



一般社団法人

# 国際数理科学協会会報

No.98/2016.4

編集委員：藤井淳一（委員長）

目次

\*追悼文

\*総会議事録

\*理事会議事録

\*貸借対照表

\*決算予算表

\*寄稿

## 追悼文

一般社団法人 国際数理科学協会  
代表理事 植松 康祐

2016年1月、大阪府立大学名誉教授・佐藤優子先生がご逝去されました。

佐藤優子先生は、これまで長年に渡り、国際数理科学協会の理事を勤められ、本協会に多大なるご尽力を頂きました。この追悼文を書くにあたり、私は佐藤優子先生の研究分野や教育現場でのことを詳しく存じ上げておりませんでした。ネット上からの情報を中心に書かせて頂いたために、佐藤優子先生のご経歴を良くご存じの先生方には、不十分と思われるかも知れませんが、ご容赦頂ければ幸いです。

佐藤優子先生は、九州大学理学部数学科、更に大学院にご進学され、理学博士の学位を取得されました。九州大学での助手の後、大阪府立大学、助手・講師・助教授・教授とご昇進され、長年に渡り大阪府立大学での教育と研究に邁進されました。大阪府立大学評議員、大阪府教育審議委員などの要職にも就かれました。

佐藤優子先生が所属されていた学会は、日本数学会、アメリカ数学会、数理生物学会、日本人工知能学会、組合せ論とその応用協会、統計科学研究会、日本数理科学協会（現国際数理科学協会）と幅広い分野に跨っておられました。また、日本数学会では評議員、統計科学研究会では監事、国際数理科学協会では理事とそれぞれの学会の運営と発展にも貢献されました。

ここで、大阪府立大学ご退官直前の研究業績の一部をご紹介します。

1. J.Uemura, M.Sato 「Learning of erasing primitive formal systems from positive examples」  
Theoretical Computer Science 364(1) pp.98–114, 2006 年
2. S.Hamano, M.Sato 「Semantic Analysis of Association Rules via Item Response Theory」  
Lecture Notes in Artificial Intelligence 3587 pp.641–650, 2005 年
3. Y.Mukouchi, M.Sato 「Properties of SH Systems and Their Languages」  
Tamsui Oxford Journal of Mathematical Sciences 20(2), 2004 年

また、退官直前に採択された科学研究費のテーマの一部をご紹介します。

1. 「正規パターン上の決定木及び正規基本形式体系の帰納学習に関する研究」
2. 「正規パターン上の決定木の帰納学習とゲノム情報科学への応用」
3. 「推論による知識発見に関する研究」
4. 「擬微分作用素の構造とその偏微分方程式への応用」
5. 「格子経路組合せ理論の研究」

上記の研究業績から、佐藤優子先生が晩年目指されていた研究目標を知ることができると思われます。2008年3月20日の最終講義では、カタラン数について数学的性質に源氏香之図などのお話を織り交ぜて行われたそうです。カタラン数 (Catalan number) は、ベルギーの数学者 E.C.Catalan(1814-1894) によって定義された  $C_n = \frac{2n}{n+1}C_{n-1}$  です。彼は、 $n$  個の数値  $a_1, a_2, \dots, a_n$  の積の計算  $a_1 a_2 \dots a_n$  を2つの積の計算の繰り返しで行うと、何通りの計算方法があるのかという問題から、この数に到達したと言われています。カタラン数は、不思議な数で、 $n = 2^k - 1$  (メルセンヌ数) のときのみ奇数となり、それ以外は偶数となるという性質を持っています。この数からお話しを始められたのは、佐藤優子先生がこれまで研究されてきた random walk と密接に関係しているものと推察できます。原点を出発した random walk が  $2m$  ステップで初めて原点に戻る lattice path (最短折線) の総数が、 $2C_{m-1}$  のカタラン数で表されることに関係しています。また、源氏香とは室町時代に確立された香道の一つである「組香」から生まれたとされています。源氏香之図とは、5回の香りを聞いた後、縦に5本の線を引き同じ香であったものを横線で結ぶという図のことです。このルールでは、全部で52通りのつなぎ方があり、源氏物語の全54帖の桐壺と夢の浮橋の2帖を除く52帖の巻の名前が付けられているものです。



紅葉賀



朝顔



蜻蛉

このような模様が52種類作られています。源氏香之図は、家紋として使用され、着物や重箱などの紋様としても活用されています。佐藤優子先生が最終講義で、カタラン数とこの源氏香之図をどのように結び付けてお話しになったのかは分かりませんが、非常に興味深いお話であったに違いないと思われます。

私が佐藤優子先生にお目にかかったのは15年ぐらい前の事で、堺東での協会事務所(移転前)でのことでした。その当時、私は、この協会の歴史も人脈も全く知らないままで、幹事の役を仰せ付けられました。佐藤優子先生は、温和で優しく私に色々なことをお教え下さいました。その時は、事務所の運営についてしかお話しをしませんでしたが、今となれば、もっと御専門のことをお伺い出来たらよかったです悔やまれます。佐藤優子先生のご冥福をお祈り申し上げます。

# 社員総会議事録

2016年3月27日

場所：大阪教育大学 天王寺キャンパス 本館 306 号室

時間：14：00～15：00

議長：代表理事 植松康祐

出席者（代議員）：植松康祐、藤井淳一、富永雅、石井博昭、 田悦生、地道正行、菊田健作、会沢成彦、北條仁志、（顧問）藤井正俊

総社員数 28 名、出席者数 28 名（委任状含め）、総会は成立。

## 議題

### 1. 報告事項

#### (1) 決算関連

- ① 年会費の値上げにより会費は、増収となりました。
- ② 海外書籍取次業者（EBSCO）から予想外に注文がありました。
- ③ 電子書籍導入により印刷費を引き下げることができるようになってきました。最終的には、年間 50 万円以下を目指しています。（以前は 100 万円超でした）
- ④ 事務所移転により、家賃がかなり削減できるようになりました。旧事務所より 40 万円から 50 万円程度削減できています。

