

基于 html5 中 Canvas 元素实现的时空图动态演示

物理学系 胡耕源 17307110016

摘要: 本实验运用 HTML5 网页技术, 制作了一个可以在用户控制下生成动态时空图的网站, 并以车库佯谬为例, 利用该网站制作了对该佯谬的时空图详解, 示范了该网站的使用方式。

一、引言

在狭义相对论的学习过程中, 时空图是直观展示不同惯性系下, 不同物体运动状况的良好工具。借助时空图, 可以简化很多问题的分析难度。但是, 时空图的几何空间不是我们熟悉的欧氏空间, 而是抽象的闵氏空间, 这使得我们在绘制和分析时空图时, 有一定的困难。此外, 我们有时候希望在不同的惯性坐标系下分析同一个问题, 而手动绘制相应的时空图时, 需要一定的计算来适应新的坐标系, 且在这个过程中我们无法直观地体会不同的坐标系之间的切换过程。而利用计算机技术, 我们可以实现时空图的动态演示, 能够给学习者展现出参考系变换非常直观的几何图像, 并且极大地降低了绘制时空图的难度。本实验中, 作者利用 html5 制作了一个能动态演示时空图的网站, 实现了绘制时空图和动态切换坐标系的功能, 并制作了一个关于车库佯谬的例子, 来展示时空图的作用。

二、实验原理¹

1. 狭义相对论和洛伦兹变换

洛伦兹变换是在对麦克斯韦方程的研究过程中产生的。19 世纪末, 随着对麦克斯韦方程组的研究, 人们发现它隐含了光速不变的推论。这一推论也得到了实验的验证, 而这与经典力学中的伽利略变换相矛盾。而此时, 洛伦兹发现, 用他提出的一种新的变换方式替代伽利略变换, 可以使光速不变性自动满足, 并成功解释了一系列由光速不变导致的佯谬。爱因斯坦发展和推广了这一成果, 得到了著名的狭义相对论。

洛伦兹变换的数学公式为:

$$\begin{cases} t' = \gamma(t - vx) \\ x' = \gamma(x - vt) \\ \gamma = 1/\sqrt{1 - v^2} \end{cases} \quad (1.1)$$

为了简化叙述, 这里仅给出了 1+1 维情形。其中 x 是空间坐标, v 是 x 方向上的速度标量, t 是时间坐标。本实验报告后续内容中, 也仅对 1+1 维情形进行阐述。

事实上, 该变换是 x 与 t 张成的二维线性空间 (闵氏时空) 中的线性变换, 因而可以写成矩阵形式, 它对应一个 2×2 的矩阵:

$$\gamma \begin{pmatrix} 1 & -\beta \\ -\beta & 1 \end{pmatrix} \quad (1.2)$$

上式中 $\beta = \frac{v}{c}$, 即 $c=1$ 的自然单位制下的速度。

2. 时空图

¹ 主要参考梁灿彬著《微分几何与广义相对论·上册》第二版第六章内容, 此处仅给出基本概念, 更严谨的表述和更多具体的理论内容请参考原著

我们可以用时空图描述 1+1 维时空中的运动。

下图展示了时空图中的基本元素,包含坐标轴,描述物体运动的世界线,和光的世界线(辅助线),下图由本文将介绍的网页程序生成。

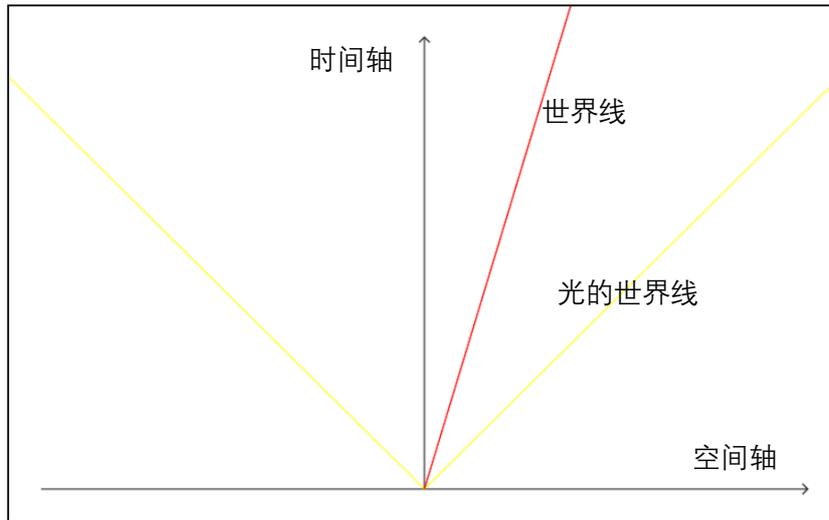


图 1 时空图概览

图中,横坐标描述空间维度,纵坐标描述时间维度。对同样的运动,不同惯性参考系下的时空图不同,这些时空图之间的变换即洛伦兹变换。做一维运动的物体的时空轨迹称为世界线,而匀速运动的物体在时空图上的轨迹是一条直线。

特别注意的是,当我们采用自然单位制,即 $c=1$ 时,光线在图中表现为倾角 45° 的直线,而其他任何匀速运动的世界线都不可能越过这一直线。在上图中,黄色的线即表示从原点出发的光线

3. 时空间隔和校准曲线

前面提到,时空图所在的空间是 1+1 维的闵氏空间,因而它的几何也是闵氏几何。在闵氏几何中,最重要的量之一是“空间线元”,也就是我们物理上所说的时空间隔,定义为:

$$ds^2 = (dx)^2 - (dt)^2 \quad (1.3)$$

这是一个洛伦兹不变量,即在不同坐标系间变换时,该时空间隔不变。从闵氏时空中的“距离”为

$$l_{op} = \sqrt{-t^2 + x^2} \quad (1.4)$$

上式中的 l_{op} 表示从原点 O 到时空图上任一点 p 的距离。于是我们发现,不同于欧氏时空中,等距离线是一个圆,闵氏时空的等距离线是一对双曲线:

$$l_{op}^2 = -t^2 + x^2 \quad (1.5)$$

因此,在比较时空图上的线长时,我们不能用欧氏几何的观念比较,要借助等距离双曲线的帮助。我们称该等距离线为“校准曲线”,图 2 为校准曲线的示意图,其中橙色的双曲线是校准曲线,黄色的直线是光的世界线。不难看出,校准曲线是以光的世界线为渐近线的等轴双曲线。

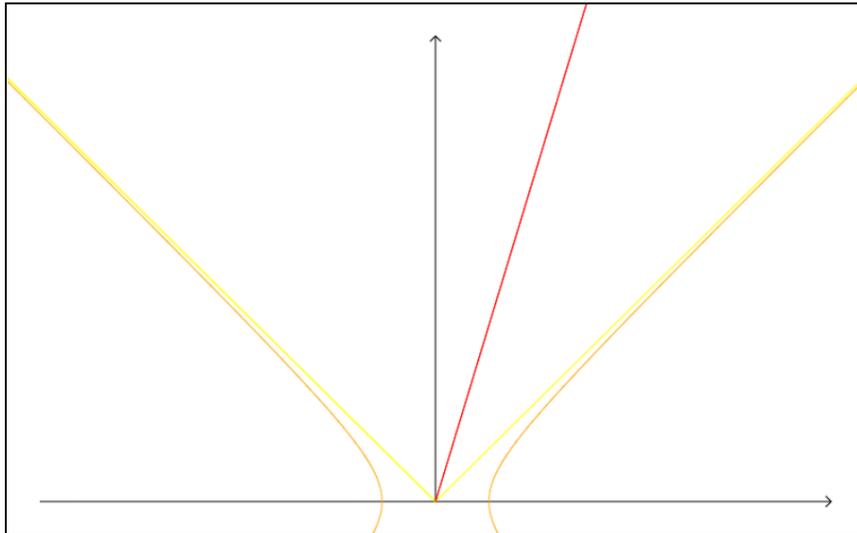


图 2 校准曲线示意图

4. Html5²与 Canvas³简介

HTML5 是 HTML 最新的修订版本，由万维网联盟（W3C）于 2014 年 10 月完成标准制定。目标是取代 1999 年所制定的 HTML 4.01 和 XHTML 1.0 标准，以期能在互联网应用迅速发展的时候，使网络标准达到符合当代的网络需求。广义论及 HTML5 时，实际指的是包括 HTML、CSS 和 JavaScript 在内的一套技术组合。

