主専攻実習（定理証明班）

第六回課題レポート（再提出）

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 26 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

```
% [定理　２５] ----------------------------------------------------------;
% △ABC の頂点 A,B,C から直線 g に降ろした垂線の足を D,E,F とすると、
% 点 D,E,F からそれぞれ BC, CA, AB におろした垂線 DP,EQ,FR は一点で交わる
% A(u1, u2) B(0, 0) C(u3, 0)
% D(x1, x2) E(x3, x4) F(x5, x6)
% P(x7, x8) Q(x9, x10) R(x11, x12)
%---------------------------------------------------------------------;

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

load_package groebner;
on comp,gcd,ezgcd;
off allfac,pwrds;

in cal_sys_relations$


% ２点間のユークリッド距離 D^2 ----------------------------------------------;
procedure squared_euclid(a1,a2,b1,b2)$
begin
 scalar d;
 d:=(a1-b1)^2+(a2-b2)^2;
 return d
end$
%---------------------------------------------------------------------;

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

order x12, x11, x10, x9, x8, x7, x6, x5, x4, x3, x2, x1, u3, u2, u1;
factor x12, x11, x10, x9, x8, x7, x6, x5, x4, x3, x2, x1;

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

% D-E-F は共線となっている
h1:=collinear(x1, x2, x3, x4, x5, x6);
```

```
% AĐ⊥EF , BE⊥FĐ , CF⊥ĐE
h2:=vertically(u1, u2, x1, x2, x3, x4, x5, x6);
h3:=vertically(0, 0, x3, x4, x5, x6, x1, x2);
h4:=vertically(u3, 0, x5, x6, x1, x2, x3, x4);

% 点Đ,E,FからそれぞれBC,CA,ABに下ろした垂線の足をP,Q,Rとする
% ĐP⊥BC かつ B-P-C
h5:=vertically(x1, x2, x7, x8, 0, 0, u3, 0);
h6:=collinear(0, 0, x7, x8, u3, 0);

% EQ⊥CA かつ C-Q-A
h7:=vertically(x3, x4, x9, x10, u3, 0, u1, u2);
h8:=collinear(u3, 0, x9, x10, u1, u2);

% FR⊥AB かつ A-R-B
h9:=vertically(x5, x6, x11, x12, u1, u2, 0, 0);
h10:=collinear(u1, u2, x11, x12, 0, 0);

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

% ĐP, EQ, FRの共点を言うためには、§6例3により、
% ĐB^2 - ĐC^2 + EC^2 -EA^2 + FA^2 -FB^2 = 0 を示せれば良い

db2:=squared_euclid(x1, x2, 0, 0);
dc2:=squared_euclid(x1, x2, u3, 0);
ec2:=squared_euclid(x3, x4, u3, 0);
ea2:=squared_euclid(x3, x4, u1, u2);
fa2:=squared_euclid(x5, x6, u1, u2);
fb2:=squared_euclid(x5, x6, 0, 0);

conclusion:=db2-dc2+ec2-ea2+fa2-fb2;


%-----------------------------------------------------------------------;
% 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({x12, x11, x10, x9, x8, x7, x6, x5, x4, x3, x2, x1}, 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};

%----------------------------------------------------------;
% 「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

```
% [定理 ２５] ----------------------------------------------------------;
% △ABC の頂点 A,B,C から直線 g に降ろした垂線の足を Đ,E,F とすると、
% 点 Đ,E,F からそれぞれ BC, CA, AB におろした垂線 ĐP,EQ,FR は一点で交わる
% A(u1, u2) B(0, 0) C(u3, 0)
% Đ(x1, x2) E(x3, x4) F(x5, x6)
% P(x7, x8) Q(x9, x10) R(x11, x12)
%----------------------------------------------------------------------;

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

load_package groebner;

on comp,gcd,ezgcd;

off allfac,pwrds;


in cal_sys_relations$


% ２点間のユークリッド距離 Đ^2 ------------------------------------------------;
procedure squared_euclid(a1,a2,b1,b2)$
begin
 scalar d;
 d:=(a1-b1)^2+(a2-b2)^2;
 return d
end$

