TD2 - Polynômes

Nicolas Estibals*

15 octobre, 22 octobre, 19 novembre 2008

Exercice 1 Racines

1. Trouver les racines de $X^5 - 3X^3 - 4X^2 + 2X + 4$.
2. Décomposer ce polynôme en produit de facteurs premiers dans $\mathbb{R}[X]$, puis dans $\mathbb{C}[X]$.
3. Essayer de faire de même avec $X^5 + 12X^4 - 3X^3 - 4X^2 + 2X + 4$.
4. Approcher les racines à 10^{-12} près.

Exercice 2 Polynômes de Чебышёв

On définit la suite de polynômes suivante :

$$\begin{cases} T_0(X) = 1 \\ T_1(X) = X \\ T_{n+2}(X) = 2X \cdot T_{n+1}(X) - T_n(X) \end{cases}$$

1. Calculer $T_2, T_3 \dots T_{10}$ (et l'exprimer de façon lisible !)
2. Tracer le graphe de ces polynômes sur l'intervalle $[-1, 1]$.
3. Calculer $T_{10}(\cos x)$.
4. Montrer par récurrence que l'on peut généraliser cette identité. (Indication : montrer que $\forall n \in \mathbb{N}, \forall x \in \mathbb{R}, T_n(\cos x) = \cos nx$.)
5. Montrer graphiquement que T_5 ne s'annule qu'entre -1 et 1.
6. *(facultatif)* Montrer que cela est vrai pour tout n.
7. Savez-vous traduire Чебышёв ? (Indication : l'exercice et ses notations aident !)

Exercice 3 A la recherche d'une racine cubique

On pose $P = X^9 + 6X^8 + 12X^7 + 17X^6 + 36X^5 + 36X^4 + 27X^3 + 54X^2 + 27$.

1. Chercher un polynôme Q tel que $Q^3 = P$. (Indication : il sera de bon ton de limiter ses recherches grâce à des considérations sur le degré d'un tel Q.)

Exercice 4 Algorithme d'Euclide

À ne traiter que si il vous reste du temps.

1. Écrire le pseudo-code pour l'algorithme d'Euclide pour le calcul du PGCD.
2. Écrire une procédure Maple qui implémente cet algorithme pour les polynômes.
3. Vérifier son bon fonctionnement sur quelques exemples grâce à la commande `gcd` (consulter l'aide pour connaître son fonctionnement).

*N'hésitez pas à m'écrire à l'adresse `Nicolas.Estibals@ens-lyon.fr`! Vous pouvez retrouver les sujets sur `http://perso.ens-lyon.fr/nicolas.estibals/`.

```
> # Exercice 1 : Racines
  restart;
> # Question 1/
  p:=X^5-3*X^3-4*X^2+2*X+4;
```

$$p := X^5 - 3\,X^3 - 4\,X^2 + 2\,X + 4$$

```
> # Pour afficher la séquence de toutes les racines d'un polynôme on peut
  utiliser :
  allvalues(RootOf(p,X));
```

$$-1,\ 1,\ 2,\ -1 + I,\ -1 - I$$

```
> # Question 2/
  # fatorisation dans R[X]
  factor(p);
```

$$(X - 1)\,(X - 2)\,(X + 1)\,(X^2 + 2\,X + 2)$$

```
> # factorisation dans C[X]
  # Pour factoriser dans C[X], il faut dire à Maple qu'il "a le droit"
  d'utiliser I.
  factor(p,I);
```

$$(X + (1 + I))\,(X + (1 - I))\,(X + 1)\,(X - 1)\,(X - 2)$$

```
> # Question 3/
  q:=X^5+12*X^4-3*X^3-4*X^2+2*X+4;
```

$$q := X^5 + 12\,X^4 - 3\,X^3 - 4\,X^2 + 2\,X + 4$$

```
> # Essayons comme précédement.
  allvalues(RootOf(q,X));
```