HTML5 中提供了<Canvas>标签。网站设计者可以通过 JavaScript 在 Canvas 元素上绘画，它可以用于动画、游戏画面、数据可视化、图片编辑以及实时视频处理等方面。

5. Canvas 动画

Canvas 可以被用于制作动画，不过 Canvas 本身并不提供内部图形的动态变化功能，而是需要以循环的方式反复重绘图像来模拟内部图形的变形和运动。JavaScript 内置了循环绘图的函数，但是我们仍需设计算法，使程序能够计算出每一帧的图形并生成绘制方法，再由循环绘图实现动画效果。

三、程序设计 with 过程

1. 时空图的基本元素

一副完整的时空图必须具备两类基本元素：坐标系和世界线。

坐标系包含坐标轴和坐标网格，无论在什么惯性系下，本惯性系的坐标轴和坐标网格都是水平和垂直方向正交的直线，因此可以提前将它们保存为 Path2D 对象。考虑到有时候我们只显示坐标轴不显示网格，将两者分开保存以便单独绘制。

世界线因物体运动速率的不同，在图中表现为不同斜率的直线，此外不同物体的出发点可能不一致，因此世界线与空间轴的交点也可能不一致。因此，绘制世界线需要两个参数：速率和出发位置。

2. 设计目标

通过 Canvas，实现一个可以帮助用户绘制时空图的页面。时空图可以同时展示多个物体的世界线，并且可以在不同的惯性参考系中切换，包含坐标

² 引用自维基百科 <https://zh.wikipedia.org/wiki/HTML5>

³ 引用自 MDN 网页文档 https://developer.mozilla.org/zh-CN/docs/Web/API/Canvas_API

原点的平移和不同速度的惯性系间的洛伦兹变换。当参考系进行切换时，时空图中的所有世界线同步地进行协变。

此外，时空图中的元素可以以任意的颜色呈现，以便区分不同物体和坐标轴。当用户添加新的世界线时，可以指定其颜色和名字，并会自动生成图例。

3. 程序设计的总框架

a) 分层设计

Canvas 无法修改已经绘制的内容，只能擦除后重绘来达到动画效果。因此，为了减小运算量，避免很多不必要的重绘，将 Canvas 分层。

在本实验所设计的网页中，重叠放置了 3 个大小相同的 Canvas 元素。底层为背景层，负责防止不参与动画绘制的静态元素；中间层为时空层，负责绘制动态的时空图；顶层为 UI 层，负责执行用户交互。

b) 统一重绘

在播放动画时，由于 Canvas 本身的特性，不可以对原有的图形进行修改，需要全部重绘。因此，我们需要另外储存图中所有的元素的绘制参数，当重绘时，绘图函数读取我们保存的数据，自动生成相应的动画。

对于两类基本元素，我们分别定义一个类（class）保存。对于坐标系，类包含四个属性：坐标系名称，原点位置坐标 x ，运动速度 v ，是否显示网格 `showGrid`。对于世界线，包含世界线名称，原点坐标，运动速度三个属性。在物理上，这里的运动速度隐含了一个已经规定的“静止参考系”，上文中提到的速度均是相对该静止参考系的速度。

定义了两个类以后，每条世界线或每个坐标系都是类的一个实例。每当添加一条世界线时，即生成一个实例，保存在一个 `map` 容器里。所有绘图操作由一个统一的函数完成，这个函数自动遍历保存元素数据的 `map` 并完成绘图。

c) 用户接口

除了实现基本功能的函数外，对用户接口进行额外的封装，尽量避免用户直接操作时空图对象的属性，从而减少程序出错的可能。用户可以在网页上通过控制面板按钮，执行预先设计的操作。

The image shows a user interface control panel for adding worldlines. It contains the following elements:

- 名称:** A text input field.
- 类型:** A dropdown menu with "世界线" selected.
- 起始位置:** A text input field.
- 速度:** A text input field with "(自然单位制)" next to it.
- 选择颜色:** Two color selection boxes, each with a black color swatch.
- 透明度:** Two sliders, each with a white knob on a horizontal track.
- 模式:** A dropdown menu with "添加" selected.
- 执行:** A button.
- 生成代码:** A button.
- 执行代码:** A button.
- 代码:** A large text area for entering code.

