Mathematicaを利用した物理 双曲線

H.Yanase

このシリーズはMathematicaを利用すると自分で変更し、試行錯誤できるようになっているがMathematicaがなくても理解で きるように作成されているMathematicaのプログラムに興味がない読者はプログラム部分はとばして結果だけを見ればよ い。Mathematicaの入力は【Mathematicaの入力】で区別する。Mathematicaがインストールされていれば全てのコマンドは 修正、実行が可能であるが理解に必要な結果はCDF(WolframのHPから無償ダウンロード可能)をインストールすれば Webページからも実行できる。これを【実行可能】で表す。(PCの環境によってはプログラムロード、実行に多少時間が かかる場合があります。)

Mathematicaを利用する場合は事前にMathematicaの基礎(本HPからダウンロード可能)を読んでおくとよい。

本章ではMathematicaVer8から機能強化されたマウスなどによるダイレクトな入力方法の学習もできるようにしてある。

より細かなプログラムの説明はMathematicaのHPにチュートリアルとして豊富なコンテンツがあるので参考にするとよい。

Mathematica HP:http://www.wolfram.com/

筆者HP:http://www.ne.jp/asahi/buturiwa/ai/EducationM1.htm

2次曲線

楕円(Ellipse)、放物線(Parabola)、双曲線(Hyperbola)は物理の運動の分野や波の分野でとても重要な曲線である。 数学的にはこれらは1つの仲間で次のようなパラメタで統一的に表現できることを示そう。 図のように直線Lとある決まった点Fがあるとする。Qから直線Lへの垂線の足をHとしよう。



ここでQF/QHの比をeとして例えばe=1となるようにQ点を動かすとどんな線になるだろうか。

 $e = \frac{\rm QF}{\rm QH} \ \dots (1)$

【Mathematica入力】

H=(0,y) Q=(x,y) F=(a,b) とおこう。

すると線分 QF = $\sqrt{(x-a)^2 + (y-b)^2}$ QH = x とすると $e = \frac{\sqrt{(x-a)^2 + (y-b)^2}}{x}$ となるので (x-a)² + (y-b)² = $e^2 x^2$ から (y - b)² = $e^2 x^2 - (x-a)^2$ よって y = $\pm \sqrt{e^2 x^2 - (x-a)^2}$ +b 作図すると次のようになる。

下記の例でClearというコマンドは記号を初期化させるためのものでなくてもよいが、Mathematicaでこのコマンドを実行す る以前にこの記号に値を代入したりしているとそれを記憶しているので初期化している。はじめて使う場合は必要無いが 長く計算させていると以前の結果が代入されていることがあるので間違いを減らすためのおまじないみたいなものであ る。;があるとそのコマンド結果は表示されない、すっきりさせたい時に利用する。実行はShift+Enterキーだ。



これは普段に見るものと90度回転しているが放物線である。e=1なのでルートの中を展開すると

 $\sqrt{x^2 - (x - a)^2} = \sqrt{2xa + a^2}$ となるので $(y - b)^2 = 2xa + a^2$ となり、xとyを変えれば見慣れた放物線になる。

そこでこのeのことを離心率と呼ぶ。

ではe<1の時はどうなるだろうか。この時ルートの中は x² が正として残り e>1の時はルートの中の x²が負として残ること がわかるだろう。この違いはe<1の時に、楕円。e>1の時に双曲線という結果になる。そこでeを変化させてグラフを描か せてみる。

【Mathematica入力】

次のようにMathematicaではManipulateのコマンドで変数を動的に変えることができる。 最後のAppearance→"Labeled"は離心率の値を画面に表示させるオプションである。

```
Clear[a]; Clear[b]; Clear[e];

Manipulate[

a = -1; b = 0;

y1 = \sqrt{e^2 x^2 - (x - a)^2} + b;

y2 = -\sqrt{e^2 x^2 - (x - a)^2} + b;

Plot[{y1, y2}, {x, -4, 4}],

{{e, 1, "離心率"}, 0, 2, Appearance → "Labeled"}]
```



【実行可能CDF1】

楕円、放物線、双曲線がeをパラメタとして連続的に変化することが確認できただろうか。

これからe=1で放物線、0<e<1で楕円、e>1で双曲線になることがわかる。

これは物理においても例えば人工衛星は地球の周りを楕円軌道している。この衛星に搭載しているエンジンを噴射さ せ、地球の軌道を離れ、他の惑星に行かせたい時にはその軌道をいったん双曲線になるようにしないといけない。楕円で ある限り閉じた曲線になるわけだから必ず戻ってきてしまうわけである。さらに地球上で石を投げるとこれは高校で学習 したように放物線を描く。その結果地球に落ちてくる。では地球に落ちるよりはやく地球を回るようにすごい速さで投げ 出したらどうなるか。答えは人工衛星のになり楕円軌道を描くのである。楕円軌道を描くのに必要な速度、そして双曲 線、つまり引力圏を離脱するのに必要な速度も高校の物理学で簡単に求まるのである。

次にこの関係を立体的に見てみよう。非常におもいろい図形「円錐」をもってきてこれを平面で切り、その切り口を観測 するとここにも楕円(円)、放物線、双曲線が登場する。

