An Efficient *Maple* Program for Calculating Adomian Polynomials

Mariam Al-Mazmumy

Department of Mathematics and Statistics, College of Science, University of Jeddah, Jeddah, Saudi Arabia
Email: mhalmazmumy@uj.edu.sa

Abstract

The immense quest for proficient numerical schemes for the solution of mathematical models featuring nonlinear differential equations led to the realization of the Adomian decomposition method (ADM) in the 80th. Undoubtedly, the solution of nonlinear differential equations using ADM is presided over by the acquisition of Adomian polynomials, which are not always easy to find. Thus, the present study proposes easy-to-implement *Maple* programs for the computation of Adomian polynomials. In fact, the proposed algorithms performed remarkably on several test functions, consisting of one- and multi-variable nonlinearities. Moreover, the introduced programs are advantageous in terms of simplicity; coupled with the requirement of less computational time in comparison with what is known in the literature.

Keywords

Nonlinear Operators, Adomian Polynomials, Adomian Decomposition Method, Multi-Variable Nonlinearities, *Maple* Package

1. Introduction

The Adomian decomposition method (ADM) [1] [2] [3] [4] is one of the prominent methods for the treatment of both the nonlinear or linear functional equations; including mainly the integral equations, partial differential equations (PDEs), ordinary differential equations (ODEs), integro-differential equations, and the blazing fractional differential equations (FDEs) among others. Certainly, ADM has been significantly utilized to solve a variety of emerging real-life problems in contemporary engineering and applied sciences. Amazingly, ADM competently solves assorted initial-value problems (IVPs) and boundary-value problems (BVPs) [2] [5]-[11] straightaway without the need for linearization, discretization, perturbation, or even needless preventive postulations. In addition, the ap-

proach divulges exact analytical solutions whenever reachable or closed-form series solutions whenever exact solutions are unreachable. Undeniably, the solution of nonlinear functional equations using ADM is presided over by the acquisition of Adomian polynomials, which are not always easy to find for all forms of nonlinearities.

In this regard, Adomian and his team members have successfully made use of the ADM to tackle various forms of nonlinear models endowed with varying nonlinearities. In particular, one may find the application of ADM in solving ODEs prescribed with product nonlinearity in [12], exponential nonlinearity in [13], composite nonlinearity in [14], polynomial nonlinearity in [15], trigonometric nonlinearity in [16], negative-power nonlinearity in [17], radical nonlinearity in [18], hyperbolic nonlinearity in [19], and lastly, decimal-power nonlinearity in [20] just to mention but a few. Besides, ADM rapidly solves nonlinear equations featuring any form of analytic nonlinearity; thereby revealing exact or optimal approximate solutions via convergent series.

On the other hand, several researchers have theoretically examined the convergence of ADM [21] [22] [23]. Cherruault [21] sufficiently addressed the convergence issue of ADM via the application of fixed point theorems; in fact, Abbaoui and Cherruault [22] gave a yet different version of Equation (5) for the computation of Adomian polynomials; while [23] gave a simple but elegant convergence proof of the ADM. More so, various other researchers have proposed several modifications for ADM by suitably proposing various ways for easy acquisition of Adomian polynomials, including for instance, Adomian and Rach [24], Wazwaz [25], Abdelwahid [26], and Rach [27] to mention a few; interested reader(s) can equally read [28] [29] [30] [31] [32] and the references therewith. In the same vein, the recent work by Duan [33] [34] [35] on the modification of ADM stands out; indeed, yet reliable algorithms were proposed for ADM through the modification of the Adomian polynomials for both the one- and multi-variable nonlinearities.

However, as symbolic computation using mathematical software for the computation of Adomian polynomials has taken center stage in this arena—solely for the purpose of pass-tracking convergence—various researchers have greatly utilized these softwares to devise fast converging schemes. For instance, see Choi and Shin [36] used *MATHEMATICA* to implement the algorithm introduced in [25]; Chen and Lu [37] made use of *Maple* to compute A_n so easily. Pourdarvish [38] utilized *MATHEMATICA* software for the coding of modification reported in [28]; see also Azreg-Aïnou [32], Duan [39] [40], Hendi *et al.* [41] and Bairagi [42] among others utilized *the MATHEMATICA* for the computation of Adomian polynomials; while Fatoorehchi and Abolghasemi [43] made use of the *MATLAB* program for their algorithm.

Nevertheless, based on the aforementioned relevance of symbolic computation in the acquisition of Adomian polynomials, and the quest for a fast and reliable algorithm for this computational task, the current study proposes easy-to-implement algorithms via the application of *Maple* software, for the construc-

tion of Adomian polynomials in one- and multi-variable nonlinearity settings. Moreover, it is our aim to propose improved and equally simple *Maple* algorithms that will swiftly compute the Adomian polynomials for all forms of nonlinearities. Moreover, we arrange the paper as follows: In the second section, we present the standard Adomian decomposition method. In Sections 3 and 4, we will *Maple* programs for the one- and multi-variable Adomian polynomials, respectively. Section 5 gives the application of the algorithms given in Sections 3 and 4; while Section 6 gives certain concluding points.