$$\begin{aligned}&\mathrm{RootOf}(\_Z^5 + 12\,\_Z^4 - 3\,\_Z^3 - 4\,\_Z^2 + 2\,\_Z + 4,\ index = 1),\\ &\quad \mathrm{RootOf}(\_Z^5 + 12\,\_Z^4 - 3\,\_Z^3 - 4\,\_Z^2 + 2\,\_Z + 4,\ index = 2),\\ &\quad \mathrm{RootOf}(\_Z^5 + 12\,\_Z^4 - 3\,\_Z^3 - 4\,\_Z^2 + 2\,\_Z + 4,\ index = 3),\\ &\quad \mathrm{RootOf}(\_Z^5 + 12\,\_Z^4 - 3\,\_Z^3 - 4\,\_Z^2 + 2\,\_Z + 4,\ index = 4),\\ &\quad \mathrm{RootOf}(\_Z^5 + 12\,\_Z^4 - 3\,\_Z^3 - 4\,\_Z^2 + 2\,\_Z + 4,\ index = 5)\end{aligned}$$

```
> # Question 4/
  # Comme Maple n'arrive pas à exprimer simplement les racines de q, il les
  laisse sous une forme formelle.
  # mais on peut quand même les approcher grâce à 'evalf'!
  # Petit rappel : on peut choisir la précision à laquelle Maple fait ses
  calculs grâce a Digits
  Digits:=12;
  evalf(allvalues(RootOf(q,X)));
```

$$Digits := 12$$

$$\begin{aligned}&0.669824081467 + 0.490501329078\,I,\ -0.560909288942 + 0.400464556951\,I,\\ &\quad -12.2178295851,\ -0.560909288942 - 0.400464556951\,I,\\ &\quad 0.669824081467 - 0.490501329078\,I\end{aligned}$$

```
> # Remarque : on peut aussi trouver une factorisation approchée de q grâce
  aux mots-clés 'real' et 'complex'.
  factor(q,real);
  factor(q,complex);
```

$$(X + 12.2178295851)\,(X^2 + 1.12181857788\,X + 0.474991091795)\,(X^2 - 1.33964816293\,X + 0.689255853940)$$

$$\begin{aligned}&(X + 12.2178295851)\,(X + (0.560909288942 + 0.400464556951\,I))\,(X\\ &\quad + (0.560909288942 - 0.400464556951\,I))\,(X + (-0.669824081467 + 0.490501329078\,I))\\ &\quad (X + (-0.669824081467 - 0.490501329078\,I))\end{aligned}$$

```
> # Exercice 2 : Polynômes de Tchebychev
  # Toujours commencer un exercice par la commande restart, cela permet de
  "dépoluer" l'espace des noms de variables.
  restart;
```

```
> # Question 1/
  # Préparons un tableau pour contenir les 11 premières itérations de la
  suite et initialisons les deux premières valeurs.
  T:=array(0..10);
  T[0]:=1;
  T[1]:=X;
```

$$T := \mathit{array}(0 \,..\, 10, [])$$

$$T_0 := 1$$

$$T_1 := X$$

```
> # Calculons les itérations 2 à 10 grâce à la formule de récurrence.
  for i from 2 to 10 do
          T[i]:=2*X*T[i-1]-T[i-2];
  od;
```

$$T_2 := 2X^2 - 1$$

$$T_3 := 2X(2X^2 - 1) - X$$

$$T_4 := 2X(2X(2X^2 - 1) - X) - 2X^2 + 1$$

$$T_5 := 2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X$$

$$T_6 := 2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) - 2X(2X(2X^2 - 1) - X) + 2X^2 - 1$$

$$T_7 := 2X(2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) - 2X(2X(2X^2 - 1) - X) + 2X^2 - 1) - 2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) + 2X(2X^2 - 1) - X$$

$$T_8 := 2X(2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) - 2X(2X(2X^2 - 1) - X) + 2X^2 - 1) - 2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) + 2X(2X^2 - 1) - X) - 2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) + 2X(2X(2X^2 - 1) - X) - 2X^2 + 1$$

$$T_9 := 2X(2X(2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) - 2X(2X(2X^2 - 1) - X) + 2X^2 - 1) - 2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) + 2X(2X^2 - 1) - X) - 2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) + 2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) + 2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) + 2X(2X^2 - 1) - X) + 2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) - 2X(2X(2X^2 - 1) - X) + 2X^2 - 1$$