【Mathematica入力】

はじめに次のように切り取る板を0の関数で定義する。 例では外枠、座標軸、等高線を全て消すように指示している。 次のことを考えて、面の角度をしていし、描画領域を決めておく。 (参照)

http://demonstrations.wolfram.com/にはWeb上でも実行できるプログラムが多く掲載されている。また数学のデータベースも 充実している。Mathematicaがあればこれらは「ヘルプ」メニューから検索ができる。からないコマンドが出てきたらその 単語の上にカーソルを移動し、f1キーを押すとMathematicaの説明が見られる。 "Conic Sections: The Double Cone" from the Wolfram Demonstrations Project

http://demonstrations.wolfram.com/ConicSectionsTheDoubleCone/

 $\begin{aligned} & \text{plane}[\theta_{-}] := \text{ParametricPlot3D}[\{\text{s} \cos[\theta], \text{y}, 1 - \text{s} \sin[\theta]\}, \{\text{s}, -10, 10\}, \\ & \{\text{y}, -10, 10\}, \\ & \text{Boxed} \rightarrow \text{False}, \text{Axes} \rightarrow \text{False}, \text{PlotRange} \rightarrow 7, \text{Mesh} \rightarrow \text{None}] \end{aligned}$

【Mathematica入力】 次のように入力するとマウスで自由に回転する板が得られる。[]の中には0からπまでの値を入れると面が傾く。

 $plane[\pi]$



【Mathematica入力】 次のように入力するとマウスで自由に回転する3次元の円錐が得られる。

 $\begin{aligned} &\texttt{ParametricPlot3D[\{z \ \texttt{Cos}[\theta], \ z \ \texttt{Sin}[\theta], \ z\}, \ \{\theta, \ 0, \ 2 \ \pi\}, \ \{z, \ -7, \ 7\}, \ \texttt{Boxed} \rightarrow \texttt{False}, \\ &\texttt{Axes} \rightarrow \texttt{False}, \ \texttt{Mesh} \rightarrow \texttt{False}] \end{aligned}$



【Mathematica入力】

先の円錐のプログラムを切り取る板の角度に依存させ、次のように定義する。MeshFunctionを利用し、切り口の部分を太線で強調させる。

次にManipulateでこの円錐と板を両方描かせ、角度をマウスで指定できるようにする。ここまでのプログラムを以下には 1つにまとめてある。

```
\begin{aligned} & \text{plane}[\theta_{-}] := \text{ParametricPlot3D}[\{\text{s} \text{Cos}[\theta], \text{y}, 1 - \text{s} \text{Sin}[\theta]\}, \{\text{s}, -10, 10\}, \\ & \{\text{y}, -10, 10\}, \\ & \text{Boxed} \rightarrow \text{False}, \text{Axes} \rightarrow \text{False}, \text{PlotRange} \rightarrow 7, \text{Mesh} \rightarrow \text{None}]; \\ & \text{corn}[\alpha_{-}] := \text{ParametricPlot3D}[\{\text{z} \text{Cos}[\theta], \text{z} \text{Sin}[\theta], \text{z}\}, \{\theta, 0, 2\pi\}, \{\text{z}, -5, 5\}, \\ & \text{Boxed} \rightarrow \text{False}, \text{Axes} \rightarrow \text{False}, \text{Mesh} \rightarrow \{\{0.\}\}, \\ & \text{MeshFunctions} \rightarrow \{\text{Function}[\{\text{x}, \text{y}, \text{z}, \theta, \text{zz}\}, (1 - \text{z}) \text{Cos}[\alpha] - \text{x} \text{Sin}[\alpha]]\}, \\ & \text{MeshStyle} \rightarrow \text{AbsoluteThickness}[2], \text{PlotPoints} \rightarrow 50, \text{SphericalRegion} \rightarrow \text{True}, \\ & \text{PlotRange} \rightarrow 7]; \\ & \text{Manipulate}[\text{Show}[\{\text{corn}[\alpha], \text{plane}[\alpha]\}, \text{ImageSize} \rightarrow \{500, 400\}], \end{aligned}
```

```
{\{\alpha, 0, " = n \leq n, n, \text{Appearance} \rightarrow "Labeled"\}, SaveDefinitions \rightarrow True
```



マウスで全体を動かし、スライダーで切り取る口の角度を変更してみよう。特に放物線を見つけるのは難しい。 どんなときに円、楕円、双曲線、放物線になるかが確認できただろうか。円錐の頂点から直線をまっすぐおろした線を母 線という。この母線に平行な平面で切った切り口が放物線になる。このことを確かめてみるために先のプログラムを変更 してみよう。

【Mathematica入力】

各関数に平面の高さhを変数に加えて定義しなおす。そして答えを表示させるチェックボックスをつけてチェックがある 時にはIf[]関数で分岐する処理を加えた。これらの処理もMathematicaでは極力簡単にできるように工夫されている。 関数の定義は適当な名前[a_,b_]:=f(a,b)という形で引数をいくつにしても任意の関数を定義できるがManipulateを使う場合 はInitialization:→以下に定義しておく。Appearance→"Labeled"オプションをつかうと例のように選んでいる数値が横に表示

