	0.0013	0.0095
7	0.3619	-0.0518	-0.0539	0.0008	0.1049
8	0.3710	0.0769	-0.0361	0.0005	-0.0414
9	0.3958	0.0944	-0.0018	0.0007	-0.0933
10	0.2061	0.2522	0.0964	0.0004	-0.3490
11	0.2507	0.2001	0.0939	0.0004	-0.2944
12	0.2384	0.2568	0.0640	0.0005	-0.3213
13	0.1185	0.3967	0.1280	0.0002	-0.5249
14	0.1637	0.3089	0.1090	0.0004	-0.4183
15	0.1105	0.4692	0.1073	0.0001	-0.5766
16	0.2019	0.3083	0.0448	0.0002	-0.3533
17	0.1675	0.4233	-0.0021	0.0003	-0.4215
18	0.2473	0.3115	0.0228	0.0005	-0.3348
19	0.3721	0.1597	-0.0017	0.0009	-0.1589
20	0.3167	0.2258	0.0153	0.0005	-0.2417
21	0.2893	0.2090	0.0408	0.0009	-0.2507

Bersambung...

Lanjutan Tabel F.1

P	γ	h_{obj} (BBCA)	h_{obj} (ADRO)	J_{obj} (BC-AD)	C_{obj}
22	0.3507	0.1404	0.0337	0.0009	-0.1750
23	0.3430	0.0808	0.0881	0.0006	-0.1695
24	0.2669	0.1539	0.0734	0.0001	-0.2274
25	0.3791	0.0562	0.0591	-0.0000	-0.1153
26	0.3913	-0.0204	0.0641	0.0001	-0.0439
27	0.2983	-0.1528	-0.0048	0.0001	0.1575
28	0.3275	-0.0839	0.0555	0.0002	0.0282
29	0.2587	-0.1776	0.0339	0.0001	0.1435
30	0.2631	-0.1661	0.0357	0.0003	0.1301
31	0.3124	-0.0798	0.0653	0.0003	0.0142
32	0.4218	0.0295	0.0409	0.0004	-0.0707
33	0.3860	0.0406	0.0591	0.0000	-0.0996
34	0.3804	-0.0646	-0.0150	-0.0000	0.0796
35	0.3418	0.1067	-0.0148	0.0001	-0.0920

F.2 Riwayat Iterasi SBR N=2

Proses pencarian ekuilibrium menggunakan metode SBR dicatat pada tabel berikut. *Swap* menunjukkan indeks bit yang ditukar, *Utility* adalah nilai utilitas yang dicapai, dan *Bitstring* merepresentasikan portofolio strategi.

Tabel F.2: Riwayat Iterasi SBR N=2

Date	Iteration	Bitstring	Utility	Swap
2021-01-04	0	10	0.3498	Initial
2021-02-02	0	10	0.1132	Initial
2021-03-04	0	10	0.0767	Initial
2021-04-06	0	10	0.0605	Initial
2021-04-06	1	01	0.1058	0<->1
2021-05-05	0	10	0.1091	Initial
2021-05-05	1	01	0.1240	0<->1
2021-06-10	0	10	-0.0643	Initial
2021-06-10	1	01	0.0480	0<->1
2021-07-09	0	10	-0.1020	Initial
2021-08-10	0	10	0.1549	Initial
2021-09-10	0	10	0.1904	Initial
2021-10-11	0	10	0.5052	Initial
2021-11-10	0	10	0.4010	Initial
2021-12-09	0	10	0.5145	Initial
2022-01-10	0	10	0.7940	Initial
2022-02-09	0	10	0.6185	Initial
2022-03-14	0	10	0.9386	Initial
2022-04-12	0	10	0.6170	Initial
2022-05-23	0	10	0.8472	Initial
2022-06-23	0	10	0.6240	Initial
2022-07-22	0	10	0.3213	Initial
2022-08-23	0	10	0.4527	Initial
2022-09-21	0	10	0.4198	Initial
2022-10-20	0	10	0.2827	Initial

Bersambung...

Lanjutan Tabel F.2

Date	Iteration	Bitstring	Utility	Swap
2022-11-18	0	10	0.1628	Initial
2022-11-18	1	01	0.1773	0<->1
2022-12-19	0	10	0.3080	Initial
2023-01-17	0	10	0.1124	Initial
2023-01-17	1	01	0.1181	0<->1
2023-02-16	0	10	-0.0405	Initial
2023-02-16	1	01	0.1285	0<->1
2023-03-17	0	10	-0.3054	Initial
2023-03-17	1	01	-0.0093	0<->1
2023-04-27	0	10	-0.1674	Initial
2023-04-27	1	01	0.1114	0<->1
2023-05-30	0	10	-0.3549	Initial
2023-05-30	1	01	0.0681	0<->1
2023-07-05	0	10	-0.3317	Initial
2023-07-05	1	01	0.0719	0<->1
2023-08-04	0	10	-0.1590	Initial
2023-08-04	1	01	0.1314	0<->1
2023-09-05	0	10	0.0597	Initial
2023-09-05	1	01	0.0825	0<->1
2023-10-05	0	10	0.0811	Initial
2023-10-05	1	01	0.1181	0<->1
2023-11-03	0	10	-0.1293	Initial
2023-11-03	1	01	-0.0299	0<->1
2023-12-04	0	10	0.2136	Initial

F.3 Perbandingan Konvergensi VQE Berdasarkan Depth N=2

Tabel berikut menunjukkan perbandingan energi akhir (konvergensi) yang berhasil dicapai oleh algoritma VQE pada berbagai kedalaman sirkuit di setiap periode jendela waktu.

Tabel F.3: Perbandingan Energi Akhir VQE Berdasarkan Depth N=2

P	Date	D1	D2	D3	D4	D5	D6
1	2021-01-04	-0.1890	-0.3217	-0.3491	-0.3474	-0.3497	-0.3498
2	2021-02-02	-0.1111	4.7859	-0.1132	-0.1111	4.7760	-0.1132
3	2021-03-04	-0.0737	4.8600	-0.0767	-0.0737	4.8484	-0.0767
4	2021-04-06	-0.0601	-0.1019	-0.0890	-0.0898	-0.0996	-0.0994
5	2021-05-05	-0.1088	-0.1199	-0.1197	-0.1177	-0.1191	-0.1188
6	2021-06-10	0.0640	-0.0054	-0.0479	-0.0458	-0.0392	-0.0427
7	2021-07-09	0.1062	5.2114	0.1020	0.1062	5.1883	0.1020
8	2021-08-10	0.0711	-0.1221	-0.1548	-0.1549	-0.1549	-0.1549
9	2021-09-10	0.0022	-0.0816	-0.1902	-0.1902	-0.1902	-0.1903
10	2021-10-11	-0.1939	-0.2310	-0.5028	-0.5050	-0.5051	-0.5052
11	2021-11-10	-0.1886	-0.2333	-0.2752	-0.4009	-0.4009	-0.4010
12	2021-12-09	-0.1290	-0.2419	-0.5055	-0.4940	-0.5020	-0.5073
13	2022-01-10	-0.2892	-0.7889	-0.7934	-0.7940	-0.7939	-0.7940

Bersambung...

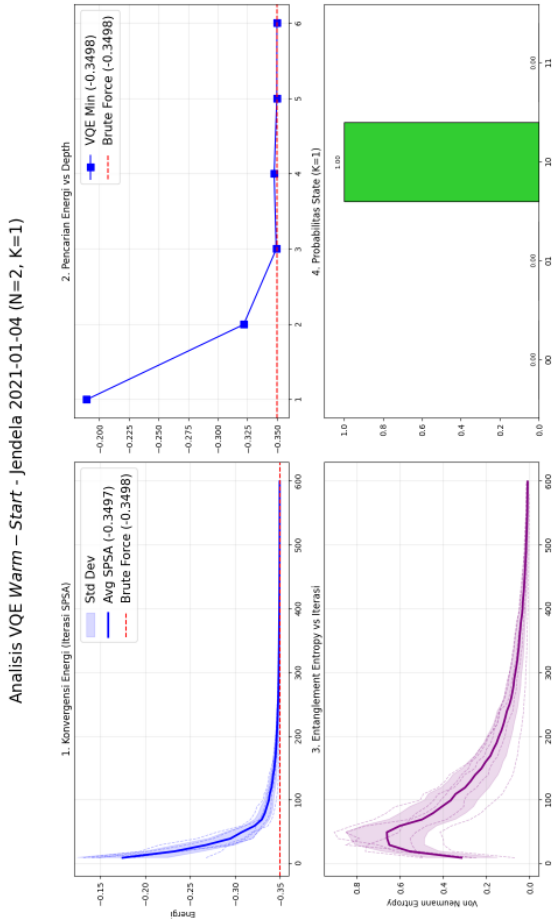
Lanjutan Tabel F.3

P	Date	D1	D2	D3	D4	D5	D6
14	2022-02-09	-0.2213	-0.5994	-0.6168	-0.6185	-0.6184	-0.6183
15	2022-03-14	-0.6649	-0.9386	-0.9386	-0.9378	-0.9386	-0.9386
16	2022-04-12	-0.1394	-0.6044	-0.6152	-0.6170	-0.6170	-0.6170
17	2022-05-23	-0.8328	-0.8472	-0.8472	-0.8472	-0.8472	-0.8472
18	2022-06-23	-0.1876	-0.6234	-0.6222	-0.6239	-0.6240	-0.6240
19	2022-07-22	0.0008	-0.3133	-0.3212	-0.3213	-0.3213	-0.3213
20	2022-08-23	-0.0380	-0.4508	-0.4524	-0.4527	-0.4527	-0.4527
21	2022-09-21	-0.0843	-0.4021	-0.4198	-0.4197	-0.4198	-0.4198
22	2022-10-20	-0.0692	0.0564	-0.2780	-0.2345	-0.2666	-0.2628
23	2022-11-18	-0.1625	-0.1733	-0.1729	-0.1711	-0.1724	-0.1722
24	2022-12-19	-0.1470	-0.2771	-0.3074	-0.3062	-0.3079	-0.3080
25	2023-01-17	-0.1122	-0.1161	-0.1162	-0.1154	-0.1157	-0.1156
26	2023-02-16	-0.1040	-0.1216	-0.1285	-0.1284	-0.1283	-0.1285
27	2023-03-17	0.3034	0.0954	0.0094	0.0096	0.0093	0.0093
28	2023-04-27	0.1639	-0.0866	-0.1113	-0.1113	-0.1114	-0.1112
29	2023-05-30	0.0065	-0.0681	-0.0681	-0.0681	-0.0681	-0.0681
30	2023-07-05	0.0620	-0.0719	-0.0718	-0.0719	-0.0719	-0.0719
31	2023-08-04	0.1511	-0.1312	-0.1313	-0.1313	-0.1313	-0.1313
32	2023-09-05	-0.0592	-0.0781	-0.0765	-0.0731	-0.0764	-0.0759
33	2023-10-05	-0.0806	-0.1133	-0.1078	-0.1047	-0.1118	-0.1112
34	2023-11-03	0.1294	0.0375	0.0313	0.0366	0.0977	0.0301
35	2023-12-04	0.0292	-0.1775	-0.2135	-0.2135	-0.2135	-0.2136

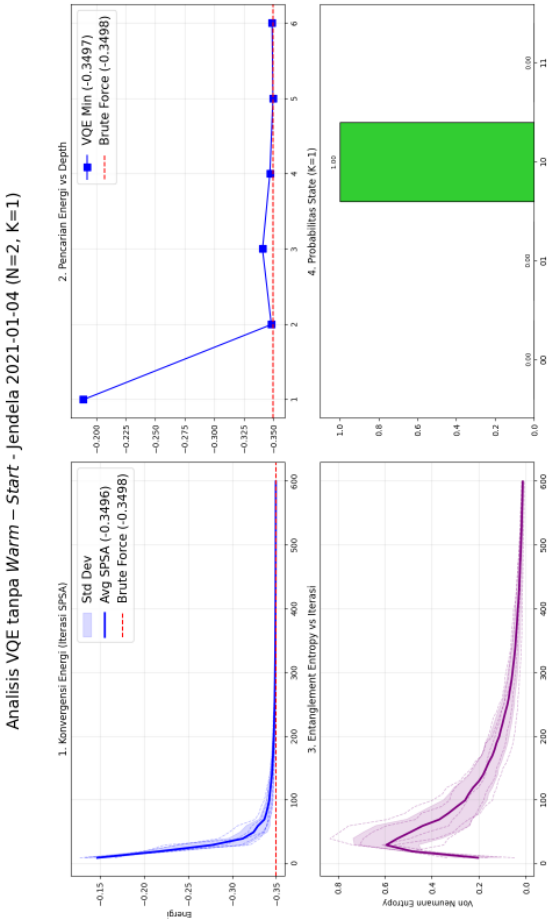
Lampiran G

Visualisasi Perbandingan Alokasi Portofolio N=2 (GT vs No-GT)

Lampiran ini menampilkan grafik alokasi aset pada setiap jendela waktu untuk sistem dengan $N = 2$ aset, membandingkan pengaruh model *Game Theory* terhadap keputusan investasi.