2. Adomian Decomposition Method

We make consideration to the following nonhomogeneous nonlinear ODE to highlight the ADM procedure:

$$Lu + Ru + Nu = g , \tag{1}$$

with L and R, $(L > R)$ representing the linear order derivatives, Nu denotes the nonlinear term present, g is the forcing term, while $u = u(x)$ is the solution function.

Next, upon deploying the inverse operator L^{-1} to both sides of the above equation, one obtains:

$$u = \Phi(x) + L^{-1}g - L^{-1}[Ru] - L^{-1}[Nu], \tag{2}$$

where $\Phi(x)$ is a function emanating from the prescribed initial data. Additionally, ADM decomposes the solution $u(x)$ in form of a series as follows:

$$u(x) = \sum_{n=0}^{\infty} u_n(x), \tag{3}$$

and further decomposes the nonlinear term Nu in form of a polynomial series as follows:

$$Nu = \sum_{n=0}^{\infty} A_n , \tag{4}$$

where A_n, which depends on $u_0, u_1, \cdots, u_n$, are referred to as Adomian polynomials. In fact, given any form of nonlinearity $Nu = f(u)$, Adomian polynomials are sequentially determined using the following précised relation [1] [2] [3]:

$$A_n = \frac{1}{n!} \frac{d^n}{d\lambda^n} \left[F\left(\sum_{i=0}^{n} \lambda^i u_i \right) \right]_{\lambda=0} , \quad n = 0, 1, 2, \cdots \tag{5}$$

Furthermore, one may make use of the above relation to explicitly express some of the Adomian polynomials corresponding to one-variable generalized analytic nonlinearity $Nu = f(u(x))$, as follows:

$$A_0 = F(u_0),$$
$$A_1 = u_1 F'(u_0),$$
$$A_2 = u_2 F'(u_0) + \frac{1}{2!} u_1^2 F''(u_0),$$
$$A_3 = u_3 F'(u_0) + u_1 u_2 F''(u_0) + \frac{1}{3!} u_1^3 F'''(u_0),$$
$$A_4 = u_4 F'(u_0) + \left(\frac{1}{2!} u_2^2 + u_1 u_3 \right) F''(u_0) + \frac{1}{2!} u_1^2 u_2 F'''(u_0) + \frac{1}{4!} u_1^4 F^{(4)}(u_0),$$
$$\vdots \tag{6}$$

Thus, upon substituting Eqs. (3) and (4) into Equation (2), one acquires:

$$\sum_{n=0}^{\infty} u_n = \Phi(x) - L^{-1} g(x) - L^{-1} R\left(\sum_{n=0}^{\infty} u_n\right) - L^{-1}\left(\sum_{n=0}^{\infty} A_n\right),$$

which subsequently gives the expected solution $u(x)$ recursively as follows:

$$\begin{aligned}
u_0(x) &= \Phi(x) - L^{-1} g(x), \\
u_{n+1}(x) &= -L^{-1}(Ru_n) - L^{-1}(A_n), \ \ n \geq 0.
\end{aligned} \tag{7}$$

Lastly, having found the solution components $u_n(x)$, for $n \geq 0$ above, the expected closed-form series solution of the governing IVP then takes the following form:

$$\psi_n = \sum_{k=0}^{n-1} u_n, \ \ \text{where } u = \lim_{n\to\infty} \psi_n,$$

that is, by taking the net sum of n-term partial sum; this is indeed the approximate solution.

3. *Maple* Program for Computing One-Variable Adomian Polynomials

This section presents an easy-to-implement *Maple* program for the computation of one-variable Adomian polynomials through the application of the Adomian relation earlier given in Equation (5). Moreover, the program inputs are the nonlinear term $N(u)$ and the number of the expected Adomian polynomials K. In essence, the program is written in simple and plain *Maple* language as follows:

PROGRAM FOR ONE VARIABLE

INPUT

> restart

> $k := K$

> $f := u \to N(u)$

> for j from 0 to k do $A[j] := subs\left(\lambda = 0, \dfrac{1}{j!}\dfrac{\mathrm{d}^j}{\mathrm{d}\lambda^j} f\left(\sum_{i=0}^{20} \lambda^i u[i](x)\right)\right)$ od

4. *Maple* Program for Computing Multi-Variable Adomian Polynomials

In the same manner, the current section presents an easy-to-implement *Maple* program for the computation of multi-variable Adomian polynomials through the application of the generalized version of the Adomian relation earlier expressed in Equation (5). Furthermore, the program inputs are the nonlinear term $N(u)$, the number of the unknown variables N, and the number of the resulting Adomian polynomials K. Ultimately, we write the program in simple and clear *Maple* language as follows:

PROGRAM FOR n-VARIABLE

INPUT

> restart

> $n := N$

```
>   k := K
>   f := (u_1, u_2, ···, u_n) → N(u_1, u_2, ···, u_n)
>   for j from 0 to k do
```

$$A[j] := subs\left(\lambda = 0, \frac{1}{j!}\frac{\mathrm{d}^j}{\mathrm{d}\lambda^j} f\left(seq\left(\sum_{i=0}^{20} \lambda^i u[m,i](x), m = 1, ···, n\right)\right)\right) \quad \text{od}$$

5. Applications

The present section portrays the application of the introduced one- and multi-variable *Maple* programs for the acquisition of Adomian polynomials. In fact, various forms of nonlinearities will be scrutinized to exhibit the competency of the devised associated algorithms.

Example 1. (Polynomial nonlinearity): Consider the polynomial nonlinearity in one-variable of the form $Nu = u^3$.

Applying the *Maple* program for the computation of one-variable Adomian polynomials on the given polynomial nonlinearity, we set $K = 3$ and then obtain the associated Adomian polynomials as follows:

$$A_0 := u_0(x)^3$$

$$A_1 := 3u_0(x)^2 u_1(x)$$

$$A_2 := 3u_0(x)u_1(x)^2 + 3u_0(x)^2 u_2(x)$$

$$A_3 := u_1(x)^3 + 6u_0(x)u_1(x)u_2(x) + 3u_0(x)^2 u_3(x)$$

Example 2. (Product nonlinearity): Consider the derivative nonlinearity in one-variable of the form $Nu = u\dfrac{\mathrm{d}u}{\mathrm{d}x}$.

Applying the *Maple* program for computing one-variable Adomian polynomials on the mentioned product nonlinearity, we set $K = 3$ and obtain the associated Adomian polynomials as follows:

$$A_0 := u_0(x)\left(\frac{\mathrm{d}}{\mathrm{d}x}u_0(x)\right)$$

$$A_1 := u_1(x)\left(\frac{\mathrm{d}}{\mathrm{d}x}u_0(x)\right) + u_0(x)\left(\frac{\mathrm{d}}{\mathrm{d}x}u_1(x)\right)$$

$$A_2 := u_2(x)\left(\frac{\mathrm{d}}{\mathrm{d}x}u_0(x)\right) + u_1(x)\left(\frac{\mathrm{d}}{\mathrm{d}x}u_1(x)\right) + u_0(x)\left(\frac{\mathrm{d}}{\mathrm{d}x}u_2(x)\right)$$

$$A_3 := u_3(x)\left(\frac{\mathrm{d}}{\mathrm{d}x}u_0(x)\right) + u_2(x)\left(\frac{\mathrm{d}}{\mathrm{d}x}u_1(x)\right) + u_1(x)\left(\frac{\mathrm{d}}{\mathrm{d}x}u_2(x)\right)$$
$$+ u_0(x)\left(\frac{\mathrm{d}}{\mathrm{d}x}u_3(x)\right)$$

Example 3. (Hyperbolic nonlinearity): Consider the hyperbolic nonlinearity in one-variable of the form $Nu = \sinh(u)$.

Equally, we apply the *Maple* program for the computation of one-variable Adomian polynomials on the prescribed nonlinearity. Indeed, we $K = 3$ and

further obtain the associated Adomian polynomials as follows:

$$A_0 := \sinh\big(u_0(x)\big)$$

$$A_1 := \cosh\big(u_0(x)\big)u_1(x)$$

$$A_2 := \frac{1}{2}\sinh\big(u_0(x)\big)u_1(x)^2 + \cosh\big(u_0(x)\big)u_2(x)$$

$$A_3 := \frac{1}{6}\cosh\big(u_0(x)\big)u_1(x)^3 + \sinh\big(u_0(x)\big)u_1(x)u_2(x) + \cosh\big(u_0(x)\big)u_3(x)$$

Example 4. (Inverse trigonometric nonlinearity): Consider the inverse trigonometric nonlinearity in one-variable of the form $Nu = \tan^{-1}(u)$.

