|  |
| --- |
| 主専攻実習（定理証明班） |
| 第八回課題レポート |
| 担当：森継 修一 |

|  |
| --- |
| 知識情報システム主専攻 201611502久保川一良  2018年11月19日 |

* 接続環境
  + 自分のローカル環境にReduceをインストールして利用した。
    - 使用したPCのスペックについて： <https://bit.ly/2Cg7hqb>
  + OSやReduceのオプション設定などについては以下の通りである。

username@my\_computer:~ [HH:MM:SS]

$ lsb\_release -a

No LSB modules are available.

Distributor ID: Ubuntu

Description: Ubuntu 16.04.5 LTS

Release: 16.04

Codename: xenial

username@my\_computer:~ [HH:MM:SS]

$ reduce # alias reduce='redcsl -v -w -k 4000 --nogui'

Codemist Standard Lisp revision 4765 for linux-gnu:x86\_64: Sep 19 2018

Created: Wed Sep 19 15:57:15 2018

Reduce (Free CSL version, revision 4765), 19-Sep-18 ...

Memory allocation: 4168 Mbytes

There are 8 processors available

* 入力ファイル

% オイラー線(Euler line) の証明 -------------------------------------------;

% △ABCの各頂点から辺BC,CA,ABに向かって下ろした垂線の足をそれぞれD,E,Fとする

% さらに、辺BC,CA,ABの中点をL,M,Nとする

% このとき、△ABCの垂心V, 重心G, 外心Xは一直線上に存在する

%

% A(0, 0) B(u1, u2) C(u3, u4) D(x01, x02) E(x03, x04) F(x05, x06)

% L(x07, x08) M(x09, x10) N(x11, x12) V(x13, x14) G(x15, x16) X(x17, x18)

%----------------------------------------------------------------------;

% 関数定義読み込み） ----------------------------------------------------;

in cal\_sys\_relations$

% ---------------------------------------------------------------------;

% ＜証明＞ -------------------------------------------------------------;

order x18, x17, x16, x15, x14, x13, x12, x11, x10, x09, x08, x07, x06, x05, x04, x03, x02, x01, u4, u3, u2, u1;

factor x18, x17, x16, x15, x14, x13, x12, x11, x10, x09, x08, x07, x06, x05, x04, x03, x02, x01;

% ---------------------------------------------------------------------;

% 仮定 ----------------------------------------------------------------;

% まず、垂心について考えてみる

% △ABCの各頂点から辺BC,CA,ABに向かって下ろした垂線の足をそれぞれD,E,Fとする

% B-D-C かつ AD⊥BC

h1:=collinear(u1, u2, x01, x02, u3, u4);

h2:=vertically(0, 0, x01, x02, u1, u2, u3, u4);

% C-E-A かつ BE⊥CA

h3:=collinear(u3, u4, x03, x04, 0, 0);

h4:=vertically(u1, u2, x03, x04, u3, u4, 0, 0);

% A-F-B かつ CF⊥AB

h5:=collinear(0, 0, x05, x06, u1, u2);

h6:=vertically(u3, u4, x05, x06, 0, 0, u1, u2);

% 垂心VはAD, BE, CFの交点

% A-V-D かつ B-V-E かつ C-V-F

h7:=collinear(0, 0, x13, x14, x01, x02);

h8:=collinear(u1, u2, x13, x14, x03, x04);

h9:=collinear(u3, u4, x13, x14, x05, x06);

%----------------------------------------------------------------------;

% ここで一旦、垂心Vについて仮定が成り立っているか確認してみる -----------------------;

%----------------------------------------------------------------------;

%----------------------------------------------------------------------;

% Groebner Basis: 結果が1となったら、仮定が誤っている可能性が高い ---------------;

%%% 変数を定義し、lex形式で並べる ------------------------------------------;

torder({x14, x13, x06, x05, x04, x03, x02, x01}, lex)$

%%% 仮定において定義した式からGroebner Basisを求める -------------------------;

gb:=groebner{h1, h2, h3, h4, h5, h6, h7, h8};

