

流星の密度計算の予察的研究

滋賀県立米原高等学校 地学班 山本直矢 市橋秀一 小倉匠翼 山田直人

動機 自分達で観測した流星は肉眼では白色に見えたのに対し、撮影した画像は複数の色に見られた。そこで、R.G.B.の光の強度を調べることで、流星の光の特徴がわかるのではないかと考え研究を行った。

目的 流星の色別輝度変化の傾向を調べる。また、流星の密度を調べて、輝度変化のグラフとの関係性を調べる。

研究1 流星の観測

機材 ・NIKON D3200 ・Canon EOS20Da

方法 露光時間を30秒に固定し、ISO設定を撮影状況により変更する。

結果 ペルセウス座流星群を4枚、オリオン座流星群を1枚撮影することができた(図1.2.3)。



図1.ペルセウス座流星群の流星1 (NIKON, ISO3200)

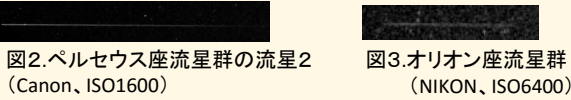


図2.ペルセウス座流星群の流星2 (Canon, ISO1600)

図3.オリオン座流星群 (NIKON, ISO6400)

研究2 色別輝度変化グラフの作成

方法 ①各画像をR・G1・G2・Bの色別に分ける。
②画像処理ソフトマカリを用いて測光を行う。

結果 流星群別に、輝度変化の傾向に違いがあることがわかった(図5.6.7)。

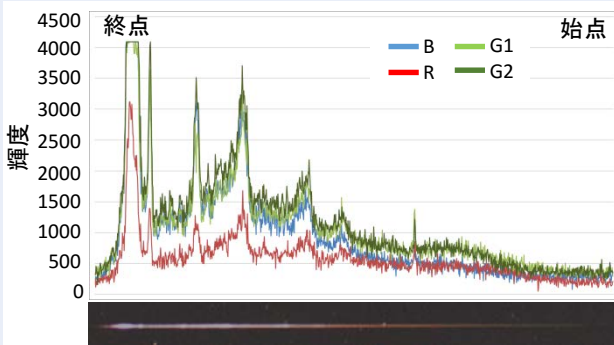


図5. ペルセウス座流星群の流星1の色別輝度グラフ

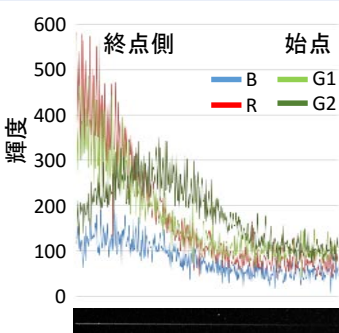


図6. ペルセウス座流星群の流星2の色別輝度グラフ

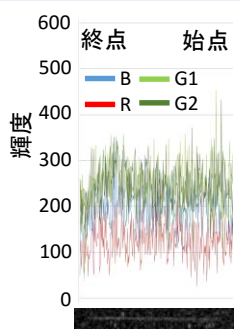


図7. オリオン座流星群の流星の色別輝度グラフ

考察

- 輝度が急激に増加している部分は、流星塵が分裂し瞬間的に燃焼したと考えられる(図5)。
- G2の光のみが増加している部分は、流星本体の光ではなく、短痕の光と考えられる(図6)。
- 輝度変化の傾向は、ペルセウス座流星群は山型、オリオン座流星群は平型であると考えられる。

研究3 流星の測光質量を求める

方法

①流星の見かけの等級を求める

- 各写真で基準星を決め、画像処理ソフトマカリを用いて、各明るさを測定する。
- I式より、流星の見かけの等級を求める。

$$\text{流星全体の明るさが基準星の明るさに対する倍率} \\ = 10^{\frac{2}{5}(\text{基準星の等級} - \text{流星の見かけの等級})} \dots \text{I式}$$

②流星の絶対等級を求める

II式より、流星の絶対等級を求める。

$$M = m + (10 - 5 \log L) - (2.5p) \log(\sin \theta) \dots \text{II式}$$

③ペルセウス座流星群の流星の測光質量を求める

III式より、ペルセウス座流星群の測光質量を求める。

$$\log m_p = -0.39M - 1.69 \dots \text{III式}$$

M=絶対等級、m=見かけの等級
m_p=測光質量(g)、p=相反則不規係数
L=観測者と流星の距離(km)・・・100kmとする
θ=視線方向と流星の進行方向のなす角度

