

Um diagrama 3D do cone de luz para visualizar linhas de universo

Cassiano Faria Inacio

Universidade Federal de Alfenas – Minas Gerais, Brasil

cassiano.inacio@sou.unifal-mg.edu.br

Samuel Bueno Soltau

Universidade Federal de Alfenas – Minas Gerais, Brasil

samuel.soltaul@unifal-mg.edu.br

Data de Envio: 10 de dezembro de 2024

Resumo

Desenvolvemos um *software* para a visualização de linhas de universo no contexto das Teorias da Relatividade Restrita (TRR) e Geral (TRG). Através de um diagrama tridimensional do cone de luz, o código escrito em Python permite a classificação e localização de eventos facilitando a compreensão dos conceitos de espaço-tempo e causalidade. A implementação do código foi realizada de forma a simplificar os cálculos e a visualização, utilizando um trivector (t, x, y) sem perda de generalidade do modelo. Os resultados mostraram a eficácia da ferramenta na promoção de uma abordagem intuitiva do estudo da TRR, com potencial para ser utilizada em metodologias de ensino em cursos de graduação e no ensino médio.

Palavras-chave: Teorias da Relatividade; Cone de Luz; Causalidade; Linhas de Universo; Python.

Abstract

A Spacetime Diagram Plotter for Visualizing Worldlines

We developed software for visualizing worldlines in the context of the Special (TRR) and General (TRG) Theories of Relativity. Through a three-dimensional diagram of the lightcone, the code written in Python allows events classification and location, helping understanding spacetime and causality concepts. The code implementation was carried out in order to simplify calculations and visualization, using a trivector (t, x, y) without loss of generality of the model. The results showed the effectiveness of the tool in promoting an intuitive approach TRR study, with the potential to be used in teach methodologies in undergraduate and high school courses.

Keywords: Theories of Relativity; Lightcone; Causality; Worldlines; Python.

1 Introdução

As Teorias da Relatividade Restrita (TRR) e Geral (TRG) transformaram profundamente nossa compreensão dos conceitos

de espaço, tempo e gravidade, substituindo as noções absolutas da Mecânica Clássica por uma visão dinâmica e interdependente dessas dimensões.

No entanto, a natureza contraintuitiva das teorias de Einstein, em comparação à abordagem intuitiva da Mecânica Newtoniana, impõe novos desafios para a compreensão, especialmente para físicos iniciantes que estão apenas começando a explorar os conceitos abstratos das Teorias da Relatividade.

Diante disso, nossa motivação foi facilitar a visualização dos diferentes tipos de *eventos* e a sua localização nas distintas regiões no *cone de luz* por meio de um *software*. Utilizando a linguagem *Python*, foi desenvolvido um diagrama tridimensional capaz de exibir a posição e a classificação de cada evento dentro do cone.

A utilização de códigos computacionais para estudar fenômenos físicos é uma ferramenta valiosa. Merma et al. (2024) utilizou uma simulação numérica para modelar trajetórias de projéteis em condições com e sem resistência do ar. Enquanto Sherin et al. (2016) desenvolveu jogos interativos com código aberto para simular efeitos relativísticos, oferecendo uma experiência de laboratório virtual. Müller and Bobblest (2014) exploraram a aparência visual de objetos em movimentos relativísticos, usando *Python* e técnicas de visualização em estrutura de arame, ele representou a distorção visual de linhas e esferas em movimento próximo a velocidade da luz. Como alternativa de visualização dos conceitos da Teoria da Relatividade Especial Sumardani et al. (2020) desenvolveu uma aplicação de realidade virtual como mídia de aprendizagem para TRR. Kuwahara et al. (2024) realizou um trabalho sobre o limite da velocidade de propagação de informação, onde ele desenvolveu um algoritmo eficiente para simular sistemas de bósons integrantes. Takiwaki et al. (2009) desenvolveu um código computacional embasado na TRR, analisando o núcleo de estrelas massivas, magnetizadas e com alta rotação para estudar as propriedades das explosões magnéticas ao longo de um período prolongado de evolução.

Neste artigo expomos o trabalho do seguinte modo. Na seção 2 apresentamos os conceitos mínimos da TRR empregados na construção do *software*. Os procedimentos metodológicos estão descritos na seção 3. Os resultados alcançados foram dis-

cutidos na seção 4. Concluímos com uma síntese das contribuições do estudo e sugerimos direções para a pesquisa futura.

2 Teoria

A classificação dos eventos no espaço-tempo, em relação ao cone de luz, depende diretamente do posicionamento do cone de um *evento* em determinado referencial. O cone de luz de um *evento* contém as áreas que podem ser conectadas causalmente com este evento. A região interna ao cone abarca os *eventos* cuja conexão causal pode ser alcançada ou influenciada sem ultrapassar a velocidade da luz ($v \leq c$) e a região externa é composta pelos *eventos* que não podem ser conectados causalmente ao evento.