#### (2) 協会活動関連

- ① 2015 年 3 月 14 日大阪国際大学にて、第 1 回国際数理科学シンポジウムが行われました。11 件の研究発表と 1 件の特別講演となりました。
- ② SCMJ 誌の発行については、既に、Vol.78-1.2.3 を発行しました。2015 年度中別冊 1 冊（FIM）を予定していましたが別冊発行が 2016 年度にずれ込みました。
- ③ 投稿総数、国内・海外、採択率の報告
- ④ 2015 年度新規会員 6 名となりました。
- ⑤ 旧事務所所在地ビルオーナー（株式会社瓦町ビル）と保証金返還を巡って簡易訴訟を行い、当協会の主張が全面的に受け入れられた和解となりました。
- ⑥ 投稿募集と新会員募集を前面に出したホーム・ページに改めるべく作成中です。

### 2. 審議事項

#### (1) 平成 28 年度事業計画

- ・ 代表理事に関しまして、植松康祐先生（大阪国際大学）が二年の任期満了となりますが、引き続き二年間の代表理事をお引受けいただきました。
- ・ 代議員選挙を行い、従来より継続の方、28 人の新代議員が選任されました。さらに、投票があった会員で 2 人新代議員を予定しており、本人の応諾により新代議員と致したいと思います。
- ・ 長年当協会の理事をされ当協会の発展に貢献された佐藤優子先生（大阪府立大学名誉教授）が、2 月にご逝去されました。追悼号発行を協会で検討中です。

- ・過去の SCMJ 掲載論文の PDF 化の希望が寄せられた。
- ・今年度は、SCMJ・Vol.79-1.2.3 の三冊の発行を予定する。
- ・在庫として保有している SCMJ 雑誌に関しては、5～10 冊を残して全て廃棄し、保管コスト・事務の削減を行う。
- ・交換雑誌についても、ウェブ環境が整備され従来ほどの重要性はないため、基本的に一部を残し、打ち切りをすすめる。
- ・有名無実化している海外会員については、本人に会員としても継続意思を確認の後、継続希望でない者に対しては、PW を利用して HP による論文閲覧を不可能にする。

#### (2) 平成 28 年度予算

- ・米国債売却後の余裕資金約 700 万円は、引き続き積極運用しないで様子を見ることになりました。
- ・新 HP 完成後、SE 費用の削減のため、辻本氏と月額経費の削減交渉をする。
- ・長年にわたり当協会に貢献いただきました佐藤俊輔先生、田畑吉雄先生、寺岡義伸先生、高橋渉先生 の四名を新たに名誉会員と決定いたしました。

以上

---

## 理事会議事録

総会終了後、ただちに理事会が開催された。理事 10 名（委任状を含む）、監事 1 名が出席されたので、理事会は成立し、総会での決定事項は全て全員の賛成が得られ承認された。

以下に財務諸表（2016 年 3 月 23 日付監事会沢成彦監査済）を掲載する（2015 年度貸借対照表、及び 2015 年度決算予算表）

2015年度 貸借対照表  
(15/1/1-15/12/31)

(¥)会計

借方			貸方		
科目	期首	期末	科目	期首	期末
固定資産(保証金)	1,077,615	0	協会活動予備資金		
流動資産	1,741,906	1,807,060	出版基盤強化積立金	500,000	500,000
(定期預金)	0	0	TOTAL INDEX 積立金	414,993	414,993
(普通預金)	1,741,906	1,807,060		0	
(現金)	0	0	IT機器積立金		
安全資産ファンド	6,869,024	6,872,879	事務所移転積立金	1,077,615	0
			事務機購入積立金	0	
			減価償却積立金	0	
			回転資金	0	
			繰越金	7,695,937	7,764,946
合計	9,688,545	8,679,939	合計	9,688,545	8,679,939

外貨会計

借方			貸方		
科目	期首	期末	科目	期首	期末
固定資産			協会活動予備資金	\$52,297.86	\$47,499.65
流動資産			IT機器積立金	\$0.00	
定期預金(★)	\$1,069.89	\$1,070.01	\$-¥準備金		
普通預金(★)	\$51,227.97	\$46,429.64	繰越金	\$0.00	
\$国債2(★)	\$0.00	\$0.00	合計 \$	\$52,297.86	\$47,499.65
合計 \$	\$52,297.86	\$47,499.65			
(ユーロ)(★)	€ 2,074.40	€ 2,074.40	(ユーロ)	€ 2,074.40	€ 2,074.40
¥マルチマネー	11,667,694	10,515,782	¥マルチマネー	¥11,667,694	10,515,782
¥普通預金	184,836	1,234,840	¥普通預金	¥184,836	1,234,840

数理科学推進基金会計

借方			貸方		
科目	期首	期末	科目	期首	期末
清水基金	1,000,000	1,000,000	ISMS受賞基金	1,000,000	1,000,000
功力基金	100,000	100,000	国際研究交流基金	1,737,510	1,737,510
石原	2,000,000	2,000,000	通信費	0	0
その他	539,580	538,580	交通費	0	0
			繰越金	902,070	902,070
合計	3,639,580	3,638,580	合計	3,639,580	3,639,580

★印は、為替相場変動リスクあり

\* 2015年度決算予算表  
(2015年/1/1-14/12/31)

<b>収入</b>			
科 目	15年度実績	15年度予算	16年度予算
前年度繰越金	-		
	-		
刊行物頒布代(書店)	222,300	650,000	420,000
刊行物頒布代(書店)海外\$より	1,406,199	800,000	800,000
会費			
機関会員 A(旧協力校)	0		
機関会員 B(交換誌)	-	0	-
賛助会員(機関会員)	397,676	470,000	380,000
正会員(国内)	743,800	670,000	750,000
ページチャージ・別刷(¥)	79,368	100,000	100,000
ページチャージ(\$→¥)			-
事務所解約保証金(特別収入項目)	183,733	200,000	-
設備更新積立金			
(イ)減価償却積立金取り崩し分			
(ロ)回転資金取り崩し分			
(ハ)事務機購入積立金取り崩し分			
預金利子	-		
(\$→¥:調整項目)	1,542,185	1,318,000	1,620,000
雑収入	103,515		-
合 計	4,678,776	4,208,000	4,070,000
<b>支出</b>			
科 目	15年度実績	15年度予算	16年度予算
通信交通輸送費(イ+ロ+ハ)	162,003	250,000	150,000
(イ)編集通信交通費	-	50,000	-
(ロ)査読通信費			
(ハ)抜刷等輸送費	162,003	200,000	150,000
講演依頼料		10,000	
租税公課	47,950		-
印刷費	750,060	850,000	500,000
組版委託費・書籍整理費	200,000	120,000	180,000
SE委託費(辻本氏)	434,760	350,000	450,000
消耗品代	4,646	30,000	10,000
備品代(OA機器soft,本代,rental server)	66,859		50,000
人件費	1,924,812	1,600,000	1,700,000
借事務所代	813,072	804,000	810,000
電話代	70,334	50,000	70,000
振込料・手数料	12,088	10,000	10,000
電気代	28,951	30,000	30,000
	-	0	
保険料	17,241	4,000	10,000
税金	70,000	70,000	70,000
会費(学術団体)	30,000	30,000	30,000
雑費	46,000		
合 計	4,678,776	4,208,000	4,070,000

