Experiments with a cubic polynomial equation

Consider the cubic polynomial equation

$$f(x) = x^3 - 44\,x^2 + 564\,x - 1728 = 0. \tag{1}$$

1. Prove that equation (1) has three real roots. Exhibit three sign change intervals, one for each root.

2. Write a Maple program to implement Newton's method.

3. Compute numerical approximations, of up to 10 decimal digits of accuracy, to all three real roots of equation (1), using Newton's method.

4. Denote by ρ_1, ρ_2, ρ_3 the three real roots of equation (1).

   (a) Find the values of the three quantities: $\rho_1 + \rho_2 + \rho_3$, $\rho_1\rho_2 + \rho_2\rho_3 + \rho_1\rho_3$, $\rho_1\rho_2\rho_3$
   Viète's formulae, Elementary Symmetric Functions

   (b) Compute the *exact* values of the sums $\rho_1^2 + \rho_2^2 + \rho_3^2$, $\rho_1^3 + \rho_2^3 + \rho_3^3$, $\dfrac{1}{\rho_1^2} + \dfrac{1}{\rho_2^2} + \dfrac{1}{\rho_3^2}$
   Fundamental Theorem of Symmetric Functions, `simplify(expr,[siderels]);`

5. **Basins of Attraction for Newton's Method**
   The question which initial guesses z_0 in Newton's method lead to convergence in which one of the roots ρ_1, ρ_2, ρ_3 of equation (1), is the question of computing the basins of attraction of each one of the three roots. More specifically, the three basins of attraction of the roots ρ_1, ρ_2, ρ_3 are defined as:

   $$B_{\rho_1} = \{z_0 \in \mathbf{C} \mid \text{ Newton's method with initial value } z_0 \text{ converges to } \rho_1\}.$$

   $$B_{\rho_2} = \{z_0 \in \mathbf{C} \mid \text{ Newton's method with initial value } z_0 \text{ converges to } \rho_2\}.$$

   $$B_{\rho_3} = \{z_0 \in \mathbf{C} \mid \text{ Newton's method with initial value } z_0 \text{ converges to } \rho_3\}.$$

   Our objective is to produce a relatively accurate drawing of the three basins of attraction B_{ρ_1}, B_{ρ_2}, B_{ρ_3}, in the region of the complex plane defined by a square of side 60, centered around $(0,0)$.

   (a) Consider the region of the complex plane, defined by a square of side 60, centered around $(0,0)$. Consider the x-coordinate values $x = -30, -29, \ldots -1, 0, 1, \ldots 29, 30$ and the y-coordinate values $y = -30, -29, \ldots -1, 0, 1, \ldots 29, 30$. For each pair of x, y values, make up the complex number $z_0 = x \ + \ y\ i$ and use it as an initial guess for Newton's method for equation (1).

   (b) For each Newton's method from the previous step, observe whether the method converges to ρ_1, ρ_2, or ρ_3 or if it doesn't converge. Use the same accuracy ε for all Newton's methods. Using these observations, classify each z_0 into one of the sets (basins of attraction) B_{ρ_1}, B_{ρ_2}, B_{ρ_3}, or disregard it (case of non-convergence).

(c) Assign a different color to each basin of attraction B_{ρ_1}, B_{ρ_2}, B_{ρ_3} and plot the points belonging to these sets, using your color scheme. This should give you an approximate picture of the basins of attraction.

(d) Repeat the process with refining further the subdivision of the intervals on the x and y axes, to include for instance points with floating point coordinates, e.g. -30, -29.5, -29, ..., -1, -0.5, 0, 0.5, 1, ..., 29.5, 30. This should yield a more accurate picture of the basins of attraction.

6. **Solving polynomial equations by homotopies**
The cubic equation (1) can be solved numerically by the method of homotopies as follows. Consider the cubic polynomial equation $g(x) = (x-1)(x-2)(x-3) = 0$, with the three obvious roots $x = 1, x = 2, x = 3$. Consider a parameter λ, taking values in the unit interval $[0, 1]$ and define the *convex homotopy* function

$$H(x, \lambda) = \lambda f(x) + (1 - \lambda)g(x).$$

Notice that we have, $H(x, 0) = g(x)$ and $H(x, 1) = f(x)$. Take n equally spaced values of λ in the unit interval $[0, 1]$, $\lambda_1, \ldots, \lambda_n$ and solve the n equations $H(x, \lambda_1) = 0, \ldots,$ $H(x, \lambda_n) = 0$ by Newton's method, using for initial conditions: 1, 2, 3 for $H(x, \lambda_1) = 0$, the 3 roots of $H(x, \lambda_1) = 0$ as initial conditions for $H(x, \lambda_2) = 0$ and so forth. Finally use the 3 roots of $H(x, \lambda_n) = 0$, to solve $f(x) = 0$. The parameter n can be taken to be 10, 100, 1000, successively.