される。

オプションのControlType→は選択ボタンをチェックボックスやラジオボックス等選べる。この場合は排他的に選択するの でSetterBarとした。複数選択可にする場合はTogglerBarを使うとよい。これらの機能の紹介もヘルプやチュートリアルに詳 しい。

```
Manipulate [
      If [chk == 1, shw[0, h], If [chk == 2, shw[\pi / 8, h],
                  If [chk = 3, shw[3\pi/8, h], If [chk = 4, shw[\pi/4, h], shw[\alpha, h]]]]],
       {\{\alpha, 0, " \square nn \in \mathbb{R}^n\}, 0, \pi, Appearance \rightarrow "Labeled"\},
       {{h, 1, "高さ"}, -4, 4, Appearance → "Labeled"},
       {{chk, 0, "例の表示:"}, {0 → Dynamic["選択なし"], 1 → Dynamic["円"],
                  2 → Dynamic["楕円"],
                  3 → Dynamic["双曲線"], 4 → Dynamic["放物線"]}, ControlType → SetterBar},
      Initialization \Rightarrow
              (plane2[\theta_{h_{1}}, h_{1}] := ParametricPlot3D[{s Cos[\theta], y, h - s Sin[\theta]},
                              \{s, -10, 10\}, \{y, -10, 10\}, Boxed \rightarrow False, Axes \rightarrow False, PlotRange \rightarrow 7,
                              Mesh \rightarrow None];
                  \operatorname{corn2}[\alpha, h] := \operatorname{ParametricPlot3D}[\{ \operatorname{z} \operatorname{Cos}[\theta], \operatorname{z} \operatorname{Sin}[\theta], \operatorname{z} \}, \{\theta, 0, 2\pi \}, \{
                              \{z, -5, 5\}, Boxed \rightarrow False, Axes \rightarrow False, Mesh \rightarrow \{\{0.\}\},\
                             MeshFunctions \rightarrow {Function[{x, y, z, \theta, zz}}, (h - z) Cos[\alpha] - x Sin[\alpha]]},
                             MeshStyle \rightarrow AbsoluteThickness[2], PlotPoints \rightarrow 50, SphericalRegion \rightarrow True,
                             PlotRange \rightarrow 7];
                  \operatorname{shw}[\alpha_{, h_{]}} := \operatorname{Show}[\operatorname{corn2}[\alpha, h], \operatorname{plane2}[\alpha, h]], \operatorname{ImageSize} \rightarrow \{500, 400\}];)
```



【実行可能CDF2】

選択なしを選んでおくと角度も自由に変えられる。例えば放物線を選ぶと円錐の母線と面が平行になるので高さを変化さ せれば放物線が変化するのがよくわかるだろう。マウスで全体をドラッグして平面が線になる角度でためしてみるとよ い。この変化の中に宇宙の運動の全てが記述されるわけだ、小宇宙をじっくり観察してほしい。

2距離の和が一定な曲線

では次に平面に点A(-p,0)、B(p,0)を決めて適当な距離Q(x,y)を動かす。この時、距離AQ、BQの和が一定になるようなQの 軌跡はどんな曲線になるだろうか。考えてみてほしい。



これは実験すれば解が得られる。たるんだ糸のはしをA,Bとして固定し、細いペンで糸をピンと張るようにして線を引いていけばいい。結果はやってみると次の図のような楕円ができる。



そこで図のように半長径をa、半短径をbとする。そして糸を固定した点A、点Bを焦点と呼ぶことにする。

惑星の運動は太陽を1つの焦点としてこの楕円軌道を描いている。また、作図した楕円の線に沿って壁を作成し、1つの焦 点からビー玉をはじくと必ずもう一つの焦点を通る。さらにこの壁を内側を鏡にして1つの焦点に電球をおけば、もう一 つの焦点に光が集まり明るくなることからこの「焦点」という名前が使われている。

【宿題1】

焦点からスタートし、楕円面にビー玉が衝突する時の入射角はどこでも等しくなることを示せ。

この作図で本当に楕円ができるなら高校でならった楕円の方程式が導けるはずである。それを示そう。

ペンをS点に持ってくると糸の長さは2p+2(a-p)=2aとおける。

図の線分ATの長さが丁度糸の半分のaになることがわかる。よって三平方の定理から次が成り立つ $p^2 = a^2 - b^2 \dots (2)$

次に線分AQ = $\sqrt{(x+p)^2 + y^2}$,線分BQ = $\sqrt{(x-p)^2 + y^2}$ だからAQ + BQ = 2aだから $\sqrt{(x+p)^2 + y^2} + \sqrt{(x-p)^2 + y^2} = 2a$ となる $\sqrt{(x+p)^2 + y^2} = 2a \cdot \sqrt{(x-p)^2 + y^2}$ の両辺を2乗し、 $(x + p)² + y² = 4 a² - 4 a \sqrt{(x - p)² + y²} + (x - p)² + y²$ よって a $\sqrt{(x - p)^{2} + y^{2}} = (a^{2} - x p)$ となるのでもう一度両辺を2乗すると a²((x - p)² + y²) = a⁴ - 2 x p a² + x² p² となるから簡単にすると a² x² - x² p² + a² y² = a⁴ ... (3) を得る。これに式(2)を代入すると