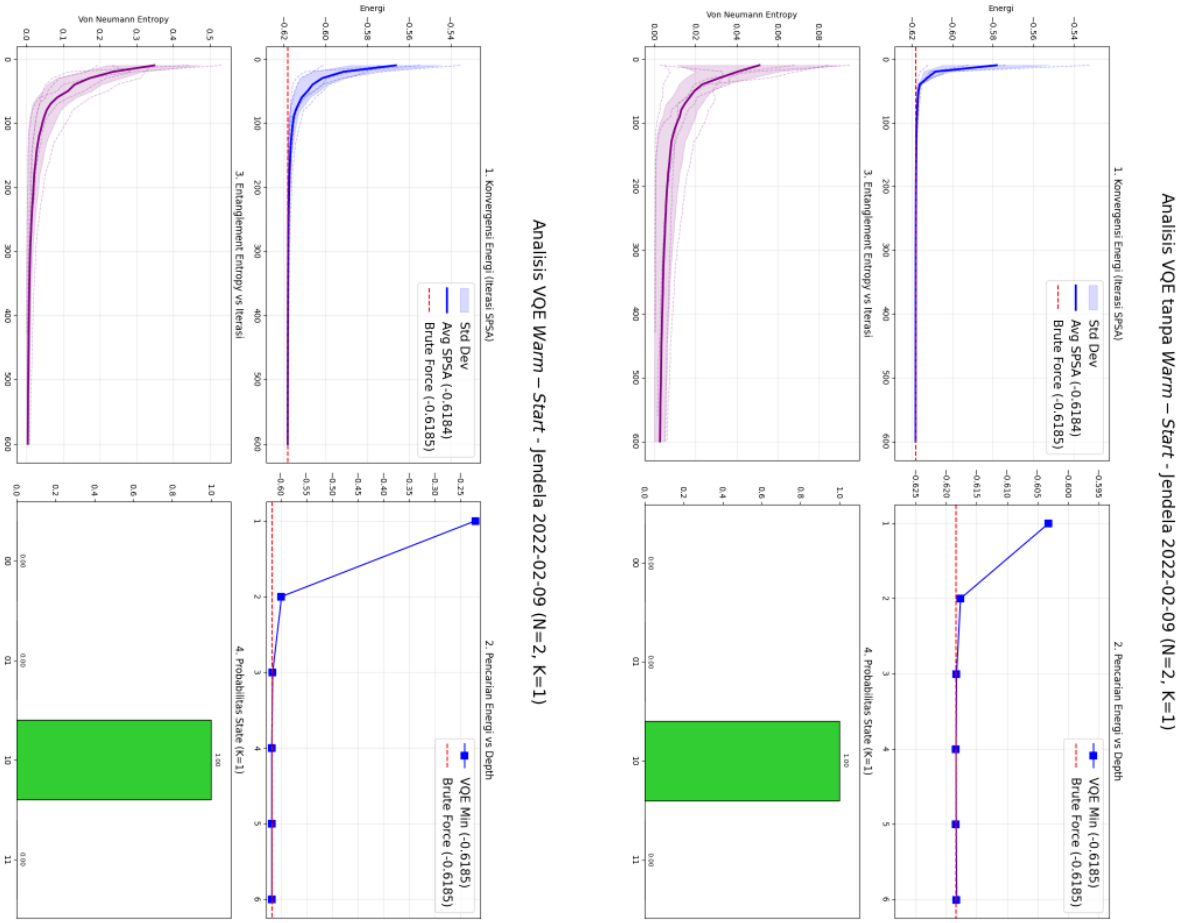


(a) Dengan *Game Theory*



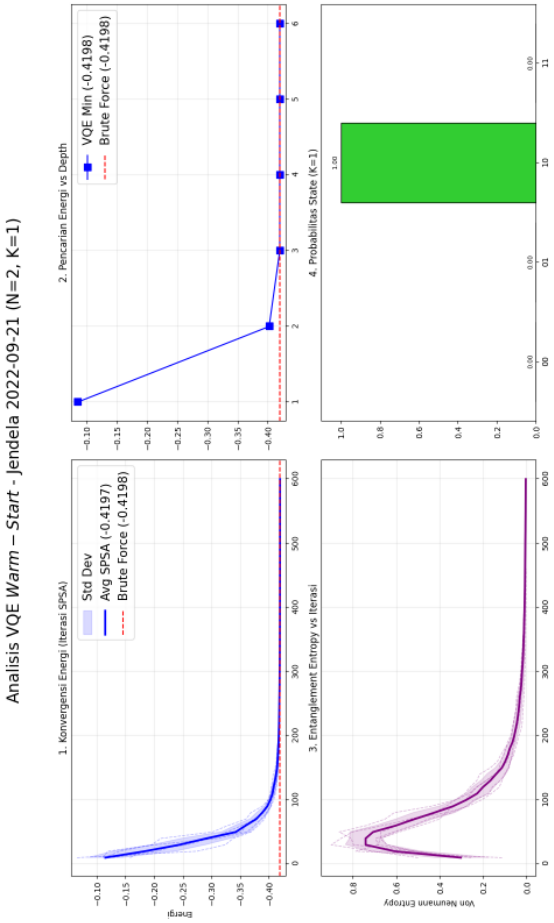
(b) Tanpa *Game Theory*

Gambar G.1: Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2021-01-04

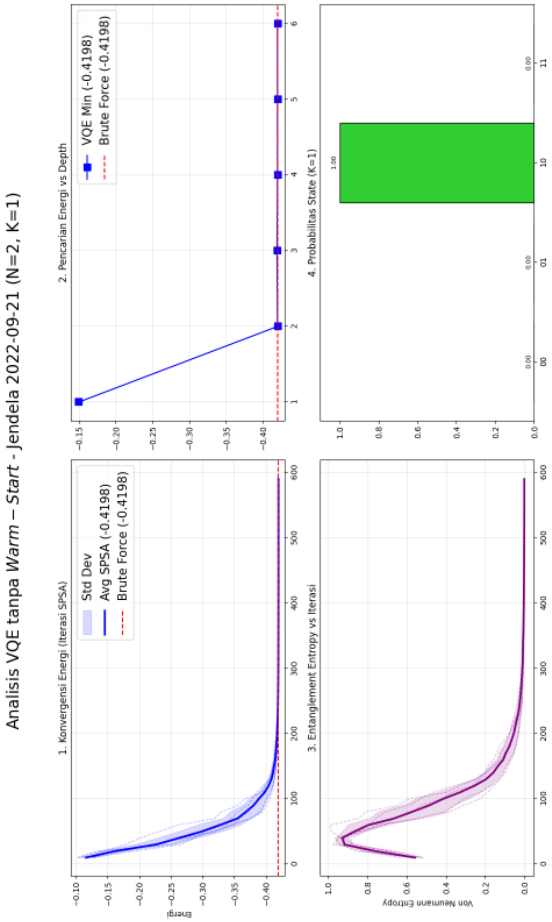


(a) Dengan *Game Theory* (b) Tanpa *Game Theory*

Gambar G.2: Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2022-02-09

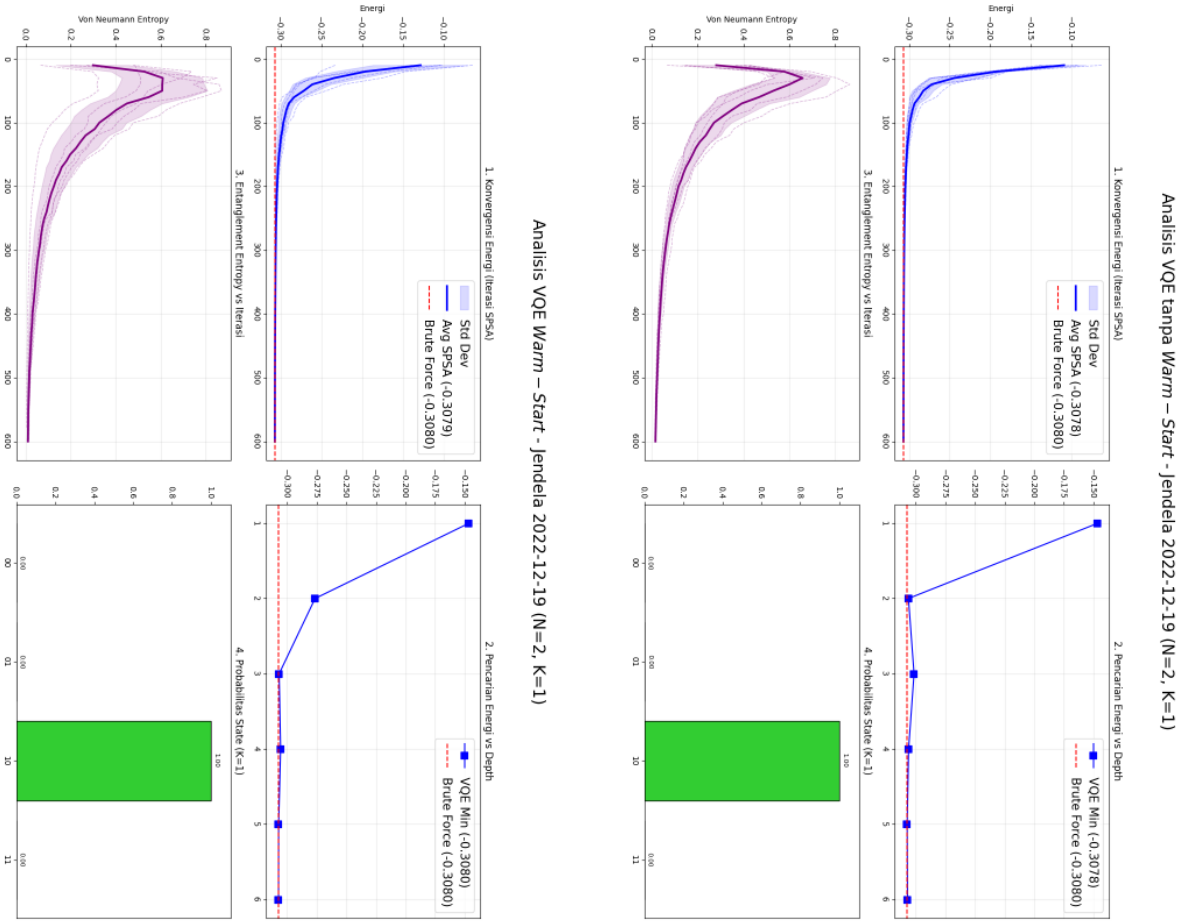


(a) Dengan *Game Theory*



(b) Tanpa *Game Theory*

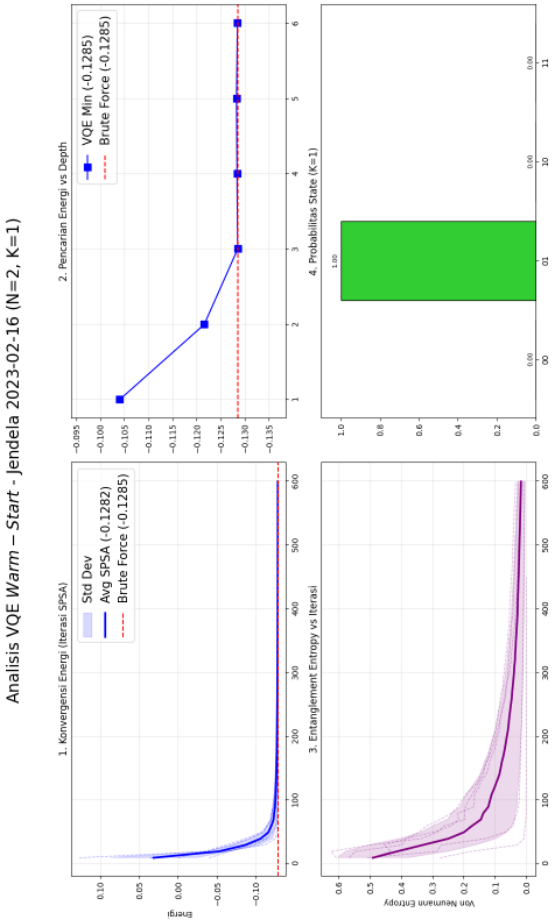
Gambar G.3: Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2022-09-21



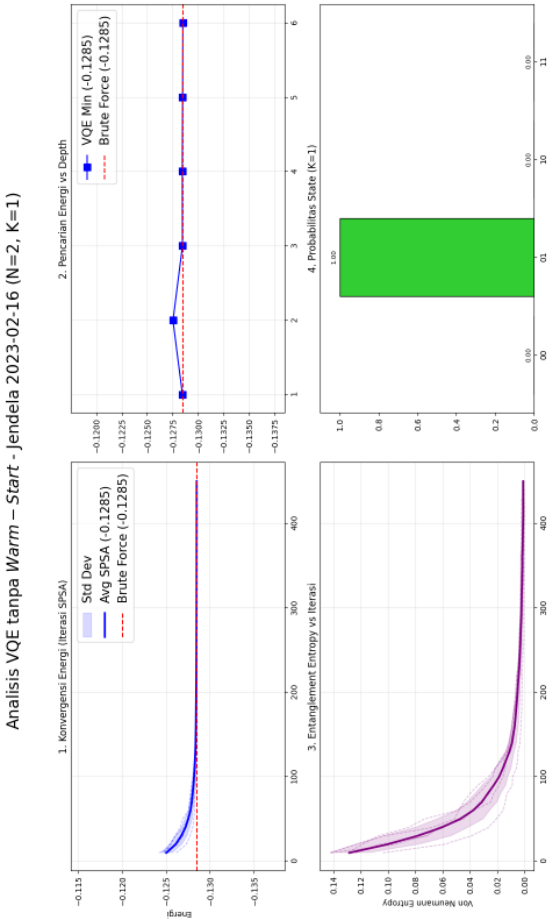
(a) Dengan *Game Theory*

(b) Tanpa *Game Theory*

Gambar G.4: Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2022-12-19

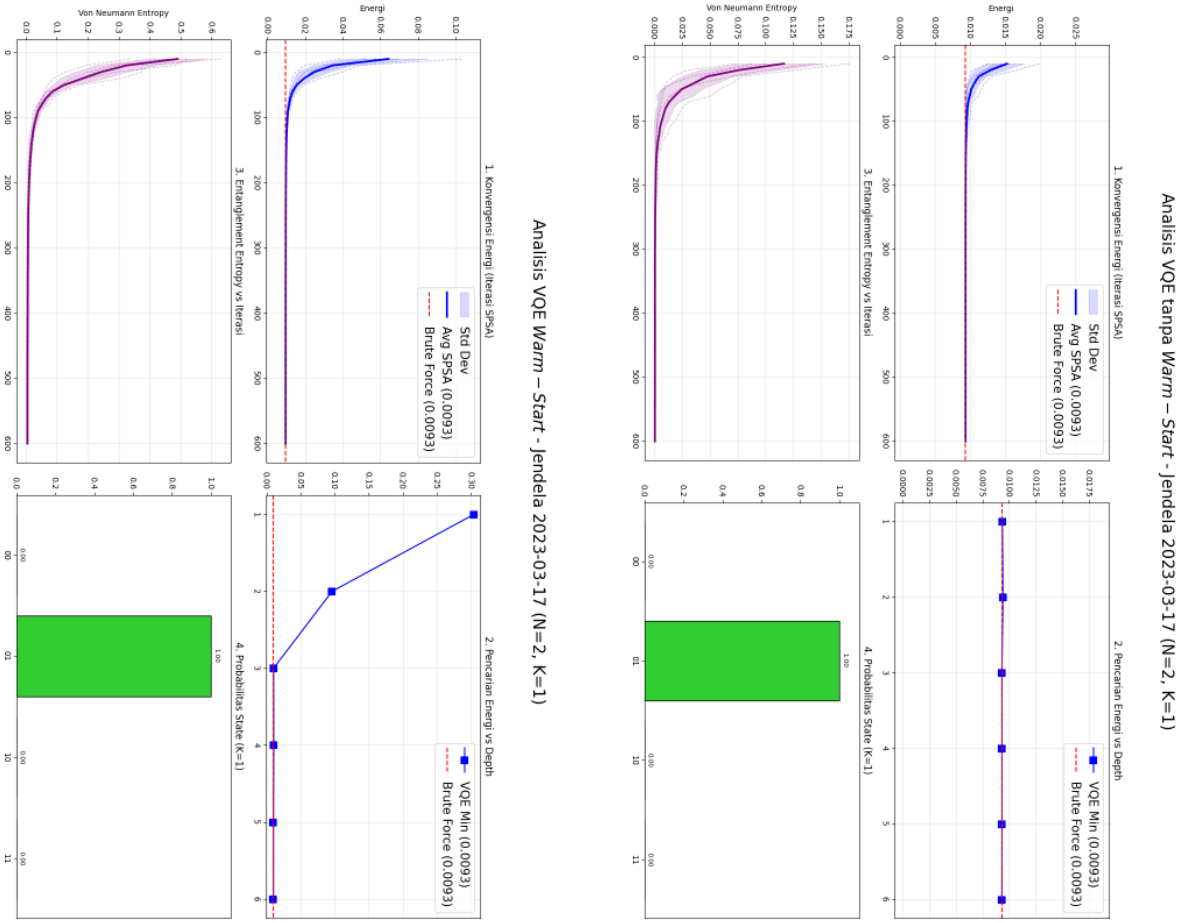


(a) Dengan *Game Theory*



(b) Tanpa *Game Theory*

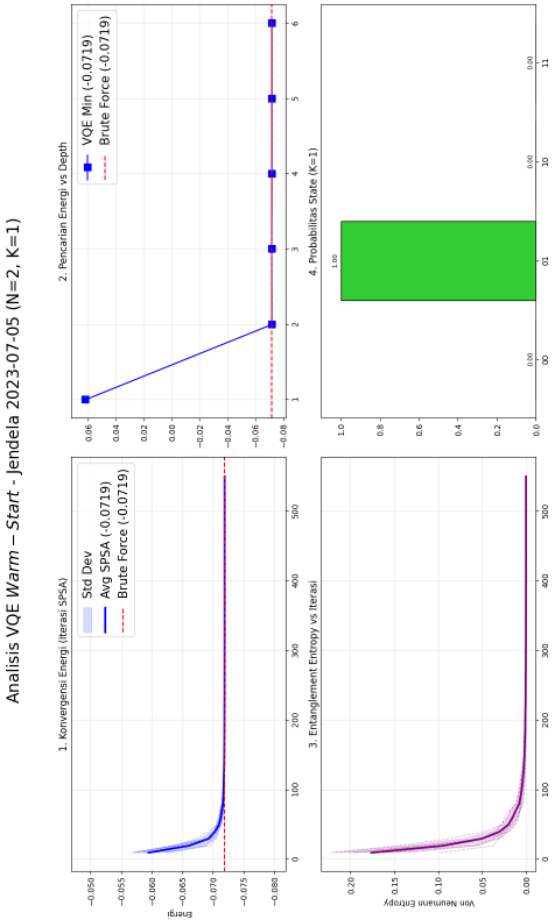
Gambar G.5: Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2023-02-16



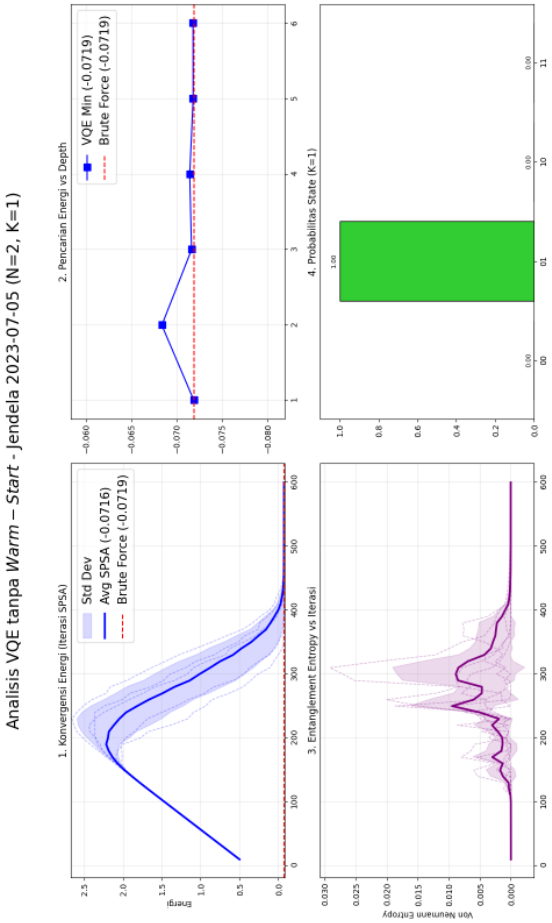
(a) Dengan *Game Theory*

(b) Tanpa *Game Theory*

Gambar G.6: Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2023-03-17

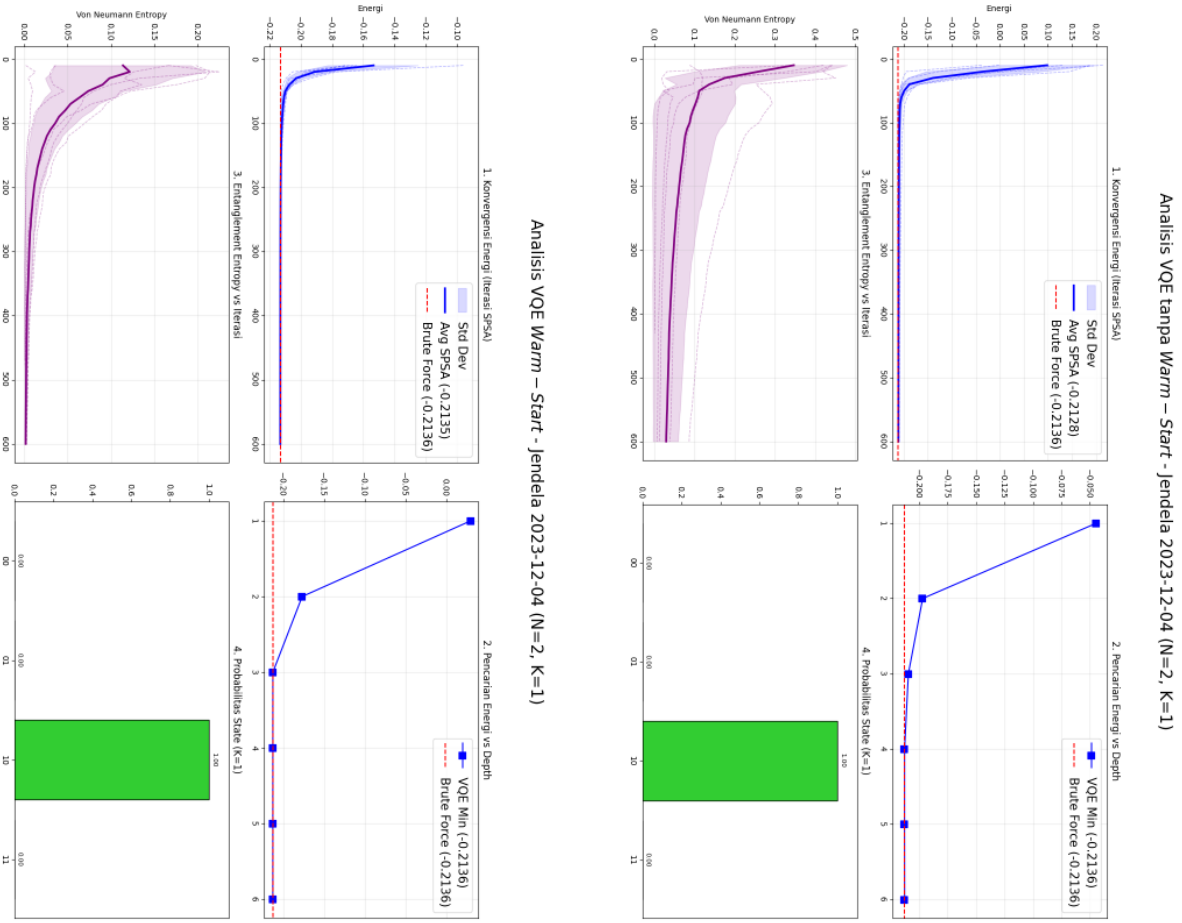


(a) Dengan *Game Theory*



(b) Tanpa *Game Theory*

Gambar G.7: Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2023-07-05



(a) Dengan *Game Theory*

(b) Tanpa *Game Theory*

Gambar G.8: Perbandingan Alokasi Portofolio N=2 Jendela Waktu 2023-12-04

Lampiran H

Data Numerik Hasil Simulasi N=4

Lampiran ini menyajikan dataset numerik untuk simulasi dengan $N = 4$ aset. Untuk data harga harian saham untuk periode simulasi, dapat merujuk pada Tabel E.1 di Lampiran E.

H.1 Parameter Dinamis Hamiltonian N=4

Tabel berikut menyajikan hasil komputasi harian yang diakumulasikan secara bulanan untuk mendapatkan bias linier h_{obj}, offset C_{obj}, dan nilai γ dinamis pada sistem N=4.