Accordingly, we implement the *Maple* program for the computation of one-variable Adomian polynomials on the present nonlinearity with $K = 3$ to obtain the resulting polynomials as follows:

$$A_0 := \arctan\big(u_0(x)\big)$$

$$A_1 := \frac{u_1(x)}{1 + u_0(x)^2}$$

$$A_2 := \frac{u_2(x)}{1 + u_0(x)^2} - \frac{u_1(x)^2 u_0(x)}{\big(1 + u_0(x)^2\big)^2}$$

$$A_3 := \frac{u_3(x)}{1 + u_0(x)^2} - \frac{2u_0(x)u_1(x)u_2(x)}{\big(1 + u_0(x)^2\big)^2} + \frac{4}{3}\frac{u_1(x)^3 u_0(x)^2}{\big(1 + u_0(x)^2\big)^3} - \frac{1}{3}\frac{u_1(x)^3}{\big(1 + u_0(x)^2\big)^2}$$

Example 5. (Logarithmic nonlinearity): Consider the logarithmic nonlinearity in one-variable of the form $Nu = \ln(1 + u)$.

In the same manner, the application of the devised *Maple* program for the computation of one-variable Adomian polynomials on the current logarithmic nonlinearity when setting $K = 3$ yields the following polynomials:

$$A_0 := \ln\big(1 + u_0(x)\big)$$

$$A_1 := \frac{u_1(x)}{1 + u_0(x)}$$

$$A_2 := \frac{u_2(x)}{1 + u_0(x)} - \frac{1}{2}\frac{u_1(x)^2}{\big(1 + u_0(x)\big)^2}$$

$$A_3 := \frac{u_3(x)}{1 + u_0(x)} - \frac{u_2(x)u_1(x)}{\big(1 + u_0(x)\big)^2} + \frac{1}{3}\frac{u_1(x)^3}{\big(1 + u_0(x)\big)^3}$$

Example 6. (Negative-power nonlinearity): Consider the negative-power nonlinearity in one-variable of the form $Nu = u^{-m}$, where $m \in \mathbb{R}$.

Here, we evenly implement the *Maple* program for computing one-variable Adomian polynomials on the negative-power nonlinearity by setting $K = 3$ and subsequently acquire the following Adomian polynomials:

$$A_0 := u_0(x)^{-m}$$

$$A_1 := -u_0(x)^{-m-1} m u_1(x)$$

$$A_2 := \frac{1}{2} u_0(x)^{-m-2} m^2 u_1(x)^2 - u_0(x)^{-m-1} m u_2(x) + \frac{1}{2} u_0(x)^{-m-2} m u_1(x)^2$$

$$A_3 := -\frac{1}{6} m \Big(u_0(x)^{-m-3} m^2 u_1(x)^3 + 3 u_0(x)^{-m-3} m u_1(x)^3 + 2 u_0(x)^{-m-3} u_1(x)^3$$
$$- 6 u_0(x)^{-m-2} m u_1(x) u_2(x) - 6 u_0(x)^{-m-2} u_1(x) u_2(x) + 6 u_0(x)^{-m-1} u_3(x) \Big)$$

Example 7. (Composite Nonlinearity): Consider the composite nonlinearity in one-variable of the form $Nu = e^{-\sin\left(\frac{1}{2}u\right)}$.

Accordingly, the application of the *Maple* program for one-variable nonlinearity on the present nonlinearity when $K = 3$ yields the following Adomian polynomials:

$$A_0 := e^{-\sin\left(\frac{1}{2}u_0(x)\right)}$$

$$A_1 := -\frac{1}{2} \cos\left(\frac{1}{2}u_0(x)\right) u_1(x) e^{-\sin\left(\frac{1}{2}u_0(x)\right)}$$

$$A_2 := \frac{1}{8} u_1(x)^2 \sin\left(\frac{1}{2}u_0(x)\right) e^{-\sin\left(\frac{1}{2}u_0(x)\right)} - \frac{1}{2} \cos\left(\frac{1}{2}u_0(x)\right) u_2(x) e^{-\sin\left(\frac{1}{2}u_0(x)\right)}$$
$$+ \frac{1}{8} \cos\left(\frac{1}{2}u_0(x)\right)^2 u_1(x)^2 e^{-\sin\left(\frac{1}{2}u_0(x)\right)}$$

$$A_3 := \frac{1}{48} \cos\left(\frac{1}{2}u_0(x)\right) u_1(x)^3 e^{-\sin\left(\frac{1}{2}u_0(x)\right)} + \frac{1}{4} \sin\left(\frac{1}{2}u_0(x)\right) u_1(x) u_2(x) e^{-\sin\left(\frac{1}{2}u_0(x)\right)}$$
$$- \frac{1}{16} \sin\left(\frac{1}{2}u_0(x)\right) \cos\left(\frac{1}{2}u_0(x)\right) u_1(x)^3 e^{-\sin\left(\frac{1}{2}u_0(x)\right)}$$
$$- \frac{1}{2} \cos\left(\frac{1}{2}u_0(x)\right) u_3(x) e^{-\sin\left(\frac{1}{2}u_0(x)\right)} + \frac{1}{4} \cos\left(\frac{1}{2}u_0(x)\right)^2 u_1(x) u_2(x) e^{-\sin\left(\frac{1}{2}u_0(x)\right)}$$
$$- \frac{1}{48} \cos\left(\frac{1}{2}u_0(x)\right)^3 u_1(x)^3 e^{-\sin\left(\frac{1}{2}u_0(x)\right)}$$

Example 8. (Radical nonlinearity): Consider the radical function nonlinearity in one-variable of the form $Nu = \sqrt{\alpha u^2 + \beta}$, where $\alpha, \beta \in \mathbb{R}$.

