

Asymptote によるアニメーション画像の作成とその活用

埼玉大学大学院理工学研究科 中川 幸一¹

1 はじめに

文科省発表の「GIGA スクール構想の実現に向けた ICT 環境整備 (端末) の進捗状況について (確定値)」[1]によると, 全自治体等 (1,812 自治体) のうち 1,748 自治体等 (96.5%) が令和 2 年度内に納品を完了する見込みであると回答されており, 2021 年度はまさに ICT 教育元年となるであろう.

今までは, (L^AT_EX などを用いて) 作成してきた (図入りの) PDF 文章の大半は, 印刷のための PDF で十分であった. そのため, 図は静止画として描くだけで十分であった.

しかしこれからの時代を考慮すると PDF 文章は, 電子文章のための PDF として作成する必要が出てくる可能性が大いにある. ならば PDF 内で扱う図も, 静止画だけでなくアニメーション画像も埋め込んでいくことが望ましいと思われる.

2 Asymptote とは



：自然な座標系で記述する (強力な) ベクターグラフィック言語 ²

Asymptote には以下のような特徴がある.

- 単なるグラフィック言語ではなく, プログラミング言語という側面もある
- 高品質の PostScript, PDF, SVG, WebGL³, および PRC⁴ ベクターグラフィックスなどを生成
- 主要なプラットフォーム (UNIX, MacOS, Microsoft Windows) で実行可能
- 3D ベクター画像の WebGL グラフィックスを生成して, HTML ファイル内に埋め込める
- 3D ベクター画像の PRC グラフィックスを生成して, PDF ファイルに埋め込める
- MetaPost にインスパイアされており, はるかにクリーンで強力な C++ のようなプログラミング構文

¹E-mail: k-nakagawa@h6.dion.ne.jp

²ロゴは <https://asymptote.sourceforge.io/gallery/> から引用

³互換性のある任意のウェブブラウザ上で, プラグインを使用せずにインタラクティブな 2 次元および 3 次元のコンピュータグラフィックスをレンダリングするための JavaScript API

⁴ISO 標準のファイルフォーマット. 仏 TTF 社 (2006 年にアドビが買収) が開発した 3D フォーマット.

3 アニメーションの仕組みについて

アニメーション（画像）とは、画像の位置・形などを少しずつ連続的に変化させ、複数の静止画を一コマずつ写した映像手法のことである。これが動いているように見える理由は、ファイ現象と呼ばれる、静止画の連続によって引き起こされる仮想の運動を知覚する現象によるものであり、具体例としては、パラパラ漫画⁵や GIF アニメーション⁶等が挙げられる。

Asymptote のアニメーションには `animate` モジュールを用いて作る。構造のすべては \LaTeX の `animate.sty` オプションを受け入れている。

4 Asymptote によるアニメーション画像の作り方

4.1 \LaTeX 側

\LaTeX のソース内についての説明をする。

\LaTeX の処理系（エンジン）の選択として、インライン PDF アニメーションを行うには「PDF ファイルを直接出力する」タイプの処理系で行う必要がある。具体的には

- `pdf \LaTeX / Xe \LaTeX / Lua \LaTeX`
- `pdf \TeX / Lua \TeX`

などである。

パッケージとして以下の 2 つを入れる。

- `\usepackage[inline]{asymptote}`
- `\usepackage{animate}`

そして、(アニメーションなどの) 図を入れたい場所で、環境内に Asymptote のコード (`\begin{asy} ... \end{asy}`) を書く。

4.2 Asymptote 側

Asymptote のコード内についての説明をする。アニメーションを作成するためには、大きく分けて冒頭部分、パラパラ漫画部分、最後の部分の 3 つを書く必要がある。

冒頭部分には

```
import animate;  
animation A;
```

とモジュールの宣言を行い、アニメーションを行うための構造体の宣言もしておく。

次にパラパラ漫画部分であるが `picture pic;` を宣言してから

⁵少しずつ異なる絵が描かれたページを次々にめくることで、残像によって絵が動いて見えるもの。

⁶GIF のマルチイメージと呼ばれる複数のフレームを順に表示できる機能を使ったアニメーション。

```

for(int i=0; i < 枚数; ++i) {
    (i によるパラメーターで1枚ずつの図を描く)
    A.add(pic);
}