Tabel H.1: Parameter Dinamis Portofolio N=4 (γ, h_{obj}, C_{obj})

P	Tanggal	γ	h_{obj} (BB)	h_{obj} (AD)	h_{obj} (SM)	h_{obj} (TL)	C_{obj}
1	2021-01-04	0.2984	0.0875	0.0278	0.0753	0.0465	-0.2399
2	2021-02-02	0.4001	0.0284	0.0178	0.0561	0.0269	-0.1338
3	2021-03-04	0.4285	0.0192	0.0407	0.0218	0.0176	-0.1041
4	2021-04-06	0.3754	0.0158	0.0635	0.0471	0.0261	-0.1558
5	2021-05-05	0.3886	0.0274	0.0536	0.0239	0.0305	-0.1386
6	2021-06-10	0.4286	-0.0158	0.0116	-0.0233	0.0116	0.0134
7	2021-07-09	0.3231	-0.0247	-0.0209	-0.0765	-0.0265	0.1471
8	2021-08-10	0.3562	0.0396	0.0143	-0.0573	-0.0178	0.0200
9	2021-09-10	0.3675	0.0487	0.0171	-0.0517	-0.0007	-0.0152
10	2021-10-11	0.2440	0.1269	0.0457	-0.0510	0.0482	-0.1708
11	2021-11-10	0.3056	0.1013	0.0525	-0.0047	0.0469	-0.1971
12	2021-12-09	0.2640	0.1298	0.0443	-0.0557	0.0321	-0.1515
13	2022-01-10	0.1735	0.1987	0.0737	-0.0470	0.0639	-0.2898
14	2022-02-09	0.2103	0.1552	0.0715	-0.0356	0.0545	-0.2463
15	2022-03-14	0.1535	0.2353	0.0852	-0.0581	0.0536	-0.3163
16	2022-04-12	0.2446	0.1553	0.0507	-0.0412	0.0224	-0.1877
17	2022-05-23	0.1995	0.2126	0.0344	-0.0907	-0.0011	-0.1559
18	2022-06-23	0.3274	0.1572	0.0057	-0.0159	0.0112	-0.1592
19	2022-07-22	0.3962	0.0831	0.0213	-0.0079	-0.0007	-0.0971
20	2022-08-23	0.3495	0.1156	0.0329	-0.0126	0.0077	-0.1445
21	2022-09-21	0.3532	0.1061	0.0060	0.0274	0.0204	-0.1611
22	2022-10-20	0.3464	0.0727	-0.0130	0.0579	0.0172	-0.1358
23	2022-11-18	0.3620	0.0425	0.0073	0.0451	0.0443	-0.1396
24	2022-12-19	0.3270	0.0780	-0.0279	-0.0044	0.0367	-0.0827

Bersambung...

Lanjutan Tabel H.1

P	Tanggal	γ	h_{obj} (BB)	h_{obj} (AD)	h_{obj} (SM)	h_{obj} (TL)	C_{obj}
25	2023-01-17	0.3697	0.0300	-0.0208	0.0370	0.0297	-0.0761
26	2023-02-16	0.3415	-0.0081	-0.0491	0.0392	0.0325	-0.0145
27	2023-03-17	0.3173	-0.0751	-0.0197	-0.0398	-0.0022	0.1366
28	2023-04-27	0.3330	-0.0405	0.0057	-0.0590	0.0280	0.0655
29	2023-05-30	0.2944	-0.0877	0.0145	-0.0510	0.0171	0.1068
30	2023-07-05	0.3068	-0.0820	0.0289	-0.0187	0.0180	0.0535
31	2023-08-04	0.3777	-0.0389	-0.0030	0.0151	0.0328	-0.0065
32	2023-09-05	0.4263	0.0164	0.0093	-0.0120	0.0207	-0.0349
33	2023-10-05	0.4014	0.0217	-0.0055	0.0222	0.0297	-0.0685
34	2023-11-03	0.3617	-0.0307	-0.0324	0.0234	-0.0073	0.0466
35	2023-12-04	0.3681	0.0543	-0.0064	0.0298	-0.0072	-0.0709

Selanjutnya adalah parameter interaksi J antar masing-masing aset pada setiap jendela waktu.

Tabel H.2: Kompilasi Parameter Bias h_{obj}, Konstanta C_{obj}, dan Koefisien γ N=4

P	Tanggal	BB-TL	BB-SM	BB-AD	TL-SM	TL-AD	SM-AD
1	2021-01-04	0.0004	0.0004	0.0006	0.0004	0.0005	0.0005
2	2021-02-02	0.0006	0.0007	0.0011	0.0006	0.0007	0.0009
3	2021-03-04	0.0006	0.0008	0.0011	0.0007	0.0007	0.0009
4	2021-04-06	0.0004	0.0005	0.0007	0.0004	0.0005	0.0007
5	2021-05-05	0.0004	0.0005	0.0007	0.0005	0.0005	0.0007
6	2021-06-10	0.0003	0.0005	0.0005	0.0005	0.0004	0.0005
7	2021-07-09	0.0002	0.0004	0.0002	0.0003	0.0002	0.0003
8	2021-08-10	0.0001	0.0003	0.0002	0.0003	0.0002	0.0002
9	2021-09-10	0.0002	0.0004	0.0003	0.0004	0.0002	0.0003
10	2021-10-11	0.0001	0.0001	0.0002	0.0002	0.0002	0.0002
11	2021-11-10	0.0001	0.0000	0.0002	0.0003	0.0002	0.0003
12	2021-12-09	0.0001	-0.0000	0.0003	0.0002	0.0002	0.0002
13	2022-01-10	0.0001	-0.0001	0.0002	0.0001	0.0001	0.0001
14	2022-02-09	0.0001	-0.0001	0.0002	0.0001	0.0001	0.0002
15	2022-03-14	0.0000	-0.0002	0.0001	0.0001	0.0001	0.0001
16	2022-04-12	0.0001	-0.0002	0.0002	0.0002	0.0000	0.0002
17	2022-05-23	0.0001	-0.0000	0.0001	0.0002	0.0001	0.0001
18	2022-06-23	0.0002	-0.0001	0.0002	0.0004	0.0002	0.0002
19	2022-07-22	0.0002	-0.0000	0.0000	0.0005	0.0002	0.0002
20	2022-08-23	0.0001	-0.0001	0.0000	0.0004	0.0002	0.0001
21	2022-09-21	0.0003	0.0002	-0.0000	0.0003	0.0003	0.0001
22	2022-10-20	0.0002	0.0001	-0.0001	0.0003	0.0003	0.0001
23	2022-11-18	0.0001	0.0001	-0.0001	0.0001	0.0002	0.0000
24	2022-12-19	0.0000	-0.0001	-0.0001	0.0001	0.0002	0.0001
25	2023-01-17	-0.0000	-0.0002	0.0000	0.0001	0.0002	0.0001
26	2023-02-16	0.0000	-0.0002	-0.0000	0.0001	0.0002	0.0001
27	2023-03-17	0.0000	-0.0002	-0.0000	0.0001	0.0002	0.0001
28	2023-04-27	0.0001	-0.0001	0.0000	0.0001	0.0001	0.0001
29	2023-05-30	0.0000	-0.0001	0.0000	0.0001	0.0001	0.0001
30	2023-07-05	0.0001	0.0000	0.0000	0.0001	0.0001	0.0000
31	2023-08-04	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001
32	2023-09-05	0.0001	0.0002	0.0001	0.0000	0.0001	0.0001

Bersambung...

Lanjutan Tabel H.2

P	Tanggal	BB-TL	BB-SM	BB-AD	TL-SM	TL-AD	SM-AD
33	2023-10-05	0.0000	0.0001	0.0001	0.0000	0.0001	0.0001
34	2023-11-03	-0.0000	0.0001	0.0000	0.0000	0.0000	0.0001
35	2023-12-04	0.0000	0.0001	0.0001	0.0001	0.0000	0.0002

H.2 Riwayat Iterasi SBR N=4

Proses pencarian ekuilibrium menggunakan metode SBR dicatat pada tabel berikut. extitSwap menunjukkan indeks bit yang ditukar, extitUtility adalah nilai utilitas yang dicapai, dan extitBitstring merepresentasikan portofolio strategi.

Tabel H.3: Riwayat Iterasi SBR N=4

Date	Iteration	Bitstring	Utility	Swap
2021-01-04	0	1100	0.2717	Initial
2021-01-04	1	1010	0.3294	1<->2
2021-02-02	0	1100	0.1168	Initial
2021-02-02	1	1010	0.1756	1<->2
2021-03-04	0	1100	0.0801	Initial
2021-03-04	1	1001	0.1259	1<->3
2021-03-04	2	0011	0.1316	0<->2
2021-04-06	0	1100	0.0882	Initial
2021-04-06	1	0101	0.1836	0<->3
2021-04-06	2	0011	0.2257	1<->2
2021-05-05	0	1100	0.1200	Initial
2021-05-05	1	0101	0.1726	0<->3
2021-06-10	0	1100	-0.0049	Initial
2021-06-10	1	0101	0.0499	0<->3
2021-07-09	0	1100	-0.1002	Initial
2021-07-09	1	1001	-0.0892	1<->3
2021-08-10	0	1100	0.0455	Initial
2021-08-10	1	1001	0.1093	1<->3
2021-09-10	0	1100	0.0984	Initial
2021-09-10	1	1001	0.1337	1<->3
2021-10-11	0	1100	0.3516	Initial
2021-11-10	0	1100	0.2977	Initial
2021-11-10	1	1001	0.3088	1<->3
2021-12-09	0	1100	0.3251	Initial
2021-12-09	1	1001	0.3492	1<->3
2022-01-10	0	1100	0.5259	Initial
2022-01-10	1	1001	0.5453	1<->3
2022-02-09	0	1100	0.4201	Initial
2022-02-09	1	1001	0.4540	1<->3
2022-03-14	0	1100	0.5779	Initial
2022-03-14	1	1001	0.6409	1<->3
2022-04-12	0	1100	0.3559	Initial
2022-04-12	1	1001	0.4123	1<->3
2022-05-23	0	1100	0.4239	Initial
2022-05-23	1	1001	0.4947	1<->3
2022-06-23	0	1100	0.3382	Initial

Bersambung...

Lanjutan Tabel H.3

Date	Iteration	Bitstring	Utility	Swap
2022-07-22	0	1100	0.1664	Initial
2022-07-22	1	1001	0.2102	1<->3
2022-08-23	0	1100	0.2478	Initial
2022-08-23	1	1001	0.2979	1<->3
2022-09-21	0	1100	0.2545	Initial
2022-09-21	1	1010	0.2683	1<->2
2022-10-20	0	1100	0.1811	Initial
2022-10-20	1	1010	0.2623	1<->2
2022-11-18	0	1100	0.1743	Initial
2022-11-18	1	0110	0.1796	0<->2
2022-12-19	0	1100	0.2298	Initial
2023-01-17	0	1100	0.1197	Initial
2023-01-17	1	1010	0.1343	1<->2
2023-02-16	0	1100	0.0488	Initial
2023-02-16	1	0110	0.1434	0<->2
2023-03-17	0	1100	-0.1544	Initial
2023-03-17	1	0101	-0.0435	0<->3
2023-04-27	0	1100	-0.0247	Initial
2023-04-27	1	0101	0.0680	0<->3
2023-05-30	0	1100	-0.1408	Initial
2023-05-30	1	0101	0.0637	0<->3
2023-07-05	0	1100	-0.1275	Initial
2023-07-05	1	0101	0.0942	0<->3
2023-08-04	0	1100	-0.0115	Initial
2023-08-04	1	0110	0.0965	0<->2
2023-09-05	0	1100	0.0750	Initial
2023-10-05	0	1100	0.1035	Initial
2023-10-05	1	0110	0.1042	0<->2
2023-11-03	0	1100	-0.0753	Initial
2023-11-03	1	0110	0.0327	0<->2
2023-12-04	0	1100	0.0947	Initial
2023-12-04	1	1010	0.1689	1<->2

H.3 Perbandingan Konvergensi VQE Berdasarkan Depth

Tabel berikut menunjukkan perbandingan energi akhir (konvergensi) yang berhasil dicapai oleh algoritma VQE pada berbagai kedalaman sirkuit (Depth 1 hingga 8) di setiap periode jendela waktu.

Tabel H.4: Perbandingan Energi Akhir VQE Berdasarkan Depth N=4

P	Date	D1	D2	D3	D4	D5	D6	D7	D8
1	2021-01-04	-0.2717	4.8223	4.6980	4.6738	4.9412	-0.2340	4.5782	-0.2099
2	2021-02-02	-0.1168	4.9384	4.7936	4.8495	4.9590	-0.0981	4.7716	-0.1539
3	2021-03-04	-0.1204	2.5696	-0.1026	-0.1092	-0.1167	-0.0952	-0.0896	-0.0936
4	2021-04-06	-0.1620	0.8819	-0.1805	-0.1583	-0.1995	-0.1288	-0.1477	-0.1418
5	2021-05-05	-0.1128	4.9364	19.7290	-0.1071	-0.1200	4.9422	4.7809	4.7737

Bersambung...

Lanjutan Tabel H.4

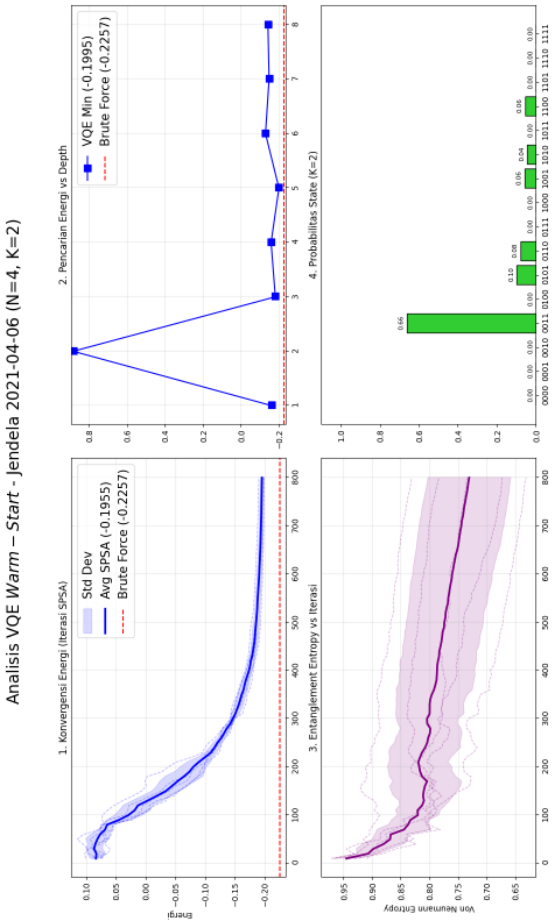
P	Date	D1	D2	D3	D4	D5	D6	D7	D8
6	2021-06-10	0.0202	4.9745	20.0318	0.0748	0.0049	5.0291	4.9977	4.9823
7	2021-07-09	0.0967	0.1717	0.1464	0.1247	0.1482	0.0964	0.1332	0.1496
8	2021-08-10	-0.0200	0.0051	0.0069	-0.0210	-0.0558	-0.0212	-0.0443	-0.0627
9	2021-09-10	-0.0667	-0.0089	-0.0311	-0.1055	-0.0748	-0.0554	-0.0656	-0.0861
10	2021-10-11	-0.3511	-0.0295	-0.3449	-0.1902	-0.3036	-0.1953	-0.3275	-0.2275
11	2021-11-10	-0.2484	-0.1485	-0.2080	-0.2324	-0.2216	-0.2070	-0.2583	-0.2917
12	2021-12-09	-0.2375	-0.1548	-0.3286	-0.2518	-0.1965	-0.2059	-0.3126	-0.2954
13	2022-01-10	-0.5141	-0.5341	-0.5433	-0.5432	-0.5420	-0.5399	-0.5405	-0.5349
14	2022-02-09	-0.3344	-0.2350	-0.2984	-0.3264	-0.3020	-0.3388	-0.3992	-0.3362
15	2022-03-14	-0.5777	-0.6102	-0.4650	-0.5749	-0.4060	-0.6367	-0.5505	-0.5655
16	2022-04-12	-0.2511	-0.1718	-0.2280	-0.2333	-0.3719	-0.3384	-0.2657	-0.3463
17	2022-05-23	-0.4235	-0.4593	-0.4936	-0.4907	-0.4768	-0.4760	-0.4713	-0.4744
18	2022-06-23	-0.3265	-0.0082	-0.3262	-0.1559	-0.3200	-0.3181	-0.3002	-0.2753
19	2022-07-22	-0.1047	-0.0879	-0.0872	-0.1056	-0.1696	-0.1522	-0.1671	-0.1293
20	2022-08-23	-0.1647	-0.1392	-0.1900	-0.2031	-0.2265	-0.2123	-0.2422	-0.2272
21	2022-09-21	-0.2545	4.7870	4.8914	4.7338	4.9872	-0.2258	4.6914	-0.0684
22	2022-10-20	-0.1811	4.8539	4.8753	4.8451	5.0253	-0.1210	4.7039	-0.0911
23	2022-11-18	0.0255	-0.1759	-0.0933	-0.1372	-0.1480	-0.1615	-0.1459	-0.1465
24	2022-12-19	-0.1468	0.0633	-0.1005	-0.0195	-0.1914	-0.1007	-0.0853	-0.0349
25	2023-01-17	-0.1197	4.9404	4.9086	4.9221	5.0409	-0.0186	4.8060	-0.0326
26	2023-02-16	0.2185	-0.0924	0.0626	-0.0730	-0.0272	-0.0415	-0.0506	-0.0712
27	2023-03-17	0.0837	5.0038	20.2735	0.2293	0.1545	5.1505	5.1237	5.1940
28	2023-04-27	0.0617	4.9434	20.1315	0.1984	0.0247	5.0811	5.0507	5.0134
29	2023-05-30	0.0675	4.9653	20.2141	0.2769	0.1408	5.1753	5.0387	5.1120
30	2023-07-05	0.0011	4.9636	20.1076	0.2009	0.1275	5.1637	4.9435	5.0699
31	2023-08-04	-0.0243	-0.0476	-0.0469	-0.0563	-0.0578	-0.0456	-0.0264	0.0068
32	2023-09-05	-0.0744	0.0025	-0.0522	-0.0605	-0.0545	-0.0529	-0.0446	-0.0364
33	2023-10-05	0.0883	-0.0967	-0.0417	-0.0507	-0.0819	-0.0880	-0.0621	-0.0866
34	2023-11-03	0.0446	-0.0068	0.0397	0.0710	-0.0280	0.0045	-0.0100	0.0244
35	2023-12-04	-0.0947	4.8910	4.9673	4.9179	5.0122	-0.0965	4.8457	-0.0473

Halaman ini sengaja dikosongkan

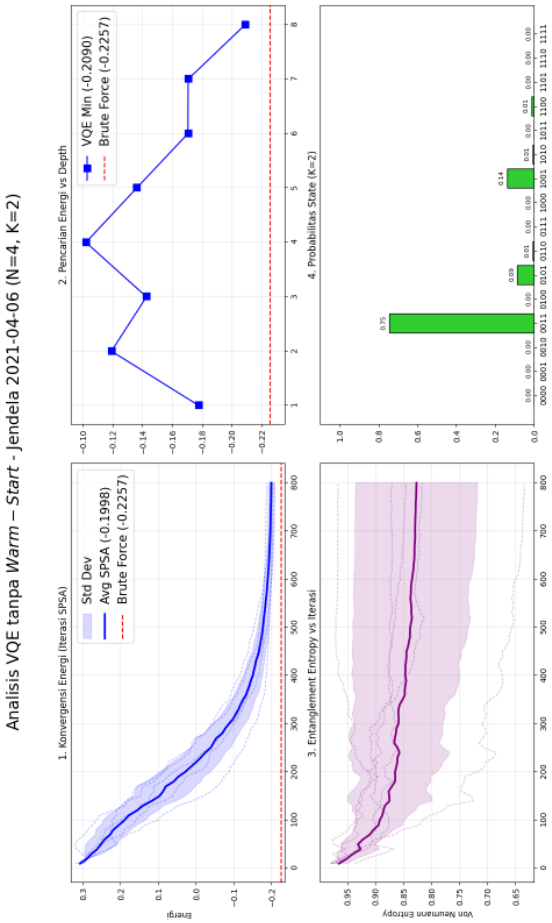
Lampiran I

Visualisasi Perbandingan Alokasi Portofolio N=4 (GT vs No-GT)

Lampiran ini menampilkan grafik alokasi aset pada setiap jendela waktu untuk sistem dengan $N = 4$ aset, membandingkan pengaruh model *Game Theory* terhadap keputusan investasi.