結果 ペルセウス座流星群については、表1のようになった。

表1.ペルセウス座流星群の各流星の等級と測光質量

	ペルセウス座流星群			
	流星1	流星2	流星3	流星4
見かけの等級	-3.50	-0.85	0.05	-0.04
絶対等級	-1.06	1.59	2.50	2.40
測光質量(g)	5.27×10^{-2}	4.88×10^{-3}	2.17×10^{-3}	2.36×10^{-3}

- 絶対等級は、見かけの等級と比べると、2等級ほど暗くなることが分かった。
- ペルセウス座流星群の流星の測光質量は、 $2.00 \times 10^{-3} \sim 5.00 \times 10^{-2}g$ 程度であることがわかった。

結論

- 流星の輝度変化の傾向は、ペルセウス座流星群は山型、オリオン座流星群は平型であると考えられる。
- ペルセウス座流星群の流星の測光質量は、 $2.00 \times 10^{-3} \sim 5.00 \times 10^{-2}g$ 程度であることがわかった。

今後の課題

- 次の撮影の際には、流星の発光時間が測定できる方法で撮影を行い、流星物質の密度を求める。
- データの信憑性、正確性を上げるために、一枚でも多くの流星画像からデータを測る。

参考文献 流星II 解析と理論 斉藤馨治・長沢工 (1984)

正多面体に張るシャボン膜の最小面積

滋賀県立米原高等学校 数学班 藪内駿 谷本晃太 丸本啓樹 三浦康平

動機

シャボン液に立体の骨組みをつけたとき、その内部に膜が張る。我々はこの膜が自身の面積の最小値を取ろうとするとされていることに興味をもった。本当に膜の面積を最小にするように張るのか調べるため、膜の面積を数学的視点から考察した。

方法

1. ゼムツールをシャボン液※につける
2. ゼムツールに張った膜を観察する
3. 膜の面積を求める

※シャボン液(台所用洗剤:洗濯のり:水=1:5:10)

結果

正四面体、正六面体、正八面体に張った膜の様子は図1のとおりである。

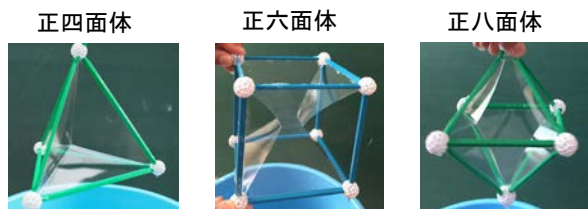


図1.各多面体に張った膜の様子

正六面体、正八面体においては内部に平面が見られた。変数 x を用いて、 x の値による面積の変化を調べた。以下の2点を条件とする。

- ①内部の膜には表面張力が働いているため、膜は曲面となる。今回は数学的に考察するため表面張力の影響を考慮せず、膜が多角形の集合であるとして考える。
- ②計算を簡略化するため各正多面体の一辺を10とする。

- 正四面体
膜が内部の1点に集まった。
膜の面積は $75\sqrt{2}$ となった。

• 正六面体

内部の平面を1辺が $x(0 \leq x \leq 10)$ の正方形とした。膜の面積を S_1 とすると、以下のように数式化できた。

$$S_1 = x^2 + 10\sqrt{2}(10 - x) + 2(x + 10)\sqrt{x^2 - 20x + 200}$$

$x \doteq 0.73$ のとき最小値をとった。

• 正八面体

内部の平面をひし形とし、その短い対角線を $x(0 \leq x \leq 10)$ とおいた。その膜の面積を S_2 とすると、以下のように数式化できた。

$$S_2 = 10\sqrt{3x^2 - 40x + 400} + 5(\sqrt{2} - 1)x + 50\sqrt{3} + 50$$

$x \doteq 5.53$ のとき最小値をとった。

考察

正四面体の様子から正六面体においても膜が1点に集まる、つまり $x=0$ になると予想した。しかし、 $x \doteq 0.73$ となり内部に平面ができるときの方が、面積が小さくなることが分かった。