$$a^{2} x^{2} - x^{2} (a^{2} - b^{2}) + a^{2} y^{2} = a^{4}$$

 $b^{2} x^{2} + a^{2} y^{2} = a^{4}$

a>0,b>0として両辺を a² b² で割ると

 $\frac{x^2}{a^2} + \frac{y^2}{b^2} = \frac{a^2}{b^2} \dots (4)$

となり見慣れた楕円の方程式(a=bならば円の方程式)が得られる。

円はa=bと考えることができるからこの時、式(2)からp=0、すなわち円は2つの焦点が重なり1つになった状態と考えられる。

【問題1】

さきほどの離心率eを長半径aと短半径bで表してみよ。

【解答1】

下図のようにQがTの位置にきた時をT=Qとし、図のSの位置にきた時をS=Q'とする。

また、SH'=cとするとTH=a+cである。はじめにT=Qの時

先と同様にQF=TB = $\sqrt{p^2 + b^2}$ = aである。よって式(1)より次のように離心率が求まる。



$$e = \frac{QF}{QH} = \frac{QB}{QH} = \frac{a}{a+c} \dots (5)$$

次にQがSにある時は次のように離心率が求まる。

 $e = \frac{BQ'}{Q'H'} = \frac{a-p}{c}\dots(6)$

離心率はどこでも等しいから式(5),(6)より

$$\frac{a-p}{c} = \frac{a}{a+c}$$

$$c = \frac{a(a-p)}{p} \dots (7)$$

この結果を再び式(6)に入れると式(2)より次のように解が得られる。

$$e = \frac{p}{a} = \frac{\sqrt{a^2 - b^2}}{a} \dots (8)$$

またこれから離心率は焦点の位置pが

p = a e...(9)

で与えられることがわかる。

【問題2】

問題1の結果を利用してe=1となるのはどんな特徴の曲線か考えてみよ。

【解答2】

問題1の結果からe=1とするとp=aとなる。すなわち図の焦点は曲線上に移動してしまうのでもはや楕円ではない。式(8)からこの時、b=0になってしまうので横幅が無限に長いような細長い楕円になってしまう。しかしe=1となるのは楕円であることをあきらめれば曲線の外に焦点を持つ場合で答えがみつかる。

これは下図のようにはじめに触れた放物線である。原点をOとして高さhのところに焦点Pがあったとしよう。

曲線上の点をQ(x,y)とおく。直線までの距離QHとPQが等しくなるような曲線を求めればよい。



そこでQ(x,y)とするとPQ=QHが成り立つので図のように点Pと直線との距離をhとすると

$$PQ = \sqrt{x^2 + (y - h)^2}$$
, $QH = y$

だから $PQ^2 = QH^2$ より

$$x^{2} + y^{2} - 2hy + h^{2} = y^{2}$$
$$y = \frac{x^{2}}{2h} + \frac{h}{2}$$

となるので答えは放物線であり、しかも焦点の位置は頂点と原点までの2倍の高さの位置にとればいいことがわかる。

このようにp=aとならないところに奥深さがある。

放物線はPQとQHが等しくなるようにQを動かした時にできる曲線であるといえる。放物線や双曲線と異なり焦点は1つである。

ある意味で2つあった楕円の焦点が無限大まで離れて1つしか見えてない状態といえる。

【Mathematica入力】

ここでMathematicaを利用して焦点の位置を自由に動かした時、ダイレクトに楕円の変化がわかるプログラムを作成してみよう。

楕円の作図には様々な方法がある。極座標というのを用いるのが簡単だがここでは焦点の位置をはっきりさせるため式 (4)を用いよう。 式(2)(4)から

$$y^{2} = a^{2} - \frac{x^{2}b^{2}}{a^{2}}$$
$$y = \pm \sqrt{a^{2} - \frac{x^{2}b^{2}}{a^{2}}} = \pm \sqrt{a^{2} - \frac{x^{2}(a^{2} - p^{2})}{a^{2}}} = 7 \pm \sqrt{a^{2} - x^{2} + \frac{p^{2}}{a^{2}}}$$

焦点の座標は離心率を用いて式(9)で決まるから式()からbを消して

$$y = \pm \sqrt{\frac{p^2}{e^2} + e^2 - x^2}$$

となることを利用する。

```
Clear[x1, y1]; Clear[e]; Clear[x, y]; Clear[xx, yy]; Clear[p];
Manipulate[
xx = Part[p, 1];
yy = Part[p, 2];
ans = Solve[\frac{xx^2}{e^2} + e^2 - x^2 = 0, x];
x1 = Simplify[Part[x /. ans, 1]];
x2 = Simplify[Part[x /. ans, 2]];
y = Sqrt[\frac{xx^2}{e^2} + e^2 - k^2];
Dynamic[LocatorPane[Dynamic[p],
Dynamic@Plot[{Evaluate[y] + yy, -Evaluate[y] + yy}, {k, x1, x2},
PlotRange → {{-2, 2}, {-2, 2}}]],
{[e, 0.7, "離心率"}, 0.1, 1, Appearance → "Labeled"},
{[p, {1, 0}, "position of source 1"}, {-5, -5}, {5, 5}, ControlType -> None}]
```



【実行可能CDF3】

焦点を表す小さな灰色の円(ロケーター)が現れたはずだ。これをマウスで直接ドラッグしてみよう。

楕円と焦点の関係を色々動かしてためしてみるとよい。

2距離の差が一定な曲線

では次に平面に点A(-p,0)、B(p,0)を決めて適当な距離Q(x,y)を動かす。この時、距離AQ、BQの差が一定になるようなQの 軌跡はどんな曲線になるだろうか。考えてみてほしい。