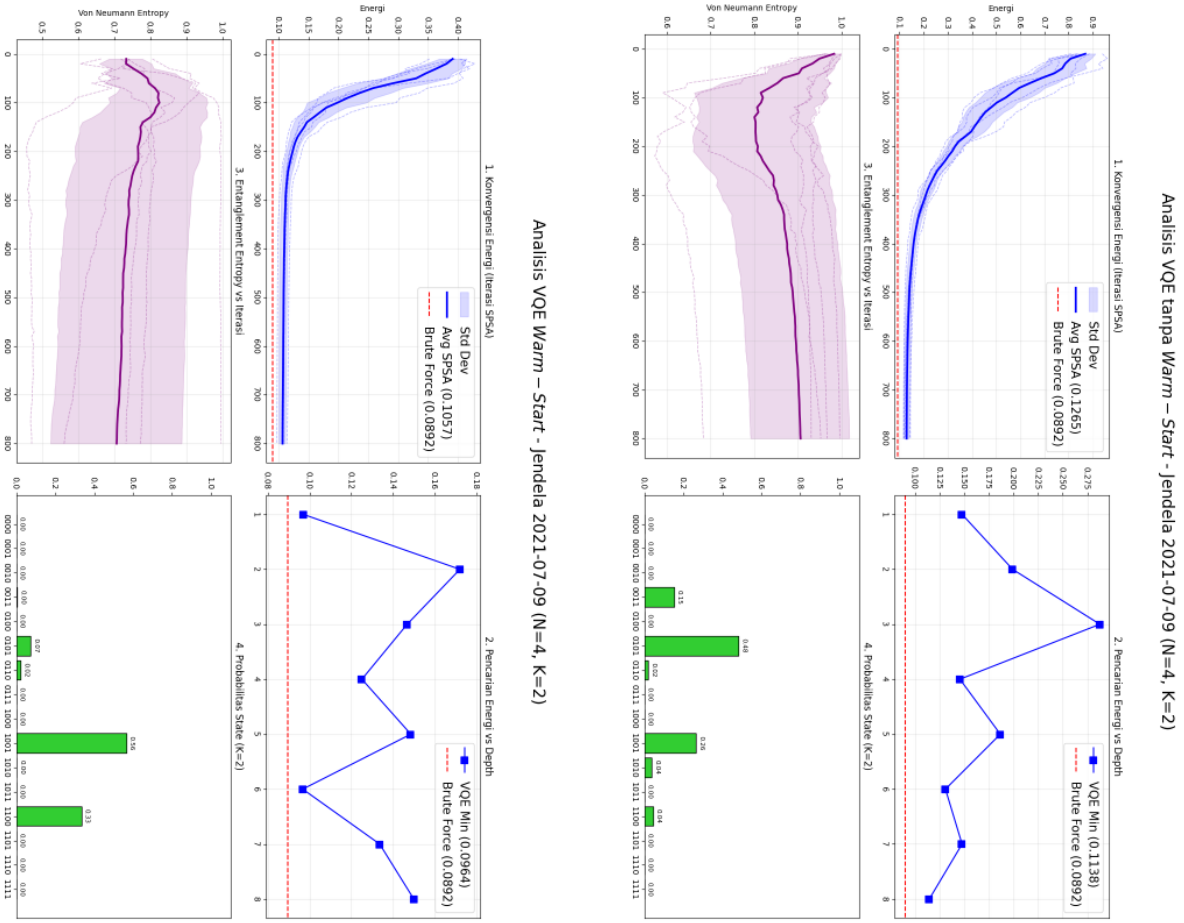


(a) Dengan *Game Theory*



(b) Tanpa *Game Theory*

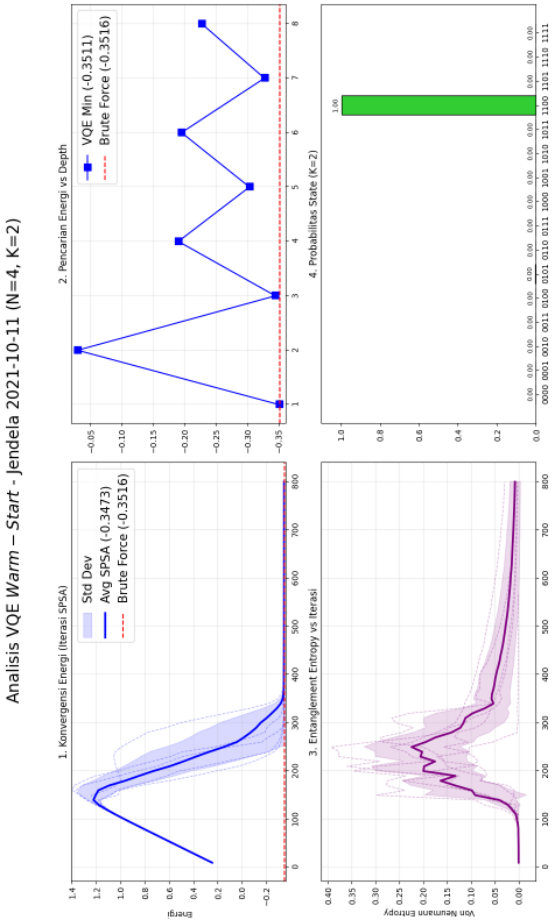
Gambar I.1: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2021-04-06



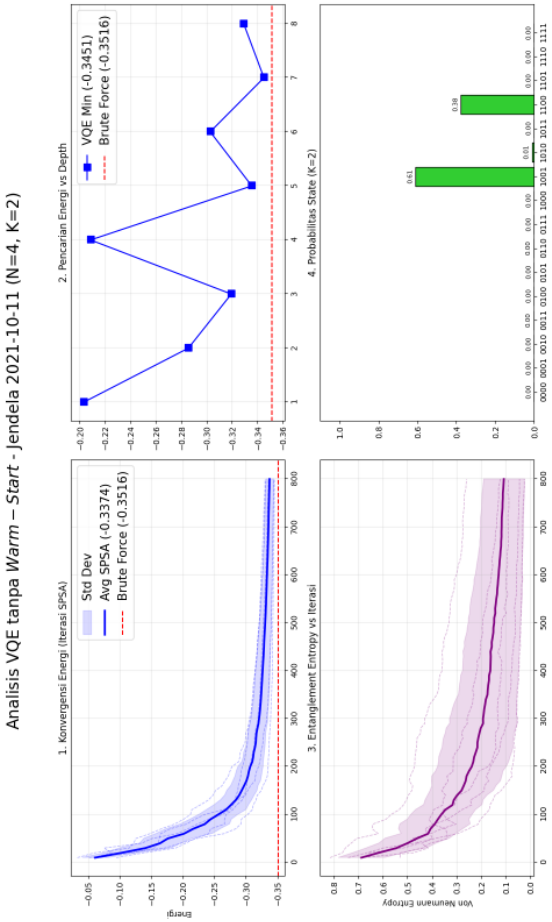
(a) Dengan *Game Theory*

(b) Tanpa *Game Theory*

Gambar I.2: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2021-07-09

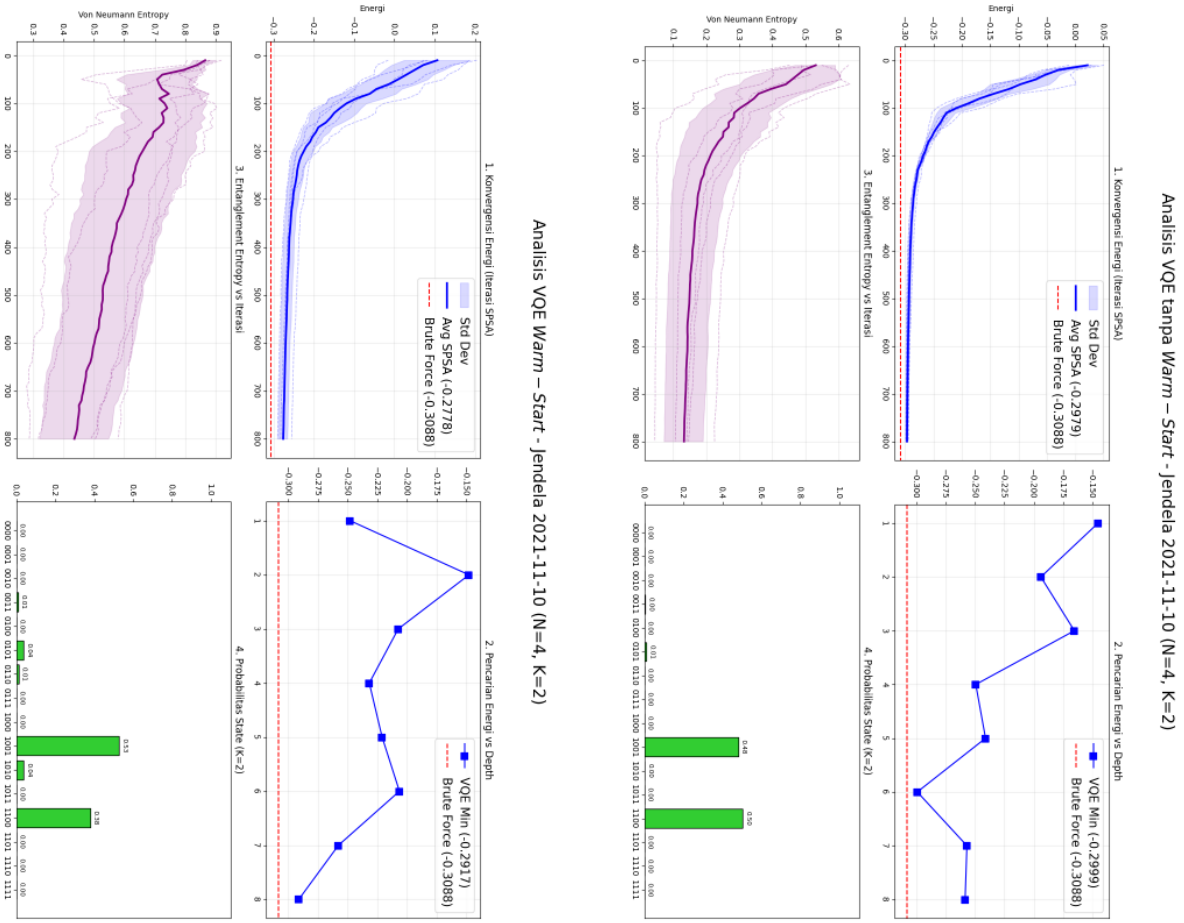


(a) Dengan *Game Theory*



(b) Tanpa *Game Theory*

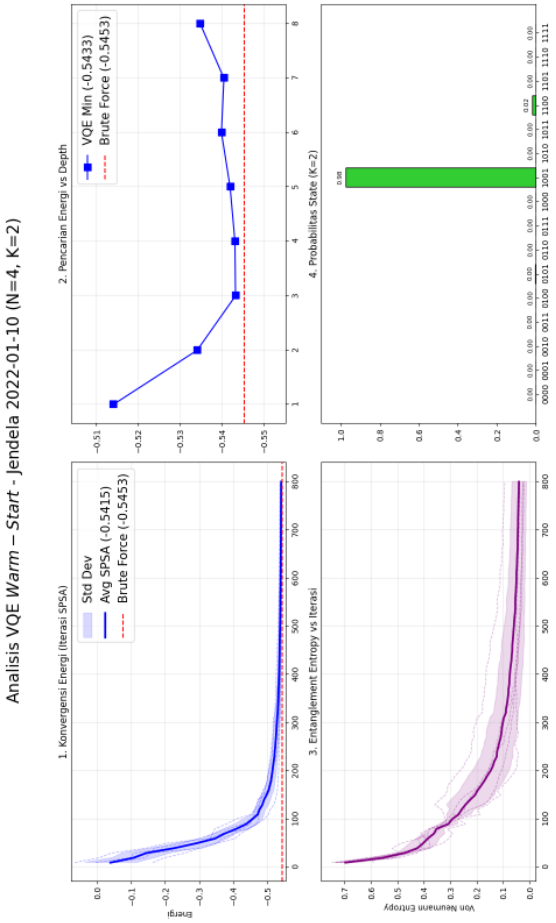
Gambar I.3: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2021-10-11



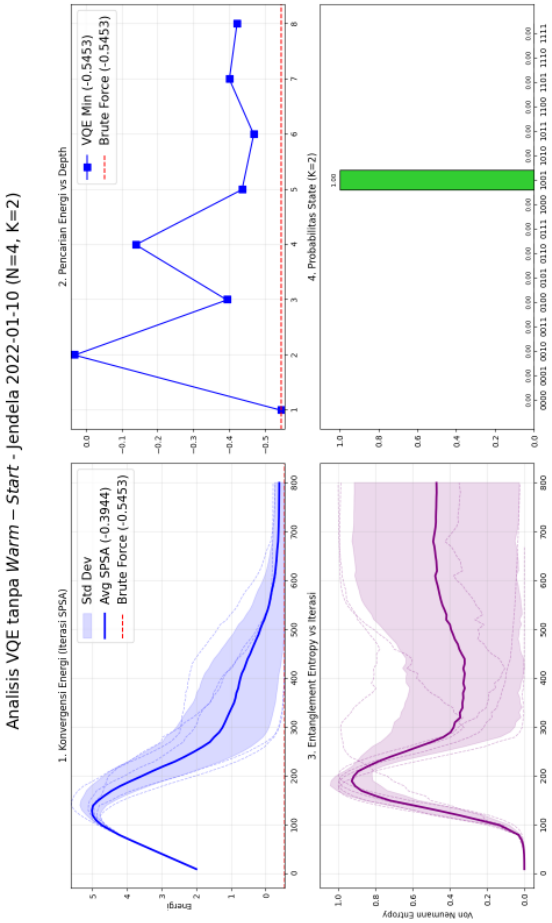
(a) Dengan *Game Theory*

(b) Tanpa *Game Theory*

Gambar I.4: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2021-11-10

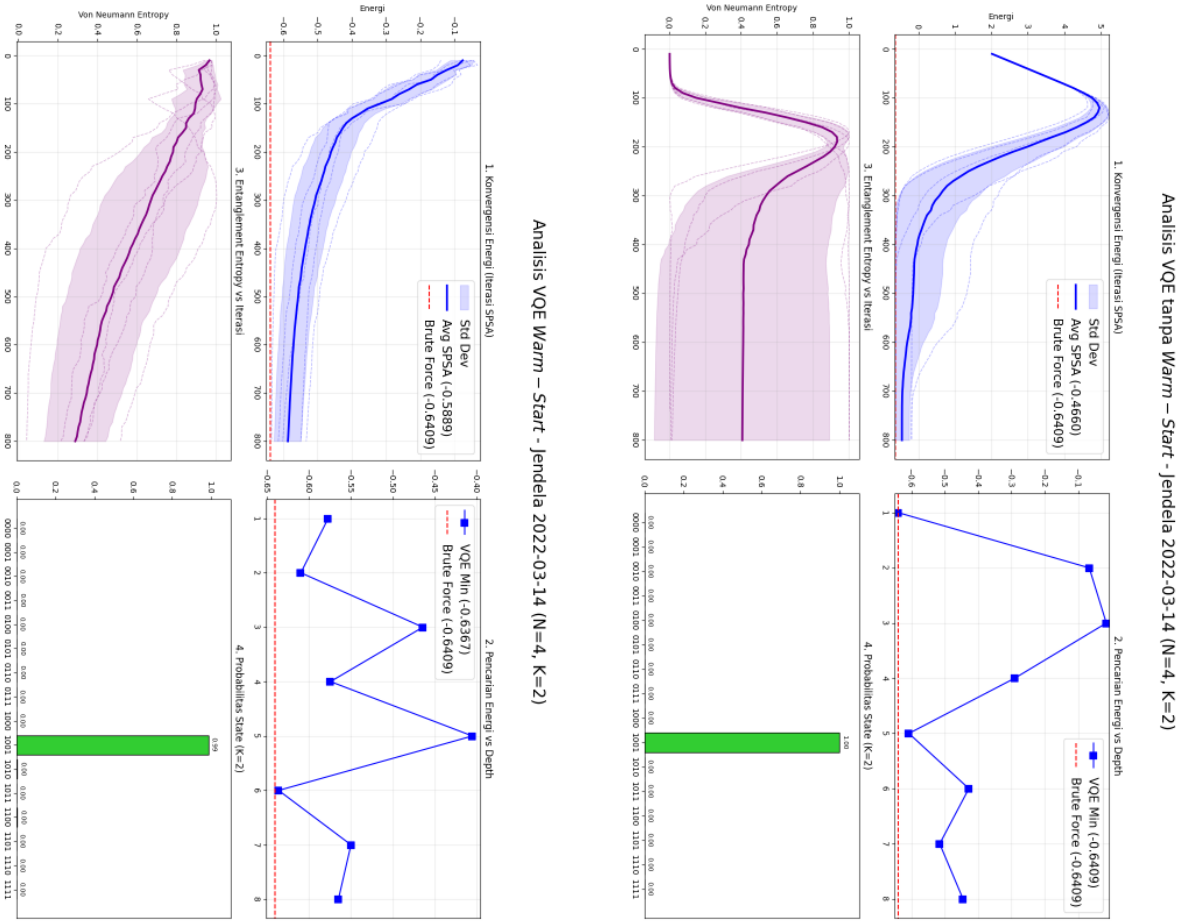


(a) Dengan *Game Theory*



(b) Tanpa *Game Theory*

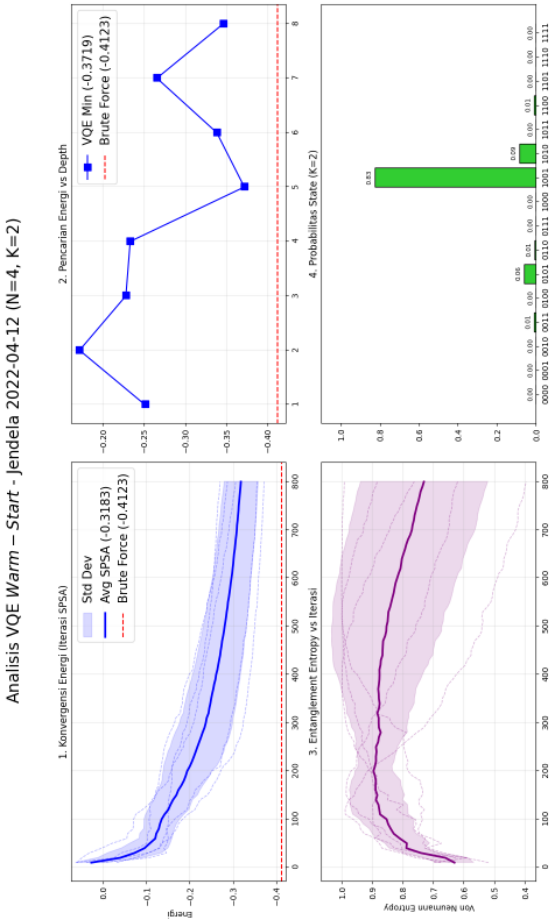
Gambar I.5: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-01-10



