Contents

Widening the Scope of Extrema Problems

Giora Mann and Nurit Zehavi
Weizmann Institute of Science, ISRAEL
email: gioram@mofet.macam98.ac.il
email: ntzehavi@weizmann.weizmann.ac.il

Introduction

What mathematical knowledge will students obtain in our new technological age when Computer Algebra Systems take over performing of routine tasks? How will the mathematics curriculum and didactics change? These questions are frequently asked, not only by educators and policy makers who are reluctant to adjust to the changing times, but also, and with much thought, by CAS enthusiasts. In this paper we will discuss some answers and conclusions that were reached in attracting teachers to use CAS as their professional tool. During the last three years we have designed and implemented professional development courses with the main goal of enabling teachers to take advantage of the possibilities offered by CAS. In preparing problems to be solved by the teachers, we have explored the options for widening the scope of the traditional repertoire of mathematical problems. In the following we will explain this in detail with examples.

Four Extrema problems

Clearly, CAS technology has changed the way in which we solve Extrema problems. While the software is performing tedious calculations we can devote ourselves to the main algorithm:

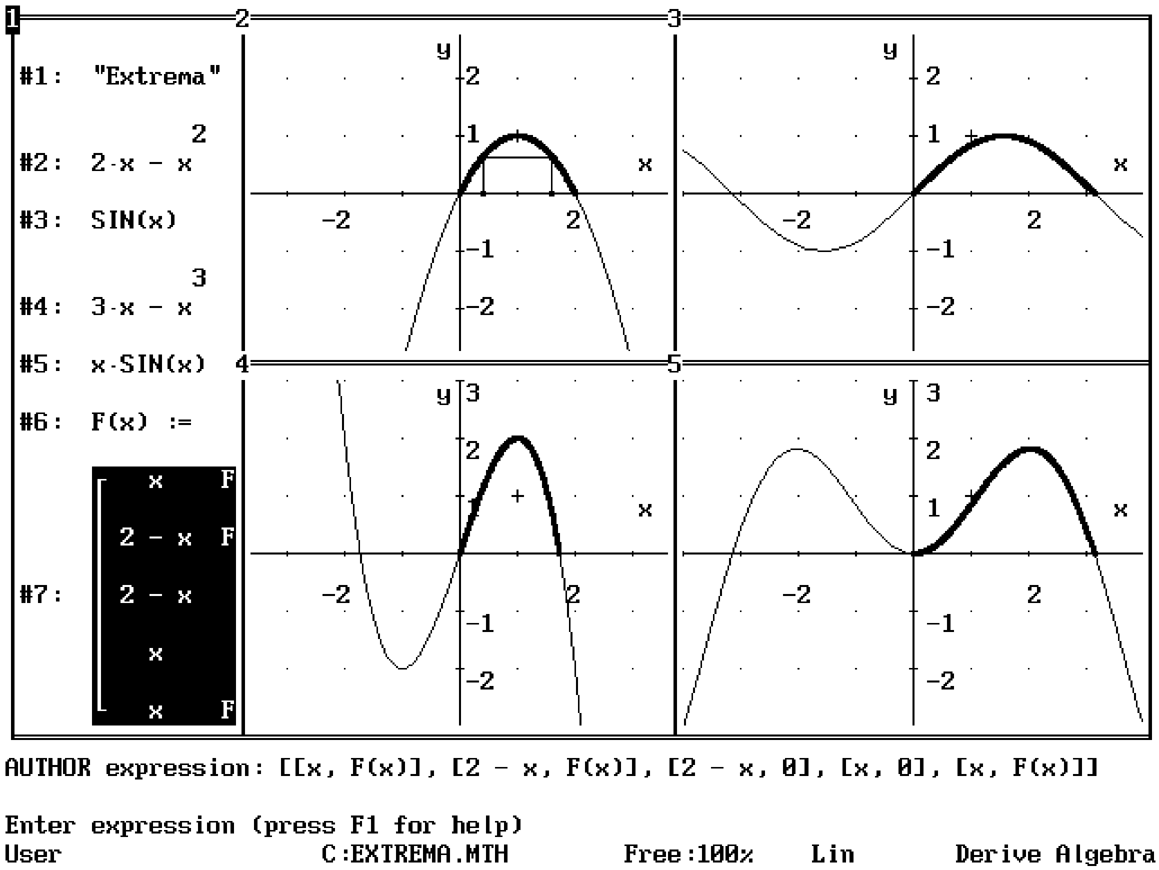
(One) Defining the *target function* including the *possible domain*;
(Two) Tracing an *approximate solution*;
(Three) Calculating a *refined solution*.