そこでQ(x,y)とするとAQ-BQが一定になるわけだから

$$AQ = \sqrt{(x+p)^2 + y^2}$$
, $BQ = \sqrt{(x-p)^2 + y^2}$

より次のようにおける。

AQ - BQ =
$$\sqrt{(x+p)^2 + y^2} - \sqrt{(x-p)^2 + y^2} = k$$

2乗すると

$$(x+p)^{2} + y^{2} = (x-p)^{2} + y^{2} + k^{2} - 2k\sqrt{(x-p)^{2} + y^{2}}$$
$$2k\sqrt{(x-p)^{2} + y^{2}} = k^{2} - 4xp$$

再び2乗すると

$$4k^{2}((x-p)^{2} + y^{2}) = k^{4} - 8xpk^{2} + 16x^{2}p^{2}$$
$$x^{2} - 2xp + p^{2} + y^{2} = \frac{k^{2}}{4} - 2xp + 4x^{2}\frac{p^{2}}{k^{2}}$$
$$y^{2} = \frac{k^{2}}{4} - p^{2} + x^{2}\left(\frac{4p^{2}}{k^{2}} - 1\right)...(10)$$

ここで簡単にするために $k^2 = 4$ を選ぼう。すると

$$(p^2 - 1)x^2 - y^2 = p^2 - 1$$

となる。これは $p^2 \ge 1$ の時は

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} = r^2$$

となる。これは一般に双曲線の方程式である。 $p^2 < 1$ の時は次の楕円の式になる。

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = r^2$$

kをもどしてyについて解くと次を得る。

$$y = \pm \sqrt{\frac{k^2}{4} - p^2 + x^2 \left(\frac{4 p^2}{k^2} - 1\right)}$$

【Mathematica入力】

これをpを変数にしてMathematicaに描かせてみよう。簡単に視覚に訴えるプログラムがMathematicaの特徴だ。

だたこの場合は単純にyの値を描画させてしまうとルート内の符号によって楕円も描いてしまうのでPlotのオプションにある

RegionFunctionをつかう。このオプションは描画領域を指定することができる。まず次のコマンドでこの式の×軸との交点を

みつける。もちろんy=0として×について解けばいいので手でやっても簡単な結果が得られる。

Mathermaticaでは方程式を解くには次のようにSolveコマンドを使う、下記の例ではさらにSimplifyに条件を入れて簡単にした結果を得られるようにしている。

Clear[y, x];
ans = Solve
$$\left[0 = \frac{k^2}{4} - p^2 + x^2 \left(4 \frac{p^2}{k^2} - 1\right), x\right];$$

Simplify[ans, {k > 0, p > 0}]
 $\left\{\left\{x \rightarrow -\frac{k}{2}\right\}, \left\{x \rightarrow \frac{k}{2}\right\}\right\}$

Mathermaticaでは方程式を解くには次のようにSolveコマンドを使う、下記の例ではさらにSimplifyに条件を入れて簡単にした結果を得られるようにしている。条件は式以外にも実数や複素数を指定することができる。

ではプログラムを書くと±の2つの解を{}で同時に描かせる。ただし、焦点の位置pと距離の差kは選べるようにする。

pについては画面から直接入力できるようにLocatorを使って指定する。ただし、2つの焦点は別々に動かすことができるのでそれぞれの焦点について2種類の双曲線が描かれるようになっている。1つにしたい時はpをp[[1]]に置き換えると1成分だけをとるので1組の双曲線しか現れない。RegionFunctionを用いて先の結果から×が-k/2とk/2の範囲は楕円になるので描画しないようにしている。

```
Clear[y]; Clear[p]; Clear[k];

Manipulate[

Plot\left[\left\{Sqrt\left[\frac{k^2}{4} - p^2 + x^2\left(\frac{4 p^2}{k^2} - 1\right)\right], -Sqrt\left[\frac{k^2}{4} - p^2 + x^2\left(\frac{4 p^2}{k^2} - 1\right)\right]\right\}, \{x, -6, 6\},
PlotRange \rightarrow \{\{-6, 6\}, \{-4, 4\}\}, RegionFunction \rightarrow Function[\{x\}, Abs[x] > k / 2]\right],
```

{{k, 4, "距離の差"}, 0.1, 8, Appearance → "Labeled"},

 $\{\{p, \{\{-2, 0\}, \{2, 0\}\}\}, \{-4, -0\}, \{4, 0\}, Locator\}, Deployed \rightarrow True\}$



【実行可能CDF4】

波の干渉と双曲線

さて、物理を学習しているならこのグラフを見てあることに気づくはずだ。水面波が2点ABから同位相で出ていると き、十分時間がたつと合成波ができる。このとき距離の差が行路差に相当しこれが波長の整数倍なら強めあうという干渉 条件があった。つまり、この双曲線は節、腹線そのものだ。

距離の差kがn=1,2,3.....として

 $\mathbf{k} = \mathbf{A}\mathbf{Q} - \mathbf{B}\mathbf{Q} = \mathbf{n}\lambda...(11)$