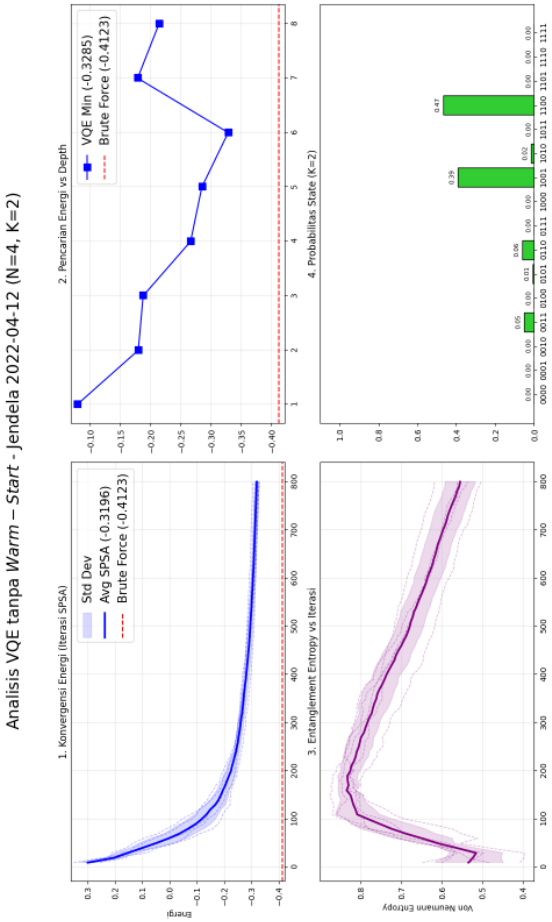
(a) Dengan *Game Theory*

(b) Tanpa *Game Theory*

Gambar I.6: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-03-14

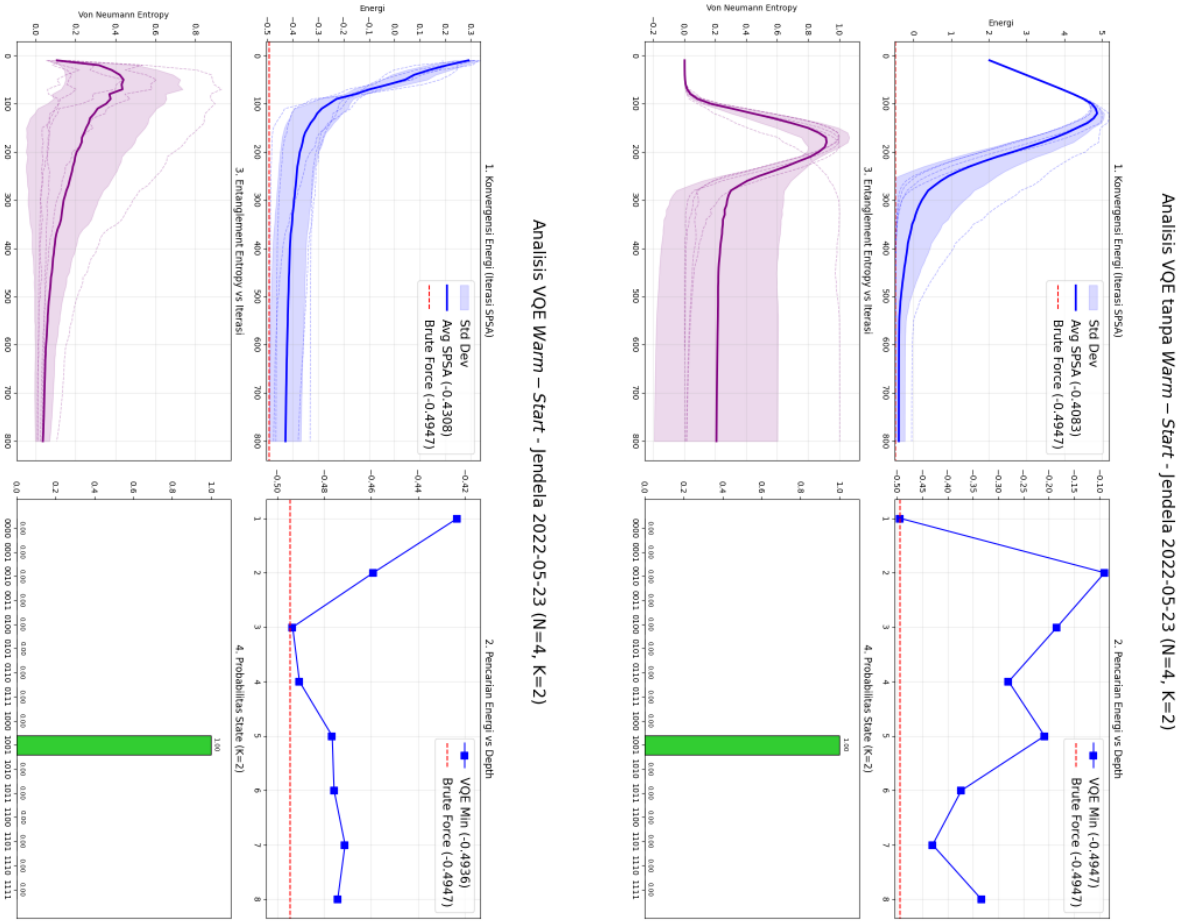


(a) Dengan *Game Theory*



(b) Tanpa *Game Theory*

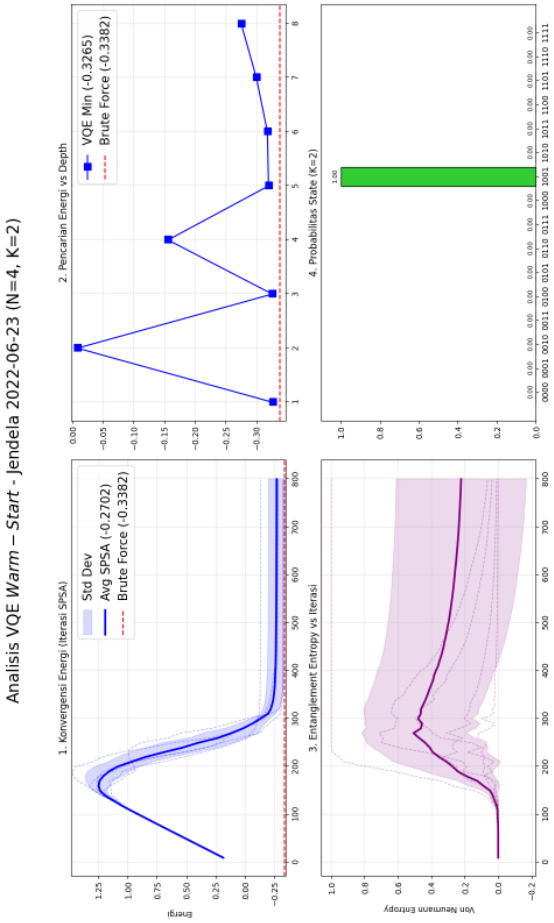
Gambar I.7: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-04-12



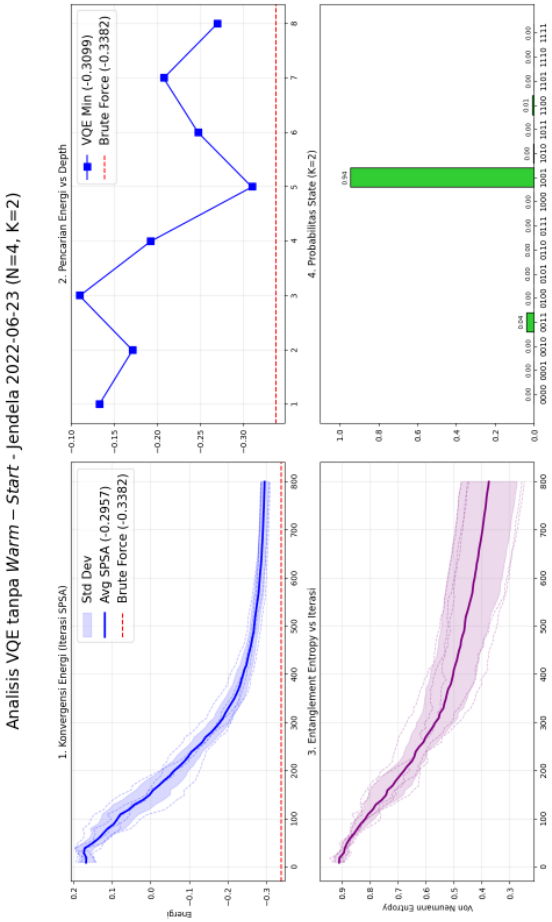
(a) Dengan *Game Theory*

(b) Tanpa *Game Theory*

Gambar I.8: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-05-23

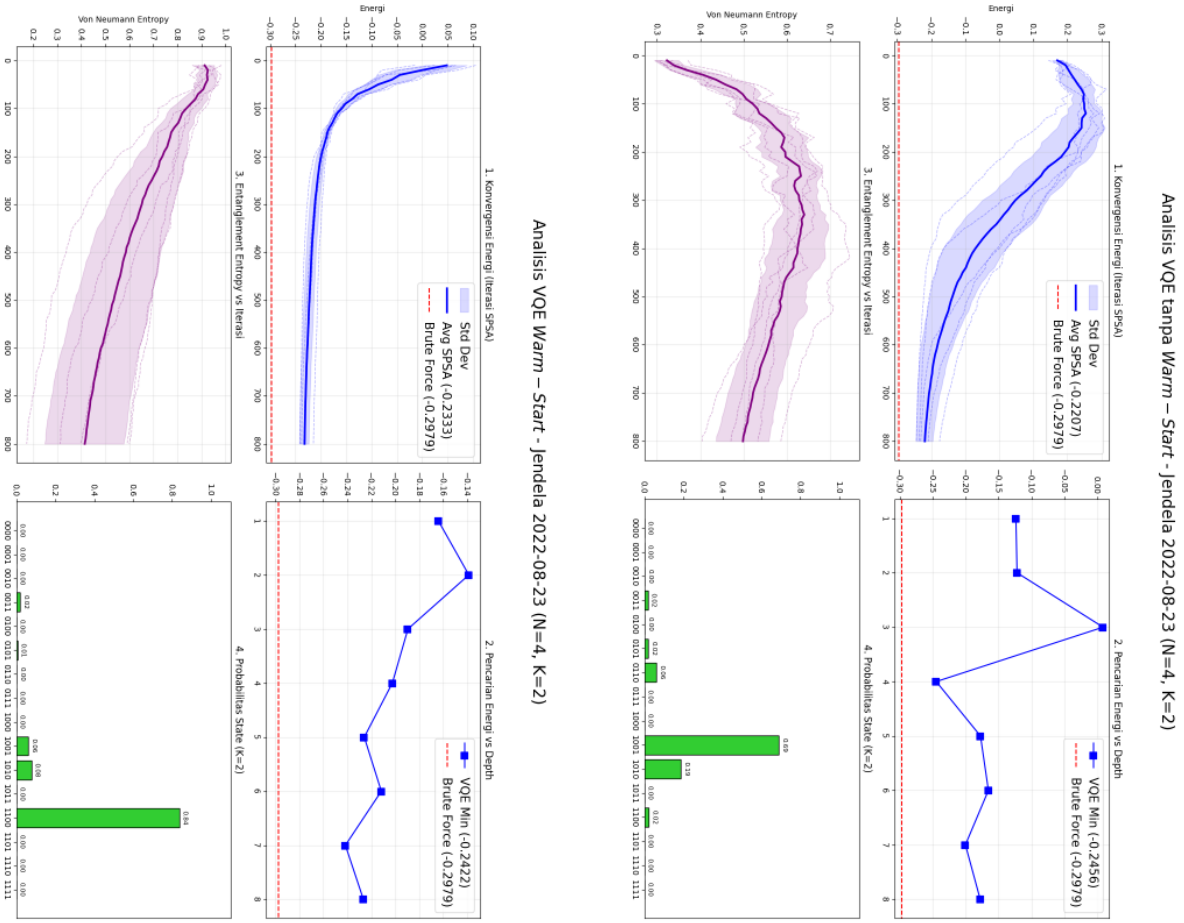


(a) Dengan *Game Theory*



(b) Tanpa *Game Theory*

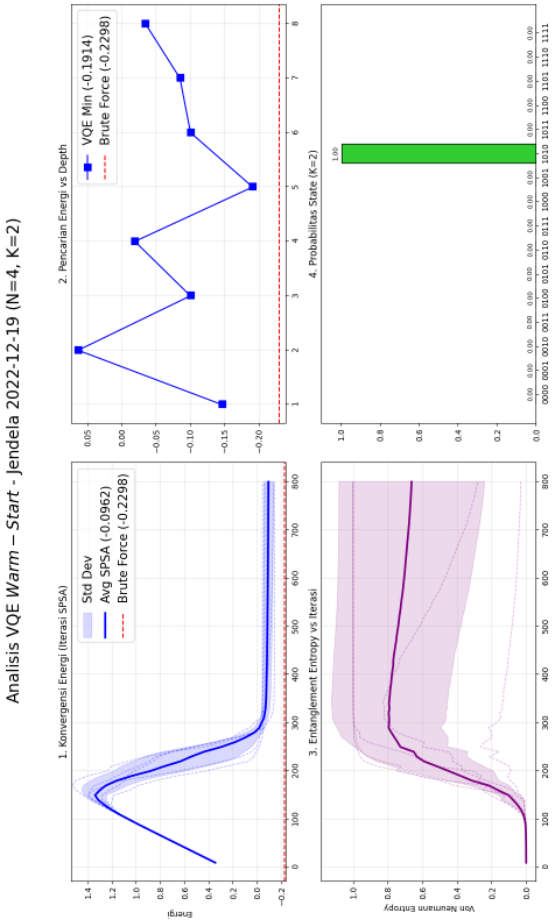
Gambar I.9: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-06-23



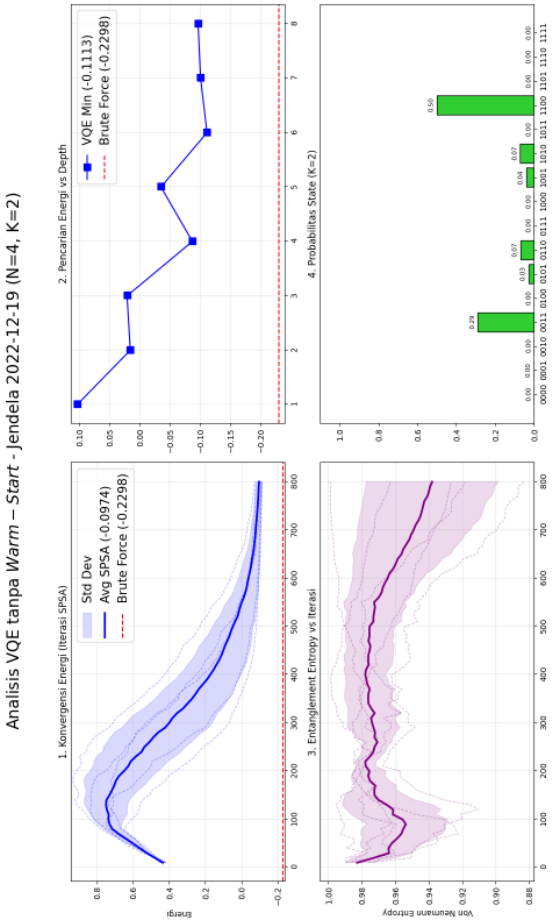
(a) Dengan *Game Theory*

(b) Tanpa *Game Theory*

Gambar I.10: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-08-23

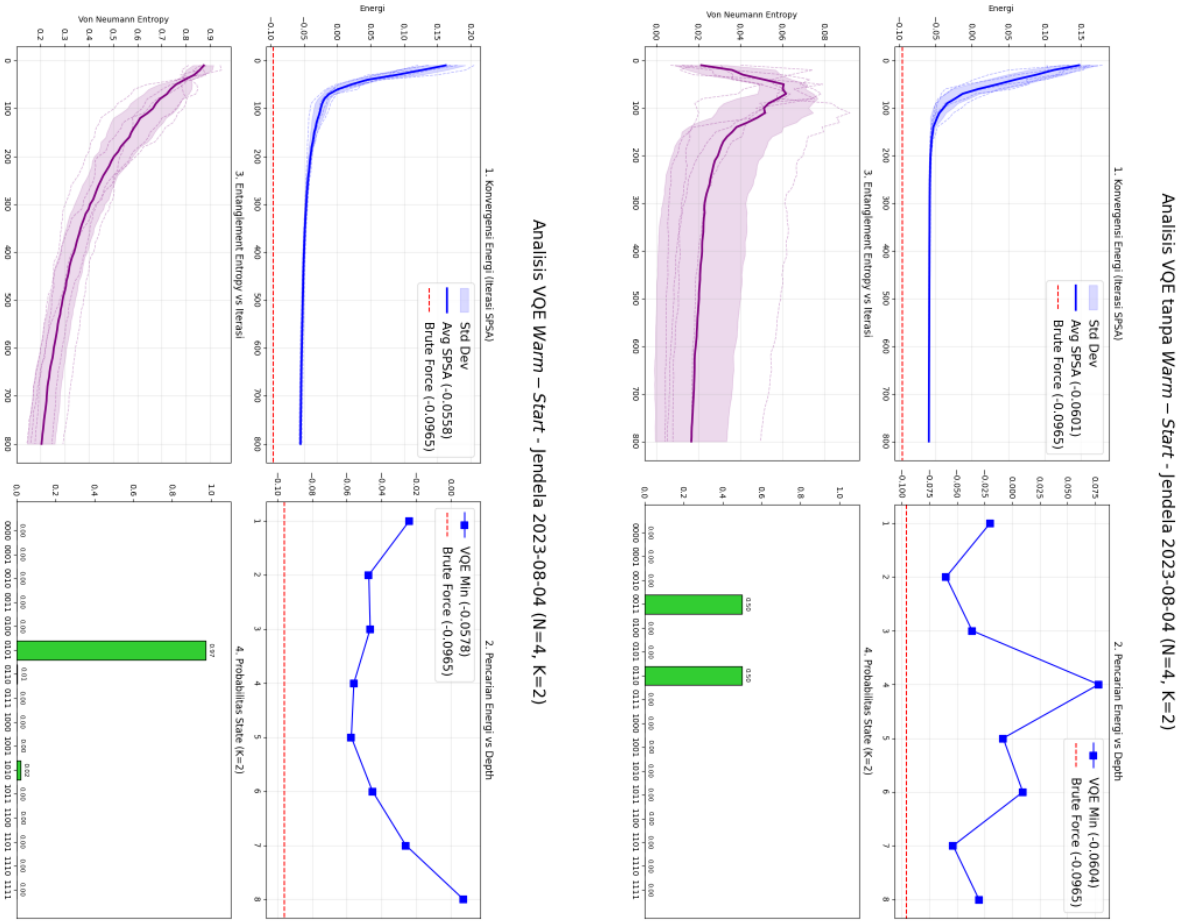


(a) Dengan *Game Theory*



(b) Tanpa *Game Theory*

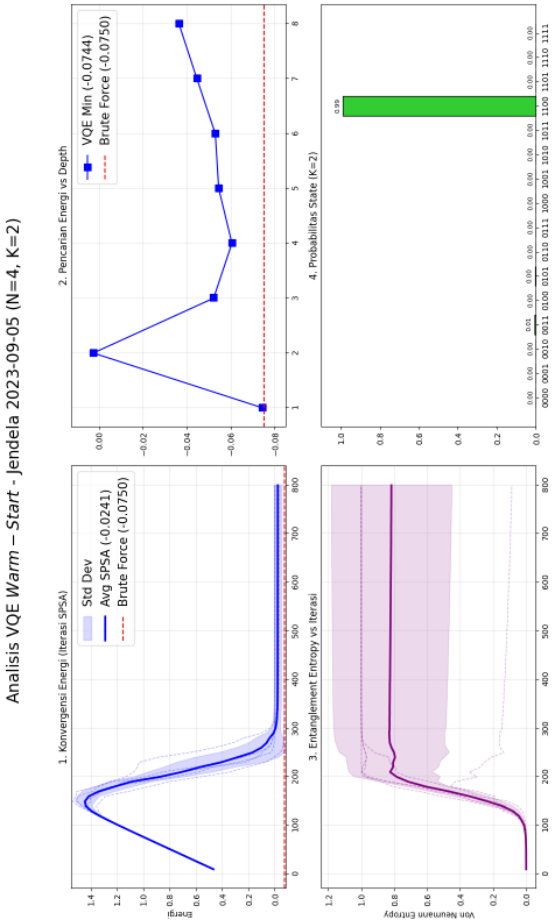
Gambar I.11: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2022-12-19