除了使用按钮控制外，用户还可以直接输入操作代码。使用代码的优势在于可以一次性编辑多行，描述全部要执行的动作，方便用户之间交流共享

所绘制的时空图。用户不需要事先学习代码的语法，可以通过“生成代码”功能用控制按钮生成代码。

4. 绘图函数的具体实现⁴

a) 绘图参考系

在 Canvas 中，有一个预设的坐标系，用户通过该坐标系的坐标来控制绘制图形的位置。这一坐标系的初始定义比较奇怪，以 Canvas 元素的左上角为原点，向右为 x 轴正方向，向下为 y 轴正方向。在绘制时空图前，通过坐标变换将其变换到我们常用的笛卡尔坐标系。

b) 坐标平移和洛伦兹变换

坐标平移较为简单，遍历所有元素，将其 x 属性加减即可。

根据式 (1.2)，洛伦兹变换可以用一个线性变换矩阵来表示，而 Canvas 中提供了“应用变换矩阵”的功能 `transform(a,b,c,d,e,f)`，该功能对 Canvas 的绘图参考系做变换。其中前四个参数即对应了二维平面上的线性变换矩阵 $\begin{pmatrix} a & c \\ b & d \end{pmatrix}$ ，e 和 f 表示水平和垂直方向的平移，因此我们讲 abcd 用式 (1.2) 代入，并将 e/f 设为 0 即可实现时空图上的洛伦兹变换。

c) 静态元素的绘制

前面提到，我们将所有时空图元素放入 map 容器中，并循环读取绘制。对于静态元素的绘制，将绘图函数作为元素类的方法，读取自身属性并绘制相应的图形。

在绘制过程中，不计算元素在图中的倾角等等，而是总将绘图坐标系根据速度参数进行洛伦兹变换后，画出其坐标系内的静止的元素。物理上，就是采用被动变化的观点，令坐标系发生变化，永远以绘制物体自身为参考系绘图。这样可以避免计算对象在不同坐标系中的图像的麻烦和可能出现的错误。

d) 辅助元素

辅助元素包含光的世界线和校准曲线。它们大多数时候仅仅帮助我们观察已画好的图形，在重新绘图或者变换视角后，就不再需要，因此我们不保存这些元素，仅在调用相关函数的时候直接画出，当背景图层发生任何变换后便被擦除。

四、实验结果与分析

1. 时空图程序的应用：以车库佯谬为例⁵

狭义相对论中，对于高速运动的惯性系，存在“尺缩钟慢”的效应。这一效应会造成很多与我们日常经验相冲突的佯谬，车库佯谬便是这类佯谬的代表之一，通过理解车库佯谬，我们可以对狭义相对论中有更深入的理解。

车库佯谬的内容为：假设有一列长度为 a 的列车，以非常接近光速的速度 v 开进一个车库。车库的长度也为 a，与这列列车相同。对于列车参考系上的人而言，由于尺缩效应，他们看到的车库长度应该小于 a，从而这列列车无法全部停进车库；而对于车库参考系的人而言，这列火车相对他们运动，由于尺缩效应，长度小于 a，从而可以完全停进车库。从日常经验而言，这两个结论时相矛盾的，而这是这是相对论可以推出的严格结论，这似乎表明相