Here we shall encounter Extrema problems usually not found in a traditional Calculus curriculum, because at least one of the three above steps cannot be performed in the traditional way. Therefore we must use the *soLve* command (in *Derive*), either in *Exact* mode or in *approXimate* mode, in step (a); We perform step (b) by using tables or graphs, and we construct a new MAX-SELECT algorithm to calculate a refined solution, or apply Newton's method to solve the equation *target'(x)=0*.

In professional development courses we present to the teachers the following family of maximum problems:

What is the maximum area of a rectangle inscribed
between the x-axis and the graph of a function f(x)?

Contents



We start from the standard problem where $F(x) := 2 \cdot x - x^2$: In step (a) we use the symmetry of F (x) at x=1 to conclude that if the first coordinate of the left side of the rectangle is x, then the first coordinate of the right side of the rectangle is 2 - x. Thus the *target function* is $A(x) := IF(0 < x < 1, (2 - 2 \cdot x) \cdot F(x))$. In step (b) we can see *by tracing the graph* of A (x) that the maximum area occurs at $x \approx 0.42$; in step (c) we *soLve* in *Exact* mode the equation A' (x)=0 (don't forget the possible domain) to get a *refined solution*, $x = 1 - \sqrt{3} / 3 \approx 0.422649$ (PrecisionDigits: =6).

Next, we let F (x): =SIN (x) and deal with a similar problem that requires, for step (a) some knowledge of circular functions - the graph of sin (x) is symmetric at $x = \pi / 2$, thus the *target function* is $A(x) := IF(0 < x < \pi / 2, (\pi - 2 \cdot x) \cdot F(x))$. In step (b) by *tracing the graph* of A (x) we find that the maximum area occurs at $x \approx 0.7$. In step (c) we get into trouble, since we cannot *soLve* the equation A' (x)=0 in *Exact* mode. By going over to *approXimate* mode and remembering the domain, we get a refined solution: $x \approx 0.710462$.

We face another difficulty with $F(x):=3\cdot x-x^3$ in step (a). The symmetry in the first two problems is gone! In order to find the first coordinate of the right side of the rectangle, given that the first coordinate of the left side of the rectangle is x, we *soLve***,** in *Exact* mode, the equation F(x) = F(t):

$$\mathrm{SOLVE}(F(x)-F(t),t)=\left[t=x,t=-\frac{\sqrt{3}\cdot\sqrt{(4-x^2)}+x}{2},t=\frac{\sqrt{3}\cdot\sqrt{(4-x^2)}-x}{2}\right].$$

Only the third solution applies, therefore:

$Z(x):=\dfrac{\sqrt{3}\cdot\sqrt{(4-x^2)}-x}{2}$ and the *target function* is:

$A(x):=\mathrm{IF}(0<x<1,F(x)\cdot(Z(x)-x))$.

In step (b) by *tracing the graph* of A(x) we find that the maximum area occurs at $x\approx 0.47$.

In step (c) we get into trouble, since we cannot *soLve* the equation A'(x)=0 in *Exact* mode. By changing to *approXimate* mode and remembering the domain, we obtain a *refined solution*, $x\approx 0.474791$.