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


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

order x12, x11, x10, x9, x8, x7, x6, x5, x4, x3, x2, x1, u3, u2, u1;


factor x12, x11, x10, x9, x8, x7, x6, x5, x4, x3, x2, x1;



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

% Đ-E-F は共線となっている
h1:=collinear(x1, x2, x3, x4, x5, x6);


h1 := x6*x3 - x6*x1 - x5*x4 + x5*x2 + x4*x1 - x3*x2


% AĐ⊥EF , BE⊥FĐ , CF⊥ĐE
h2:=vertically(u1, u2, x1, x2, x3, x4, x5, x6);


h2 := x6*x2 - x6*u2 + x5*x1 - x5*u1 - x4*x2 + x4*u2 - x3*x1 + x3*u1

h3:=vertically(0, 0, x3, x4, x5, x6, x1, x2);


h3 :=  - x6*x4 - x5*x3 + x4*x2 + x3*x1

h4:=vertically(u3, 0, x5, x6, x1, x2, x3, x4);


h4 := x6*x4 - x6*x2 + x5*x3 - x5*x1 - x3*u3 + x1*u3


% 点 Đ,E,F からそれぞれ辺 BC,CA,AB に下ろした垂線の足を P,Q,R とする
% ĐP⊥BC かつ B-P-C
h5:=vertically(x1, x2, x7, x8, 0, 0, u3, 0);
```

```
h5 := x7*u3 - x1*u3

h6:=collinear(0, 0, x7, x8, u3, 0);


h6 :=  - x8*u3


% EQ⊥CA かつ C-Q-A
h7:=vertically(x3, x4, x9, x10, u3, 0, u1, u2);


h7 := x10*u2 + x9*( - u3 + u1) - x4*u2 + x3*(u3 - u1)

h8:=collinear(u3, 0, x9, x10, u1, u2);


h8 := x10*(u3 - u1) + x9*u2 - u3*u2


% FR⊥AB かつ A-R-B
h9:=vertically(x5, x6, x11, x12, u1, u2, 0, 0);


h9 :=  - x12*u2 - x11*u1 + x6*u2 + x5*u1

h10:=collinear(u1, u2, x11, x12, 0, 0);


h10 := x12*u1 - x11*u2


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

% ÐP, EQ, FR の共点を言うためには、§６例３により、
% ÐB^2 - ÐC^2 + EC^2 -EA^2 + FA^2 -FB^2 = 0 を示せれば良い

db2:=squared_euclid(x1, x2, 0, 0);


         2     2
db2 := x2  + x1

dc2:=squared_euclid(x1, x2, u3, 0);


         2     2               2
dc2 := x2  + x1  - 2*x1*u3 + u3

ec2:=squared_euclid(x3, x4, u3, 0);


         2     2               2
ec2 := x4  + x3  - 2*x3*u3 + u3

ea2:=squared_euclid(x3, x4, u1, u2);


         2               2               2     2
ea2 := x4  - 2*x4*u2 + x3  - 2*x3*u1 + u2  + u1

fa2:=squared_euclid(x5, x6, u1, u2);


         2               2               2     2
fa2 := x6  - 2*x6*u2 + x5  - 2*x5*u1 + u2  + u1

fb2:=squared_euclid(x5, x6, 0, 0);


         2     2
fb2 := x6  + x5


conclusion:=db2-dc2+ec2-ea2+fa2-fb2;
```

```
conclusion :=  - 2*x6*u2 - 2*x5*u1 + 2*x4*u2 + x3*( - 2*u3 + 2*u1) + 2*x1*u3



%----------------------------------------------------------------------;
% 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({x12, x11, x10, x9, x8, x7, x6, x5, x4, x3, x2, x1}, 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};

             2     2          2
gb := { - (u1  + u2 )*x12 + u2 *x4 + (u1*u2 - u2*u3)*x3 + (u2*u3)*x1,

           2     2                        2
       (u1  + u2 )*x11 - (u1*u2)*x4 - (u1  - u1*u3)*x3 - (u1*u3)*x1,

           2              2     2          2
       (u1  - 2*u1*u3 + u2  + u3 )*x10 - u2 *x4 - (u1*u2 - u2*u3)*x3