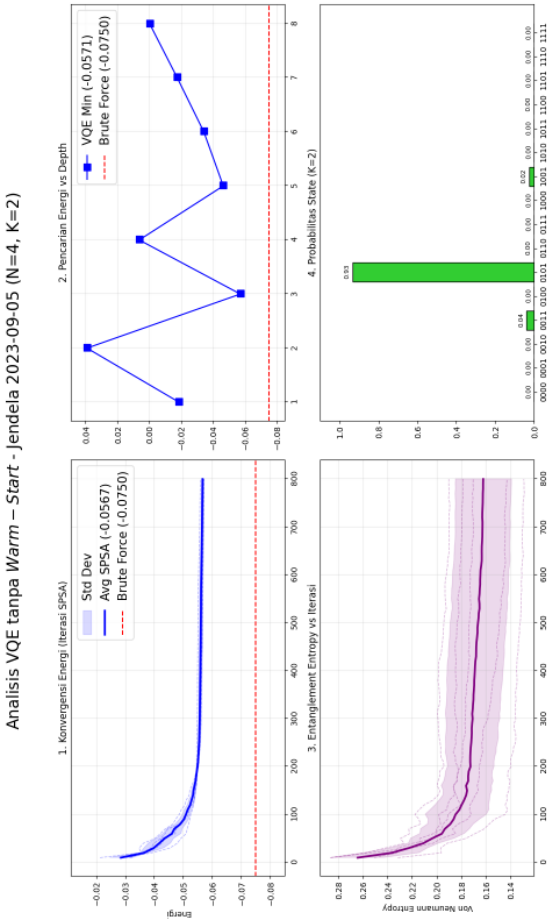
(a) Dengan *Game Theory*

(b) Tanpa *Game Theory*

Gambar I.12: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2023-08-04

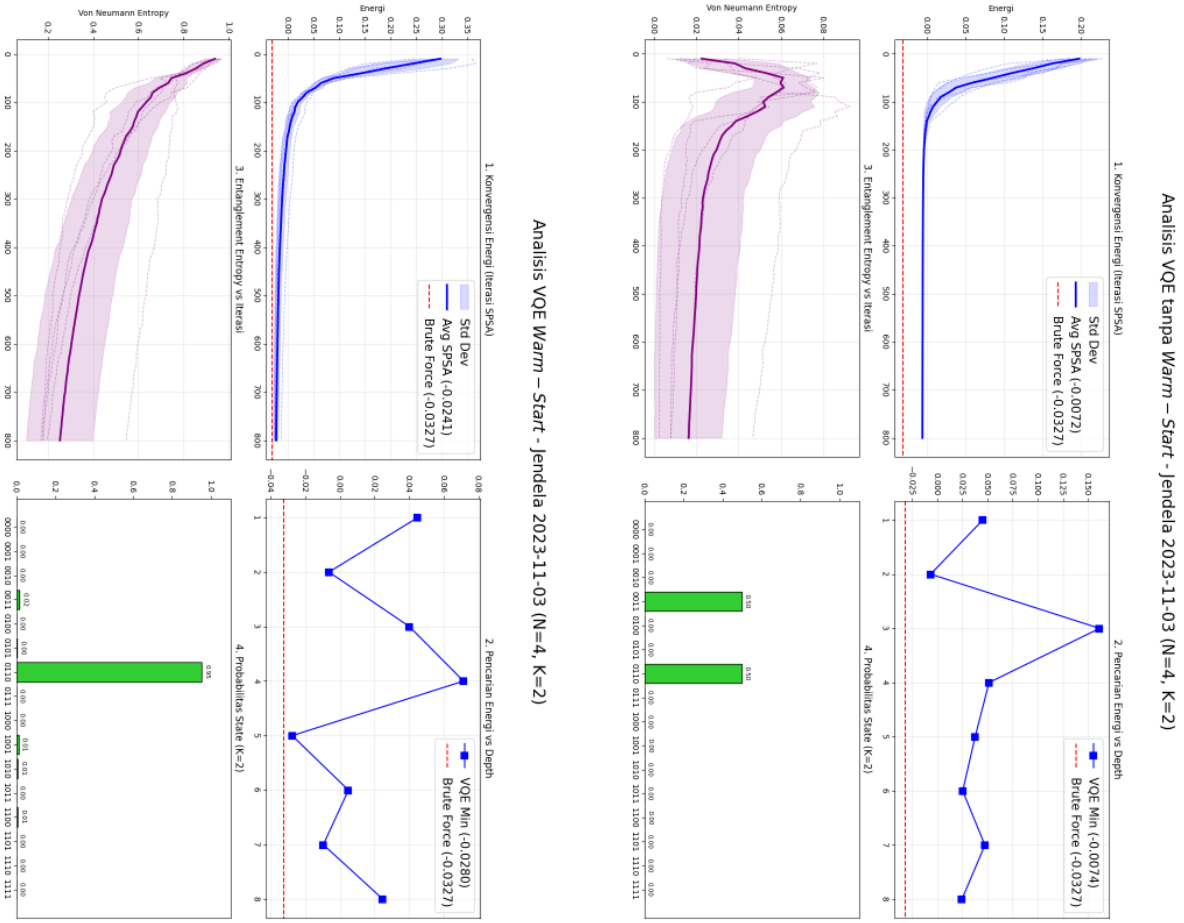


(a) Dengan *Game Theory*



(b) Tanpa *Game Theory*

Gambar I.13: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2023-09-05



(a) Dengan *Game Theory*

(b) Tanpa *Game Theory*

Gambar I.14: Perbandingan Alokasi Portofolio N=4 Jendela Waktu 2023-11-03

Lampiran J

Skrip Visualisasi dan Pembangkitan Grafik Hasil Simulasi

Pada bagian ini, dilampirkan kumpulan skrip pemrograman *Python* yang secara khusus dikembangkan untuk mengeksekusi operasi pembuatan visualisasi data kuantitatif dari hasil simulasi *Variational Quantum Eigensolver* (VQE) dan ekuilibrium teori permainan. Skrip-skrip ini berfungsi sebagai instrumen konversi dari representasi data komputasional mentah menjadi bentuk diagram grafis yang komprehensif, mencakup pembentukan kurva konvergensi energi historis, histogram sebaran probabilitas status kuantum, grafik perbandingan imbal hasil portofolio, hingga komposisi metrik rasio absolut Sharpe. Rangkaian rutinitas komputasional visual ini merupakan tulang punggung utama dari analisis observabel data hasil evaluasi performa model yang memfasilitasi pelacakan konvergensi lanskap energi portofolio lintas dimensi aset.

```
import matplotlib
matplotlib.use('Agg')  # Menggunakan backend non-interaktif
import matplotlib.pyplot as plt
import numpy as np
import pennylane as qml
import pandas as pd
import os

# Import Qiskit untuk perenderan rangkaian yang lebih baik
from qiskit import QuantumCircuit, QuantumRegister, ClassicalRegister

def plot_all(results, data_clean, value_benchmark_idx, config, filenames):
    """
    Menghasilkan semua grafik yang diperlukan untuk analisis.
    """
    tickers = config['tickers']
    N = len(tickers)
    benchmark_ticker = config['benchmark_ticker']

    # 1. Folder Khusus Analisis Jendela
    output_dir = f"Analisis_Window_N{N}"
    if not os.path.exists(output_dir):
        os.makedirs(output_dir)

    print(f"\n--- Menghasilkan Grafik Analisis Per Jendela di '{output_dir}' ---")
    for i, win_data in enumerate(results['window_analysis_history']):
        win_filename = os.path.join(output_dir, f"{win_data['date']}_window.png")
        plot_window_analysis(win_data, n_qubits=N, filename=win_filename)

        # Render sirkuit per jendela
        circ_filename = os.path.join(output_dir, f"{win_data['date']}_circuit.png")
        plot_quantum_circuit(N, win_data['best_depth'], circ_filename, params=win_data['
    best_params'])
```

```
    # 2. Update nama file rangkaian untuk depth terakhir (untuk folder Hasil_NX)
    final_depth = results['depths_history'][-1] if len(results['depths_history']) > 0 else 1
    suffix = f"_N{N}"
    filenames['circuit_png'] = f'rangkaian_kuantum_depth{final_depth}{suffix}.png'

    # Rangkaian Kuantum Window Terakhir dengan Parameter
    final_win = results['window_analysis_history'][-1]
    plot_quantum_circuit(N, final_depth, filenames['circuit_png'], params=final_win['
    best_params'])

    # 4. Hasil Backtest VQE vs Benchmarks
    # Cari data classic jika ada
    classic_file = f"classic_compare/Hasil_Classic_Compare/equity_history_classic_N{N}.csv"
    value_classic = None
    if os.path.exists(classic_file):
        try:
            df_classic = pd.read_csv(classic_file)
            value_classic = df_classic['Equity'].values
        except:
            pass

    plot_equity_growth(data_clean, results['value_vqe'], results['value_bench'],
                       value_benchmark_idx, benchmark_ticker, 'Quantum VQE',
                       'blue', filenames['backtest_vqe_png'], value_classic=value_classic)

    # 5. Hasil Backtest Nash vs Benchmarks
    plot_equity_growth(data_clean, results['value_nash'], results['value_bench'],
                       value_benchmark_idx, benchmark_ticker, 'Nash Equilibrium (Klasik)',
                       'orange', filenames['backtest_nash_png'], value_classic=value_classic)

    # 6. Depth per Window
    plot_depth_per_window(results['rebalance_dates'], results['depths_history'], filenames['
    depth_png'])

def plot_window_analysis(win_data, n_qubits, filename):
    """
    Merender 6 panel analisis untuk satu jendela waktu tertentu.
    """
    date = win_data['date']
    best_history = win_data['best_history']
    best_ent_hist = win_data['best_ent_hist']
    depth_energies = win_data['depth_energies']
    lr_data = win_data['lr_data']
    probs = win_data['winning_probs']
    K = win_data['K']
    use_warm_start = win_data.get('use_warm_start', True)

    # Hitung Brute Force khusus untuk jendela ini (dengan suffix N)
    bf_energy = calculate_bf_for_window(date, n_qubits)

    fig, axes = plt.subplots(3, 2, figsize=(16, 15))

    # Judul Dinamis berdasarkan Warm-Start
    ws_text = "Warm-Start" if use_warm_start else "tanpa Warm-Start"
    fig.suptitle(f"Analisis VQE {ws_text} - Jendela {date} (N={n_qubits}, K={K})", fontsize
    =20, fontweight='bold')
    axes = axes.flatten()

    # 1. Konvergensi Energi
    iters = np.arange(1, len(best_history) + 1) * 10 # batch_size=10
    final_e_spsa = best_history[-1]
    axes[0].plot(iters, best_history, marker='o', markersize=3, color='blue', label=f'SPSA
    Progress ({final_e_spsa:.4f})')
    if bf_energy is not None:
        axes[0].axhline(y=bf_energy, color='red', linestyle='--', label=f'Brute Force ({
    bf_energy:.4f})')
        all_e = list(best_history) + [bf_energy]
        axes[0].set_ylim(min(all_e)-0.01, max(all_e)+0.01)
    axes[0].set_title('1. Konvergensi Energi (Iterasi SPSA)')
    axes[0].set_ylabel('Energi')
    axes[0].legend()
    axes[0].grid(True, alpha=0.3)
```

```
    # 2. Energi vs Depth
    d_list = [d for d, e, it in depth_energies]
    e_list = [e for d, e, it in depth_energies]
    vqe_min_depth = min(e_list)
    axes[1].plot(d_list, e_list, marker='s', markersize=8, color='blue', label=f'VQE Min ({
    vqe_min_depth:.4f})')
    if bf_energy is not None:
        axes[1].axhline(y=bf_energy, color='red', linestyle='--', label=f'Brute Force ({
    bf_energy:.4f})')
        all_e = e_list + [bf_energy]
        axes[1].set_ylim(min(all_e)-0.01, max(all_e)+0.01)
    axes[1].set_title('2. Pencarian Energi vs Depth')
    axes[1].set_xticks(d_list)
    axes[1].legend()
    axes[1].grid(True, alpha=0.3)

    # 3. Entanglement Entropy
    if len(best_ent_hist) > 0:
        axes[2].plot(iters, best_ent_hist, color='purple', marker='^', markersize=3)
        axes[2].set_title('3. Entanglement Entropy vs Iterasi')
        axes[2].set_ylabel('Von Neumann Entropy')
        axes[2].grid(True, alpha=0.3)

    # 4. Learning Rate Finder
    test_a, test_e, best_a = lr_data
    axes[3].plot(test_a, test_e, marker='o', color='green')
    axes[3].axvline(best_a, color='red', linestyle='--', label=f'Best a: {best_a:.4f}')
    axes[3].set_xscale('log')
    axes[3].set_title('4. LR Finder (Calibration)')
    axes[3].legend()
    axes[3].grid(True, alpha=0.3)

    # 5. Distribusi Probabilitas (Bar Chart)
    bitstrings = [format(i, f'0{n_qubits}b') for i in range(2**n_qubits)]
    colors = ['limegreen' if b.count('1') == K else 'salmon' for b in bitstrings]
    axes[4].bar(bitstrings, probs, color=colors, edgecolor='black')
    axes[4].set_title(f'5. Probabilitas State (K={K})')
    axes[4].set_ylim(0, 1.1)
    # Tambahkan label teks di atas batang
    for idx, p in enumerate(probs):
        axes[4].text(idx, p + 0.02, f"{p:.2f}", ha='center', fontsize=8)

    # 6. Variansi Gradien (Diagnostic untuk Barren Plateaus)
    best_gvar_hist = win_data.get('best_gvar_hist', [])
    if len(best_gvar_hist) > 0:
        axes[5].plot(iters, best_gvar_hist, color='orange', marker='x', markersize=3)
        axes[5].set_yscale('log') # Gunakan skala log karena variansi bisa sangat kecil
        axes[5].set_title('6. Variansi Gradien vs Iterasi (Log)')
        axes[5].set_ylabel('Var(Grad)')
        axes[5].grid(True, alpha=0.3)
    else:
        axes[5].text(0.5, 0.5, 'Data Gradien Tidak Tersedia', ha='center')

    plt.tight_layout(rect=[0, 0.03, 1, 0.95])
    plt.savefig(filename)
    plt.close()

def plot_quantum_circuit(N, depth, filename, params=None, title=None):
    """
    Merender rangkaian kuantum menggunakan Qiskit untuk hasil yang lebih berwarna,
    informatif (label parameter), dan otomatis dibagi baris (folding).
    """
    qr = QuantumRegister(N, 'q')
    cr = ClassicalRegister(N, 'c')
    qc = QuantumCircuit(qr, cr)

    if params is None:
        params = np.zeros(N * 2 * (depth + 1))

    # Reshape parameter sesuai struktur Ansatz (layer, qubit, gate_type)
    w = params.reshape((depth + 1, N, 2))
```

```
    for layer in range(depth + 1):
        for q_idx in range(N):
            # Ambil nilai theta dan bulatkan agar tidak terlalu panjang di gambar
            theta_ry = np.round(float(w[layer, q_idx, 0]), 2)
            theta_rz = np.round(float(w[layer, q_idx, 1]), 2)

            # Tambahkan gate RY dan RZ dengan label parameter numerik
            qc.ry(theta_ry, qr[q_idx])
            qc.rz(theta_rz, qr[q_idx])

        if layer < depth:
            # Tambahkan Barrier antar layer untuk kejelasan visual
            qc.barrier()
            # Chain CNOT (Entanglement)
            for q_idx in range(N - 1):
                qc.cx(qr[q_idx], qr[q_idx + 1])
            qc.cx(qr[N - 1], qr[0])
            qc.barrier()

    # Tambahkan Pengukuran di akhir
    qc.measure(qr, cr)

    # Pengaturan Visual Qiskit
    # style='iqp' memberikan skema warna modern/IBM standard
    # fold=20 disetel agar muat sekitar 5 depth per baris (mencegah terlalu panjang ke samping
    )
    fig = qc.draw(output='mpl', style='iqp', plot_barriers=True, justify='left', fold=20)

    if title:
        fig.suptitle(title, fontsize=16, fontweight='bold', y=1.05)
    else:
        fig.suptitle(f"VQE Optimized Circuit (N={N}, Depth={depth})", fontsize=16, fontweight=
    'bold', y=1.05)

    fig.savefig(filename, bbox_inches='tight', dpi=150)
    plt.close(fig)
    print(f"Gambar rangkaian kuantum (Qiskit) disimpan sebagai '{filename}'.")

def calculate_bf_for_window(date, n_qubits):
    """Fungsi pembantu untuk hitung BF on-the-fly dengan Penalty Annealing Target."""
    suffix = f"_N{n_qubits}"
    try:
        # 1. Ambil Bias (h)
        h_df = pd.read_csv(f'bias_h_total{suffix}.csv')
        h_win = h_df[h_df['Date'] == str(date)]
        h_obj = h_win['Bias_h_Obj'].values
        h_pen = h_win['Bias_h_Pen'].values
        n = len(h_obj)

        # 2. Ambil Interaksi (J)
        J_df = pd.read_csv(f'interaksi_J_total{suffix}.csv')
        J_win = J_df[J_df['Date'] == str(date)]
        J_mat_obj = np.zeros((n, n))
        J_mat_pen = np.zeros((n, n))
        tickers = h_win['Ticker'].values
        t2i = {t: i for i, t in enumerate(tickers)}
        for _, r in J_win.iterrows():
            i, j = t2i[r['Ticker_i']], t2i[r['Ticker_j']]
            J_mat_obj[i,j] = J_mat_obj[j,i] = r['Interaction_J_Obj']
            J_mat_pen[i,j] = J_mat_pen[j,i] = r['Interaction_J_Pen']

        # 3. Ambil Konstanta C dan Target Penalty
        c_df = pd.read_csv(f'parameter_pendamping{suffix}.csv')
        c_win = c_df[c_df['Date'] == str(date)].iloc[0]
        c_obj = c_win['C_Obj']
        c_pen = c_win['C_Pen']
        penalty_A = c_win['Penalty_A_Target']

        # 4. Kombinasikan menjadi Hamiltonian Total Akhir
        h_total = h_obj + penalty_A * h_pen
        J_total = J_mat_obj + penalty_A * J_mat_pen
```

```
        c_total = c_obj + penalty_A * c_pen

        from itertools import product
        min_e = float('inf')
        for b in product([0, 1], repeat=n):
            Z = 1 - 2 * np.array(b)
            # E = sum(h_i * Z_i) + sum(J_ij * Z_i * Z_j) + C
            e = np.sum(h_total * Z) + sum(J_total[i,j]*Z[i]*Z[j] for i in range(n) for j in
    range(i+1, n)) + c_total
            if e < min_e: min_e = e
        return min_e
    except:
        return None

    plt.tight_layout()
    plt.savefig(filename)
    print(f"Grafik konvergensi detail disimpan sebagai '{filename}'.")

def plot_equity_growth(data_clean, value_strategy, value_bench, value_benchmark_idx,
    benchmark_ticker, strategy_label, color, filename, value_classic=None):
    plt.figure(figsize=(12, 6))
    plt.plot(data_clean.index[:len(value_strategy)], value_strategy, label=strategy_label,
    linewidth=2.5, color=color)
    plt.plot(data_clean.index[:len(value_bench)], value_bench, label='Benchmark (Equal Weight)
    ', linestyle=':', color='black', alpha=0.6)
    plt.plot(value_benchmark_idx.index, value_benchmark_idx.values, label=f'Indeks ({
    benchmark_ticker})', linestyle='-.', color='magenta', linewidth=2)

    if value_classic is not None:
        plt.plot(data_clean.index[:len(value_classic)], value_classic, label='Classic
    Markowitz (Mean-Var)', linestyle='--', color='green', linewidth=2)

    plt.title(f'Perbandingan Pertumbuhan Ekuitas: {strategy_label} vs Benchmarks')
    plt.ylabel('Total Ekuitas (Rupiah)')
    plt.xlabel('Tanggal')
    plt.legend(loc='upper left')
    plt.grid(True, linestyle=':', alpha=0.7)
    plt.tight_layout()
    plt.savefig(filename)
    print(f"Grafik {strategy_label} disimpan sebagai '{filename}'.")

def plot_depth_per_window(rebalance_dates, depths_history, filename):
    plt.figure(figsize=(10, 5))
    avg_depth_floor = int(np.floor(np.mean(depths_history)))
    plt.plot(rebalance_dates, depths_history, marker='o', linestyle='-', color='purple', label
    ='Depth Terpilih')
    plt.axhline(y=avg_depth_floor, color='red', linestyle='--', label=f'Rata-rata Depth (Floor
    ): {avg_depth_floor}')
    plt.title('Depth Terpilih vs Rebalance Window')
    plt.ylabel('Depth')
    plt.xlabel('Tanggal Rebalance')
    plt.xticks(rotation=45)
    plt.legend()
    plt.grid(True, linestyle=':', alpha=0.7)
    plt.tight_layout()
    plt.savefig(filename)
    print(f"Grafik depth per window disimpan sebagai '{filename}'. Rata-rata (floor): {
    avg_depth_floor}")
```