を満たすのが強め合う腹線なのでこれをMathematicaに描かせる。kをnλに置き換える。焦点は1つを選ぶようにしてンnと λの組み合わせによっては楕円が描かれることを許している。

```
Clear[y]; Clear[p]; Clear[nmax]; Clear[λ];
Manipulate
```

tb = Table $\left[\operatorname{Sqrt} \left[\frac{(n \lambda)^2}{4} - p[[1]]^2 + x^2 \left(\frac{4 p[[1]]^2}{(n \lambda)^2} - 1 \right) \right], \{n, 1, nmax\} \right];$ Plot [{tb, -tb}, {x, -4, 4}, PlotRange → {{-6, 6}, {-6, 6}}, { {nmax, 2, "n"}, 0, 8, 1, Appearance → "Labeled"}, { {{\lambda, 1, "波長\lambda"}, 0.5, 4, 0.5, Appearance → "Labeled"}, { {{p, {-2, 0}}, {-4, -0}, {4, 0}, Locator}, Deployed → True}



では最終的にA,Bからの波の波面を同時に描かせてみよう。これには以下のようにプログラムを変える。

【Mathematica入力】

g1に先の双曲線を入れて、g2にはA,Bから半径がλの整数倍になる円形波を描かせる。そしてチェックボックスckがチェックされるとこの円と双曲線が両方表示されるようにする。円が正しくみられるようにオプションとしてx、yが1対1になるようにしている。

```
Clear[y]; Clear[p]; Clear[nmax]; Clear[\lambda, tb]; Clear[g1, g2];

Manipulate[

tb = Table[Sqrt[\frac{(n \lambda)^2}{4} - p[[1]]^2 + x^2 \left(\frac{4 p[[1]]^2}{(n \lambda)^2} - 1\right)], {n, 1, nmax}];

g1 = Plot[{tb, -tb}, {x, -4, 4}, PlotRange \rightarrow {{-6, 6}, {-6, 6}},

PlotStyle \rightarrow Thick];

g2 = Table[{Circle[p, n \lambda], Circle[-p, n \lambda]}, {n, 1, 2 nmax}];

If[ck, Show[{g1, Graphics[g2]}, PlotRange \rightarrow {{-6, 6}, {-6, 6}},

AspectRatio \rightarrow 1],

Show[g1, PlotRange \rightarrow {{-6, 6}, {-6, 6}}, AspectRatio \rightarrow 1]],

{{nmax, 2, "n"}, 0, 8, 1, Appearance \rightarrow "Labeled"},

{{\lambda, 1, "k \in \lambda"}, 0.5, 4, 0.5, Appearance \rightarrow "Labeled"},

{{p, {-2, 0}, {-4, -0}, {4, 0}, Locator},

{{ck, False, "k \in \pi r, {True, False}]]
```



【実行可能CDF5】

このプログラムでは選び方では楕円も描画される。しかし、波源を動かせば双曲線がすぐに現れると思うので大きなn と小さなλを選んで双曲線が多く描けたら波面表示のチェックボックスをマウスクリックするとみごと両方の波の円の交 点、つまり山と山が重なっている点を双曲線の群が通過していることがわかる。いろいろ確かめてほしい。 さて、実際にお風呂場で水面を2カ所でとんとんとたたいてみると円形波ができて、しばらくすると合成波になる。 1箇所をとんとんとたたいた時はきれいな円形波が波紋として広がっていくのがわかる。しかし、2カ所でたたい合成波 をみた場合、

ここであつかったような節線、腹線を見ることができるだろうか。2カ所にすると波紋のふち、つまり円形波の円周部分 も波打ち、節線、腹線を見るのは難しい。しかし腹線は常に振動しているため見るのは難しいが節線であれば確認できそ うなものである。