\* 寄稿

## TeX 用描画ライブラリ TikZ のすすめ

藤井 淳一 (大阪教育大学 教養学科 情報科学講座)

### はじめに

私は Window のマシンで Word より TeX を使うことが多く、DOS の plainTeX の時代から作ってきた大量のノート類を TeX ファイルとして溜め込んできました。初期のころは、LaTeX と云えども picture 環境で描ける図は非常に限定的でした。今は少しましにはなりましたが、それでも LaTeX で図を描くのは多少ハードルが高めで、支援ソフトがないと難しかったりします。正確な関数計算を必要とする描画では GNUplot が定番ですが、それ以外のラフな描画では私は WinTPIC というフリーソフトやその研究室改良版 FKtpic (指導院生の作品<sup>1)</sup>) を使っていました。これらは、プレビューアとして重要だった DVIOUT もサポートしている TPIC という仕組みを利用したシンプルなもの、ある程度手入力で描けたり修正したりできるものでした。とはいえ絶対座標を使って描画し、変数は基本使えないので、拡大縮小回転など座標計算が必要なものは前述のような支援ソフトがないと苦しくなります。

そこで、Xy-pic と呼ばれる優れたライブラリも併用していましたが、DVIOUT と PDF での差が気になるので、結局あまり使うこともなく、FKtpic を使うことがほとんどでした。もともと曲線において座標値が膨大になる WinTPIC を改良して Bezier 曲線を利用して座標値を減らしたもののなのですが、それでも Xy-pic などよりはどうしても多くなります。私は普段から大量の TeX ファイルを扱うので、\input 等で外部ファイルを使うことなく、できれば一つのファイルに収めたい主義で、これは気になっていました。

表題の TikZ も目にとまり、気になる存在ではあったのですが、DVIOUT では表示できず、Xy-pic より困った状況で、さらに Web ページで挙がっていたり解説やマニュアルに載っているような例示ファイルが走らないことがしばしばでしたので、二の足を踏んでいました。そうこうしているうちに、ひどかった pdfLaTeX もちゃんと走るようになり、それが前提の TeX エディタ texworks も普及するにつれ、PDF 主体になってきて、DVIOUT の必要性がかなり薄れました。

TikZ は TeX プレゼンテーションで今や世界標準となった Beamer と同じくドイツ・リューベック大学の Till Tantau の作で、“TikZ ist kein Zeichenprogramm” (TikZ はただの描画プログラムとは違うよ) から名づけられています<sup>2)</sup>。Beamer と同様、ベースとなるグラフィックライブラリ PGF に基づいて作成されています。とはいえ、日本での Windows マシンでの通常の TeX インストールでは、[8]にある作者たちの膨大な作品集は、そのまま走るという意味ではほぼ全滅だと思いますので、その直し方も含めて簡単な紹介をしたいと思います。TikZ の簡単な解説は [5]にも載っていますし、HP 上にもいくつかの解説 [3, 6, 10] やマニュアル [7, 1] があります。

### 1. インストールと描画の基本

正確には切り替えの時期がわかりかねますが、5、6年前以降に新しい TeX をインストールされた方であれば、自動的に入っています。もしなければ、公開されているライブラリ [9] から DL して解凍

<sup>1)</sup> <http://ex.osaka-kyoiku.ac.jp/~fujii/dl/DL.html> で公開しています。

<sup>2)</sup> 計算機に詳しい方ならば、Richard Stallman 創設の (GNUplot も含まれる) フリーソフト総体 GNU = GNU is not UNIX (GNU はただの UNIX とは違うよ) を思い出し、その真似をしているのに気づかれたでしょう。

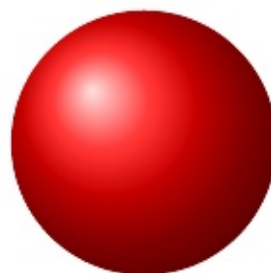
すれば、ディレクトリの構造ごと解凍されますので、 $\text{T}_{\text{E}}\text{X}$ が入っているところに移すだけでインストール完了です（ただし、`xcolor`が入っていない場合にはスタイルファイルが必要です）。

次の宣言をしさえすれば、実際の命令は `\tikz` の後で1文なら実行可能で、多くの処理をしたいときは `tikzpicture` 環境 (`\begin{tikzpicture}` と `\end{tikzpicture}` の間) の中に命令文を書きます。Window の標準的な  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  で使うときは、次の宣言文を `\documentclass` もしくは `\documentstyle` の宣言の直後に入れてください。

```
\usepackage[dvipdfm]{graphicx,color}
\usepackage{tikz}
```

その設定で、本文 (document 環境内) に記述します。例えば、Beamer の列挙アイテムにも使われている立体的なボールも、短い文章で描けます：

```
\documentclass{article}
\usepackage[dvipdfm]{graphicx,color}
\usepackage{tikz}
\begin{document}
\tikz\shade[ball color=red] (0,0) circle (1);
\end{document}
```



`shade` はグラデーションの命令で、色と、中心と半径を指定しているだけです。デフォルトの単位は `cm` です。

線を引くのは本文内に

```
\tikz\draw(0,0)--(1,0)--(.7,.5)--cycle;
```

と書くと、3角形が得られます。`cycle` は折れ線を閉じる命令で、上の例では座標 (0,0) を意味します。長さの基本単位は `cm` で、個別の数値にも単位がつけられます。冒頭部分で縮尺も設定できるので、デフォルトは `\tikz` が `\tikz[x=1cm,y=1cm]` と同等です。線の太さや色・縮尺を変えたり、頂点に記号を付けたりするには