Kode J.1: Skrip utama generator grafik evaluasi portofolio (`plot_generator.py`)

```
import matplotlib
matplotlib.use('Agg')
import matplotlib.pyplot as plt
import numpy as np
import pennylane as qml
import pandas as pd
import os

def render_vqe_probability_bar(n_qubits, K):
    # 1. Ambil Parameter Terbaik (Theta) dari hasil run sebelumnya
    # Kita asumsikan file 'theta_final_all_depths.csv' ada
```

```
if not os.path.exists('theta_final_all_depths.csv'):
    print("Error: File theta_final_all_depths.csv tidak ditemukan. Jalankan simulasi
terlebih dahulu.")
    return

df_theta = pd.read_csv('theta_final_all_depths.csv')

# Ambil depth terakhir (max depth) yang tercatat
max_depth_in_file = df_theta['Depth'].max()
params = df_theta[df_theta['Depth'] == max_depth_in_file]['Theta_Value'].values

print(f"Merender Probabilitas untuk Depth: {max_depth_in_file}")

# 2. Definisikan Sirkuit Kuantum untuk mendapatkan Probabilitas
dev = qml.device("default.qubit", wires=n_qubits)

@qml.qnode(dev)
def prob_circuit(p):
    depth = max_depth_in_file
    w = p.reshape((depth + 1, n_qubits, 2))
    for layer in range(depth + 1):
        for q in range(n_qubits):
            qml.RY(w[layer, q, 0], wires=q)
            qml.RZ(w[layer, q, 1], wires=q)
        if layer < depth:
            for q in range(n_qubits - 1):
                qml.CNOT(wires=[q, q + 1])
            qml.CNOT(wires=[n_qubits - 1, 0])
    return qml.probs(wires=range(n_qubits))

# Hitung Probabilitas
probs = prob_circuit(params)
bitstrings = [format(i, f'0{n_qubits}b') for i in range(2**n_qubits)]

# 3. Visualisasi Grafik Batang
plt.figure(figsize=(10, 6))

# Tentukan warna: Hijau untuk yang valid (sum=K), Merah untuk yang melanggar
colors = []
for bs in bitstrings:
    if bs.count('1') == K:
        colors.append('limegreen')
    else:
        colors.append('salmon')

bars = plt.bar(bitstrings, probs, color=colors, edgecolor='black', alpha=0.8)

# Tambahkan Label Probabilitas di atas batang
for bar in bars:
    yval = bar.get_height()
    plt.text(bar.get_x() + bar.get_width()/2, yval + 0.01, f'{yval:.3f}', ha='center', va=
'bottom', fontsize=10)

# Tambahkan keterangan/legend manual
from matplotlib.lines import Line2D
legend_elements = [
    Line2D([0], [0], color='limegreen', lw=4, label=f'Valid (K={K})'),
    Line2D([0], [0], color='salmon', lw=4, label='Invalid (Penalty Applied)')
]
plt.legend(handles=legend_elements, loc='upper right')

plt.title(f"Distribusi Probabilitas State VQE (N={n_qubits}, K={K})", fontsize=14)
plt.ylabel("Probabilitas", fontsize=12)
plt.xlabel("Bitstring (Spin Interpretation)", fontsize=12)
plt.ylim(0, 1.1) # Beri ruang untuk label angka
plt.grid(axis='y', linestyle='--', alpha=0.6)

# Tandai Pemenang
winner_idx = np.argmax(probs)
plt.annotate('Kandidat Terpilih\n(Energi Terendah)',
             xy=(winner_idx, probs[winner_idx]),
             xytext=(winner_idx, probs[winner_idx] + 0.15),
```

```
                arrowprops=dict(facecolor='black', shrink=0.05),
                ha='center', fontweight='bold')

    filename = "visualisasi_probabilitas_vqe.png"
    plt.savefig(filename, dpi=300)
    plt.close()
    print(f"Grafik probabilitas berhasil disimpan sebagai: {filename}")

if __name__ == "__main__":
    # Gunakan parameter yang sesuai dengan run terakhir Anda
    render_vqe_probability_bar(n_qubits=2, K=1)
```

Kode J.2: Skrip visualisasi distribusi probabilitas status VQE (`render_vqe_probs.py`)

```
import marimo

__generated_with = "0.23.5"
app = marimo.App()


@app.cell
def _():
    import marimo as mo
    import os
    import re

    return mo, os, re


@app.cell
def _(mo, os, re):
    # Fungsi untuk mencari file apa saja yang di-import oleh sebuah file
    def scan_dependencies(file_path, project_files):
        found_deps = set()
        try:
            with open(file_path, 'r', encoding='utf-8') as f:
                content = f.read()
                # Mencari pola 'import nama_file' atau 'from nama_file import ...'
                for other_file in project_files:
                    module_name = other_file.replace('.py', '')
                    # Regex untuk memastikan kita tidak salah mencocokkan kata di dalam string
                    pattern = rf'\b(import|from)\s+{module_name}\b'
                    if re.search(pattern, content):
                        found_deps.add(other_file)
        except:
            pass
        return found_deps

    def draw_project_map():
        root = os.getcwd()
        # Filter: Hanya ambil file .py, abaikan project_graph.py dan main_N...py
        all_py_files = [
            f for f in os.listdir(root)
            if f.endswith('.py')
            and f != 'project_graph.py'
            and not re.match(r'main_N\d+\.py', f)
        ]

        mermaid_lines = ["graph TD"]

        # Tambahkan gaya visual yang lebih kaya
        mermaid_lines.append("    classDef core fill:#FF9800,stroke:#E65100,stroke-width:2px,
    color:white;")
        mermaid_lines.append("    classDef strategy fill:#2196F3,stroke:#0D47A1,stroke-width:2
    px,color:white;")
        mermaid_lines.append("    classDef engine fill:#4CAF50,stroke:#1B5E20,stroke-width:2px
    ,color:white;")
        mermaid_lines.append("    classDef math fill:#9C27B0,stroke:#4A148C,stroke-width:1px,
    color:white;")

        # Kategorisasi File
        categories = {
```

```
            'core': ['main.py', 'config.py', 'backtest_runner.py', 'report_generator.py', '
    plot_generator.py', 'data_downloader.py'],
            'strategy': ['run_strategy_step.py', 'rebalance_portfolio.py'],
            'engine': ['run_vqe_adaptive.py', 'find_nash_sbr.py', 'brute_force_validator.py',
    'find_optimal_lr_spsa.py', 'run_spsa_test.py'],
            'math': [f for f in all_py_files if f.startswith(('calc_', 'compute_', 'build_'))]
        }

        def get_style(filename):
            for cat, files in categories.items():
                if filename in files:
                    return f":::{cat}"
            return ""

        has_connections = False
        seen_edges = set()

        for file in all_py_files:
            deps = scan_dependencies(os.path.join(root, file), all_py_files)

            # Styling node berdasarkan kategori
            style = get_style(file)
            mermaid_lines.append(f'    {file.replace(".", "_")}["{file}"]{style}')

            for dep in deps:
                if file != dep:
                    src = file.replace(".", "_")
                    dst = dep.replace(".", "_")
                    edge = (src, dst)
                    if edge not in seen_edges:
                        mermaid_lines.append(f"    {src} --> {dst}")
                        seen_edges.add(edge)
                        has_connections = True
        if not has_connections:
            return mo.md("###  Tidak ditemukan hubungan import antar file.\nPastikan file Anda
     menggunakan `import nama_file_lain`.")

        return mo.vstack([
            mo.md("#  Peta Hubungan Kode Proyekmu"),
            mo.md("Garis panah menunjukkan file mana yang 'memanggil' file lainnya."),
            mo.mermaid("\n".join(mermaid_lines))
        ])

    draw_project_map()
    return


if __name__ == "__main__":
    app.run()
```

Kode J.3: Skrip visualisasi grafik interaksi antar aset (`project_graph.py`)

```
import os
import re
import base64
import requests # Pastikan library requests tersedia

def scan_dependencies(file_path, project_files):
    found_deps = set()
    try:
        with open(file_path, 'r', encoding='utf-8') as f:
            content = f.read()
            for other_file in project_files:
                module_name = other_file.replace('.py', '')
                pattern = rf'\b(import|from)\s+{module_name}\b'
                if re.search(pattern, content):
                    found_deps.add(other_file)
    except:
        pass
    return found_deps

def generate_mermaid():
```

```
    root = os.getcwd()
    all_py_files = [
        f for f in os.listdir(root)
        if f.endswith('.py')
        and f not in ['project_graph.py', 'export_graph.py']
        and not re.match(r'main_N\d+\.py', f)
    ]

    mermaid_lines = ["graph TD"]
    mermaid_lines.append("    classDef core fill:#FF9800,stroke:#E65100,stroke-width:2px,color
    :white;")
    mermaid_lines.append("    classDef strategy fill:#2196F3,stroke:#0D47A1,stroke-width:2px,
    color:white;")
    mermaid_lines.append("    classDef engine fill:#4CAF50,stroke:#1B5E20,stroke-width:2px,
    color:white;")
    mermaid_lines.append("    classDef math fill:#9C27B0,stroke:#4A148C,stroke-width:1px,color
    :white;")

    categories = {
        'core': ['main.py', 'config.py', 'backtest_runner.py', 'report_generator.py', '
    plot_generator.py', 'data_downloader.py'],
        'strategy': ['run_strategy_step.py', 'rebalance_portfolio.py'],
        'engine': ['run_vqe_adaptive.py', 'find_nash_sbr.py', 'brute_force_validator.py', '
    find_optimal_lr_spsa.py', 'run_spsa_test.py'],
        'math': [f for f in all_py_files if f.startswith(('calc_', 'compute_', 'build_'))]
    }

    def get_style(filename):
        for cat, files in categories.items():
            if filename in files:
                return f":::{cat}"
        return ""

    seen_edges = set()
    for file in all_py_files:
        style = get_style(file)
        mermaid_lines.append(f'    {file.replace(".", "_")}["{file}"]{style}')
        deps = scan_dependencies(os.path.join(root, file), all_py_files)
        for dep in deps:
            if file != dep:
                src = file.replace(".", "_")
                dst = dep.replace(".", "_")
                if (src, dst) not in seen_edges:
                    mermaid_lines.append(f"    {src} --> {dst}")
                    seen_edges.add((src, dst))

    return "\n".join(mermaid_lines)

def export_png(mermaid_text, output_file="project_graph.png"):
    print("Mengonversi Mermaid ke PNG via mermaid.ink...")
    # Encode ke base64
    graphbytes = mermaid_text.encode("ascii")
    base64_bytes = base64.b64encode(graphbytes)
    base64_string = base64_bytes.decode("ascii")

    url = "https://mermaid.ink/img/" + base64_string

    try:
        response = requests.get(url)
        if response.status_code == 200:
            with open(output_file, "wb") as f:
                f.write(response.content)
            print(f"Berhasil! Gambar disimpan sebagai: {output_file}")
        else:
            print(f"Gagal mengunduh gambar. Status code: {response.status_code}")
    except Exception as e:
        print(f"Terjadi kesalahan: {e}")

if __name__ == "__main__":
    mermaid_str = generate_mermaid()
```

```
export_png(mermaid_str)
```

Kode J.4: Skrip utilitas untuk pengeksporan hasil graf fisis (`export_graph.py`)

Lampiran K

Skrip Utama Orkestrasi dan Algoritma Hibrida Kuantum

Pada bagian ini, dilampirkan kumpulan skrip pemrograman utama yang mengatur alur logika eksekusi algoritma GT-VQE secara terintegrasi.

```
import warnings
import pandas as pd
from data_downloader import download_market_data
from backtest_runner import run_backtest
from report_generator import generate_report
from plot_generator import plot_all
from config import get_config, clean_workspace

warnings.filterwarnings('ignore')

tickers = ['BBCA.JK', 'TLKM.JK', 'SMGR.JK', 'ADRO.JK']
N = len(tickers)

# 0. Bersihkan Workspace
clean_workspace(N)

config = get_config(tickers)

suffix = f"_N{N}"
filenames = {
    'daily_prices_csv': f'harga_harian_saham{suffix}.csv',
    'report_txt': f'laporan_backtest{suffix}.txt',
    'circuit_png': f'rangkaian_kuantum{suffix}.png',
    'convergence_png': f'grafik_konvergensi_detail{suffix}.png',
    'backtest_vqe_png': f'hasil_backtest_vqe{suffix}.png',
    'backtest_nash_png': f'hasil_backtest_nash{suffix}.png',
    'depth_png': f'grafik_depth_per_window{suffix}.png'
}

data_clean, benchmark_data, benchmark_rets = download_market_data(
    config['tickers'], config['benchmark_ticker'],
    config['start_date'], config['end_date'], config['start_bt_date']
)
data_clean.to_csv(filenames['daily_prices_csv'])
print(f"Data harga harian telah diekspor ke '{filenames['daily_prices_csv']}'.")

results = run_backtest(data_clean, config['tickers'], config)
value_benchmark_idx = generate_report(results, data_clean, benchmark_data, benchmark_rets,
    config['tickers'], config, filenames['report_txt'])
plot_all(results, data_clean, value_benchmark_idx, config, filenames)
print(f"\nSeluruh proses N={N} selesai.")
```