And indeed, while in *Exact* mode:

$$\mathrm{SOLVE}\left[\frac{d}{dx}(F(x)\cdot(Z(x)-x)),x\right]=[]$$

When we change to *approXimate,* we get a *refined solution*:

$$\mathrm{SOLVE}\left[\frac{d}{dx}(F(x)\cdot(Z(x)-x)),x,0,1\right]=[x=0.474791]$$

Last but not least is the case of $F(x):=x\cdot\mathrm{SIN}(x)$. Here we have serious difficulties from the beginning. Step #2 in the last problem does not apply here (no solution for the equation $x\cdot\mathrm{SIN}(x)=t\cdot\mathrm{SIN}(t)$ in *Exact* mode). Therefore, after locating p, where F(x) gets its maximum, we define Z(x) as the numerical solution of F(x)=F(t) to the right of p (done in *approXimate* mode):

#1 $F(x):=x\cdot\mathrm{SIN}(x)$

#2 $p:=\mathrm{RHS}\left[\left[\mathrm{SOLVE}\left[\frac{d}{dx}F(x),x,1,3\right]\right]_1\right]$

#3 $Z(x):=\mathrm{RHS}((\mathrm{SOLVE}(F(x)=F(t),t,p,\pi))_1)$

#4 $A(x):=\mathrm{IF}(0<x<p,F(x)\cdot(Z(x)-x))$

Step (b) poses a new problem - *DERIVE* will not plot A(x). A simple solution for this is to compute a few values of A(x) from which we determine that the *approximate solution* is $x \approx 1.2$.

Now comes step (c) and the software is unable to calculate the derivative of A(x). This calls for some ingenuity. Here are three different ways, not applicable without CAS: The primitive, tedious and time-consuming way is to create a sequence of tables, each of them a refinement of the previous one.

#5 $V(u,d) := \text{VECTOR}\left[[x, A(x)], x, u - \frac{6\cdot d}{10}, u + \frac{6\cdot d}{10}, \frac{d}{10}\right]$

#6 PrecisionDigits:=16

#7 $V(1.24,10^{-2}) = \begin{vmatrix} * \\ 1.237 \quad 1.701207318498363 \\ 1.238 \quad 1.701208676001700 \\ 1.239 \quad 1.701204252297197 \\ * \end{vmatrix}$

#8 $V(1.238,10^{-3}) = \begin{vmatrix} * \\ 1.2376 \quad 1.701208826540270 \\ 1.2377 \quad 1.701208875610957 \\ 1.2378 \quad 1.701208866880534 \\ * \end{vmatrix}$ etc.

The biggest problem with this method is that the user, himself, has to find the value of x for which A(x) is maximum. A *recursive formula* would be helpful.

#1 $F(x) := x \cdot \text{SIN}(x)$

#2 $p := \text{RHS}\left[\left[\text{SOLVE}\left[\frac{d}{dx} F(x), x, 1, 3\right]\right]_1\right]$

#3 Precision := Approximate

#4 PrecisionDigits := 24

#5 p := 2.02875783811043422357697 (In order to save time in future calculations).

#6 $Z(x) := \text{RHS}((\text{SOLVE}(F(x) = F(t), t, p, \pi))_1)$

#7 $A(x) := \text{IF}(0 < x < p, F(x) \cdot (Z(x) - x))$

Next comes the algorithm for selecting the value of x for which A(x) is maximum. The following formula assumes that this value is between u and u+d:

#8 $G(u,d) := \text{MAX}\left[\text{SELECT}\left[A(x - 0.000001) \le A(x + 0.000001), x, u - \frac{d}{10}, u + d, \frac{d}{10}\right]\right]$

The next step will be to create a recursive formula

#9 $H(u,n) := \text{IF}(n = 1, G(u, 10^{-1}), G(H(u, n-1), 10^{-n}))$

#10 H(1.2,1)=1.23 ....Compute time 14.6 seconds

#11 H(1.2,2)=1.237 ....Compute time 28.9 seconds

#12 H(1.2,5)=1.237734 ....Compute time 72.3 seconds

#13 H(1.2,8)=1.237734896 ....Compute time 115.5 seconds

A third approach is to use a variation of *Newton's method for solving equations,* which is again a recursive formula. As we know by now we cannot use the built-in algorithms of Calculus. We can overcome this obstacle by using a *Numeric Derivative.*

#1 $F(x) := x \cdot \text{SIN}(x)$

#2 $p := \text{RHS}\left[\left[\text{SOLVE}\left[\frac{d}{dx} F(x), x, 1, 3\right]\right]_1\right]$

#3 Precision := Approximate

#4 PrecisionDigits := 14

#5 p := 2.0287578381104 (In order to save time in future calculations).