Maple code accompanying

"Experiments with a cubic polynomial equation"

very useful resource for Maple enthusiasts

http://www.maplesoft.com/applications/

maple APPLICATION CENTER

offers over 1500 Maple applications

```
> restart;
> f := x -> x^3 - 44*x^2 + 564*x-1728;
```

$$f := x \to x^3 - 44\,x^2 + 564\,x - 1728$$

```
> Digits := 15;         # an outome of a FAPP thinking process
```

$$Digits := 15$$

```
> epsilon := 10^(-6);   # another outome of a FAPP thinking process
```

$$\varepsilon := \frac{1}{1000000}$$

```
> Df := D(f);
```

$$Df := x \to 3\,x^2 - 88\,x + 564$$

the fundamental Newton iteration

```
> na:= x -> evalf(x-f(x)/Df(x));
```

$$na := x \to \mathrm{evalf}\left(x - \frac{\mathrm{f}(x)}{\mathrm{Df}(x)}\right)$$

a procedure to implement Newton's iteration

```
> nit := proc(xinit)
     local xcurrent,xnext;
      xcurrent := xinit;
      xnext := na(xinit);
      while abs(f(xcurrent)-f(xnext)) > epsilon do
          xcurrent := xnext;
          xnext := na(xnext);
      od;
      xnext;
  end proc;
```

nit := **proc**(*xinit*)

local *xcurrent*, *xnext*;

xcurrent := *xinit*; *xnext* := na(*xinit*); **while** $\varepsilon <$ abs(f(*xcurrent*) – f(*xnext*)) **do** *xcurrent* := *xnext*; *xnext* := na(*xnext*) **end do**; *xnext*

end proc

compute numerical approximations to all three real roots rho1, rho2, rho3

(some reverse engineering is lurking here)

```
> nit(1);  nit(14);   nit(30);
```

$$4.45599625468248$$

$$18.0000000000004$$

$$21.5440037453173$$

check the result of these computations using Maple's solve and fsolve

```
> solve(f(x));
  fsolve(f(x));
```

$$18,\ 13+\sqrt{73},\ 13-\sqrt{73}$$

$$4.45599625468247,\ 18.0000000000000,\ 21.5440037453175$$

Maple allows for multiple assignment

```
> rho1,rho2,rho3 := %%;
```

$$\rho1,\ \rho2,\ \rho3 := 18,\ 13+\sqrt{73},\ 13-\sqrt{73}$$

Viete's formulae, Fundamental Theorem of Symmetric Functions

```
> simplify(rho1+rho2+rho3);
  simplify(rho1*rho2+rho2*rho3+rho3*rho1);
  simplify(rho1*rho2*rho3);
```

$$44$$

$$564$$

$$1728$$

```
> simplify(rho1^2+rho2^2+rho3^2);
  simplify(r1^2+r2^2+r3^2,[r1+r2+r3=e1,r1*r2+r2*r3+r3*r1=e2,r1*r2*r3=e3]);
```

$$808$$

$$e1^2 - 2\ e2$$

```
> simplify(rho1^3+rho2^3+rho3^3);
  simplify(r1^3+r2^3+r3^3,[r1+r2+r3=e1,r1*r2+r2*r3+r3*r1=e2,r1*r2*r3=e3]);
```

$$15920$$

$$e1^3 - 3\ e2\ e1 + 3\ e3$$

```
> simplify((1/rho1^2)+(1/rho2^2)+(1/rho3^2));  numer(%)/expand(denom(%));
  simplify((1/rho1^2)+(1/rho2^2)+(1/rho3^2),[r1+r2+r3=e1,r1*r2+r2*r3+r3*r1=e2,r1*r2*r3=e3]);
```

$$\frac{4612}{9\ (13+\sqrt{73}\ )^2\ (-13+\sqrt{73}\ )^2}$$

$$\frac{1153}{20736}$$

$$\frac{1153}{20736}$$

build and plot the basins of attraction for the three roots rho1, rho2, rho3

```
> rho1,rho2,rho3 := fsolve(f(x));

  BArho1 := {}:
  BArho2 := {}:
  BArho3 := {}:

  square := 30;
  step := 0.5;

  for xi from -square by step to square do
  for yi from -square by step to square do

  # build a complex number x0, to be used as an initial condition
  x0 := evalf(xi+yi*I);

  # run Newton's method with this initial condition
  aux := nit(x0);

   if abs(aux - rho1) < epsilon then BArho1 := BArho1 union {x0}; end if;
   if abs(aux - rho2) < epsilon then BArho2 := BArho2 union {x0}; end if;
   if abs(aux - rho3) < epsilon then BArho3 := BArho3 union {x0}; end if;

  # note that abs can deal with complex numbers
  od;
  od;
```

$$\rho1, \rho2, \rho3 := 4.45599625468247, 18.0000000000000, 21.5440037453175$$

$$square := 30$$

$$step := 0.5$$

consistency check

```
> nops(BArho1);nops(BArho2);nops(BArho3);
  %+%%+%%%;
  (2*square*(1/step)+1)^2;
```