Kode K.1: Skrip utama orkestrasi keseluruhan parameter dan eksekusi pada sistem N=4 (`main_N4.py`)

```python
import numpy as np
import pandas as pd
import os
import pennylane as qml
from compute_metrics import calculate_entanglement_entropy

def run_vqe_adaptive(H_obj, H_pen, n_qubits, curr_date, ne_bitstring=None, K=2, target_penalty
    =5.0,
                     max_depth=4, maxiter=100, max_total_iter=500,
                     batch_size=10, conv_window=4, conv_tol=1e-3,
                     best_a_base=0.1, seed=42, file_config=None):
    """
    Menjalankan VQE dengan Penalty Annealing.
    H_total(t) = H_obj + penalty(t) * H_pen
    """
    if file_config is None:
        file_config = {
            'riwayat_iterasi': 'riwayat_iterasi_vqe.csv',
            'theta_final': 'theta_final_all_depths.csv',
            'depth_vs_energi': 'hasil_depth_vs_energi.csv'
        }

    dev = qml.device("default.qubit", wires=n_qubits)
    rng = np.random.default_rng(seed)

    # --- RESET RIWAYAT ITERASI ---
    if os.path.exists(file_config['riwayat_iterasi']):
        os.remove(file_config['riwayat_iterasi'])

    def make_circuit(depth):
        def Ansatz(params):
            w = params.reshape((depth + 1, n_qubits, 2))
            for layer in range(depth + 1):
                for q in range(n_qubits):
                    qml.RY(w[layer, q, 0], wires=q)
                    qml.RZ(w[layer, q, 1], wires=q)
                if layer < depth:
                    for q in range(n_qubits - 1):
                        qml.CNOT(wires=[q, q + 1])
                    qml.CNOT(wires=[n_qubits - 1, 0])

        @qml.qnode(dev)
        def cost_circuit(params, p_scale=1.0):
            Ansatz(params)
            # H_total dinamis: H_obj + scale * H_pen
            return qml.expval(H_obj + p_scale * H_pen)

        @qml.qnode(dev)
        def prob_circuit(params):
            Ansatz(params)
            return qml.probs(wires=range(n_qubits))

        @qml.qnode(dev)
        def state_circuit(params):
            Ansatz(params)
            return qml.state()

        return cost_circuit, prob_circuit, state_circuit

    def run_spsa(cost_circuit, state_circuit, n_params, init_params=None, a_base=0.1, c_base
    =0.1, depth_label=1, target_p=5.0):
        a, c      = a_base, c_base
        A_coeff   = maxiter * 0.1
        alpha_exp = 0.602
        gamma_exp = 0.101

        if init_params is not None:
            params = init_params.copy()
        else:
            params = rng.uniform(0, 2 * np.pi, n_params)

        energy_history = []
```

```
    entropy_history = []
    grad_var_history = []
    total_iters     = 0

    while total_iters < max_total_iter:
        # --- PENALTY ANNEALING TERKALIBRASI ---
        # Pinalti mencapai 100% pada 70% total budget iterasi.
        # Sisa 30% digunakan untuk "fine-tuning" dengan pinalti konstan.
        anneal_progress = min(1.0, total_iters / (0.7 * max_total_iter))
        p_scale = target_p * (0.1 + 0.9 * anneal_progress)

        current_grad_vars = []
        for _ in range(batch_size):
            k     = total_iters
            a_k   = a / (A_coeff + k + 1) ** alpha_exp
            c_k   = c / (k + 1)           ** gamma_exp
            delta = 2 * rng.integers(0, 2, size=n_params) - 1

            cost_plus  = float(cost_circuit(params + c_k * delta, p_scale=p_scale))
            cost_minus = float(cost_circuit(params - c_k * delta, p_scale=p_scale))
            grad       = (cost_plus - cost_minus) / (2 * c_k * delta)

            # [Phase 3] Gradient Clipping
            grad       = np.clip(grad, -1.0, 1.0)

            params     = params - a_k * grad
            current_grad_vars.append(np.var(grad))
            total_iters += 1

        current_energy = float(cost_circuit(params, p_scale=p_scale))
        energy_history.append(current_energy)
        grad_var_history.append(np.mean(current_grad_vars))

        # Hitung Entropi
        try:
            current_psi = state_circuit(params)
            current_entropy = calculate_entanglement_entropy(current_psi)
            entropy_history.append(current_entropy)
        except:
            entropy_history.append(0.0)

        iter_df = pd.DataFrame({
            'Depth': [depth_label],
            'Iteration': [total_iters],
            'Energy': [current_energy],
            'Entropy': [entropy_history[-1]],
            'Grad_Var': [grad_var_history[-1]],
            'Penalty_Scale': [p_scale]
        })
        iter_df.to_csv(file_config['riwayat_iterasi'], mode='a', header=not os.path.exists
(file_config['riwayat_iterasi']), index=False)

        if total_iters >= maxiter and len(energy_history) >= conv_window:
            # [Phase 4] Entropy Protection & Annealing Protection
            # Jangan berhenti jika:
            # 1. Entropi masih tinggi (mencegah superposisi macet)
            # 2. Annealing pinalti belum mencapai 100%
            if (n_qubits <= 4 and entropy_history[-1] > 0.1) or (anneal_progress < 1.0):
                continue

            E_old, E_now = energy_history[-conv_window], energy_history[-1]
            if abs(E_old - E_now) / (abs(E_old) + 1e-12) < conv_tol:
                break
    return params, energy_history[-1], energy_history, entropy_history, grad_var_history,
total_iters

best_energy    = np.inf
best_params, best_depth, best_history, best_ent_hist, best_grad_var_hist = None, 1, [],
[], []
depth_energies = []
prev_params    = None
all_theta_data = []
```

```
for depth in range(1, max_depth + 1):
    n_params = n_qubits * 2 * (depth + 1)
    cost_fn, prob_fn, state_fn = make_circuit(depth)

    if prev_params is not None and len(prev_params) < n_params:
        # Warm-start yang diperbaiki: gunakan distribusi normal sangat kecil
        # agar layer baru mendekati identitas, menjaga stabilitas energi.
        old_w = prev_params.reshape((depth, n_qubits, 2))     # (depth, n, 2)
        new_layer = rng.normal(0, 1e-4, (1, n_qubits, 2))    # layer baru (dekat identitas
)
        # Sisipkan layer baru sebelum output layer terakhir
        new_w = np.concatenate([old_w[:-1], new_layer, old_w[-1:]], axis=0)  # (depth+1, n
, 2)

        # [Phase 2] Asymmetric Jitter: Pemutus simetri agar tidak terjebak superposisi
        asymmetric_jitter = rng.normal(0, 1e-4, n_params) * np.linspace(1, 1.5, n_params)
        init_p = new_w.flatten() + asymmetric_jitter
    else:
        if ne_bitstring is not None:
            init_w = np.zeros((depth + 1, n_qubits, 2))
            for q in range(n_qubits):
                init_w[0, q, 0] = np.pi if int(ne_bitstring[q]) == 1 else 0.0

            # [Phase 2] Asymmetric Jitter pada inisialisasi Nash
            asymmetric_jitter = rng.normal(0, 1e-3, n_params) * np.linspace(1, 1.5,
n_params)
            init_p = init_w.flatten() + asymmetric_jitter
        else:
            init_p = rng.uniform(0, 2 * np.pi, n_params)

    # Skala learning rate dinamis: semakin dalam sirkuit, langkah semakin kecil (presisi)
    current_a_base = best_a_base / np.sqrt(depth)

    params, energy, e_hist, ent_hist, g_var_hist, n_iters = run_spsa(
        cost_fn, state_fn, n_params, init_params=init_p,
        a_base=current_a_base, c_base=0.1/np.sqrt(depth),
        depth_label=depth, target_p=target_penalty
    )
    print(f"    [Depth {depth}] Konvergen dalam {n_iters} iterasi | E = {energy:.6f}")
    depth_energies.append((depth, energy, n_iters))
    prev_params = params

    # --- KUMPULKAN THETA (Poin 1) ---
    for idx, val in enumerate(params):
        all_theta_data.append({'Depth': depth, 'Theta_Index': idx, 'Theta_Value': val})

    if energy < best_energy:
        best_energy, best_params, best_depth, best_history, best_ent_hist,
best_grad_var_hist = energy, params, depth, e_hist, ent_hist, g_var_hist

# --- EKSPOR THETA GABUNGAN (Poin 1) ---
pd.DataFrame(all_theta_data).to_csv(file_config['theta_final'], index=False)

# --- EKSPOR DEPTH VS ENERGI (Poin 3) ---
depth_df = pd.DataFrame(depth_energies, columns=['Depth', 'Energy', 'Iterations'])
depth_df.insert(0, 'Date', curr_date.date())
depth_df.to_csv(file_config['depth_vs_energi'], mode='a', header=not os.path.exists(
file_config['depth_vs_energi']), index=False)

_, prob_fn, _ = make_circuit(best_depth)
probs = prob_fn(best_params)
sorted_indices = np.argsort(probs)[::-1]
best_bitstring = None
for idx in sorted_indices:
    bs = format(idx, f'0{n_qubits}b')
    if bs.count('1') == K:
        best_bitstring = bs
        break
if best_bitstring is None:
    top_k = np.argsort(probs)[-K:]
    bs_list = list('0' * n_qubits)
```

```
        for idx in top_k: bs_list[idx] = '1'
        best_bitstring = ''.join(bs_list)

    return [i for i, bit in enumerate(best_bitstring) if bit == '1'], best_depth, best_energy,
     best_history, best_ent_hist, best_grad_var_hist, depth_energies, probs, best_params
```

Kode K.2: Skrip inti loop VQE adaptif (`run_vqe_adaptive.py`)

```
import numpy as np
import pandas as pd
import os
from compute_endogenous_lambda import compute_endogenous_lambda
from find_nash_sbr import find_nash_sbr
from brute_force_validator import run_brute_force_window
from build_hamiltonian_total import build_hamiltonian_total
from find_optimal_lr_spsa import find_optimal_lr_spsa
from run_vqe_adaptive import run_vqe_adaptive
from calc_simple_return import calculate_simple_return
from calc_log_return import calculate_log_return
from calc_expected_returns import calculate_expected_simple_return,
    calculate_expected_log_return_with_drag
from calc_expected_returns import calculate_expected_simple_return,
    calculate_expected_log_return_with_drag

def run_strategy_step(lookback_data, tickers, curr_date, K=2, penalty_A=5.0,
                      max_depth=6, maxiter=100,
                      max_total_iter=500, batch_size=10,
                      conv_window=4, conv_tol=1e-3,
                      use_warm_start=True,
                      file_config=None):
    """
    Eksekusi satu periode pembelajaran dengan pencatatan tanggal.
    """
    if file_config is None:
        file_config = {
            'metrics': 'metrik_return_dan_lambda.csv',
            'bias_h': 'bias_h_total.csv',
            'interaksi_J': 'interaksi_J_total.csv',
            'pencarian_lr': 'hasil_pencarian_lr.csv',
            'parameter_pendamping': 'parameter_pendamping.csv'
        }

    n_assets = len(tickers)
    # ... rest of calculations ...
    # Simple Return dihitung manual via fungsi baru
    simple_rets = calculate_simple_return(lookback_data)
    # Log Return dihitung manual via fungsi baru
    log_rets  = calculate_log_return(lookback_data)

    binary_st = (log_rets <= 0).astype(int)   # 1 = turun, 0 = naik

    # --- PERHITUNGAN EXPECTED RETURN (Rumus Volatility Drag) ---
    mu_R_daily = calculate_expected_simple_return(simple_rets)
    # Variance log return sebagai dasar volatility drag
    var_r_daily = log_rets.var()
    mu_r_daily = calculate_expected_log_return_with_drag(mu_R_daily, var_r_daily)

    # Menggunakan return periode (126 hari)
    mu_simple_period = mu_R_daily.values * 126
    mu_log_period = mu_r_daily.values * 126

    sigma_log = log_rets.std().values
    sigma_period_log = sigma_log * np.sqrt(126)

    # Matriks kovariansi tetap berbasis log return
    sigma_period_matrix = log_rets.cov().values * 126

    # gamma disini mewakili degree of risk-aversion, menggunakan mu_log_period (dengan drag)
    dan sigma_period_log
    gamma = compute_endogenous_lambda(mu_log_period, sigma_period_log)
    lam = penalty_A # penalty lambda untuk pembatas kardinalitas
    metrics_df = pd.DataFrame({
```

```
    'Date': [curr_date.date()] * n_assets,
    'Ticker': tickers,
    'Mu_Simple_Period': mu_simple_period,
    'Mu_Log_Period': mu_log_period,
    'Sigma_Log': sigma_log,
    'Sigma_Period_Log': sigma_period_log,
    'Lambda_RiskAversion': [gamma] * n_assets
})

metrics_df.to_csv(file_config['metrics'], mode='a', header=not os.path.exists(file_config[
'metrics']), index=False)

# --- KONSTRUKSI MATRIKS Q (QUBO) TERPISAH ---
# 1. Komponen Objektif (Return & Risk)
Q_diag_obj = np.zeros(n_assets)
Q_off_obj  = np.zeros((n_assets, n_assets))
K_sq = K**2

for i in range(n_assets):
    Q_diag_obj[i] = (gamma * sigma_period_matrix[i, i]) / (2.0 * K_sq) - (mu_simple_period
[i] / K)
    for j in range(i + 1, n_assets):
        Q_val = (gamma * sigma_period_matrix[i, j]) / (2.0 * K_sq)
        Q_off_obj[i, j] = Q_val
        Q_off_obj[j, i] = Q_val

# 2. Komponen Pinalti (Constraint)
Q_diag_pen = np.zeros(n_assets)
Q_off_pen  = np.zeros((n_assets, n_assets))
for i in range(n_assets):
    Q_diag_pen[i] = (1.0 - 2.0 * K)
    for j in range(i + 1, n_assets):
        Q_off_pen[i, j] = 1.0
        Q_off_pen[j, i] = 1.0

# --- KONSTRUKSI PARAMETER ISING TERPISAH ---
# Fungsi pembantu untuk konversi Q ke h, J, C
def qubo_to_ising(Q_diag, Q_off, constant=0.0):
    h = np.zeros(n_assets)
    J = np.zeros((n_assets, n_assets))
    for i in range(n_assets):
        sum_Q_ij_half = 0.0
        for j in range(n_assets):
            if i != j:
                sum_Q_ij_half += Q_off[i, j] / 2.0
                if i < j:
                    J[i, j] = Q_off[i, j] / 2.0
                    J[j, i] = Q_off[i, j] / 2.0
        h[i] = -((Q_diag[i] / 2.0) + sum_Q_ij_half)

    sum_Q_ii_half = np.sum(Q_diag) / 2.0
    sum_Q_ij_half_total = 0.0
    for i in range(n_assets):
        for j in range(i + 1, n_assets):
            sum_Q_ij_half_total += Q_off[i, j] / 2.0
    C = sum_Q_ii_half + sum_Q_ij_half_total + constant
    return h, J, C

h_obj, J_obj, C_obj = qubo_to_ising(Q_diag_obj, Q_off_obj, 0.0)
h_pen, J_pen, C_pen = qubo_to_ising(Q_diag_pen, Q_off_pen, float(K_sq))

# --- EKSPOR BIAS & INTERAKSI TERPISAH ---
h_df = pd.DataFrame({
    'Date': [curr_date.date()] * n_assets,
    'Ticker': tickers,
    'Bias_h_Obj': h_obj,
    'Bias_h_Pen': h_pen
})
h_df.to_csv(file_config['bias_h'], mode='a', header=not os.path.exists(file_config['bias_h
']), index=False)

J_list = []
```

```
for i in range(n_assets):
    for j in range(i + 1, n_assets):
        J_list.append({
            'Date': curr_date.date(),
            'Ticker_i': tickers[i],
            'Ticker_j': tickers[j],
            'Interaction_J_Obj': J_obj[i, j],
            'Interaction_J_Pen': J_pen[i, j]
        })
J_df = pd.DataFrame(J_list)
J_df.to_csv(file_config['interaksi_J'], mode='a', header=not os.path.exists(file_config['
interaksi_J']), index=False)

# --- EKSPOR KONSTANTA C ISING TERPISAH ---
c_df = pd.DataFrame({
    'Date': [curr_date.date()],
    'C_Obj': [C_obj],
    'C_Pen': [C_pen],
    'Penalty_A_Target': [penalty_A]
})
c_df.to_csv(file_config['parameter_pendamping'], mode='a', header=not os.path.exists(
file_config['parameter_pendamping']), index=False)

# Membangun Hamiltonian terpisah (Diperlukan untuk VQE & LR Finder)
H_obj = build_hamiltonian_total(h_obj, J_obj, n_assets, offset=C_obj)
H_pen = build_hamiltonian_total(h_pen, J_pen, n_assets, offset=C_pen)

# [Tugas 4] Pencarian Nash Equilibrium (Hanya jika Warm-Start aktif)
if use_warm_start:
    ne_bitstring, ne_utility = find_nash_sbr(mu_simple_period, sigma_period_matrix, gamma,
 curr_date, N=n_assets, K=K, history_file=file_config.get('nash_history', '
riwayat_nash_sbr.csv'))
    print(f"    [Nash Eq] Warm-start aktif. Bitstring: {ne_bitstring} | Utility: {
ne_utility:.6f}")
    vqe_init_bs = ne_bitstring
else:
    ne_bitstring, ne_utility = "N/A", 0.0
    vqe_init_bs = "0" * n_assets # Mulai dari |0...0>
    print("    [VQE] Warm-start dinonaktifkan. Memulai dari $|0...0\\rangle$")

# [Validasi] Jalankan Brute Force Validation per window
# Menggunakan Hamiltonian gabungan (Total) untuk pembanding energi VQE
h_total = h_obj + penalty_A * h_pen
J_total = J_obj + penalty_A * J_pen
C_total = C_obj + penalty_A * C_pen

print(f"    [Brute Force] Validasi global minimum untuk window {curr_date.date()}...")
bf_bs, bf_e = run_brute_force_window(curr_date, n_assets, h_total, J_total, C_total, K,
file_config)
print(f"    [Brute Force] Global Min: {bf_bs} | Energy: {bf_e:.6f}")

# [LR Finder] Menggunakan H gabungan dengan pinalti target bulan ini
H_target = H_obj + penalty_A * H_pen
print("    [LR Finder] Mencari Learning Rate optimal...")
best_a, test_a_values, final_energies = find_optimal_lr_spsa(H_target, n_assets, curr_date
, ne_bitstring=vqe_init_bs, K=K, test_iters=30, file_config=file_config)
print(f"    [LR Finder] Base Learning Rate terpilih: {best_a:.4f}")

# Optimasi VQE dengan Penalty Annealing
selected_indices, depth_used, energy_final, best_history, best_ent_hist, best_gvar_hist,
depth_energies, winning_probs, best_params = run_vqe_adaptive(
    H_obj, H_pen, n_assets, curr_date, ne_bitstring=vqe_init_bs, K=K, target_penalty=
penalty_A,
    max_depth=max_depth, maxiter=maxiter,
    max_total_iter=max_total_iter, batch_size=batch_size,
    conv_window=conv_window, conv_tol=conv_tol,
    best_a_base=best_a, file_config=file_config
)

lr_data = (test_a_values, final_energies, best_a)
return selected_indices, depth_used, energy_final, best_history, best_ent_hist,
best_gvar_hist, depth_energies, lr_data, ne_bitstring, ne_utility, winning_probs,
```

```
best_params
```

Kode K.3: Skrip eksekusi langkah strategi yang memadukan teori permainan klasik dengan VQE kuantum (`run_strategy_step.py`)

BIODATA PENULIS

PUTRA NAUFAL TABASSAM lahir di Surabaya tanggal 6 Januari 2004 merupakan anak pertama dari tiga bersaudara. Penulis adalah seorang mahasiswa Program Studi Sarjana Fisika di Institut Teknologi Sepuluh Nopember (ITS) sejak tahun 2022 dengan NRP 5001221120 melalui jalur SBMPTN. Sebelum berada di ITS, penulis menempuh pendidikan formal di SD Negeri Baratajaya Surabaya, SMP Negeri 6 Surabaya, dan SMA Negeri 14 Surabaya. Di samping pendidikan formal, penulis juga pernah mengenyam pendidikan keagamaan di pondok pesantren Ummul Quroo Surabaya. Selain aktif di menjalani pendidikan, penulis pernah aktif di beberapa organisasi seperti Dewan Kerja Cabang Gerakan Pramuka Kota Surabaya Periode 2021-2026, Dewan Kerja Ranting Gerakan Pramuka Kecamatan Rungkut Periode 2020-2022, dan turut aktif menjadi panitia di banyak kegiatan baik organisasi maupun pengabdian masyarakat. Penulis memiliki minat di bidang finansial sejak akhir SMA dan memutuskan untuk mengambil fisika teori untuk mengeksplorasi lebih lanjut tentang kombinasi bidang finansial dan kuantum secara praktis. Jika pembaca memiliki pertanyaan seputar penelitian ini, penulis dapat dihubungin melalui Email: naufal.tabassam@gmail.com