この様子をMathematicaでシミュレートしてみよう。

【Mathematica入力】

Norm[A,B]関数はA,B間の距離を出す関数である。また、波の干渉ではなめらかさを出すためにPlotPointsのオプションに まるめ関数Round[]をつかって波長に応じた解像度を出すようにしている。

```
Clear[xy1]; Clear[nmax]; Clear[\lambda]; Clear[tb]; Clear[g1]; Clear[g2];
Clear[q1] Clear[q2]; clear[x]; clear[tb];
Manipulate
 g2 = ParametricPlot3D[{tb[nmax, \lambda, xy1], -tb[nmax, \lambda, xy1]}, {x, -6, 6},
    PlotStyle → {{Thick, Red}, {Thick, Red}}];
 g1 = Plot3D[q1 Cos[\frac{2\pi}{2} Norm[{x, y} - {xy1, 0}] - p] +
      q2 \cos \left[\frac{2\pi}{2} \operatorname{Norm}[\{x, y\} + \{xy1, 0\}] - p\right],
     \{x, -6, 6\}, \{y, -6, 6\}, \text{ImageSize} \rightarrow \{460, 460\}, \text{Mesh} \rightarrow \text{None},\
    ImagePadding \rightarrow None,
    ImageMargins \rightarrow None, PlotRangePadding \rightarrow None, BoundaryStyle \rightarrow None,
    Boxed \rightarrow False,
    Axes \rightarrow False, Ticks \rightarrow None, PlotPoints \rightarrow Round \left[7\frac{2\pi}{2}+25\right],
    Method \rightarrow {"ArrayPlot" \rightarrow True},
    ColorFunction \rightarrow (Blend[{RGBColor[.13, .35, .9], White}, #3] &),
    PlotRange \rightarrow {{-6, 6}, {-6, 6}, {-50, 50}};
 If [ck, Show[{g1, g2}, PlotRange \rightarrow {{-6, 6}, {-6, 6}, {-50, 50}}],
   Show[g1, PlotRange \rightarrow {{-6, 6}, {-6, 6}, {-50, 50}}]],
  {{xy1, 3, "波源1のx座標"}, 0, 5, 1, Appearance → "Labeled"},
 {{q1, 6, "波源1の振幅"}, -10, 10, 1, Appearance → "Labeled"},
 {{q2, -6, "波源2の振幅"}, -10, 10, 1, Appearance → "Labeled"},
  {{nmax, 6, "n"}, 0, 8, 1, Appearance \rightarrow "Labeled"},
 \{\{\lambda, 1, "波長\lambda"\}, 0.5, 4, 0.5, Appearance → "Labeled"\},
 {{p, 1, "位相"}, 1, 6 Pi, Appearance → "Open"},
 {{ck, False, "節腹線表示"}, {True, False}},
 AutorunSequencing \rightarrow {{1, 8}, {4, 4}, {5, 12}, {6, 10}},
 TrackedSymbols \Rightarrow {xy1, q1, q2, p, \lambda, ck, nmax},
 Initialization \Rightarrow
   \left( \text{tb}[\text{nmax}_{,\lambda_{,}},\text{xy1}_{]} := \text{Table}\left[ \left\{ x, \text{Sqrt}\left[ \frac{(n \lambda)^{2}}{4} - \text{xy1}^{2} + x^{2} \left( \frac{4 \text{xy1}^{2}}{(n \lambda)^{2}} - 1 \right) \right], 0 \right\},
         {n, 1, nmax}];
```



【実行可能CDF6】

実行画面になったらまず、位相のプレイボタンを押して水面波が動く様子を確認してみよう。マウスでドラッグすると動 画を回転させることができるが負荷が大きいので一度停止ボタンを押してから回転させてみるとよい。

このプログラムでは波源AB間を6になるように最初の設定がなっている。波源1の座標のスライダーを変化させるとこれを変えられる。また、A,Bの波源の位相が π ずれるように波源1の振幅と波源2の振幅はマイナスの関係にしてある。位相が π ずれていれば中点は必ず節になるはずだ。

さて、動きが確認できて、どこに節線があるか予想してみよう。節腹線表示チェックを入れると赤線で節線が出てくる。

節腹線は合成波の速度の向きを表していた。

確かに節線は常に高さ0になっていることが角度をかえてみると確認できる。

振幅を同じにすると節線と腹線が入れ替わることもわかるだろう。また、どちらかの波の振幅を0にするときれいな円形 波が確認できる。でももう一つの振幅を0から増やしていくと円形の淵も波打ち合成波に移行していくのがわかる。 注意:このプログラムでは楕円を消す処理をしていないのでλ、nの選び方によっては楕円が出てくる。

さて、波の干渉を注意深く観察すると波源ABを結ぶ直線上は特別であることがわかる。波源A,Bの内側ではお互いの波は 反対向きに進むので合成はは定常波となり進行しない。しかし、波源の外側ではAの波もBの波も同じ向きに進むので合成 波も同じ向きに進む。

この様子もシミュレートしてみよう。

【Mathematica入力】

先の立体図を波源ABで切った断面を表してみよう。

以下の例ではGrid[]コマンドを用いてグラフを3段で表示させている。

1段目にはAからの波とBからの波の進行波が描かれ、振動数を入力できるようにしてある。rcがかかっているのは進行方向を反転させるためである。

2段目にはAの波とBの波の足し算した波が描かれる。

3段目には波のエネルギーや強さが振幅の2乗に比例したからAとBの振幅の和の2乗を表すようにしてある。外側←→内側 のスライダーでrcを1と-1に交換させ波の向きを同じ向きか反対向きになるようにしてある。ManipulateにAppearance→"Open"のオプションを加えると例のように再生のボタン等が+をクリックしなくても表示された状態になる。

```
Manipulate[
Grid[
{{Plot[{Sin[freq1x+t], Sin[freq2x-rct]}, {x, -2π, 2π}, PlotRange → {-2, 2},
ImageSize → {500, 200}, AspectRatio → 2 / 5]},
{Plot[Sin[freq2x - rct] + Sin[freq1x+t], {x, -2π, 2π}, PlotRange → {-2, 2},
ImageSize → {500, 200}, AspectRatio → 2 / 5]},
{Plot[(Sin[freq2x - rct] + Sin[freq1x+t])<sup>2</sup>, {x, -2π, 2π},
PlotRange → {-4, 4},
ImageSize → {500, 200}, AspectRatio → 2 / 5]}}],
{{freq1, 1, "振動数 1 "}, 1, 10, 1, Appearance → "Labeled"},
{{freq2, 1, "振動数 2 "}, 1, 10, 1, Appearance → "Labeled"},
{{rc, 1, "外側↔內側"}, -1, 1, 2},
{{t, 1, "time"}, 0, 2π, Appearance → "Open"}]
```