Esta classificação, entretanto, é dependente de como o observador posiciona o cone de luz no referencial utilizado. Observadores diferentes com velocidades distintas, poderiam dispor diagramas do cone de luz de formas variadas. Portanto, a relação causal entre dois eventos não é uma propriedade absoluta, mas é uma propriedade dependente da posição do observador considerado.

O *cone de luz* é uma representação geométrica no *espaço-tempo* que evidencia o caminho de uma *linha de mundo* do passado para o futuro passando pelo presente como na figura 1, é utilizado para analisar a causalidade entre os *eventos* Rodrigues (2013).

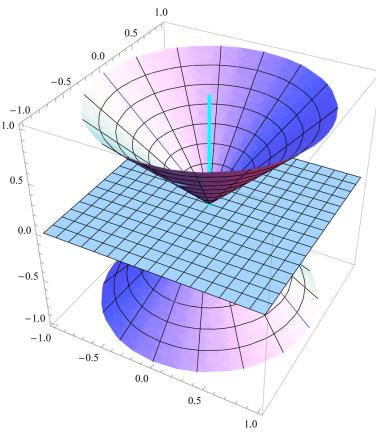


Figura 1: Ilustração do Cone de luz 3D

A Causalidade estabelece que *eventos* causam outros *eventos*, respeitando a sequência temporal. O princípio de causalidade é mantido se:

(i) Nenhuma informação ou partícula viajar mais rápido que a velocidade da luz (c).

(ii) *Eventos* localizados em Outro Lugar no cone de luz de um dado *evento* não podem ser influenciados por ele, pois estariam em uma região onde $v > c$, violando a causalidade (Hartle, 2003, cap.4).

A classificação, e consequentemente a localização dos even-

tos no cone de luz é determinada pela Eq. (1).

$$ds^2 = -c^2 dt^2 + dx^2 + dy^2 + dz^2, \quad (1)$$

que é denominada *elemento de linha* (Misner, 1973, cap.2).

O intervalo entre eventos decorrentes do elemento de linha são classificados como:

- **Tipo-tempo (time-like):** Quando $ds^2 < 0$, e nesse caso temos uma dominância temporal. Todo evento que esteja localizado dentro do cone de luz recebem essa classificação.
- **Tipo-luz ou Tipo-nulo (null/light-like):** Quando $ds^2 = 0$, o *evento* representa partículas sem massa que viajam na superfície do cone de luz na velocidade da luz.
- **Tipo-espacó (space-like):** Quando $ds^2 > 0$, há dominância espacial, nesse caso não é possível interação entre os eventos, pois a velocidade necessária para que um objeto viaje em uma linha de mundo *tipo-espacó* seria maior do que a velocidade da luz violando a causalidade na TRR (Sartori, 1996, cap.5).

Além de permitir a classificação dos eventos na tipologia acima é possível localizar a posição dos eventos nas regiões cone de luz.

- **Futuro (future):** Quando $dt > 0$ e $|dt| > dx$.
- **Passado (past):** Quando $dt < 0$ e $|dt| > dx$.

Eventos localizados nas regiões do futuro e passado podem ser vinculados ao evento presente E_0 .

- **Outro lugar (elsewhere):** Quando $|dt| < |dx|$. Utilizando unidades naturais as superfícies do cone de luz formam um ângulo de 45° graus, ou seja um *evento* localizado em Outro Lugar teria uma velocidade maior que a da luz para que sua *linha de mundo* passa-se por E_0 , por isso é impossível haver causalidade entre os *eventos* localizados dentro do cone de luz.
- **Na superfície do cone de luz:** Quando a coordenada espacial tem o mesmo valor da coordenada temporal, $|dt| = |dx|$, o evento se encontra na superfície do *cone luz*, são possíveis apenas *linhas de mundo* de partículas sem massa. A inclinação da luz forma o ângulo da superfície do cone de luz de 45°, por isso *eventos* com essa classificação estão situados nas bordas do cone de luz. (Wald, 1984, cap.1).

3 Metodologia

O *software* foi implementado em *Python*, versão 3.13.0, usando os repositórios (Matplotlib, 2024, versão 3.8.2), conforme indicado em Downey (2023), para plotar o *cone de luz* em 3 dimensões e (Numpy, 2024, versão 1.24.0) para realizar os cálculos do *elemento de linha*.

Após a leitura de um arquivo de dados contendo os parâmetros de cada evento escolhidos aleatoriamente para exemplificar todas as regiões do diagrama (cf. Tabela 1), uma função calcula o *elemento de linha* que posteriormente será usado para classificar os eventos e indicar a região do *cone de luz* na qual estão localizados.

