Reproducing Simulations for the Two-Universe Cosmological Framework

Frederico L. Cabral

December 4, 2024

Introduction

This document provides a step-by-step guide for reproducing the numerical simulations described in the paper *Entropy Dynamics and Gravitational Interactions in a Two-Universe Cosmological Framework*. The scripts are written in Python and cover scalar field dynamics, gravitational wave production, and energy transfer between the universes.

1 Requirements

To run the simulations, ensure the following requirements are met:

- Python 3.8 or later
- Libraries: `numpy`, `matplotlib`

Install the required libraries using:

```
pip install numpy matplotlib
```

2 Repository and Setup

The simulation scripts are available in the GitHub repository:

`https://github.com/fredluiscabral/2USim`

Clone the repository and navigate to the folder:

```
git clone https://github.com/fredluiscabral/2USim
cd 2USim
```

Ensure all the scripts (`setup_simulation.py`, `scalar_field_simulation.py`, `gravitational_wave_simulation.py`, `energy_transfer_simulation.py`) are in the folder.

3 Simulations

Each simulation corresponds to a physical process described in the paper. Follow the instructions for each script below.

3.1 Scalar Field Dynamics

Run the script `scalar_field_simulation.py` to simulate the evolution of the scalar field $\phi(t)$ under the influence of different entropy balance functions $f(t)$.

Steps:

1. Open `scalar_field_simulation.py`.
2. Select the desired form of $f(t)$ by modifying the line:

   ```
   f_t = config['entropy_functions']['sinusoidal']
   ```

   Options include:

   - `'constant'`
   - `'exponential'`
   - `'sinusoidal'`
   - `'stochastic'`

3. Run the script:

   ```
   python scalar_field_simulation.py
   ```

4. View the plot of $\phi(t)$ versus time.

3.2 Gravitational Wave Dynamics

Run the script `gravitational_wave_simulation.py` to simulate the amplitude of gravitational waves $h(t)$ generated by entropy transfer.

Steps:

1. Open `gravitational_wave_simulation.py`.
2. Select the desired form of $f(t)$:

   ```
   f_t = config['entropy_functions']['exponential']
   ```

3. Run the script:

   ```
   python gravitational_wave_simulation.py
   ```

4. View the plot of $h(t)$ versus time.

3.3 Energy Transfer Between Universes

Run the script `energy_transfer_simulation.py` to simulate energy density evolution in U_1 and U_2.

Steps:

1. Open `energy_transfer_simulation.py`.
2. Select the desired form of $f(t)$:

```
f_t = config['entropy_functions']['constant']
```

3. Run the script:

```
python energy_transfer_simulation.py
```

4. View the plots of $\rho_1(t)$ and $\rho_2(t)$ versus time.

4 Extending the Simulations

The scripts are modular, allowing you to modify parameters or add new entropy functions. For example:

- Modify the scalar field potential $V(\phi)$ in `setup_simulation.py`.
- Add a new form of $f(t)$:

```
def new_function(t):
    return some_expression
entropy_functions['new'] = new_function
```

- Update initial conditions for $\phi(t)$, ρ_1, or ρ_2.

5 Troubleshooting

- Ensure all required libraries are installed.
- Verify the selected form of $f(t)$ is defined in `setup_simulation.py`.
- Check the terminal output for error messages and follow debugging suggestions.

6 Support and Contributions

For support or to contribute improvements, create an issue or pull request on the GitHub repository:

`https://github.com/fredluiscabral/2USim`

Conclusion

By following this guide, you can reproduce the simulations presented in the paper and explore the dynamics of the two-universe cosmological framework. Modify and extend the scripts to investigate additional scenarios and further validate the model's predictions.