よって、正六面体の内部に平面ができるのは、膜の面積が最小値をとるように張るためだと考えられる。しかし、実際に張った膜は $x = 0.73$ ではなかった。

また、正八面体に張った膜の面積の最小値は約313だったが、膜の面積が300になる張り方を発見した。

以上の結果から、正多面体に張る膜の面積は最小値をとるわけではないということが分かった。

今後の課題

正十二面体、正二十面体についても同様の実験を行う。物理学的にも考察を試みる。

参考

平成24年度滋賀県立米原高等学校課題研究数学班
NGKサイエンスサイト 立方体で作ったシャボン膜

<http://site.ngk.co.jp/lab/no58/>

JavaScriptを用いたソフトウェアの製作

滋賀県立米原高等学校 情報班 北川 発人 寺本 大弥 那須 亮太 清水 大地

動機

JavaScriptというプログラミング言語に興味を持った。
JavaScriptは、Webブラウザさえあればどのような環境でも動作する。記述が簡単で扱いやすく、最近の要素追加により多機能になった。それらが活かせるものとして「ゲームブック」を制作することにした。
※「ゲームブック」とは、選択肢がついた紙芝居のようなもので、文章(ストーリー)とステータス、いくつかの選択肢から成る。

実装した機能

ゲームブックらしい配置と機能を目指した
•メインメニュー、画像、文章、選択肢メニューの4つすべてを iframe で配置し実装した。(図1)

•一部処理の独立化

jsファイルとして分岐、場面送り、文字表示の処理部分を独立させた

•セーブ/ロード機能、確実な分岐処理の実装

進捗の保存を localStorage へ行い、分岐処理には 入力→確定 の2段階を踏むことで、選択の確認ができるように sessionStorage を利用した。

•「GitHub」の活用

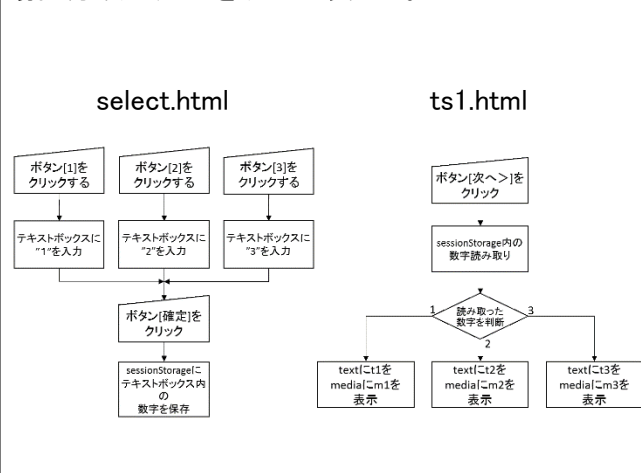
これはソフトウェア開発プロジェクトのための共有ウェブサービスである。ソースコードを管理し、Github Pagesという機能を用いてWebサイトとして実行可能にした。



図1

フローチャート

場面分岐の処理を以下に表した。



コード

選択肢

ボタン[1]

```
<button class="button" value="1">1</button>
```

ボタン[確定]

```
<button class="button" onClick="selv();">確定</button>
```

行き先・分岐先指定

```
var t1 = "https://mps2016js.github.io/text/t11a.html";
```

```
m1 = "https://mps2016js.github.io/media/m11a.html";
```

分岐処理

```
var s = sessionStorage.sel;
```

```
if (s == "1"){
```

```
parent.text.location.href= t1;
```

```
parent.media.location.href= m1;
```

```
}
```

保存

```
function psave(){
```

```
var t = parent.text.location;
```

```
var m = parent.media.location;
```

```
localStorage.tprg= t;
```

```
localStorage.mprg= m;
```

```
alert("セーブしました")
```

```
}
```

結果

今回の研究より、iframe、local・session StorageといったHTML5での追加命令を利用することで、どのような環境でも動作する進行状況の保存や分岐処理、表示を実装できた。

中間発表時は特定のWebブラウザで動作しなかったが、以降の修正によって正常に動作するようになった。テレビやほとんどのスマートフォンでも問題なく動作した。

問題点

•横長画面向けのレイアウトしか作れなかった。

•一部の携帯電話、タブレット端末では表示が崩れてしまった。

今後の課題

•画面の縦横比に合わせたメニューなどの移動

•特殊な環境を再現してデバッグを行う。

•音声、映像を用いた視覚、聴覚的な要素を取り入れる

•さいころなどランダムな選択肢を用いた分岐処理の研究

プログラム公開URL

mps2016js.github.io

ソースコード

<https://github.com/mps2016js/mps2016js.github.io>