$$T_{10} := 2X(2X(2X(2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) - 2X(2X(2X^2 - 1) - X) + 2X^2 - 1) - 2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) + 2X(2X^2 - 1) - X) - 2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) + 2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) + 2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) + 2X(2X^2 - 1) - X) + 2X(2X(2X(2X(2X^2 - 1) - X) - 2X^2 + 1) - 2X(2X^2 - 1) + X) - 2X(2X(2X^2 - 1) - X) + 2X^2 - 1$$

```
> # Le résultat n'étant pas très lisible, il faut utiliser la commande
  'simplify'.
  for i from 0 to 10 do
          T[i]:=simplify(T[i]);
  od;
```

$$T_0 := 1$$

$$T_1 := X$$

$$T_2 := 2X^2 - 1$$

$$T_3 := 4X^3 - 3X$$

$$T_4 := 8X^4 - 8X^2 + 1$$

$$T_5 := 16X^5 - 20X^3 + 5X$$

$$T_6 := 32X^6 - 48X^4 + 18X^2 - 1$$

$$T_7 := 64X^7 - 112X^5 + 56X^3 - 7X$$

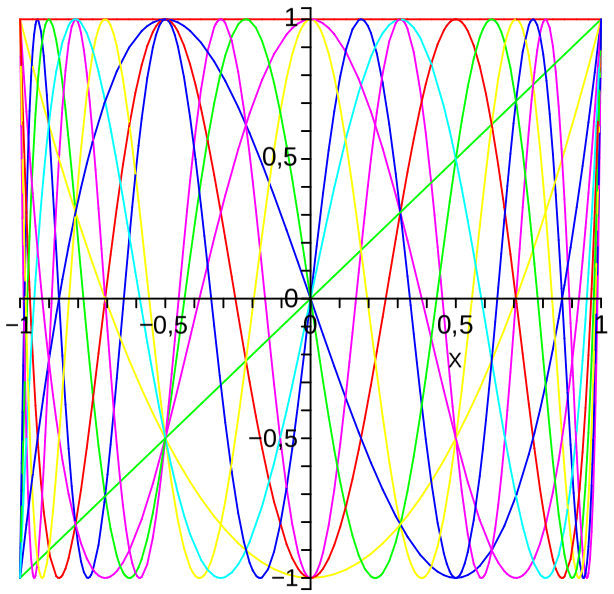
$$T_8 := 128X^8 - 256X^6 + 160X^4 - 32X^2 + 1$$

$$T_9 := 256X^9 - 576X^7 + 432X^5 - 120X^3 + 9X$$

$$T_{10} := 512X^{10} - 1280X^8 + 1120X^6 - 400X^4 + 50X^2 - 1$$

```
> # Question 2/
  # C'est la commande 'plot' qui permet de tracer les graphes de fonctions.
  # Pensez à mettre un titre : c'est plus agréable pour le correcteur et
```

```
cela vous permet de vous y retrouver.
plot(T,X=-1..1,title="Graphes des 11 premiers polynômes de Tchebychev");
```




```
> # Question 3/
  # La commande 'subs' permet de remplacer une variable dans une expressior
  subs(X=cos(x),T[10]);
```

$$512\cos(x)^{10} - 1280\cos(x)^{8} + 1120\cos(x)^{6} - 400\cos(x)^{4} + 50\cos(x)^{2} - 1$$

```
> # Nous avons ici une expression trigonométrique, la commande 'combine'
  permet de simplifier de telles expressions grâce à des identités comme
  celle que vous connaissez bien : cos(a + b) = cos(a) cos(b) - sin(a)
  sin(b).
  combine(%);
```

$$\cos(10\,x)$$

```
> # Question 4/
  # Montrons par récurrence que pour tout n et pour tout x T_n(cos x) =
  cos(n x).
  # Vérifions d'abord les cas de base (c'est à dire n=0 et n=1).
  # Assurons-nous que x n'est pas lié à une valeur dans l'environnement de
  calcul.
  unassign('x');
  # et que c'est un réel.
  assume(x::real);
  # Ce que nous venons de faire correspond à écrire dans une démonstration
  soit x un réel.

  # n=0
  evalb(subs(X=cos(x),T[0])= cos(0*x));
```

true

```
> # n=1
  evalb(subs(X=cos(x),T[1]) =cos(1*x));
```

true

```
> # Passons maintenant à la récurrence.
  # Soit n un entier >= 2,
  assume(n::integer);
  assume(n>1);

  # On suppose que T_{n-1}(cos x) = cos((n-1)x) et T_{n-2}(cos x) =
  cos((n-1)x)
  # On les remplace dans la formule de récurrence, T_n(cos x) est donc égal
  à :
  2*cos(x)*cos((n-1)*x)-cos((n-2)*x);
```

