Maple Lecture 10. Univariate and Multivariate Polynomials

We will see the relation between symbolic factorization and numerical root finding. As we have seen with algebraic numbers, the question on whether a polynomial factors or not is directly related to the choice of number field. Formally, we keep working exactly by adding roots as symbols. Numerically, we work over the complex numbers and apply the fundamental theorem of algebra.

The material of this lecture corresponds to [1, Sections 5.1 and 5.2].

10.1 Univariate Polynomials

As example, let us make a general polynomial of degree five:

```
[> p := sum('c||i*x^i','i'=0..5);   # notice the right quotes...
```

Let us first consider the basic selectors on a polynomial:

```
[> whattype(p); type(p,polynom);
```

Let us select the highest degree coefficient:

```
[> d := degree(p,x); coeff(p,x,d);
```

With coeff in a for loop, we can construct the sequence of all coefficients of `p`, but Maple provides a command to do this:

```
[> coeffs(p,x);
[> coeffs(p,x,'powers'): powers;
[> sort(p,x);
```

Besides the obvious addition and multiplication, we can also divide polynomials. Here is one problem we wish to solve: find conditions on the coefficients of p so that it is divisible by, say $x + 1$.

```
[> dv := x+1;                         # divisor
[> q := quo(p,dv,x,'rest');           # find quotient and remainder
[> rest;
[> testp := q*dv + rest;
[> expand(testp);
```

With rest we have found a condition on the coefficient of `p` for it to be divisible by $x + 1$. Here is how we can use this condition to generate a random polynomial divisible by $x + 1$.

```
[> for i from 0 to 4 do
 >    c||i := rand():
 > end do;
[> rest;
[> sol := solve({rest=0},c5);
[> assign(sol);
[> rest;
[> p;
[> gcd(p,dv);
[> rem(p,dv,x);
```

What does it mean for p to be divisible by x+1? Well, let us check the roots of `p`:

```
[> solve(p,x);
```

We know that every polynomial of degree five has five roots. Since `p` is divisible by $x + 1$, -1 is a root. The four other roots are roots of the quotient polynomial `q`. Because the coefficients were generically chosen, `q` does not factor over the rationals.

```
[> q;
[> factor(q);
```

But q has real roots:

```
[> fsolve(q);
[> Digits := 30;
[> rts := fsolve(q,x,complex);
[> q1 := product('x-rts[i]','i'=1..4);
[> lcoeff(q);
[> q2 := expand(lcoeff(q)*q1);
[> q - q2;
```

This is a numerical factorization of the polynomial `p`, using the fundamental theorem of algebra: every polynomial has exactly as many complex roots as its degree. Note that we can also work with algebraic numbers to factor the polynomial exactly.

```
[> q;
[> solsq := solve(q,x);
[> alias(alpha=RootOf(q));
[> simplify(subs(x=alpha,q));      # sanity check
[> qq := quo(q,x-alpha,x,'rq');    # compute quotient
[> simplify(qq);
[> rq;
[> simplify(rq);
[> factor(q,alpha);                # factor over field extension
```

The last command shows the factorization of `q` over the rational numbers, extended with α. We denote this field extension by $\mathbb{Q}(\alpha)$. The connection between formal and numerical approaches goes via evalf:

```
[> evalf(alpha,50);
```

10.2 Multivariate Polynomials

The general polynomial p we started with in 10.1 was actually a polynomial in several variables. The important difference is that the order of monomials in a polynomial with several variables gets more complicated:

```
[> p1 := sum(sum('c[i,j]'*x^'i'*y^'j','i'=0..2),'j'=0..2);
[> sort(p1,[x,y],plex);            # pure lexicographical order
[> sort(p1,[x,y],tdeg);            # total degree order
```

The pure lexicographical order sorts the monomials in a polynomial as words in a dictionary. The order of the letters is determined by the order in the list `[x,y]`, give as argument in **sort**. The total degree order sorts the monomials first according to their degree and then sorts monomials with the same degree lexicographically.

We can view the polynomial as a polynomial in `x` or `y`:

```
[> collect(p1,x); sort(collect(p1,y),[y,x]);
[> p2 := x - y;
```

We ask a similar question as before. Is it possible to find values of `a` and `b` such that `p1` is divisible by `p2`?

```
[> restx := rem(p1,p2,x);
[> collect(restx,y);
[> resty := rem(p1,p2,y);
[> collect(resty,x);
[> eqs := {coeffs(resty,x)};
```

Maple can factor multivariate polynomials over the complex numbers (the so-called "absolute factorization"). For example:

```
[> p := x^2 - 2*y^2;
[> f := evala(AFactor(p));          # returns the exact factorization
[> map(evalf,f);                    # see a numerical approximation
[> convert(f,radical);              # a nicer representation
```

10.3 Assignments

1. Give the Maple command to generate the polynomial $w = (x-1)(x-2)\cdots(x-20)$.
   - (a) Do $e :=$ expand(w); and let Maple find the roots of the expanded polynomial e, just use solve.
   - (b) Do $f :=$ convert(e, float); and apply fsolve(f, x, complex); to find the roots.
     Did you expect this output? Compare fsolve with solve.
2. Consider the polynomials
   $$p = 2x^5 + 11x^4 + 14x^3 + 11x^2 + 12x \quad \text{and} \quad q = 2x^5 + 5x^4 + 7x^3 + 8x^2 + 5x + 3.$$
   - (a) Give the Maple command(s) to compute the greatest common divisor of p and q.
   - (b) How can you use Maple to find polynomials k and l so that $\gcd(p,q) = kp + lq$?
   - (c) Finally, give the Maple command(s) to verify the relation $\gcd(p,q) = kp + lq$ for the k and l found.

   Hint: look at the help page for the command **gcdex**.
3. Consider the polynomial
   $$p = 2x^5 + 9x^4 + 16x^3 + 15x^2 + 12x + 9.$$
   Write p as a product of linear factors:
   - (a) symbolically, by adding sufficiently many formal roots; and
   - (b) numerically, by finding all complex roots of p.
4. Consider $f = x^2yz + x^2y - 2x^2 + xy^2z - xy^2 - xz + yz + 9$.
   Give the Maple command to transform f into
   $$((z+1)y - 2)x^2 + ((z-1)y^2 - z)x + yz + 9.$$
5. Let $p = 2yz^2 + xz^3 + 2xz^4$. Give the Maple commands to bring p in the following forms:
   (a) $2z^4x + z^3x + 2z^2y$;   (b) $2xz^4 + xz^3 + 2z^2y$;   (c) $2yz^2 + 2xz^4 + xz^3$.
6. Consider $p = xy - xy^2 + x^2y^2z - x^2yz$.
   - (a) Using the **op** command on p, draw the expression tree of p.
   - (b) Do $q :=$ collect($p, [x,z]$, recursive); and draw the expression tree of q.
   - (c) Do $r :=$ factor(p); and draw the expression tree of r.
   - (d) Which expression, p, q, or r, requires the least amount of work to evaluate? Justify by letting Maple count the number of arithmetical operations needed to evaluate p, q, and r.

References

[1] A. Heck. *Introduction to Maple*. Springer-Verlag, third edition, 2003.