```

というように、パラメーターによって変化していく図を描き、これらを順次1枚ずつ animation の構造体で宣言した変数のところに追加していく。

最後の部分では以下のようにアニメーションアイコンを表示したり、ディレイの設定などを行う。

```
label(A.pdf((オプション, )keep=!settings.inlinetex));
```

5 アニメーションの具体例（モンキーハンティング）

```

\begin{asy}
    usepackage("xcolor");
    usepackage("scsnowman");
    import animate;
    animation Anim=animation("movie3");

    for(int i=0; i < 20; ++i) {
        picture snowman;
        label(snowman,rotate(-(i/20)*360)*"\scsnowman[
            hat=red!75!black,
            muffler=green!75!black,
            arms=brown,
            snow=blue!80!green!40,
            buttons=blue!80!green!75!black,
            nose=orange,
            scale=5]",(0,-10*(i - 10)^2/10 + 100));
        label(snowman,rotate((i/20)*360)*"\scsnowman[
            hat=red!75!black,
            muffler=green!75!black,
            arms=brown,
            snow=blue!80!green!40,
            buttons=blue!80!green!75!black,
            nose=orange,
            scale=5]",(300,-10*(i - 10)^2/10 + 100));
        Anim.add(snowman);
    }
    label(Anim.pdf("autoplay,loop",delay=1000/20,keep=!settings.inlinetex));
\end{asy}

```

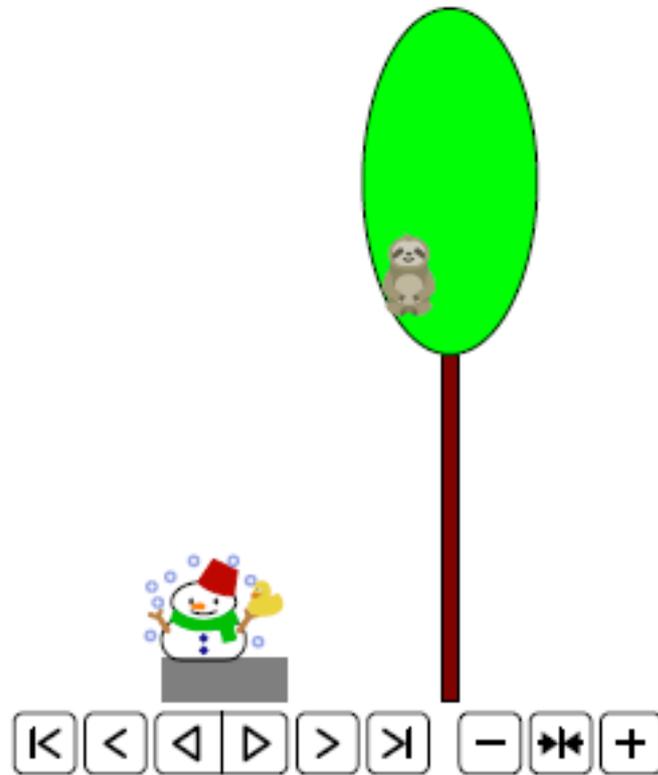


図 1: モンキーハンティング

具体的なアニメーションは <http://mathsalon.com/MonkeyHunting.pdf> を参照してください。

6 まとめと今後の展望

数ある図を作成する方法の中で、Asymptote によるアニメーション画像の作成のチュートリアルを行った。今回は PDF にアニメーション画像を埋め込む方法を紹介した。今後の展望としては、WebGL によるアニメーションの実現可能性について検討したいと思っている。

参考文献

- [1] 文部科学省『GIGA スクール構想の実現に向けた ICT 環境整備 (端末) の進捗状況について (確定値)』, https://www.mext.go.jp/content/20210518-mxt_jogai01-000009827_001.pdf