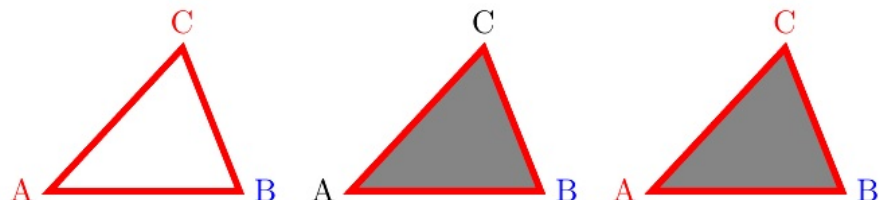
```
\tikz[line width=2pt, red, x=2cm,y=3cm]\draw(0,0)node[left]{A}
--(1,0)node[right]{\color{blue}B}--(.7,.5)node[above]{C}--cycle;
```

と書き込むだけで、三角形 ABC が書けるのは、ほぼ説明なしでもわかるでしょう。基本赤色で、Bのみ青で描画されます。`node` 命令は該当の点の [] の位置に {} の中の文字列を表示します。これはあなた任せの設定です。(縮尺の影響を受ける) 絶対座標で書きたいときは、`node at(-0.2,0){A}` のように `at` を付けた座標で記述します。また、これは全体の色を冒頭で指定しているので、文字まで同じ色になってしまいます。文字は地の色 (たいてい黒) で線のみ色を変えたいとき、更には内部を違う色で塗りたいたいときもあるでしょう。その場合、`draw` 命令より上位の (曲線も描ける) `path` 命令によって、例えば、

```
\tikz[line width=2pt, x=2cm, y=3cm]\path[draw=red,fill=gray] (0,0)node[left]{A}
--(1,0)node[right]{\color{blue}B}--(.7,.5)node[above]{C}--cycle;
```



とすると、文字は地の色、周囲は赤、内部は灰色になります (Bのみは個別設定しているの青です)。大体、`\path[draw] = \draw` なのですが、細かい設定を加えると差が生じてきます。例えば上記の部分で `\path[ ]` の部分を、`\draw[red,fill=gray]` と書き換えると、文字まで赤になってしまいます。命令がダブっていますが、`\draw[draw=red,fill=gray]` と書いてやると、`path` で書いたのと同じ結果が得られます。以上の話に出てきた設定を加えた三角形の図を下に挙げます：



ここで、TikZで採用されている `xcolor[2]` の **standard color expression** の色表示方法について触れておきましょう。基本的な色は定義済みですが、細かい色は色定義をするか混ぜないといけません。`gray` は定義済みですが実は不要です：`gray = black!50`。後者の記法は「黒 50 % (白 50 %)」を表します。したがってもっと薄いグレーを作りたいければ、`black!20` などとすればいいわけです。これは白との混合になりますが、2色以上の混合も可能です。`red!20!yellow` (赤 20 %、黄 80 %) でオレンジっぽい色が指定できます。この指定は「加色混合」(RGBの光の混合)ではなく、「減色混合」(絵具と同様のCMYKによる混合)であるので感覚的に使いやすいですし、変数を使った色混合も容易にできるようになります。一般に、 $c_j$  を定義済みの色名、 $p_j$  をパーセント (0~100) とするとき、 $c_1!p_1!c_2!p_2!\dots!p_n!c_{n+1}$  という表現は

$$\frac{p_1 p_2 \dots p_n}{100^n} c_1 + \frac{(100 - p_1) p_2 \dots p_n}{100^n} c_2 + \dots + \frac{(100 - p_1) \dots (100 - p_{n-1}) p_n}{100^n} c_n + \frac{(100 - p_1) \dots (100 - p_n)}{100^n} c_{n+1}$$

という減色混合と解釈されます。 $c_{n+1}$  が白のときは省略可能です。さらに前にハイフンを付けると「補色」になります。

このように単純なものなら基本1行で書けますが、複雑な図形は `tikzpicture` 環境で書くことになります。命令文の全種類を解説することは不可能ですので、例示にとどめざるを得ませんが。

## 2. 例文の修正

[8] の例示ファイルは、いろいろな人が作っているので、前節の冒頭部分がほとんどちゃんと設定されていないため走りません。まずはこれを設定し直しましょう。[8] の `\documentclass` も `article` 等標準的なもの以外の設定がかなりあるため走りません。そこも注意してください。このあたりを直すと、かなりのファイルがうまく走るようになります。今年オリンピックの年ですので、「clipping」の中にある五輪描画の例を引いてみましょう (冒頭部分抜粋)：

```
% Olympic rings
% Author: Paul Gaborit
\documentclass[a4paper]{article}
\usepackage{tikz}
%%<
\usepackage{verbatim}
\usepackage[active,floats,tightpage]{preview}
\PreviewEnvironment{tikzpicture}
\setlength\PreviewBorder{5pt}%
\begin{document}
```

この例は比較的ましですが、それでも `\usepackage[dvipdfm]{graphicx,color}` が抜けているので走りません。まずそれを入れますが、レイアウトの設定ファイルが4行ほど邪魔をしてちゃんと出ません。comment 環境の部分があるのですが、それも不要です。それで、全体をすっきりさせたものは、

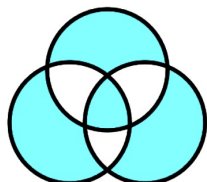
```
\documentclass[a4paper]{article}
\usepackage[dvipdfm]{graphicx,color}
\usepackage{tikz}
\begin{document}
\begin{tikzpicture}
\definecolor{r1}{RGB}{0,129,188}
\definecolor{r2}{RGB}{255,190,100}%%slightly revised
\definecolor{r3}{RGB}{35,34,35}
\definecolor{r4}{RGB}{0,157,87}
\definecolor{r5}{RGB}{238,50,78}
\begin{scope}
\clip (-6,2) rectangle (6,-.9);
\foreach \col/\xp/\yp in {r5/4/0, r4/2/-1.8, r3/0/0, r2/-2/-1.8, r1/-4/0}
{ \path[draw=white,line width=.08cm, fill=\col,even odd rule]
(\xp, \yp) circle (1.9cm) (\xp, \yp) circle (1.5cm); }
\end{scope}
\begin{scope}
\clip (-6,-.9) rectangle (6,-3.8);
\foreach \col/\xp/\yp in {r1/-4/0, r2/-2/-1.8, r3/0/0, r4/2/-1.8, r5/4/0 }
{ \path[draw=white,line width=.08cm, fill=\col,even odd rule]
(\xp, \yp) circle (1.9cm) (\xp, \yp) circle (1.5cm); }
\end{scope}
\end{tikzpicture}
\end{document}
```