7645

5882

1114

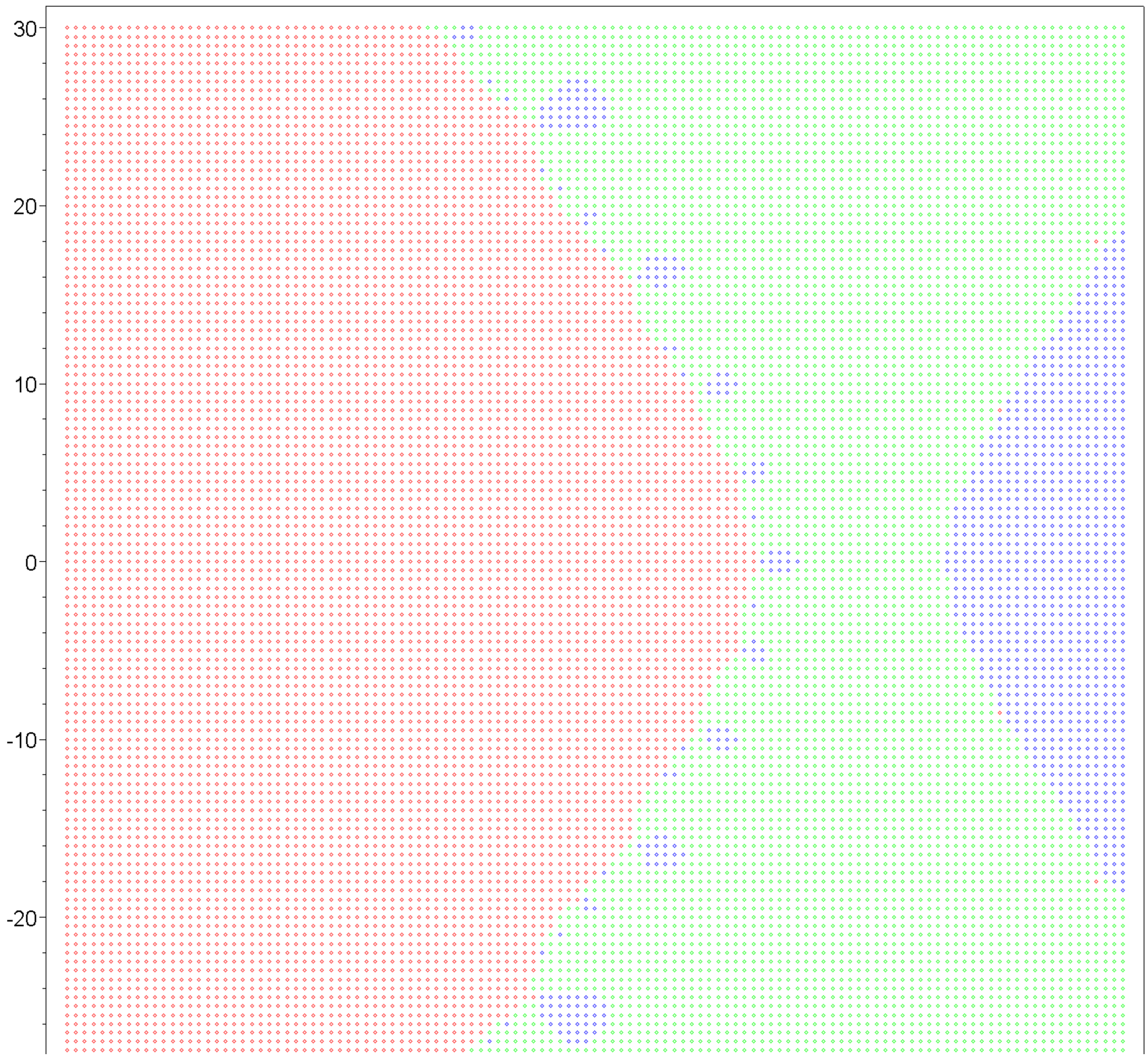
14641

14641.0000000000

```
> with(plots):

  BA1Plot := complexplot(BArho1,style=point,axes=box,color=red):
  BA2Plot := complexplot(BArho2,style=point,axes=box,color=green):
  BA3Plot := complexplot(BArho3,style=point,axes=box,color=blue):

  display({BA1Plot,BA2Plot,BA3Plot});
```


30
20
10
0
-10
-20