%---------------------------------------------------------;

% 「glterms」が出力するのは、グレブナー基底の計算過程で〈ゼロにはならない〉

% と仮定された式のリストである。

%---------------------------------------------------------;

%%% u に関する制約条件 --------------------------------------------------;

glterms;

%%% gbを法としてgを簡約 --------------------------------------------------;

preduce(h9, gb);

% ==> 0 になっているので、垂心Vについての仮定が成立 -----------------------------;

%----------------------------------------------------------------------;

%----------------------------------------------------------------------;

% 次に、重心Gについて考えてみる

% 辺BC,CA,ABの中点をL,M,Nとする

% B=L=C

h10:=midpoint(1, u1, u2, x07, x08, u3, u4);

h11:=midpoint(0, u1, u2, x07, x08, u3, u4);

% C=M=A

h12:=midpoint(1, u3, u4, x09, x10, 0, 0);

h13:=midpoint(0, u3, u4, x09, x10, 0, 0);

% A=N=B

h14:=midpoint(1, 0, 0, x11, x12, u1, u2);

h15:=midpoint(0, 0, 0, x11, x12, u1, u2);

% 重心GはAL, BM, CNの交点

% A-G-L かつ B-G-M かつ C-G-N

h16:=collinear(0, 0, x15, x16, x07, x08);

h17:=collinear(u1, u2, x15, x16, x09, x10);

h18:=collinear(u3, u4, x15, x16, x11, x12);

%----------------------------------------------------------------------;

% ここで一旦、重心Gについて仮定が成り立っているか確認してみる -----------------------;

%----------------------------------------------------------------------;

%----------------------------------------------------------------------;

% Groebner Basis: 結果が1となったら、仮定が誤っている可能性が高い ---------------;

%%% 変数を定義し、lex形式で並べる ------------------------------------------;

torder({x16, x15, x12, x11, x10, x09, x08, x07}, lex)$

%%% 仮定において定義した式からGroebner Basisを求める -------------------------;

gb:=groebner{h10, h11, h12, h13, h14, h15, h16, h17};

%%% u に関する制約条件 --------------------------------------------------;

glterms;

%%% gbを法としてgを簡約 --------------------------------------------------;

preduce(h18, gb);

% ==> 0 になっているので、重心Gについての仮定が成立 -----------------------------;

%----------------------------------------------------------------------;

%----------------------------------------------------------------------;

% 最後に、外心Xについて考えてみる

% 辺BC,CA,ABの中点をL,M,Nとする

% これは既に、 h10~h15 で定義されているので再利用する

% またXは, XL⊥BC, XM⊥CA, XN⊥AB を満たす

h19:=vertically(x17, x18, x07, x08, u1, u2, u3, u4);

h20:=vertically(x17, x18, x09, x10, u3, u4, 0, 0);

h21:=vertically(x17, x18, x11, x12, 0, 0, u1, u2);

%----------------------------------------------------------------------;

% Groebner Basis: 結果が1となったら、仮定が誤っている可能性が高い ---------------;

%%% 変数を定義し、lex形式で並べる ------------------------------------------;

torder({x18, x17, x12, x11, x10, x09, x08, x07}, lex)$

%%% 仮定において定義した式からGroebner Basisを求める -------------------------;

gb:=groebner{h10, h11, h12, h13, h14, h15, h19, h20};

%%% u に関する制約条件 --------------------------------------------------;

glterms;

%%% gbを法としてgを簡約 --------------------------------------------------;

preduce(h21, gb);

% ==> 0 になっているので、外心Xについての仮定が成立 -----------------------------;

%----------------------------------------------------------------------;

% ---------------------------------------------------------------------;

% 結論 ----------------------------------------------------------------;

% このとき、△ABCの垂心V, 重心G, 外心Xは一直線に存在する

% V-G-X

conclusion:=collinear(x13, x14, x15, x16, x17, x18);