で描けます。中身は命令を知らなくてもある程度想像可能です。ここまで凝る必要はないのですが、この作者は5輪の5色を微妙な色として `r1 ~ r5` まで define しています (RGB の光の混合)。



中身はかなり高度なテクニックを使っていますが、表現は単純です。輪っかが重なって向こう側にある部分がある程度描かない図になっています。各長方形部分を「clip」命令で限定し、その範囲が干渉しないよう scope 環境で制限しています。実は、上下にちょうど半分に分割してそれぞれ同じ5個ずつの

輪を描いているのが見て取れるでしょう。上の方は右から書くことで、右側にある輪が必ず左側の下に来るよう描かれています。下の方はその逆で、それら2つを合わせて5輪が描かれています。円周部分を白く細い線で描き、中身を指定の色で fill することで上記の重なりの実現しています。ここで重要なことは、ループとして「foreach 文」が使えることで、変数は  $\backslash x$  などのように表現します。上で使っているのは、色  $\backslash col$ 、輪の中心の  $x$  座標  $\backslash xp$ 、 $y$  座標  $\backslash yp$  です。多変数の場合は / で区切って使います。ここではその変数の領域をすべて列挙していますが、計算させることも可能です。残る疑問はおそらく even odd rule だけでしょう。下図の色の付き方がそれを示しています：



閉曲線に対して、線のない外側領域からはじめて線を横切つて図形内部に入るとき、「交差が奇数回ならその領域で色を付け、偶数回ならつけない」という規則のことを **even-odd rule** といいます。

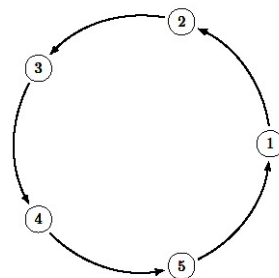
同心円の2つの円周を描いて fill するとき、挟まれた領域が奇数領域なので、両円周間のみ色がつくことになります。

foreach を使った例をもう一つ直しておきます。まさに HP の「foreach」にある Jerome Tremblay 氏の作品を書き換えたものです：

```

\documentclass{article}
\usepackage[dvipdfm]{graphicx, color}
\usepackage{tikz}
\begin{document}
\begin{tikzpicture}
\def \n {5}
\def \radius {3cm}
\def \margin {8}
\foreach \s in {1,...,\n}
{ \node[draw, circle] at ({360/\n * (\s - 1)}:\radius) {\s};
  \draw[line width=1pt, ->, >=latex] %%slightly modified
  ({360/\n * (\s - 1)+\margin}:\radius)
  arc ({360/\n * (\s - 1)+\margin}:{360/\n * (\s)-\margin}:\radius);
}
\end{tikzpicture}
\end{document}

```



領域に整数の範囲表示  $\{1, \dots, \backslash n\}$  を使っています（意味は分かるでしょう）。あと座標系に  $:$  を使っていますが、「角度 : (進む) 距離」という記法です（角度の単位はラジアンではありません）。後は説明不要かと思います。

もう一つだけ同じ HP から同様の例を手直ししてみます。同じく「clipping」にある Till Tantau の例に少し Kjell Magne Fauske 氏が手を入れたものを余計なものを省いて載せます：

```

\documentclass{article}
\usepackage[dvipdfm]{graphicx, color}
\usepackage{tikz}
\begin{document}

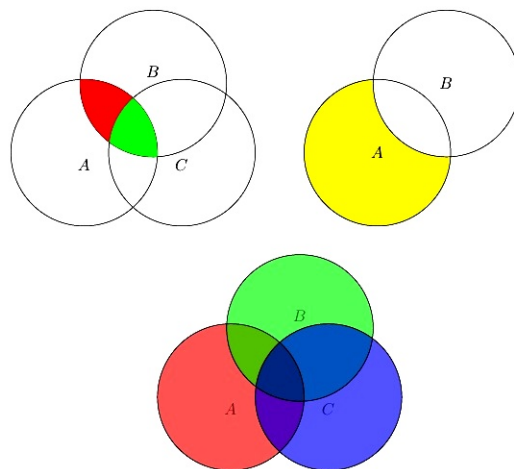
```

```

\def\firstcircle{(0,0) circle (1.5cm)}
\def\secondcircle{(45:2cm) circle (1.5cm)}
\def\thirdcircle{(0:2cm) circle (1.5cm)}
\begin{tikzpicture}
  \draw \firstcircle node[below] {$A$};
  \draw \secondcircle node [above] {$B$};
  \draw \thirdcircle node [below] {$C$};
  \begin{scope} \clip \firstcircle;
    \fill[red] \secondcircle; \end{scope}
  \begin{scope} \clip \firstcircle; \clip \secondcircle;
    \fill[green] \thirdcircle; \end{scope}
  \begin{scope}[shift={(6cm,0cm)}]
    \begin{scope}[even odd rule]
      \clip \secondcircle (-3,-3) rectangle (3,3);
      \fill[yellow] \firstcircle; \end{scope}
    \draw \firstcircle node {$A$};
  \end{scope}
  \draw \secondcircle node {$B$}; \end{scope}
  \begin{scope}[shift={(3cm,-5cm)}, fill opacity=0.5]
    \fill[red] \firstcircle;
    \fill[green] \secondcircle;
    \fill[blue] \thirdcircle;
    \draw \firstcircle node[below] {$A$};
    \draw \secondcircle node [above] {$B$};
    \draw \thirdcircle node [below] {$C$}; \end{scope}
\end{tikzpicture}
\end{document}