Tabela 1: Eventos plotados no cone de luz

Eventos	dt	dx	dy
E_1	3	4	0
E_2	6	5	0
E_3	8	8	3
E_4	9	3	4
E_5	6	7	0
E_6	7	5	2
E_7	-8	4	1
E_8	-7	4	2

Em seguida o *cone de luz* em três dimensões é plotado usando coordenadas cilíndricas. *Eventos* na região do passado são plotados na cor vermelha e *eventos* na região futuro são plotados em azul. Os *eventos* são rotulados em sequência $E_1, E_2, E_3, \dots, E_n$.

A função principal (*main*) encarrega-se de ler o arquivo de dados, processar os *eventos*, calcular o *elemento de linha*, classificar os *eventos* situados nas regiões do *cone de luz* e plota o diagrama 3D do *cone de luz*. Na figura 2, apresenta-se o diagrama de fluxo do código computacional.

Utilizou-se no código o Sistema de Unidades Naturais, no qual a velocidade da luz é igual a unidade. Devido ao fato de que na TRR espaço e tempo são unificados no *quadrivector espaço-tempo* podemos utilizar a mesma unidade para todas as coordenadas deste *quadrivector*.

Vale mencionar que no código computacional utilizou-se por simplicidade apenas o *trivector* (t, x, y) , sem perda de generalidade.

4 Resultados e Discussão

A implementação do *software* em *Python* resultou em uma ferramenta eficaz para a visualização e classificação de eventos dentro do cone de luz, conforme mostrado na Figura 3.

A análise dos *eventos* da Tabela 1 mostra que, dos oito *eventos*, cinco estão classificados como *tipo-tempo*, indicando que eles podem influenciar outros *eventos* e estão localizados dentro do *cone de luz*, dois destes *eventos* estão no passado, e três no futuro do *cone de luz*. O código classificou também três *eventos tipo-espac*, localizados fora do cone.

A escolha de utilizar um *trivector* (t, x, y) simplificou o funcionamento do código computacional, tanto nos cálculos do *elemento de linha*, quanto na plotagem do diagrama 3D, além de não comprometer a generalidade do *software*.

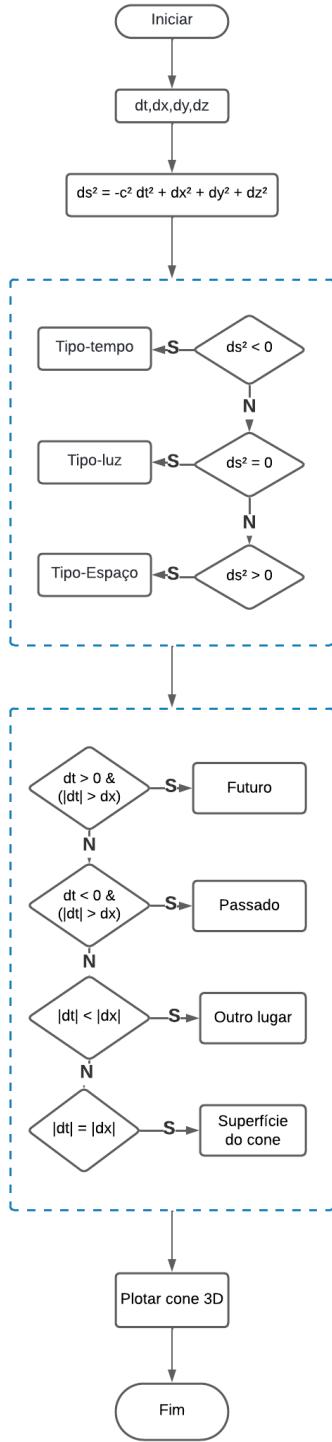


Figura 2: Diagrama de fluxo do código computacional

A flexibilidade da linguagem *Python* permitiu trabalhar com uma sintaxe simples e intuitiva, que favorece a interpretação do código computacional por interessados em estudá-lo¹. As

¹O código está disponível em <https://github.com/CFI-Physics/>.

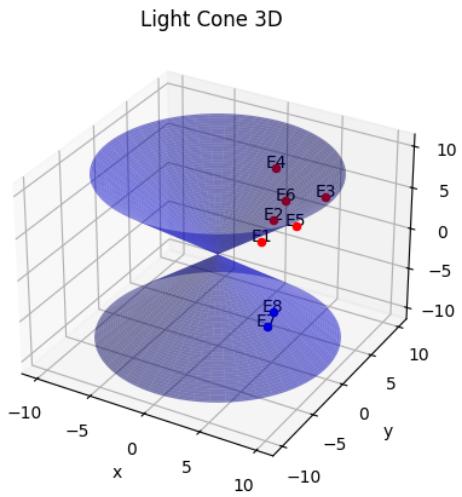


Figura 3: Cone de Luz 3D com os *eventos* plotados.

bibliotecas facilitaram a criação do diagrama e os cálculos necessários para a classificação e a localização dos *eventos* no código. Por ser amplamente usado pela comunidade acadêmica e ter um grande suporte da comunidade de desenvolvedores, *Python* se torna acessível como ferramenta para a implementação de *softwares* científicos.

