There exists no algorithm for solving the quadratic equation with real coefficients neither in $\mathbb{R}$ nor in $\mathbb{C}$

Giorgos Bougioukas*, 2016
georgebougioukas@yahoo.com

Abstract

This paper proves that the problem of the equality of two real constants is undecidable and as a consequence there exists no algorithm for solving the (univariate) quadratic equation with real coefficients neither in $\mathbb{R}$ nor in $\mathbb{C}$.

1. Definitions and assumptions

a) An algorithm is a Turing machine.

b) Assume that an algorithm (A1) for solving the quadratic equation with real coefficients in $\mathbb{R}$ necessarily gives an output that implies the decidability of the decision problem “is there a real solution for the given equation?” (P1), e.g:

   a list, empty if there exists no real solution for the given equation, with up to two (distinct) elements, otherwise.

c) Assume that an algorithm (A2) for solving the quadratic equation with real coefficients in $\mathbb{C}$ necessarily gives an output that implies the decidability of the decision problem “has the given equation exactly one complex solution?” (P2), e.g:

   a list, with one element if the given equation has exactly one complex solution, with two elements, if the given equation has exactly two (distinct) complex solutions.

*Graduate of Informatics (Hellenic Open University)
2. The problem of the equality of two real constants is undecidable (Thm1)

Proof

Let C(TM) be a string that encodes a Turing machine TM. For every Turing machine TM and for every input string (on the input alphabet of TM) x consider the following series:

$$\sum_{k=1}^{\infty} 10^{-k} h_{C(TM),x}(k),$$

wherein $h_{C(TM),x}$ a sequence of natural numbers,

for $n \geq 1$:

$$h_{C(TM),x}(n) = \begin{cases} 1, & \text{TM—encoded by } C(TM) — \text{on input } x \text{ halts after } n \text{ steps or less} \\ 0, & \text{otherwise} \end{cases}$$

The series above converges in $\mathbb{R}$ as $10^{-k} h_{C(TM),x}(k) \leq 10^{-k}$ holds for every natural number $k \geq 1$ and the series of the sequence $10^{-k}$ converges as geometric series with common ratio $10^{-1}$ for which it holds that $|10^{-1}|<1$. Therefore, for every Turing machine TM and for every input string (on the input alphabet of TM) x the following lemma holds:

$$\sum_{k=1}^{\infty} 10^{-k} h_{C(TM),x}(k) \text{ constant } \in \mathbb{R} \quad (L1)$$

Obviously, for every Turing machine TM and for every input string (on the input alphabet of TM) x the following lemma also holds:

$$\sum_{k=1}^{\infty} 10^{-k} h_{C(TM),x}(k) = 0 \iff \text{TM on input x does not halt} \quad (L2)$$

Consider a Turing machine TM$_1$ on input (on the input alphabet of TM$_1$) x$_1$. Assuming that there exists an algorithm (A3) that decides whether or not two real constants are equal, it follows that there exists another algorithm (A4) that:

a) simulating the algorithm A3 decides whether or not it holds that (the first member of the equality below is a real constant by L1):

$$2$$
\[
\sum_{k=1}^{\infty} 10^{-k} h_{c(TM_1),x_1}(k) = 0
\]

b) via the previous result and \(L_2\) decides whether or not the Turing machine \(TM_1\) on input \(x_1\) halts.

In other words, the algorithm \(A_4\) decides the halting problem, a result that contradicts Turing’s “halting theorem” (Halatsis, 2003). We reached a contradiction because we assumed that there exists an algorithm such \(A_3\). Therefore, there exists no such algorithm. □

3. There exists no algorithm such \(A_1\)

Proof

Assuming that there exists an algorithm such \(A_1\), via the assumption b) in section 1, it follows that there exists another algorithm \((A_5)\) that simulating the former, given any quadratic equation with real coefficients, decides the decision problem \(P_1\). If the form of output of the algorithm \(A_1\) is identical with the one in the example of the assumption b) in section 1, then the algorithm \(A_5\) decides \(P_1\) as follows:

- If the algorithm \(A_1\) gives as output a non-empty list – i.e there exists a real solution for the given equation – then gives as output “True”.

- Otherwise, if the algorithm \(A_1\) gives as output an empty list – i.e there exists no real solution for the given equation – then gives as output “False”.