%----------------------------------------------------------------------;

% Groebner Basis: 結果が1となったら、仮定が誤っている可能性が高い ---------------;

%%% 変数を定義し、lex形式で並べる ------------------------------------------;

torder({x18, x17, x16, x15, x14, x13, x12, x11, x10, x09, x08, x07, x06, x05, x04, x03, x02, x01}, lex)$

%%% 仮定において定義した式からGroebner Basisを求める -------------------------;

gb:=groebner{h1, h2, h3, h4, h5, h6, h7, h8, h9, h10, h11, h12, h13, h14, h15, h16, h17, h18, h19, h20, h21};

%---------------------------------------------------------;

% 「glterms」が出力するのは、グレブナー基底の計算過程で〈ゼロにはならない〉

% と仮定された式のリストである。

%---------------------------------------------------------;

%%% u に関する制約条件 --------------------------------------------------;

glterms;

%%% gbを法としてgを簡約 --------------------------------------------------;

preduce(conclusion, gb);

% ==> 0 になっていれば、定理は成立 ------------------------------------------;

showtime;

;end;

* 出力ファイル

% オイラー線(Euler line) の証明 -------------------------------------------;

% △ABCの各頂点から辺BC,CA,ABに向かって下ろした垂線の足をそれぞれD,E,Fとする

% さらに、辺BC,CA,ABの中点をL,M,Nとする

% このとき、△ABCの垂心V, 重心G, 外心Xは一直線上に存在する

%

% A(0, 0) B(u1, u2) C(u3, u4) D(x01, x02) E(x03, x04) F(x05, x06)

% L(x07, x08) M(x09, x10) N(x11, x12) V(x13, x14) G(x15, x16) X(x17, x18)

%----------------------------------------------------------------------;

% 関数定義読み込み） ----------------------------------------------------;

in cal\_sys\_relations$

% ---------------------------------------------------------------------;

% ＜証明＞ -------------------------------------------------------------;

order x18, x17, x16, x15, x14, x13, x12, x11, x10, x09, x08, x07, x06, x05, x04, x03, x02, x01, u4, u3, u2, u1;

factor x18, x17, x16, x15, x14, x13, x12, x11, x10, x09, x08, x07, x06, x05, x04, x03, x02, x01;

% ---------------------------------------------------------------------;

% 仮定 ----------------------------------------------------------------;

% まず、垂心について考えてみる

% △ABCの各頂点から辺BC,CA,ABに向かって下ろした垂線の足をそれぞれD,E,Fとする

% B-D-C かつ AD⊥BC

h1:=collinear(u1, u2, x01, x02, u3, u4);

h1 := x02\*( - u3 + u1) + x01\*(u4 - u2) - u4\*u1 + u3\*u2

h2:=vertically(0, 0, x01, x02, u1, u2, u3, u4);

h2 := x02\*(u4 - u2) + x01\*(u3 - u1)

% C-E-A かつ BE⊥CA

h3:=collinear(u3, u4, x03, x04, 0, 0);

h3 := x04\*u3 - x03\*u4

h4:=vertically(u1, u2, x03, x04, u3, u4, 0, 0);

h4 := - x04\*u4 - x03\*u3 + u4\*u2 + u3\*u1

% A-F-B かつ CF⊥AB

h5:=collinear(0, 0, x05, x06, u1, u2);

h5 := - x06\*u1 + x05\*u2

h6:=vertically(u3, u4, x05, x06, 0, 0, u1, u2);

h6 := x06\*u2 + x05\*u1 - u4\*u2 - u3\*u1

% 垂心VはAD, BE, CFの交点

% A-V-D かつ B-V-E かつ C-V-F

h7:=collinear(0, 0, x13, x14, x01, x02);

h7 := - x14\*x01 + x13\*x02

h8:=collinear(u1, u2, x13, x14, x03, x04);

h8 := - x14\*x03 + x14\*u1 + x13\*x04 - x13\*u2 - x04\*u1 + x03\*u2

h9:=collinear(u3, u4, x13, x14, x05, x06);

h9 := - x14\*x05 + x14\*u3 + x13\*x06 - x13\*u4 - x06\*u3 + x05\*u4

%----------------------------------------------------------------------;