Os resultados obtidos por meio da visualização de *eventos* no *cone de luz* mostraram ser eficazes para facilitar a compreensão dos conceitos da TRR e promover uma abordagem intuitiva e acessível ao estudo do espaço-tempo e das relações causais entre *eventos*.

A classificação de *eventos* carrega implicações profundas para a compreensão do conceito causalidade na Teoria da Relatividade Restrita. Os *eventos tipo-tempo* estão localizados dentro do cone de luz, e podem ter relação causal com o *evento de referência* (E_0) localizado no centro do cone de luz $(0, 0, 0)$. Utilizar essa ferramenta de classificação para analisar colisões de partículas elementares, onde as trajetórias de partículas dentro do cone de luz definem aquelas interações que podem ocorrer sem violar a causalidade é uma alternativa de visualização valiosa.

O cone de luz tridimensional usado neste estudo além de facilitar a visualização da estrutura causal, também promove uma compreensão intuitiva da separação entre *eventos tipo-tempo*, *tipo-espac*o e *tipo-luz*. Para físicos inciantes no estudo de TRR, essa visualização auxilia a internalizar conceitos abstratos, como a impossibilidade de influências superluminais.

A principal limitação do modelo é ser útil somente com *espaço-tempo* plano, englobando apenas a TRR. O código é uma abordagem introdutória com foco em conceitos básicos e inciais, o que limita a aplicabilidade em sistemas mais complexos e realistas. O enfoque utilizado não considera a curvatura do espaço-tempo causada por corpos massivos no contexto da

Teoria da Relatividade Geral.

No *espaço-tempo* curvo, os cones de luz não permanecem estacionários nem uniformes, eles sofrem uma deformação ocasionada pela influência gravitacional, o que altera significativamente a análise dos *eventos* e sua classificação como *tipo-tempo*, *tipo-espac*o ou *tipo-luz*. Dado isso, o modelo atual não é capaz de lidar com cenários em que a curvatura desempenha um papel fundamental, como na análise de órbitas elípticas, lentes gravitacionais ou ondas gravitacionais.

5 Conclusão

Apresentamos um código computacional escrito em *Python* com o propósito de auxiliar o estudo das *linhas de universo*. Inserimos *eventos* para serem plotados na forma de um diagrama 3D. Por meio de diversos ensaios foi possível verificar a funcionalidade e a precisão dos cálculos inseridos no código.

No futuro adaptaremos o código computacional para ser utilizado em uma metodologia de ensino para cursos de graduação em física e também na disciplina do ensino médio. Possivelmente incluindo a possibilidade de interações com o usuário em tempo real permitindo modificações de parâmetros, e a observação de como tais mudanças impactam a classificação e localização dos *eventos* no diagrama do *cone de luz*.

Referências

- Downey, A. B. (2023). *Modeling and simulation in python*.
- Hartle, J. B. (2003). *Gravity: An Introduction to Einstein's General Relativity*. Addison-Wesley.
- Kuwahara, T., Vu, T. V., and Saito, K. (2024). Effective light cone and digital quantum simulation of interacting bosons. *Nature Communications*, 15.
- Matplotlib (2024). *Matplotlib: Visualization with python*.
- Merma, M., Monroy, O., and Castillo, M. (2024). Simulación numérica de trayectorias de proyectiles: Influencia de la fuerza de arrastre. *Revista de Investigación de Física*, 27:2.
- Misner, Charles W.; Thorne, K. S. W. J. A. (1973). *Gravitation*. San Francisco.
- Müller, T. and Boblest, S. (2014). Visual appearance of wireframe objects in special relativity. *European Journal of Physics*, 35.
- Numpy (2024). *Numpy: Visualization with python*.
- Rodrigues, M. (2013). *Cone de luz 3d*.
- Sartori, L. (1996). *Understanding relativity: a simplified approach to Einstein's theories*. Univ of California Press.

Sherin, Z. W., Cheu, R., Tan, P., and Kortemeyer, G. (2016). Visualizing relativity: The openrelativity project. *American Journal of Physics*, 84:369–374.

Sumardani, D., Putri, A., Saraswati, R. R., Mulyati, D., and Bakri, F. (2020). Virtual reality media: The simulation of relativity theory on smartphone. *Formatif: Jurnal Ilmiah Pendidikan MIPA*, 10.

Takiwaki, T., Kotake, K., and Sato, K. (2009). Special relativistic simulations of magnetically dominated jets in collapsing massive stars. *Astrophysical Journal*, 691:1360–1379.

Wald, R. M. (1984). *General Relativity*. University of Chicago Press.