As proceed, the application of the one-variable *Maple* program with $K = 3$ reveals the following resulting Adomian polynomials as follows:

$$A_0 := \sqrt{\alpha u_0(x)^2 + \beta}$$

$$A_1 := \frac{\alpha u_0(x) u_1(x)}{\sqrt{\alpha u_0(x)^2 + \beta}}$$

$$A_2 := -\frac{1}{2} \frac{\alpha^2 u_0(x)^2 u_1(x)^2}{\left(\alpha u_0(x)^2 + \beta\right)^{\frac{3}{2}}} + \frac{1}{2} \frac{\alpha u_1(x)^2}{\sqrt{\alpha u_0(x)^2 + \beta}} + \frac{\alpha u_0(x) u_2(x)}{\sqrt{\alpha u_0(x)^2 + \beta}}$$

$$A_3 := \frac{1}{2}\frac{\alpha^3 u_0(x)^3 u_1(x)^3}{\left(\alpha u_0(x)^2 + \beta\right)^{\frac{5}{2}}} - \frac{1}{2}\frac{\alpha^2 u_0(x) u_1(x)^3}{\left(\alpha u_0(x)^2 + \beta\right)^{\frac{3}{2}}} - \frac{\alpha^2 u_0(x)^2 u_1(x) u_2(x)}{\left(\alpha u_0(x)^2 + \beta\right)^{\frac{3}{2}}}$$

$$+ \frac{\alpha u_1(x) u_2(x)}{\sqrt{\alpha u_0(x)^2 + \beta}} + \frac{\alpha u_0(x) u_3(x)}{\sqrt{\alpha u_0(x)^2 + \beta}}$$

Example 9. (Two-variable product nonlinearity): Consider the product nonlinearity in two-variable of the form $Nu = u_1^2 * \dfrac{du_2}{dx}$.

Applying the *Maple* program for computing multi-variable Adomian polynomials on the present two-variable product nonlinearity, we set $N = 2$, and $K = 3$ to obtain the associated Adomian polynomials as follows:

$$A_0 := u_{1,0}(x)^2 \left(\frac{d}{dx} u_{2,0}(x)\right)$$

$$A_1 := 2u_{1,0}(x)\left(\frac{d}{dx} u_{2,0}(x)\right) u_{1,1}(x) + u_{1,0}(x)^2 \left(\frac{d}{dx} u_{2,1}(x)\right)$$

$$A_2 := u_{1,1}(x)^2 \left(\frac{d}{dx} u_{2,0}(x)\right) + 2u_{1,0}(x) u_{1,1}(x)\left(\frac{d}{dx} u_{2,1}(x)\right)$$

$$+ 2u_{1,0}(x)\left(\frac{d}{dx} u_{2,0}(x)\right) u_{1,2}(x) + u_{1,0}(x)^2 \left(\frac{d}{dx} u_{2,2}(x)\right)$$

$$A_3 := 2u_{1,1}(x)\left(\frac{d}{dx} u_{2,0}(x)\right) u_{1,2}(x) + u_{1,1}(x)^2 \left(\frac{d}{dx} u_{2,1}(x)\right)$$

$$+ 2u_{1,0}(x)\left(\frac{d}{dx} u_{2,2}(x)\right) u_{1,1}(x) + 2u_{1,0}(x)\left(\frac{d}{dx} u_{2,1}(x)\right) u_{1,2}(x)$$

$$+ 2u_{1,0}(x)\left(\frac{d}{dx} u_{2,0}(x)\right) u_{1,3}(x) + u_{1,0}(x)^2 \left(\frac{d}{dx} u_{2,3}(x)\right)$$

Example 10. (Two-variable nonlinearity): Consider the nonlinearity in two-variable of the form $Nu = e^{u_1} + \cos(u_2)$.