% ここで一旦、垂心Vについて仮定が成り立っているか確認してみる -----------------------;

%----------------------------------------------------------------------;

%----------------------------------------------------------------------;

% Groebner Basis: 結果が1となったら、仮定が誤っている可能性が高い ---------------;

%%% 変数を定義し、lex形式で並べる ------------------------------------------;

torder({x14, x13, x06, x05, x04, x03, x02, x01}, lex)$

%%% 仮定において定義した式からGroebner Basisを求める -------------------------;

gb:=groebner{h1, h2, h3, h4, h5, h6, h7, h8};

2 2

gb := { - (u1\*u4 - u2\*u3)\*x14 + (u1 \*u3 + u1\*u2\*u4 - u1\*u3 - u2\*u3\*u4),

2 2

- (u1\*u4 - u2\*u3)\*x13 - (u1\*u2\*u3 - u1\*u3\*u4 + u2 \*u4 - u2\*u4 ),

2 2 2

(u1 + u2 )\*x06 - (u1\*u2\*u3 + u2 \*u4),

2 2 2

- (u1 + u2 )\*x05 + (u1 \*u3 + u1\*u2\*u4),

2 2 2

- (u3 + u4 )\*x04 + (u1\*u3\*u4 + u2\*u4 ),

2 2 2

- (u3 + u4 )\*x03 + (u1\*u3 + u2\*u3\*u4),

2 2 2 2

- (u1 - 2\*u1\*u3 + u2 - 2\*u2\*u4 + u3 + u4 )\*x02

2 2

+ (u1 \*u4 - u1\*u2\*u3 - u1\*u3\*u4 + u2\*u3 ),

2 2 2 2

- (u1 - 2\*u1\*u3 + u2 - 2\*u2\*u4 + u3 + u4 )\*x01

2 2

- (u1\*u2\*u4 - u1\*u4 - u2 \*u3 + u2\*u3\*u4)}

%---------------------------------------------------------;

% 「glterms」が出力するのは、グレブナー基底の計算過程で〈ゼロにはならない〉

% と仮定された式のリストである。

%---------------------------------------------------------;

%%% u に関する制約条件 --------------------------------------------------;

glterms;

{ - u3 + u1,

- u4 + u2,

u3,

u4,

u1,

u2,

2 2 2 2

u4 - 2\*u4\*u2 + u3 - 2\*u3\*u1 + u2 + u1 ,

2 2

u4 + u3 ,

2 2

u2 + u1 ,

u4\*u1 - u3\*u2}

%%% gbを法としてgを簡約 --------------------------------------------------;

preduce(h9, gb);

0

% ==> 0 になっているので、垂心Vについての仮定が成立 -----------------------------;

%----------------------------------------------------------------------;

%----------------------------------------------------------------------;

% 次に、重心Gについて考えてみる

% 辺BC,CA,ABの中点をL,M,Nとする

% B=L=C

h10:=midpoint(1, u1, u2, x07, x08, u3, u4);

h10 := 2\*x07 - u3 - u1

h11:=midpoint(0, u1, u2, x07, x08, u3, u4);

h11 := 2\*x08 - u4 - u2

% C=M=A

h12:=midpoint(1, u3, u4, x09, x10, 0, 0);

h12 := 2\*x09 - u3

h13:=midpoint(0, u3, u4, x09, x10, 0, 0);

h13 := 2\*x10 - u4

% A=N=B

h14:=midpoint(1, 0, 0, x11, x12, u1, u2);

h14 := 2\*x11 - u1

h15:=midpoint(0, 0, 0, x11, x12, u1, u2);

h15 := 2\*x12 - u2

% 重心GはAL, BM, CNの交点

% A-G-L かつ B-G-M かつ C-G-N

h16:=collinear(0, 0, x15, x16, x07, x08);

h16 := - x16\*x07 + x15\*x08

h17:=collinear(u1, u2, x15, x16, x09, x10);

h17 := - x16\*x09 + x16\*u1 + x15\*x10 - x15\*u2 - x10\*u1 + x09\*u2

h18:=collinear(u3, u4, x15, x16, x11, x12);

h18 := - x16\*x11 + x16\*u3 + x15\*x12 - x15\*u4 - x12\*u3 + x11\*u4

%----------------------------------------------------------------------;