```

出来上がった図は以下のものです：



### 3. その他の例

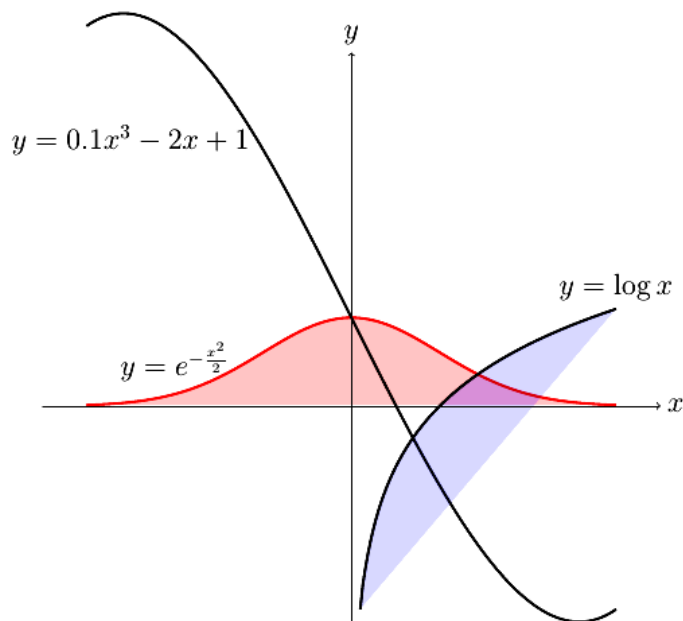
以後、前後の共通部分を除いて document 環境の中だけを挙げて、典型例をいくつか示します:

#### 例 1. 関数のグラフ

まずは関数のグラフです。plot 命令で描き、名前の付いた関数を利用するときは、 $\{\ln(\backslash x)\}$ などのように中括弧でくくることを忘れないでください。曲線は、実際には標本点をつないで折れ線で描いているので、十分すぎますが `samples=100` ぐらいで滑らかにしています (数を極端に減らせば折れ線になるのがわかります)。定義域は `domain` で指定し、これは個別にも指定出来ます (以下の対数グラフ参照)。`\draw[draw=red, fill=red!20]` とすれば、グラフは赤で、始点終点を結ぶ線分とグラフで囲まれた部分を薄い赤で塗ります。先に軸を書いていると消えますので書いていませんが、透明度を使うことで一応回避できます。ただし、全部透明度が影響しますので、対数は透明度指定 `opacity=0.2` で塗って、2度目に普通に描いています。

うれしいことに、簡単な関数や計算は用意されているので、すぐに描けます。もう少し複雑な曲線については、最後の例で述べたいと思います。

```
\begin{tikzpicture}[domain=-3:3, samples=100,
                    very thick,x=1.5cm,y=1.5cm,font=\Large]
\draw[draw=red, fill=red!20] plot(\x, {exp(-0.5*\x*\x)})
node at (-2,0.5) {$y=e^{-\frac{x^2}{2}}$};
\draw plot(\x, 0.1*\x*\x*\x-2*\x+1) node at (-2.5,3) {$y=0.1x^3-2x+1\quad$};
\draw [fill=blue!80,opacity=0.2,domain=0.1:3] plot(\x, {ln(\x)}) ;
\draw [domain=0.1:3] plot(\x, {ln(\x)}) node[above] {$y=\log x$};
\draw[thin,->] (-3.5,0)--(3.5,0) node[right] {$x$};
\draw[thin,->] (0,-2.5)--(0,4) node[above] {$y$};
\end{tikzpicture}
```



## 例 2. 図式

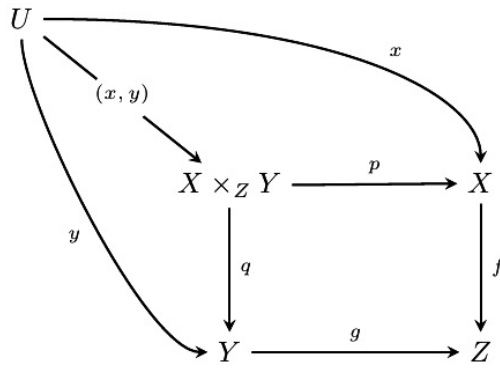
`matrix` のライブラリを使って可換図式を描いてみましょう<sup>3)</sup>。矢印はちょっと凝って `stealth` 型にしました (矢尻の変化を見てください。あと標準で、`latex` 型もありますが、`arrows` ライブラリ (cf. [7, p.173]) で様々な矢尻が使えます)。`matrix of math nodes` の宣言で、行列内の成分 `node` の文字列は数式モードになります。あと行列要素の位置の名前として `|U|` のように要素の値の前に設定でき、後で使えます。邪魔くさいので、一括で `node` の位置は `auto` に、フォントサイズは `scriptsize` に指定し、`path` も `thick` で矢尻付に指定しています (この時冒頭の矢尻指定が効いています) が、こちらの `node` は一括して数式にできないようですが、点名は単に `node(a){a}` と丸括弧で書けば設定できます (使ってないです)。

一部 3 次 Bezier 曲線を使っています。制御点  $a, b$  は、`..controls (a) and (b)..` で書きます (以下では相対的な位置指定をしています)。`draw` 命令で (実際は行列成分で途切れますが) つながった形の折れ線にすると、矢尻は最後にしか付きませんので、少し面倒ですが、`edge` で 2 点ずつ繋いでいます。特徴的なのは、`(U)` から `(XZY)` への矢印のオプションでしょう。`auto` モードでは線をよけて文字を描きますので、真上に書きたいときは `anchor=base` を指定します。後の命令は背景に白の円盤を少し大きめに描いているので、線が消えたように見えます。

```
\usetikzlibrary{matrix}
\begin{tikzpicture}[>=stealth]
\matrix [matrix of math nodes,row sep=1.5cm, column sep=1.5cm]{
|U| U & & [5mm] \\
& |(XZY)| X \times_Z Y & |(X)| X \\
& |(Y)| Y & |(Z)| Z \\
};
\tikzstyle{every node}=[auto,font=\scriptsize]
\tikzstyle{every path}=[thick,->]
\draw(U).. controls +(right:4cm) and +(up:1cm)..node {x} (X);
\draw(U).. controls +(up:-1cm) and +(left:1cm)..node[below left]{y} (Y)
(U)edge node[anchor=base,fill=white,inner sep=0.2pt,circle]{(x,y)}(XZY)
(XZY)edge node{q}(Y) (Y)edge node{g}(Z)
(XZY)edge node{p}(X) (X)edge node{f}(Z);
\end{tikzpicture}
```