In the same way, we applying the *Maple* program for the computation of multi-variable Adomian polynomials on the governing two-variable nonlinearity by setting $N = 2$, and $K = 3$ to obtain the following Adomian polynomials:

$$A_0 := e^{u_{1,0}(x)} + \cos(u_{2,0}(x))$$

$$A_1 := u_{1,1}(x) e^{u_{1,0}(x)} - \sin(u_{2,0}(x)) u_{2,1}(x)$$

$$A_2 := u_{1,2}(x) e^{u_{1,0}(x)} + \frac{1}{2} u_{1,1}(x)^2 e^{u_{1,0}(x)} - \frac{1}{2}\cos(u_{2,0}(x)) u_{2,1}(x)^2$$

$$- \sin(u_{2,0}(x)) u_{2,2}(x)$$

$$A_3 := u_{1,3}(x) e^{u_{1,0}(x)} + u_{1,2}(x) u_{1,1}(x) e^{u_{1,0}(x)} + \frac{1}{6} u_{1,1}(x)^3 e^{u_{1,0}(x)}$$

$$+ \frac{1}{6}\sin(u_{2,0}(x)) u_{2,1}(x)^3 - \cos(u_{2,0}(x)) u_{2,1}(x) u_{2,2}(x)$$

$$- \sin(u_{2,0}(x)) u_{2,3}(x)$$

Example 11. (Four-variable nonlinearity): Consider the nonlinearity in four-variable of the following form $Nu = u_1 * u_2 * \dfrac{du_3}{dx} + u_4^2$.

Accordingly, we implement the *Maple* program for the computation of multi-variable Adomian polynomials on the current four-variable nonlinearity by fixing $N = 4$, and $K = 3$ to acquire the following resulting Adomian polynomials:

$$A_0 := u_{1,0}(x) u_{2,0}(x) \left(\frac{d}{dx} u_{3,0}(x) \right) + u_{4,0}(x)^2$$

$$A_1 := u_{1,1}(x) u_{2,0}(x) \left(\frac{d}{dx} u_{3,0}(x) \right) + u_{1,0}(x) u_{2,1}(x) \left(\frac{d}{dx} u_{3,0}(x) \right)$$
$$+ u_{1,0}(x) u_{2,0}(x) \left(\frac{d}{dx} u_{3,1}(x) \right) + 2 u_{4,0}(x) u_{4,1}(x)$$

$$A_2 := u_{1,2}(x) u_{2,0}(x) \left(\frac{d}{dx} u_{3,0}(x) \right) + u_{1,1}(x) u_{2,1}(x) \left(\frac{d}{dx} u_{3,0}(x) \right)$$
$$+ u_{1,1}(x) u_{2,0}(x) \left(\frac{d}{dx} u_{3,1}(x) \right) + u_{1,0}(x) u_{2,2}(x) \left(\frac{d}{dx} u_{3,0}(x) \right)$$
$$+ u_{1,0}(x) u_{2,1}(x) \left(\frac{d}{dx} u_{3,1}(x) \right) + u_{1,0}(x) u_{2,0}(x) \left(\frac{d}{dx} u_{3,2}(x) \right)$$
$$+ u_{4,1}(x)^2 + 2 u_{4,0}(x) u_{4,2}(x)$$

$$A_3 := u_{1,3}(x) u_{2,0}(x) \left(\frac{d}{dx} u_{3,0}(x) \right) + u_{1,2}(x) u_{2,1}(x) \left(\frac{d}{dx} u_{3,0}(x) \right)$$
$$+ u_{1,2}(x) u_{2,0}(x) \left(\frac{d}{dx} u_{3,1}(x) \right) + u_{1,1}(x) u_{2,2}(x) \left(\frac{d}{dx} u_{3,0}(x) \right)$$
$$+ u_{1,1}(x) u_{2,1}(x) \left(\frac{d}{dx} u_{3,1}(x) \right) + u_{1,1}(x) u_{2,0}(x) \left(\frac{d}{dx} u_{3,2}(x) \right)$$
$$+ u_{1,0}(x) u_{2,3}(x) \left(\frac{d}{dx} u_{3,0}(x) \right) + u_{1,0}(x) u_{2,2}(x) \left(\frac{d}{dx} u_{3,1}(x) \right)$$
$$+ u_{1,0}(x) u_{2,1}(x) \left(\frac{d}{dx} u_{3,2}(x) \right) + u_{1,0}(x) u_{2,0}(x) \left(\frac{d}{dx} u_{3,3}(x) \right)$$
$$+ 2 u_{4,1}(x) u_{4,2}(x) + 2 u_{4,0}(x) u_{4,3}(x)$$

Remarkably, as portrayed in the above test examples for multi-variable nonlinearities—see examples 9, 10 and 11—it is pompous to state here that the devised *Maple* program for the construction of Adomian polynomials for multi-variable nonlinearities can equally be relevant in the computation of Adomian polynomials for *n*-variable nonlinear functional operators.