Let \(r,s\) be real constants and \(d=|s-r|\). Consider the following quadratic equation with real coefficients:

\[
x^2 + \sqrt{d}x + d = 0 \quad (E_1)
\]

Given an instance of the problem \(P_1\) consisting of \(E_1\), the algorithm \(A_5\) gives the following output:
- Either “True”, i.e. there exists a real solution for \( E_1 \) – therefore, letting \( \Delta \) be the discriminant of \( E_1 \):

\[
\Delta \geq 0 \Rightarrow (\sqrt{d})^2 - 4d \geq 0 \Rightarrow d - 4d \geq 0 \Rightarrow -3d \geq 0 \Rightarrow d \leq 0 \Rightarrow |s-r| \leq 0 \Rightarrow |s-r| = 0 \Rightarrow s=r
\]

- Or “False”, i.e. there exists no real solution for \( E_1 \) – therefore, letting \( \Delta \) be the discriminant of \( E_1 \):

\[
\Delta < 0 \Rightarrow (\sqrt{d})^2 - 4d < 0 \Rightarrow d - 4d < 0 \Rightarrow -3d < 0 \Rightarrow d > 0 \Rightarrow |s-r| > 0 \Rightarrow s > r \lor s < r \Rightarrow s \neq r
\]

The real constants \( r \) and \( s \) have no particular properties. The same is true for the algorithm \( A_5 \), except that it uses the output of the algorithm \( A_1 \), which has also no particular properties, except the assumed form of its output. Therefore, for all constants \( s, r \in \mathbb{R} \) the equation \( E_1 \) can be constructed and given as input to the algorithm \( A_5 \) whose output then implies that either \( s=r \) or \( s\neq r \), which contradicts with \( \text{Thm1} \). We reached a contradiction because we assumed that there exists an algorithm such \( A_1 \). Therefore, there exists no such algorithm. □

4. There exists no algorithm such \( A_2 \)

Proof

Assuming that there exists an algorithm such \( A_2 \), via the assumption c) in section 1, it follows that there exists another algorithm \( (A_6) \) that simulating the former, given any quadratic equation with real coefficients, decides the decision problem \( P_2 \). If the form of output of the algorithm \( A_2 \) is identical with the one in the example of the assumption c) in section 1, then the algorithm \( A_6 \) decides \( P_2 \) as follows:

- If the algorithm \( A_2 \) gives as output a list with one element, then gives as output “True”.

- Otherwise, if the algorithm \( A_2 \) gives as output a list with two elements, then gives as output “False”.

4
Let $r,s$ be real constants and $d=|s-r|$. Consider the quadratic equation with real coefficients $E_1$.

Given an instance of the problem $P_2$ consisting of $E_1$, the algorithm $A_6$ gives the following output:

- Either “True”, i.e there exists exactly one complex solution for $E_1$ in $\mathbb{C}$—therefore, letting $\Delta$ be the discriminant of $E_1$:

\[
\Delta = 0 \Rightarrow (\sqrt{d})^2 - 4d = 0 \Rightarrow d - 4d = 0 \Rightarrow -3d = 0 \Rightarrow d = 0 \\
|s-r| = 0 \Rightarrow s-r = 0 \lor r-s = 0 \Rightarrow s=r
\]

- Or “False”, i.e there exist exactly two (distinct) complex solutions for $E_1$ in $\mathbb{C}$—therefore, letting $\Delta$ be the discriminant of $E_1$:

\[
\Delta \neq 0 \Rightarrow (\sqrt{d})^2 - 4d \neq 0 \Rightarrow d - 4d \neq 0 \Rightarrow -3d \neq 0 \Rightarrow d \neq 0 \Rightarrow \\
|s-r| \neq 0 \Rightarrow s-r \neq 0 \lor r-s \neq 0 \Rightarrow s \neq r
\]

The real constants $r$ and $s$ have no particular properties. The same is true for the algorithm $A_6$, except that it uses the output of the algorithm $A_2$, which has also no particular properties, except the assumed form of its output. Therefore, for all constants $s, r \in \mathbb{R}$ the equation $E_1$ can be constructed and given as input to the algorithm $A_6$ whose output then implies that either $s=r$ or $s \neq r$, which contradicts with Thm1. We reached a contradiction because we assumed that there exists an algorithm such $A_2$. Therefore, there exists no such algorithm. □

References