% ここで一旦、重心Gについて仮定が成り立っているか確認してみる -----------------------;

%----------------------------------------------------------------------;

%----------------------------------------------------------------------;

% Groebner Basis: 結果が1となったら、仮定が誤っている可能性が高い ---------------;

%%% 変数を定義し、lex形式で並べる ------------------------------------------;

torder({x16, x15, x12, x11, x10, x09, x08, x07}, lex)$

%%% 仮定において定義した式からGroebner Basisを求める -------------------------;

gb:=groebner{h10, h11, h12, h13, h14, h15, h16, h17};

gb := { - 3\*x16 + (u2 + u4),

3\*x15 - (u1 + u3),

2\*x12 - u2,

2\*x11 - u1,

2\*x10 - u4,

2\*x09 - u3,

2\*x08 - (u2 + u4),

2\*x07 - (u1 + u3)}

%%% u に関する制約条件 --------------------------------------------------;

glterms;

{u3 + u1, - u3 + 2\*u1}

%%% gbを法としてgを簡約 --------------------------------------------------;

preduce(h18, gb);

0

% ==> 0 になっているので、重心Gについての仮定が成立 -----------------------------;

%----------------------------------------------------------------------;

%----------------------------------------------------------------------;

% 最後に、外心Xについて考えてみる

% 辺BC,CA,ABの中点をL,M,Nとする

% これは既に、 h10~h15 で定義されているので再利用する

% またXは, XL⊥BC, XM⊥CA, XN⊥AB を満たす

h19:=vertically(x17, x18, x07, x08, u1, u2, u3, u4);

h19 := x18\*( - u4 + u2) + x17\*( - u3 + u1) + x08\*(u4 - u2) + x07\*(u3 - u1)

h20:=vertically(x17, x18, x09, x10, u3, u4, 0, 0);

h20 := x18\*u4 + x17\*u3 - x10\*u4 - x09\*u3

h21:=vertically(x17, x18, x11, x12, 0, 0, u1, u2);

h21 := - x18\*u2 - x17\*u1 + x12\*u2 + x11\*u1

%----------------------------------------------------------------------;

% Groebner Basis: 結果が1となったら、仮定が誤っている可能性が高い ---------------;

%%% 変数を定義し、lex形式で並べる ------------------------------------------;

torder({x18, x17, x12, x11, x10, x09, x08, x07}, lex)$

%%% 仮定において定義した式からGroebner Basisを求める -------------------------;

gb:=groebner{h10, h11, h12, h13, h14, h15, h19, h20};

2 2 2 2

gb := { - (2\*u1\*u4 - 2\*u2\*u3)\*x18 - (u1 \*u3 - u1\*u3 - u1\*u4 + u2 \*u3),

2 2 2 2

- (2\*u1\*u4 - 2\*u2\*u3)\*x17 + (u1 \*u4 + u2 \*u4 - u2\*u3 - u2\*u4 ),

2\*x12 - u2,

2\*x11 - u1,

2\*x10 - u4,

2\*x09 - u3,

2\*x08 - (u2 + u4),

2\*x07 - (u1 + u3)}

%%% u に関する制約条件 --------------------------------------------------;

glterms;

{ - u4 + u2,u4,u4\*u1 - u3\*u2}

%%% gbを法としてgを簡約 --------------------------------------------------;

preduce(h21, gb);

0

% ==> 0 になっているので、外心Xについての仮定が成立 -----------------------------;