6. Conclusion

In conclusion, the current study proposed easy-to-implement programs via the application of *Maple* software, for the construction of Adomian polynomials. Precisely, two competent *Maple* programs were introduced to squarely deal with the one- and multi-variable nonlinearities, respectively. Indeed, this study was motivated by the immense quest for proficient numerical schemes for the solu-

tion of nonlinear mathematical models; noteworthy, most real-life scenarios are mathematically mimicked using nonlinear differential equations. Hence, our study improved the computational rapidity of the classical Adomian decomposition method by the proposed algorithms that swiftly compute the resulting Adomian polynomials. More so, the introduced programs are advantageous in terms of simplicity, and of course, in comparison with what is known in the related literature. Moreover, the generalized program for the multi-variable nonlinearities is applicable in the computation of Adomian polynomials for n-variable nonlinear functional operators.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Adomian, G. (1986) Nonlinear Stochastic Operator Equations. In: Adomian, G., Ed., *Nonlinear Stochastic Operator Equations*, Academic Press, Orlando, 88-97. https://doi.org/10.1016/B978-0-12-044375-8.50012-5

[2] Adomian, G. (1994) Solving Frontier Problems of Physics: The Decomposition Method. Kluwer Academic, Dordrecht. https://doi.org/10.1007/978-94-015-8289-6

[3] Wazwaz, A.M. (2009) Partial Differential Equations and Solitary Waves Theory. Higher Education Press, Beijing. https://doi.org/10.1007/978-3-642-00251-9

[4] Wazwaz, A.M. (2011) Linear and Nonlinear Integral Equations: Methods and Applications. Higher Education Press, Beijing.

[5] Wazwaz, A.M. (2000) The Modified Adomian Decomposition Method for Solving Linear and Nonlinear Boundary Value Problems of Tenth-Order and Twelfth-Order. *International Journal of Nonlinear Sciences and Numerical Simulation*, **1**, 17-24. https://doi.org/10.1515/IJNSNS.2000.1.1.17

[6] Ebaid, A. (2011) A New Analytical and Numerical Treatment for Singular Two-Point Boundary Value Problems via the Adomian Decomposition Method. *Journal of Computational and Applied Mathematics*, **235**, 1914-1924. https://doi.org/10.1016/j.cam.2010.09.007

[7] Bakodah, H.O., Al-Zaid, N.A., Al-Mazmumy, M. and Biswas, A. (2017) An Efficient Numerical Algorithm for Solving Dirichlet Initial-Boundary Value Problems. *Mathematics in Engineering, Science and Aerospace*, **8**, 489-510.

[8] Bakodah, H.O., Al-Mazmumy, M. and Almuhalbedi, S.O. (2019) Solving System of Integro Differential Equations Using Discrete Adomian Decomposition Method. *Journal of Taibah University for Science*, **13**, 805-812. https://doi.org/10.1080/16583655.2019.1625189

[9] Alzhrani, M.A., Bakodah, H.O. and Al-Mazmumy, M. (2019) A 3/8 Simpson's Numerical Scheme for the Classes of Volterra Integral Equations of First Kind. *Nonlinear Analysis and Differential Equations*, **7**, 99-113. https://doi.org/10.12988/nade.2019.9812

[10] Al-Mazmumy, M., Alsulami, A., Bakodah, H.O. and Alzaid, N. (2022) Modified Adomian Method for the Generalized Inhomogeneous Lane-Emden-Type Equations. *Nonlinear Analysis and Differential Equations*, **10**, 15-35. https://doi.org/10.12988/nade.2022.91142

[11] AL-Zaid, N., AL-Refaidi, A., Bakodah, H.O. and Al-Mazmumy, M. (2022) Solution of Second- and Higher-Order Nonlinear Two-Point Boundary-Value Problems Using Double Decomposition Method. *Mathematics*, **10**, Article 3519. https://doi.org/10.3390/math10193519

[12] Adomian, G. (1981) On Product Nonlinearities in Stochastic Differential Equations. *Applied Mathematics and Computation*, **8**, 35-49. https://doi.org/10.1016/0096-3003(81)90035-7

[13] Adomian, G. and Rach, R. (1985) Nonlinear Plasma Response. *Journal of Mathematical Analysis and Applications*, **111**, 114-118. https://doi.org/10.1016/0022-247X(85)90204-5

[14] Adomian, G. and Rach, R. (1986) On Composite Nonlinearities and the Decomposition Method. *Journal of Mathematical Analysis and Applications*, **113**, 504-509. https://doi.org/10.1016/0022-247X(86)90321-5

[15] Adomian, G. and Rach, R. (1985) Polynomial Nonlinearities in Differential Equations. *Journal of Mathematical Analysis and Applications*, **109**, 90-95. https://doi.org/10.1016/0022-247X(85)90178-7

[16] Adomian, G. and Rach, R. (1983) Anharmonic Oscillator Systems. *Journal of Mathematical Analysis and Applications*, **91**, 229-236. https://doi.org/10.1016/0022-247X(83)90101-4

[17] Adomian, G. and Rach, R. (1985) Nonlinear Differential Equations with Negative Power Nonlinearities. *Journal of Mathematical Analysis and Applications*, **112**, 497-501. https://doi.org/10.1016/0022-247X(85)90258-6