⁴ 注：由于代码长度过长，不便在实验报告中展示，以下内容仅介绍设计思路和展示最终效果，不包含具体代码。完整代码见附件，附件中的代码有完整注释对代码功能进行解释。

⁵ 这一示例的绘图代码也包含在了网站的脚本中，使用 `TrainandPark()` 即可调用函数生成该时空图

对论存在一定的问题。但是实际上，如果我们以相对论的思路去理解这一过程，就会发现两件事情并不矛盾。借用上面提到的时空图，我们可以很直观地去理解这一问题背后的原因。

我们首先在时空图上画出这一事件中，各个对象的世界线：⁶

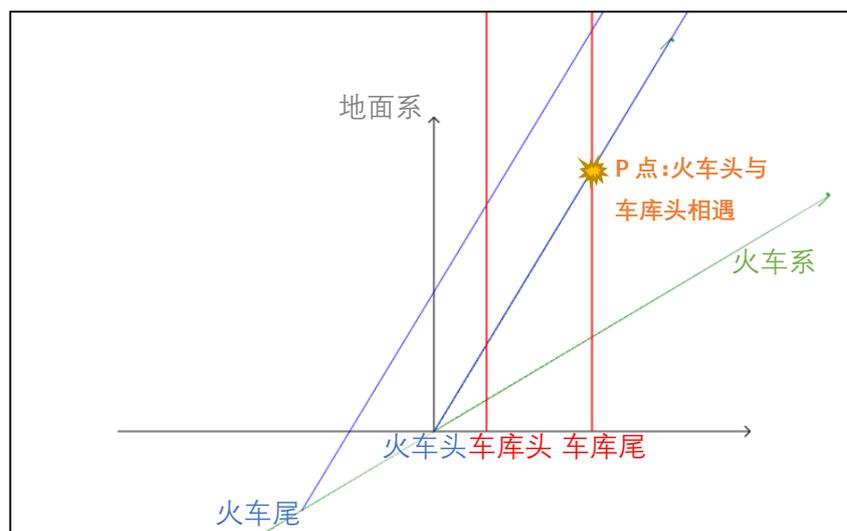


图 3 火车与车库的世界线

如图，用时空图的语言，我们可以重新表述车库佯谬：在火车系中，在过 P 点的同时面上，火车 s 头与火车尾的距离大于车库头与车库尾的距离；在地面系中，在过 P 点的同时面上，火车头与火车尾的距离小于车库头与车库尾的距离。在这一表述中，我们充分利用了一个坐标系变换下的不变量：火车头与车库头相遇这一事件，是同时同地发生的。要进一步分析这两个坐标系之间运动的关系，我们还需要添加一些辅助线：

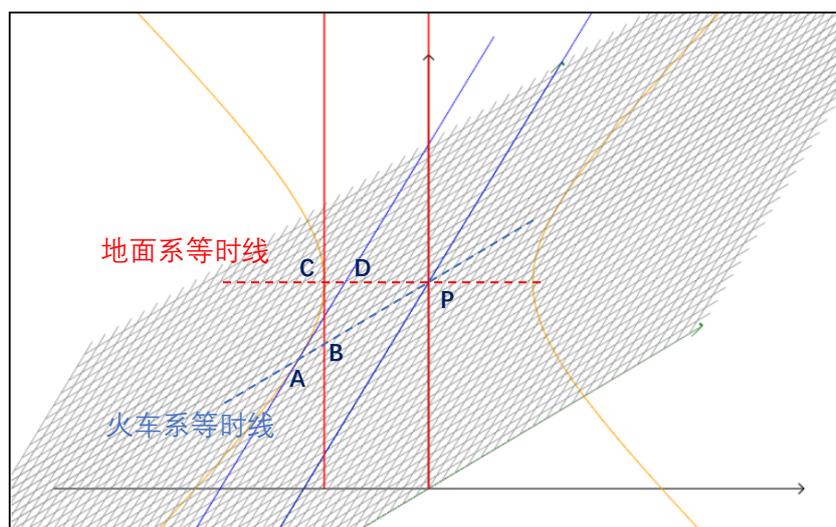


图 4 车库佯谬的传统分析

上图中，背景网格是火车系的坐标网格，因此火车的世界线和等时线分别平行与其横轴和纵轴。除 P 点外，火车系的等时线交车库尾于 B，火车尾于 A；地面系的等时线交火车尾于 D，车库尾于 C。事实上，A 点同时还是火车尾的世界线与图中的以 C 点为顶点，关于 P 的校准曲线的切点，在传统的作