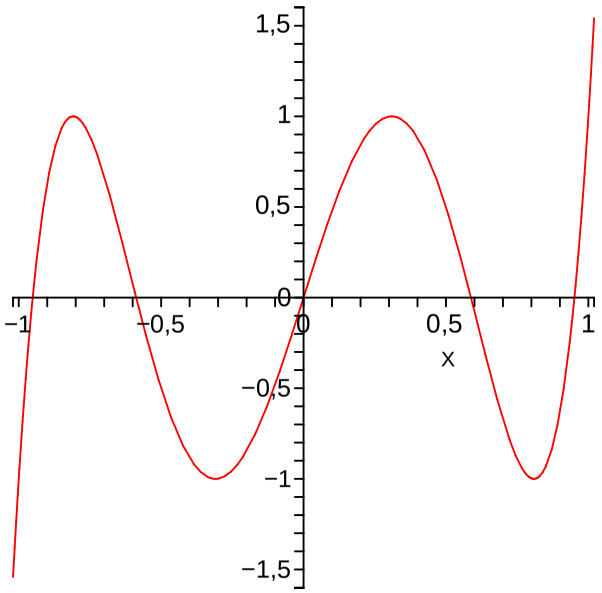
$$2\cos(x\sim)\cos((n\sim - 1)\,x\sim) - \cos((n\sim - 2)\,x\sim)$$

```
> # En réécrivant grâce à combine, on fait apparaître la solution.
  combine(%);
```

$$\cos(x\sim\, n\sim)$$

```
> # Ceci achève la preuve.
> # Question 5/
  # Traçons T_5(X) pour X dans [-1.02,1.02]
  plot(T[5],X=-1.02..1.02,title="Graphe de T_5");
```

Graphe de T_5




```
> # On peut constater que T_5 possède 5 zéros dans [-1,1], or T_5 est un
  polynôme de degré 5 et possède donc au plus 5 zéros d'après le théorème d
  d'Alembert.
  degree(T[5],X);
```

$$5$$

```
> # Question 6/
  # Cherchons des "candidats" pour être des zéros de T_n grâce à l'identité
  T_n(cos x) = cos(nx).
  allvalues(RootOf(cos(n*x),x));
```

$$\frac{\pi\,(1 + 2\,\_Z1\sim)}{2\,n\sim}$$

```
> # Si vous lisez le manuel pour la fonction allvalues (à faire!), vous
  constaterez que _Z1~ signifie une valeur entière.
  # Les valeurs suivantes sont donc des zéros de T_n:
  cos(%);
```

$$\cos\left(\frac{\pi\,(1 + 2\,\_Z1\sim)}{2\,n\sim}\right)$$

```
> # Notons d'abord que toutes ces valeurs sont dans [-1,1] (cosinus oblige!
  # Il est possible de vérifier que cos( (1+2k)/n * Pi/2) pour k = 0..n-1
  donne n zéros distincts :
  #         - 0<= (1+2k)/n*Pi/2 <= Pi,
  #         - les (1+2k)/n*Pi/2 sont distincts,
  #         - la fonction cos : [0,Pi] -> [-1,1] est bijective et donc
  surjective,
  #         - ce qui donne le résultat.
  # Comme T_n est de degré n (je vous laisse faire la démonstration par
  récurrence!),
  # T_n a n zéros au plus (d'Alembert) qui sont les cos( (1+2k)/n * Pi/2)
  pour k = 0..n-1
  # Ainsi pour tout n, T_n ne s'annule qu'entre -1 et 1.
```

```
> # Question 7/
  # L'exercice portait sur les polynômes de Tchebychev (Chebyshov en
  anglais, Tschebyscheff en vieux français), mathématicien russe également
  connu pour son inégalité en probabilité et son filtre en électronique.
```

```
> # Exercice 3 : A la recherche d'une racine cubique
> restart;
```

```
> # Donnons p à Maple.
> p:=X^9+6*X^8+12*X^7+17*X^6+36*X^5+36*X^4+27*X^3+54*X^2+27;
```