【実行可能CDF7】

実行画面になったらプレイボタンを押してみよう。

Aからの波とBからの波が同じ振動数で向かい合い、下にはその合成波が描かれている。

外側←→内側のスライダーを反対にすると同じ向きになり、外側の合成波が下に描かれる。

さらに振動数を変化させてみよう。複雑な形になる。様子が確認できたら1の振動数を10,2の振動数を9くらいにし

てみよう。高校で学習したうなりの様子がみえる。非常に激しい振動数 = (10+9)/2=8.5Hzとゆっくりの振動数 (10-9)/2=0.5Hzとなる。

人間は低い方のうなりつまり0.5Hzの方が聞こえやすい。そしていったんこの音を聞くと他の音はきこえなくなる。さらに振動数が低いと人は音の強さで聞くので3段目のグラフで見るように振動数が2倍つまり0.5×2=1Hzの音がうなりとして 聞こえるわけだ。

MathematicaにもPlayという音を出すコマンドがある。これで10ヘルツのうなりと10ヘルツのSin波の音を聞き比べてみるといい。

合成波の速度

内側外側のスライダーを入れ替えると合成波の速さも極端に変化するはずだ。注意すべきは合成波の速度は波長(振動数)が異なると速くにもみえたり、遅くにもみえたりする。しかし、どちらかの波をけせば常に波は一定の速さ進行して いるだけだ。同じ向きに速さが重なっても同じ速さの波ができる。

では波源A,Bからの任意の位置Qで合成波の速度はどうなるだろうか。波の速さはV=f λ で決まった。波源からあらゆる向きに等しくこの速度の波が出ているわけである。注意すべきは波の速度は振幅に無関係で基本的にA,Bから出た波の速さは等しく、次の図のように平行四辺形の法則で簡単に合成される。



しかし、ここで注意がいる。合成された速度の最大は元の波の速度を超えることはない。従って合成いは簡単には

$$V' = \frac{V+V}{2}$$

となることだ。これは運動量保存則を用いると媒質の質量をm、速さをVとして同じ向きなら

$$mV + mV = 2mV'$$
 $V' = V$

反対向きなら

$$mV - mV = 2mV' \quad V' = 0$$

で動かないことがわかる。(実際の波の場合はこんなに簡単ではない)

ではこの様子を Mathematica に描かせよう。

【Mathematica入力】

はじめに2つの焦点が現れるが左の焦点をマウスでドラッグすると波源の位置A,Bがともに対照的変化する。右の焦点は任 意の位置に動かすことができる。そして速度ボタンを押すと赤いベクトルで合成速度が表示される。

また高校でならったようにこの合成速度は節腹線に一致するので節腹線ボタンを押すと節腹線が現れる。

波長や数nを変えることができるが楕円を消す処理をしていないのでそのまま楕円が現れることに注意すること。

```
Clear[v]; Clear[p]; Clear[nmax]; Clear[\lambda]; Clear[tb]; Clear[g1]; Clear[g2];
Clear[g3]; Clear[chk];
```

```
Manipulate
```

```
tb = Table [Sqrt[\frac{(n \lambda)^2}{4} - p[[1]]^2 + x^2 \left(\frac{4 p[[1]]^2}{(n \lambda)^2} - 1\right)], {n, 1, nmax}];
g3 = Graphics [{Thick, Red, Arrow [{v, v + \left(\frac{v-p}{2 \operatorname{Norm}[v-p]} + \frac{v+p}{2 \operatorname{Norm}[v+p]}\right)}]]}];
g2 = Plot[{tb, -tb}, {x, -4, 4}];
g1 = Table[Graphics[{Circle[p, n \lambda], Circle[-p, n \lambda]}], {n, 1, 2 nmax}];
If[chk = 0, Show[g1, PlotRange \rightarrow {{-6, 6}, {-6, 6}}, AspectRatio \rightarrow 1],
If[chk = 1, Show[{g1, g2}, PlotRange \rightarrow {{-6, 6}, {-6, 6}}, AspectRatio \rightarrow 1],
If[chk = 2, Show[{g1, g2, g3}, PlotRange \rightarrow {{-6, 6}, {-6, 6}}, AspectRatio \rightarrow 1],
If[chk = 2, Show[{g1, g2, g3}, PlotRange \rightarrow {{-6, 6}, {-6, 6}},
AspectRatio \rightarrow 1]]],
{{nmax, 2, "n"}, 0, 8, 1, Appearance \rightarrow "Labeled"},
{{\lambda, 1, "波長\lambda"}, 0.5, 4, 0.5, Appearance \rightarrow "Labeled"},
{{v, {2, 0.01}, {-4, -0}, {4, 0}, Locator},
{{v, {2, 0.01}, {-5, -5}, {5, 5}, Locator},
{{chk, 0, "表示: "}, {0 \rightarrow "選択なし", 1 \rightarrow "節腹線", 2 \rightarrow "速度"},
TrackedSymbols \Rightarrow {p, v, \lambda, chk, nmax}}
```



【実行可能CDF8】

さて、いろいろ試してみるとおもしろいことに気づく。前にもふれたように波源AとBの直線上は興味ある現象が起きている。

A,Bを越えた瞬間に速さが0から最大になるのだ。他の位置で速さが最大になるのはどこだろう?

そう、答えは無限大遠方でしかない。つまり直線AB上でA,Bの外側の半直線は無限遠と同じなのだ!さてこれはどういうことだろう。宿題としておく。