⁶ 图中的坐标系，世界线由程序绘制，文字标签是人工添加的。程序只绘出最简的时空图，以适应图中后期添加不同解释元素的需要。

图方法中，通常是先作出粗略的校准曲线，再由相切关系得到火车尾的世界线。

通过简单的比较我们发现，在火车系中， $PA > PB$ ，故火车长于车库。而在地面系中， $PD < PC$ ，故车库长于火车。得出不同的结论是由于不同参考系下由不同的等时线，两者并不矛盾。

以上是实用程序模拟传统的解题方法。在这个过程中，我们仅仅是用程序替代了人工绘图，优势主要体现在精确度的提升：在人工绘图中，世界线和等时线的斜率可以通过一定的计算控制，但是对于校准曲线的绘制和相切点的寻找，是非常困难的。除非通过非常繁琐的计算或者很复杂的工具，我们才可以在耗费很长时间的情况下，达到非常有限的精度。而程序绘制的可以说几乎是完全精确的，并且操作非常简单，仅仅需要用代码表示出，我们想画出哪些元素。

2. 视角切换的动态展示

但是，大多数问题中，决定问题的特征量在图上足够明显，以至于我们不需要很高的精度，传统的粗略手绘方式足以胜任，我们的程序并不能很好地体现出优势。下面我用另一种方法重新阐释车库佯谬，以体现程序独特之处。

在手绘中，我们无法解决的一个难点是，绘制同一事件在不同视角下的时空图。如果不计时间消耗，我们的确可以反复地人为计算洛伦兹变换，画出几个特定视角下的时空图，但我们还是难以把握到这些时空图之间的直接联系，而我们的程序，可以很好地展现这一点：

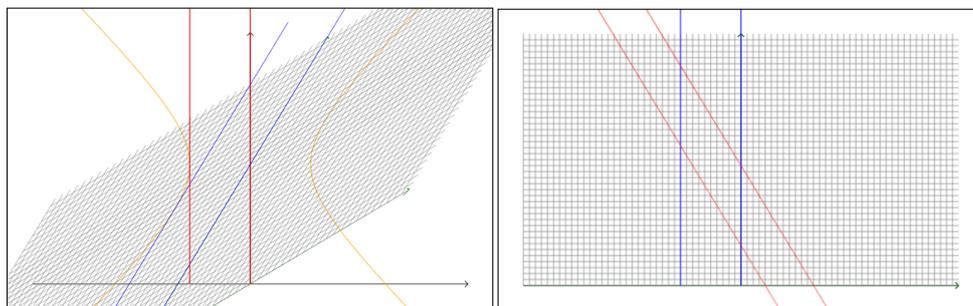


图 5a 地面系视角下的时空图

图 5b 火车系视角下的时空图

通过调用 `CentralizeWorldline(trainhead)` 和 `ChangeViewTo(trainhead)` 函数，我们无需重新设计，即可将火车头的世界线置于视野中央，并切换到火车系的视角。在火车系中我们再次清楚地看到，火车的等时面上，车库变短了。更重要的是，在网页中，我们不仅仅可以看到结果，还可以看到图 5a 是如何连续地变换到图 5b 的，整个过程，都是在洛伦兹变换下进行的。⁷

3. 程序的不足

本程序绘制时空图的核心功能已经基本完善，但还存在两个主要的缺陷：用户交互设计不完善，在网页上利用控制面板操作还较为麻烦，一些设计没能达到预期效果，体验不是很好，要取得良好的体验仍需使用代码操作；辅助元素（校准曲线，光线）还没有纳入统一绘制的框架，在一些情况下会出现 bug，有待后续改进。

⁷ 请打开附件中的网站查看动态效果。要自己尝试不同的效果，请查看 `main.js` 中的 `TrainandPark()` 函数的代码并尝试增减注释。

五、实验结果

根据设计目标,成功设计了一个扩展性非常强的框架来实现所有的预期功能,但是在用户交互上还缺少优化,有待改进。

六、参考文献

- [1] 梁灿彬, 微分几何入门与广义相对论 (上册·第二版) (页 133-151).
- [2] 维基百科, <https://zh.wikipedia.org/wiki/HTML5>
- [3] MDN, https://developer.mozilla.org/zh-CN/docs/Web/API/Canvas_API