%----------------------------------------------------------------------;

% ---------------------------------------------------------------------;

% 結論 ----------------------------------------------------------------;

% このとき、△ABCの垂心V, 重心G, 外心Xは一直線に存在する

% V-G-X

conclusion:=collinear(x13, x14, x15, x16, x17, x18);

conclusion := x18\*x15 - x18\*x13 - x17\*x16 + x17\*x14 + x16\*x13 - x15\*x14

%----------------------------------------------------------------------;

% Groebner Basis: 結果が1となったら、仮定が誤っている可能性が高い ---------------;

%%% 変数を定義し、lex形式で並べる ------------------------------------------;

torder({x18, x17, x16, x15, x14, x13, x12, x11, x10, x09, x08, x07, x06, x05, x04, x03, x02, x01}, lex)$

%%% 仮定において定義した式からGroebner Basisを求める -------------------------;

gb:=groebner{h1, h2, h3, h4, h5, h6, h7, h8, h9, h10, h11, h12, h13, h14, h15, h16, h17, h18, h19, h20, h21};

2 2 2 2

gb := { - (2\*u1\*u4 - 2\*u2\*u3)\*x18 - (u1 \*u3 - u1\*u3 - u1\*u4 + u2 \*u3),

2 2 2 2

- (2\*u1\*u4 - 2\*u2\*u3)\*x17 + (u1 \*u4 + u2 \*u4 - u2\*u3 - u2\*u4 ),

- 3\*x16 + (u2 + u4),

3\*x15 - (u1 + u3),

2 2

- (u1\*u4 - u2\*u3)\*x14 + (u1 \*u3 + u1\*u2\*u4 - u1\*u3 - u2\*u3\*u4),

2 2

- (u1\*u4 - u2\*u3)\*x13 - (u1\*u2\*u3 - u1\*u3\*u4 + u2 \*u4 - u2\*u4 ),

2\*x12 - u2,

2\*x11 - u1,

2\*x10 - u4,

2\*x09 - u3,

2\*x08 - (u2 + u4),

2\*x07 - (u1 + u3),

2 2 2

(u1 + u2 )\*x06 - (u1\*u2\*u3 + u2 \*u4),

2 2 2

- (u1 + u2 )\*x05 + (u1 \*u3 + u1\*u2\*u4),

2 2 2

- (u3 + u4 )\*x04 + (u1\*u3\*u4 + u2\*u4 ),

2 2 2

- (u3 + u4 )\*x03 + (u1\*u3 + u2\*u3\*u4),

2 2 2 2

- (u1 - 2\*u1\*u3 + u2 - 2\*u2\*u4 + u3 + u4 )\*x02

2 2

+ (u1 \*u4 - u1\*u2\*u3 - u1\*u3\*u4 + u2\*u3 ),

2 2 2 2

- (u1 - 2\*u1\*u3 + u2 - 2\*u2\*u4 + u3 + u4 )\*x01

2 2

- (u1\*u2\*u4 - u1\*u4 - u2 \*u3 + u2\*u3\*u4)}

%---------------------------------------------------------;

% 「glterms」が出力するのは、グレブナー基底の計算過程で〈ゼロにはならない〉

% と仮定された式のリストである。

%---------------------------------------------------------;

%%% u に関する制約条件 --------------------------------------------------;

glterms;

{ - u3 + u1,

- u4 + u2,

u3,

u4,

u1,

u2,

2 2 2 2

u4 - 2\*u4\*u2 + u3 - 2\*u3\*u1 + u2 + u1 ,

2 2

u4 + u3 ,

2 2

u2 + u1 ,

u4\*u1 - u3\*u2,

u3 + u1,

- u3 + 2\*u1,

- 2\*u3 + u1}

%%% gbを法としてgを簡約 --------------------------------------------------;

preduce(conclusion, gb);

0

% ==> 0 になっていれば、定理は成立 ------------------------------------------;

showtime;

Time: 80 ms plus GC time: 40 ms

;

end;