                           2
        + (u1*u2*u3 - u2*u3 ),


                                                  （中略）,


                      3   3        3   2                            3   2
        - (u1 - u3)*x3 *x2  + u2*x3 *x2 *x1 + (2*u1*u2 - 3*u2*u3)*x3 *x2

                      3      2        2                 2    3
        - (u1 - u3)*x3 *x2*x1  + (2*u1  - 2*u1*u3 - 2*u2 )*x3 *x2*x1

             3     2           2       2       3           3   3
        - (u1  - u1 *u3 + u1*u2  - 3*u2 *u3)*x3 *x2 + u2*x3 *x1

                              3   2      2                     3    3
        - (2*u1*u2 + u2*u3)*x3 *x2  + (u1 *u2 + 2*u1*u2*u3 + u2 )*x3 *x1

             2           3       3                   2   3
        - (u1 *u2*u3 + u2 *u3)*x3  + (3*u1 - 3*u3)*x3 *x2 *x1

             2            2   3            2   2   2
        - (u1  - u1*u3)*x3 *x2  - (3*u2)*x3 *x2 *x1

                                2   2         2                    2   2
        - (4*u1*u2 - 8*u2*u3)*x3 *x2 *x1 + (u1 *u2 - 2*u1*u2*u3)*x3 *x2

                          2      3        2                 2    2      2
        + (3*u1 - 3*u3)*x3 *x2*x1  - (6*u1  - 6*u1*u3 - 5*u2 )*x3 *x2*x1

               3       2           2       2       2               2       2
        + (3*u1  - 3*u1 *u3 + u1*u2  - 7*u2 *u3)*x3 *x2*x1 + (u1*u2 *u3)*x3 *x2

                   2   4                         2   3
        - (3*u2)*x3 *x1  + (6*u1*u2 + 3*u2*u3)*x3 *x1

               2                       3    2   2
        - (3*u1 *u2 + 6*u1*u2*u3 + 2*u2 )*x3 *x1

               2             3       2                         3   2
        + (3*u1 *u2*u3 + 2*u2 *u3)*x3 *x1 - (3*u1 - 3*u3)*x3*x2 *x1

               2                 3                  2   3
        + (2*u1  - 2*u1*u3)*x3*x2 *x1 + (3*u2)*x3*x2 *x1

                                   2   2        2                       2
        + (2*u1*u2 - 7*u2*u3)*x3*x2 *x1  - (2*u1 *u2 - 4*u1*u2*u3)*x3*x2 *x1

                                4        2                 2          3
        - (3*u1 - 3*u3)*x3*x2*x1  + (6*u1  - 6*u1*u3 - 4*u2 )*x3*x2*x1
```

```
              3        2             2        2              2
        - (3*u1  - 3*u1 *u3 - u1*u2  - 5*u2 *u3)*x3*x2*x1

                 2                            5                             4
        - (2*u1*u2 *u3)*x3*x2*x1 + (3*u2)*x3*x1  - (6*u1*u2 + 3*u2*u3)*x3*x1

              2                      3       3        2          3           2
        + (3*u1 *u2 + 6*u1*u2*u3 + u2 )*x3*x1  - (3*u1 *u2*u3 + u2 *u3)*x3*x1

                     3   3      2            3   2        2   4
        + (u1 - u3)*x2 *x1  - (u1  - u1*u3)*x2 *x1  - u2*x2 *x1

                     2   3      2                       2   2                 5
        + (2*u2*u3)*x2 *x1  + (u1 *u2 - 2*u1*u2*u3)*x2 *x1  + (u1 - u3)*x2*x1

              2              2       4
        - (2*u1  - 2*u1*u3 - u2 )*x2*x1

            3     2           2     2          3          2          2         6
        + (u1  - u1 *u3 - u1*u2  - u2 *u3)*x2*x1  + (u1*u2 *u3)*x2*x1  - u2*x1

                              5      2                      4      2          3
        + (2*u1*u2 + u2*u3)*x1  - (u1 *u2 + 2*u1*u2*u3)*x1  + (u1 *u2*u3)*x1 }

%---------------------------------------------------------;
% 「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,

 u2,

  - u3 + u1,

 u1,

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

   2     2
 u2  + u1 ,

                2     2
  - u3*u1 + u2  + 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: 110 ms  plus GC time: 29 ms

;

end;
```