$$p := X^9 + 6\,X^8 + 12\,X^7 + 17\,X^6 + 36\,X^5 + 36\,X^4 + 27\,X^3 + 54\,X^2 + 27$$

```
> # q est forcément de degré 3 car p=q^3 est de degré 9.
> # De plus, p est unitaire donc q l'est aussi.
> # q est donc de la forme suivante :
> q:=X^3+a*X^2+b*X+c;
```

$$q := X^3 + a\,X^2 + b\,X + c$$

```
> # Il s'agit maintenant de résoudre le système d'équations que nous
> # donne l'annulation des coefficients de p-q^3 (en effet, on veut p = q^3
  :
> {seq(coeff(p-q^3,X,i)=0,i=0..9)};
```

$$\{17 - 4\,b\,a - 3\,c - a\,(2\,b + a^2) = 0,\ 12 - 3\,b - 3\,a^2 = 0,\ 6 - 3\,a = 0,\ -3\,c^2\,b = 0,$$
$$54 - 2\,b^2\,c - c\,(2\,c\,a + b^2) - a\,c^2 = 0,$$
$$27 - b\,(2\,c\,a + b^2) - 2\,a\,c\,b - c\,(2\,c + 2\,b\,a) - c^2 = 0,$$
$$36 - b\,(2\,c + 2\,b\,a) - a\,(2\,c\,a + b^2) - c\,(2\,b + a^2) - 2\,c\,b = 0,$$
$$36 - 4\,c\,a - b\,(2\,b + a^2) - a\,(2\,c + 2\,b\,a) - b^2 = 0,\ 27 - c^3 = 0,\ 0 = 0\}$$

```
> # que l'on resoud grâce à 'solve' :
  solve(%);
```

$$\{b = 0,\ c = 3,\ a = 2\}$$

```
> #Ce qui nous donne :
> subs(%,q);
```

$$X^3 + 2\,X^2 + 3$$

```
> # Exercice 4 : Algorithme d'Euclide
> restart;
```

```
> # Question 1/
  #
  # L'algorithme d'Euclide est basé sur le fait que si P et Q sont deux
  polynômes
  # (tels que d°(P)>=d°(Q)) et si R est le reste de la division de P par Q,
  on a
  # PGCD(P,Q) = PGCD(Q,R).
  #
  # ENTREE: P, Q deux polynômes
  # SORTIE: PGCD(P,Q)
  # SI d°(P)<d°(Q) ALORS
  #     échanger P et Q
  # TANT QUE d°(Q) >= 0 FAIRE
  #     R <- RESTE_DE_DIV(P,Q)
  #     P <- Q
  #     Q <- R
  # FAIT
  # RENVOYER P
```

```
> # Question 2/
  #
  pgcd := proc(P,Q,X)
  # Il va être nécessaire de connaître l'indéterminée des polynômes.
  local p,q,r: # Il faut déclarer ses variables locales.
  if degree(P,X)<degree(Q,X) then
      p:=Q: q:=P:
  else
      p:=P: q:=Q:
  fi:
  while degree(q,X) >= 0 do
      r:=rem(p,q,X):
      p:= q / lcoeff(q,X):
    # Nous rendons ici le polynôme p unitaire étant donné que le pgcd
    # est défini à une constante multiplicative près et qu'il est plus
    # agréable de travailler avec les polynômes unitaires.
      q:= r:
  od:
  return p;
  end proc;
```

```
pgcd := proc(P, Q, X)
local p, q, r;
    if degree(P, X) < degree(Q, X) then p := Q; q := P; else p := P; q := Q; end if;
    while 0 <= degree(q, X) do r := rem(p, q, X); p := (q)/(lcoeff(q, X)); q := r; end do;
    return p;
end proc;
```

```
> # Question 3/
  gcd(X**3,X+1);
  pgcd(X**3,X+1,X);
```

$$1$$

$$1$$

```
> gcd(X**3+X**2+X+1,X**3+X);
  pgcd(X**3+X**2+X+1,X**3+X,X);
```

$$X^2 + 1$$

$$X^2 + 1$$