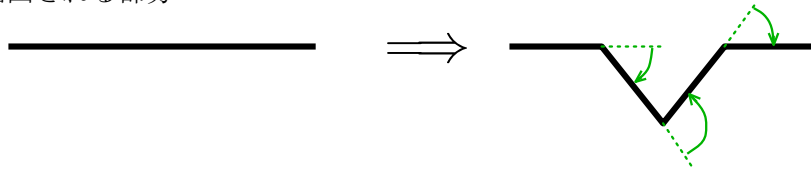
---

<sup>3)</sup> `Xy-pic` の例としてよく使われている図式ですが、似たものしかなかったので作成しました。`TikZ-cd` をインストールすれば、多少楽になりますが。



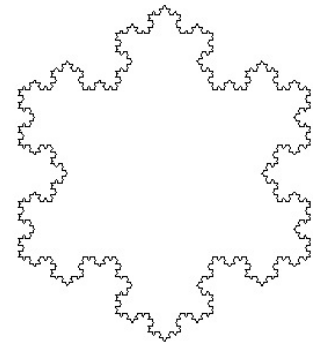
### 例 3. フラクタル (Koch 曲線)

フラクタルについては、専用の記法 **Lindenmayer Systems (L-systems)** をライブラリに採用しているので、これも簡単に描けます。`\rule{F -> F-F++F-F}` の意味するところは、自己相似の表現として、一つの線分  $F$  が「 $F$  右向  $F$  左倍向  $F$  右向  $F$ 」（向きの変更角度は 60 度に指定されている）という命令に従って描画される部分<sup>4)</sup>



に置き換えられるということで、L-system の詳細については、たとえば [4] を参照ください。元の形は `axiom=F++F++F` で指定されているように正三角形で、深さは `order=4` まで繰り返しています ([8] の Stefan Kottwitz 氏の作品から抜粋) :

```
\usetikzlibrary{lindenmeyersystems}
\usetikzlibrary[shadings]
\pgfdeclarelindenmeyersystem{Koch curve}
  {\rule{F -> F-F++F-F}}
\begin{tikzpicture}
\shadedraw[shading=color wheel]
[l-system={Koch curve, step=2pt,
  angle=60, axiom=F++F++F, order=4}]
lindenmayer system -- cycle;
\end{tikzpicture}
```



### 例 4. ツリー

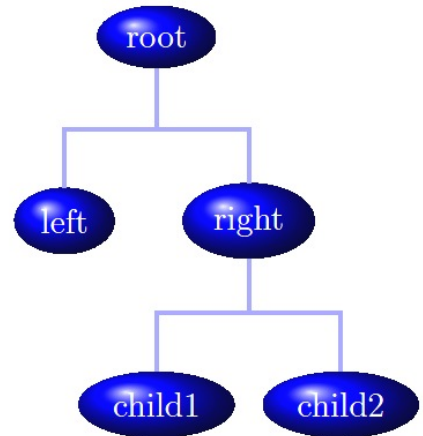
立体的なボールやツリーも簡単に書けるので、次のようなお遊びもできます。trees と shapes (楕円 shade 用) のライブラリを読み込んで `edge from parent fork down` のオプションを付けると、四角い枝のツリーになります。ツリーなので、child を入れ子にする形になっています。枝の長さは `scale` で調整できます ([7] の例を改編) :

<sup>4)</sup> 実は TikZ で下の図を書こうとしたのですが、TikZ 命令をかなり引用しているのでひどく影響して画面が乱れたので、この部分は FKTPic を使いました。

```

\usetikzlibrary{trees,shapes}
\begin{tikzpicture}
[edge from parent fork down,scale=1.5]
\tikzstyle{every node}
=[ball color=blue,ellipse,text=white]
\tikzstyle{edge from parent}=[draw,ultra thick,blue!:]
\node {root}child {node {left}} child {node {right}}
child {node {child1}} child {node {child2}};
\end{tikzpicture}

```



### 例 5. knot 処理

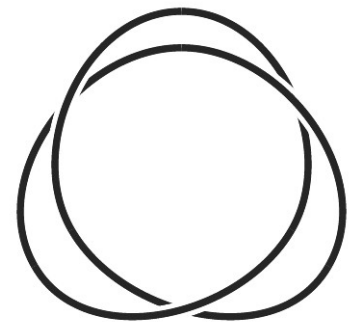
前節では 5 輪の LINK の例がありました。結び目を描く例も挙げておきます。同心の正三角形で分けられた 6 つの曲線を 120 度刻みの接線で曲線を描かせています。上に来るものは、まず太めの白で曲線を描いたのちに同じ曲線で描くマクロ `\DMAJ` によって、下にある線を分断するように見せています:

```

\def\dMAJ#1#2#3{\draw[white,line width=10pt,out=#1,in=#2] #3;
\draw[out=#1,in=#2] #3;}%%%}%%

\begin{tikzpicture}[x=1.5cm,y=1.5cm]
\tikzstyle{every path}=[line width=4pt]
\draw[out=180,in=120] (3,{sqrt(3)*12/7}) to (1,0);
\draw[out=60,in=0] (4.5,{sqrt(3)*2/7}) to (3,{sqrt(3)*2});
\draw[out=-60,in=-120] (1.5,{sqrt(3)*2/7}) to (5,0);
\dMAJ{60}{0}{(5,0) to (3,{sqrt(3)*12/7});}
\dMAJ{-60}{-120}{(1,0) to (4.5,{sqrt(3)*2/7});}
\dMAJ{180}{120}{(3,{sqrt(3)*2}) to (1.5,{sqrt(3)*2/7});}
\end{tikzpicture}

```



### 例 6. 交点処理

最後に `path`, `plot` で描かれる曲線の交点について例示しておきます。曲線が関数である場合は、交点は計算で出せないことはないですが、特に Bezier 曲線などの場合は交点を求めるのは面倒ですし、さらに範囲を絞って色付けするなど、計算のみではかなりきついです。TikZ では交点についてサポートしていますので、`\usetikzlibrary{intersections, calc}` でライブラリを読み込んだ上で、以下のよう  
に交点を拾わせて、それをもとに描画をしています:

`normal` と名付けた正規分布の密度関数まがいと、`bzier` と名付けた Bezier 曲線の 3 交点 (ここは要調整) を

```

\fill [blue, name intersections={of= normal and bezier, by={a, b, c}}]

```

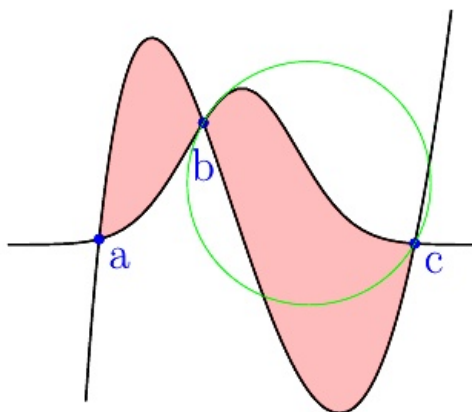
で拾って、左から `a`, `b`, `c` と名付けています (交点の個数は作者側で把握しておいてください)。この書き方はマニュアル [7] にすら載っていませんが、[10] に解説がある一番一般的な形式です。その各座標 `(a)`, `(b)`, `(c)` で青い円板を描いて交点を示しています (`[below=2mm]` で `b` のラベルをちょっと下に書いています)。

さらに、領域に色を付けるときは、まず塗る領域を閉曲線にするために余計な点を加えて全体を一



方向につなぐ必要があります。元の描画を生かすならば、関数の domain を 3:-3 のように前後逆にするのが楽でしょう。それで fill すれば、even odd rule をデフォルトとして塗ってくれます。後は点の計算ですが、\$記法で計算可能です, see [10]。平行移動については、( $(P)+(b,c)$ )によって、点 (P) をベクトル (a,b) で移動できますので、座標値を明示的に知ることなく色を塗る範囲指定をクリップする長方形領域を指定することができます：

```
\usetikzlibrary{intersections, calc}
\begin{tikzpicture}[domain=-3:3, samples=50]
\begin{scope}
\clip (-3,-3) rectangle (3,3);
\tikzstyle{every node}=[font=\LARGE]
\draw [thick,name path=normal] plot(\x, {2*exp(-\x*\x)});
\draw [thick,name path=bezier](-2,-2) .. controls (-1,13) and (1, -15) .. (3,6);
\fill [blue,name intersections={of= normal and bezier, by={a, b, c}}]
(a) circle (2pt) node [below right] at (a) {a}
(b) circle (2pt) node [below=2mm] at (b) {b}
(c) circle (2pt) node [below right] at (c) {c};
\end{scope}
\begin{scope}\clip ($(a)+(0,-3)$) rectangle ($(c)+(0,3.5)$); %%色塗り長方形
\fill[red,opacity=0.2] (-2,-2).. controls (-1,13) and (1, -15) .. (3,6)
--(3,{exp(-9)}) %%Bezier 曲線の両端に点を加える。
[domain=3:-3] plot(\x, {2*exp(-\x*\x)})--(-2,-2); %%函数逆方向描画
\end{scope}
\draw[green] let \p1 = ($(b) - (c)$),\n1 = {veclen(\x1,\y1)}
in circle [at=$(b)!0.5!(c)$],radius=\n1/2];
\end{tikzpicture}
```



長方形の左下と右上を求めるために、a,c から上下にずらして頂点を求めています。「色塗り長方形」の、a,c 部分をのどちらかを b とすれば、領域を変化させることができます。また、曲線によっては左右にずらした方がいい場合もあります。(以上のように座標値なしで変更できますが、もし座標値を知りたい場合は、点 (P) の x 座標は、下位言語の pgf を使って

```
\newdimen\X
\pgfextractx{\X}{\pgfpointanchor{P}{center}}
```

で、 $\X$  に設定することもできます。pgfextracty で  $y$  座標もとれます。）

また最後の 2 行についてですが、点に接する円を描きたいとき、中点等の内分点を取ったり、半径を求めたりする必要があるかもしれませんので、その操作について少し触れておきます。2 点  $a, c$  を 1 : 3 に内分する点は、次のいずれかで求められます：

$(\$0.75*(a)+0.25*(c)\$)$  または、  $(\$a!0.25!(c)\$)$

また、2 点間の距離は、draw 命令の中で let 文を使うと、変数として、点  $\text{p1}$ ，その座標  $\text{x1}, \text{y1}$ ，数値変数  $\text{n1}$  などが使えるようになりますので、点  $b, c$  で作られる差のベクトルの長さを veclen 関数で求めて 2 で割って半径を出し、2 点の中点を中心とした緑の円を描いています。

## おわりに

紹介は例示にとどまりましたが、一応 TikZ の特色は示せたのではないかと思います。例自体は大したものではありませんが、かなり深いところまで使えるように解説したつもりです。L<sup>A</sup>T<sub>E</sub>X で描画されるときに活用していただければ幸いです。

## 参考文献

- [1] J.Crémer, A very minimal introduction to TikZ, 2011  
<http://cremeronline.com/LaTeX/minimaltikz.pdf>
- [2] U.Kern, Extending L<sup>A</sup>T<sub>E</sub>X's color facilities: the xcolor package, 2007  
<http://mirrors.ctan.org/macros/latex/contrib/xcolor/xcolor.pdf>
- [3] Henjinonkan, TikZ 覚書  
<http://perikanfan.web.fc2.com/Manual.pdf>
- [4] G. Ochoa, An Introduction to Lindenmayer Systems  
<http://ldc.usb.ve/~gabro/lsys/lsys.html>
- [5] 奥村晴彦・黒木裕介, L<sup>A</sup>T<sub>E</sub>X<sub>2</sub>e 美文書作成入門 (改訂第 6 版), 技術評論社, 2015.
- [6] T.Soma, TikZ  
<http://www.opt.mist.i.u-tokyo.ac.jp/~tasuku/tikz.html>
- [7] T.Tantau, TikZ and PGF, 1.8, <http://www.bu.edu/math/files/2013/08/tikzpgfmanual.pdf>
- [8] T.Tantau, T<sub>E</sub>Xample.net  
<http://www.texample.net/tikz/examples/>
- [9] SourceForge: PGF and TikZ  
<https://sourceforge.net/projects/pgf/>
- [10] T<sub>E</sub>XWiki TikZ  
<https://texwiki.texjp.org/?TikZ>