#6 $Z(x) := \text{RHS}((\text{SOLVE}(F(x) = F(t), t, p, \pi))_1)$

#7 $A(x) := \text{IF}(0 < x < p, F(x) \cdot (Z(x) - x))$

Now we define B(x) to be a Numeric Derivative of A(x):

#8 $B(x) := (A(x + 0.0001) - A(x - 0.0001)) \cdot 5000$

Now we define C(x) to be a Numeric Derivative of B(x):

#9 $C(x) := (B(x + 0.0001) - B(x - 0.0001)) \cdot 5000$

D (n) is the result of n iterations according to Newton's method starting with x=1.2.

#10 $D(n) := \text{ITERATE}\left[x - \frac{B(x)}{C(x)}, x, 1.2, n\right]$

#11 D(0)=1.2

#12 D(1)=1.2382722850875

#13 D(2)=1.2377349866846

#14 D(3)=1.2377348953646

#15 D(4)=1.2377348953646

No more then 3 iterations were needed for calculating a *refined solution!*

In conclusion, we see how the mathematical problem and the limitations of the software motivated us to invent new methods of dealing with Extrema problems that could not be solved by students in the era before CAS.

Summary

Now we will clarify how the scope of Extrema problems can be widened. One obvious way is that we can solve problems that could not be solved a few years ago because of various computational difficulties. In this sense the repertoire of Extrema problems has been increased. A more important result is that new algorithms have been added for solving Extrema problems that are based on ideas that could not be applied before the age of CAS. We can now solve (non-simple) Extrema problems without calculus. This new didactic possibility has several implications: Extrema problems may be taught before the students have learned calculus. However, there is a price to pay: if the teacher wants to introduce more than the 'primitive' method of using tables, the students need to have good knowledge about Programming and Algebra (e.g., recursive formulae). The possibility that students will be able to integrate Algebra, Calculus and Programming is an intriguing pedagogical challenge. This necessitates a lot of flexibility in rethinking the pedagogical and curricular aspects of mathematics education.

In our teacher courses we present the first three problems, described above, as part of the chapter 'Calculus Concepts'. The fourth problem , which deals with $F(x) := x \cdot SIN(x)$ is introduced in the chapter 'Programming in Derive'. The teachers therefore learn the relevance of programming in solving mathematical problems. We encourage the teachers to widen the scope of other topics in the mathematics curriculum in the projects they prepare as course requirements. Thus the new technology encourages versatile thinking in mathematics.

The following table summarizes the characteristics of the four cases and the various methods of dealing with the problems using CAS.

F(x)	**Target function A(x):= IF(0<x<p,F(x)(Z(x)-x))**	**Approx. Solution**	**Refined Solution**
$2\cdot x-x^2$	Symmetry at x=1 (p=1) $Z(x):=2-x$ Exact	Plot & Trace 0.42 Exact	soLve $\frac{d}{dx}(F(x)\cdot(Z(x)-x))$ $1-\frac{\sqrt{3}}{3}\approx 0.422649$ Exact
$SIN(x)$	Symmetry at $x=\pi/2$ ($p=\pi/2$) $Z(x):=\pi-x$ Exact	Plot & Trace 0.7 Exact	soLve $\frac{d}{dx}(F(x)\cdot(Z(x)-x))$ 0.710462 approX
$3\cdot x-x^3$	No symmetry! Z(x) is one of 3 solutions of F(x)=F(t) $Z(x):=\frac{\sqrt{3}\cdot\sqrt{(4-x^2)}-x}{2}$ p=1 is the location of the max point of F(x) Exact	Plot & Trace 0.47 Exact	soLve $\frac{d}{dx}(F(x)\cdot(Z(x)-x))$ 0.474791 approX
$x\cdot SIN(x)$	No symmetry and no exact solution. Max point of F(x) is $p\approx 2.08$ $Z(x):=$ $RHS((SOLVE(F(x)=F(t),t,p,\pi))_1)$ approX	No plotting! Computing values of A(x) 1.2 approX	VECTOR([x,A(x)],..) or MAX[SELECT[A(x-0.) or NEWTON with numeric derivatives 1.237735 approX