[18] Adomian, G., Rach, R. and Sarafyan, D. (1985) On the Solution of Equations Containing Radicals by the Decomposition Method. *Journal of Mathematical Analysis and Applications*, **111**, 423-426. https://doi.org/10.1016/0022-247X(85)90226-4

[19] Adomian, G. and Rach, R. (1984) Light Scattering in Crystals. *Journal of Applied Physics*, **56**, 2592-2594. https://doi.org/10.1063/1.334291

[20] Adomian, G. and Rach, R. (1986) Solving Nonlinear Differential Equations with Decimal Power Nonlinearities. *Journal of Mathematical Analysis and Applications*, **114**, 423-425. https://doi.org/10.1016/0022-247X(86)90094-6

[21] Cherruault, Y. (1990) Convergence of Adomian's Method. *Mathematical and Computer Modelling*, **14**, 83-86. https://doi.org/10.1016/0895-7177(90)90152-D

[22] Abbaoui, K. and Cherruault, Y. (1995) New Ideas for Proving Convergence of Decomposition Methods. *Computers & Mathematics with Applications*, **29**, 103-108. https://doi.org/10.1016/0898-1221(95)00022-Q

[23] Hosseini, M.M. and Nasabzadeh, H. (2006) On the Convergence of Adomian Decomposition Method. *Applied Mathematics and Computation*, **182**, 536-543. https://doi.org/10.1016/j.amc.2006.04.015

[24] Adomian, G. and Rach, R. (1992) Generalization of Adomian Polynomials to Functions of Several Variables. *Computers & Mathematics with Applications*, **24**, 11-24. https://doi.org/10.1016/0898-1221(92)90037-I

[25] Wazwaz, A.M. (2000) A New Algorithm for Calculating Adomian Polynomials for Nonlinear Operators. *Applied Mathematics and Computation*, **111**, 33-51. https://doi.org/10.1016/S0096-3003(99)00063-6

[26] Abdelwahid, F. (2003) A Mathematical Model of Adomian Polynomials. *Applied Mathematics and Computation*, **141**, 447-453. https://doi.org/10.1016/S0096-3003(02)00266-7

[27] Rach, R. (2008) A New Definition of the Adomian Polynomials. *Kybernetes*, **37**, 910-955. https://doi.org/10.1108/03684920810884342

[28] Biazar, J., Babolian, E., Kember, G., Nouri, A. and Islam, R. (2003) An Alternate Algorithm for Computing Adomian Polynomials in Special Cases. *Applied Mathematics and Computation*, **138**, 523-529. https://doi.org/10.1016/S0096-3003(02)00174-1

[29] Abbaoui, K., Cherruault, Y. and Seng, V. (1995) Practical Formulae for the Calculus of Multivariable Adomian Polynomials. *Mathematical and Computer Modelling*, **22**, 89-93. https://doi.org/10.1016/0895-7177(95)00103-9

[30] Zhu, Y., Chang, Q. and Wu, S. (2005) A New Algorithm for Calculating Adomian Polynomials. *Applied Mathematics and Computation*, **169**, 402-416. https://doi.org/10.1016/j.amc.2004.09.082

[31] Biazar, J., Ilie, M. and Khoshkenar, A. (2006) An Improvement to an Alternate Algorithm for Computing Adomian Polynomials in Special Cases. *Applied Mathematics and Computation*, **173**, 582-592. https://doi.org/10.1016/j.amc.2005.04.052

[32] Azreg-Aïnou, M. (2009) A Developed New Algorithm for Evaluating Adomian Polynomials. *CMES-Computer Modeling in Engineering & Sciences*, **42**, 1-18.

[33] Duan, J.S. (2010) Recurrence Triangle for Adomian Polynomials. *Applied Mathematics and Computation*, **216**, 1235-1241. https://doi.org/10.1016/j.amc.2010.02.015

[34] Duan, J.S. (2010) An Efficient Algorithm for the Multivariable Adomian Polynomials. *Applied Mathematics and Computation*, **217**, 2456-2467. https://doi.org/10.1016/j.amc.2010.07.046

[35] Duan, J.S. (2011) Convenient Analytic Recurrence Algorithms for the Adomian Polynomials. *Applied Mathematics and Computation*, **217**, 6337-6348. https://doi.org/10.1016/j.amc.2011.01.007

[36] Choi, H.W. and Shin, J.G. (2003) Symbolic Implementation of the Algorithm for Calculating Adomian Polynomials. *Applied Mathematics and Computation*, **146**, 257-271. https://doi.org/10.1016/S0096-3003(02)00541-6

[37] Chen, W. and Lu, Z. (2004) An Algorithm for Adomian Decomposition Method. *Applied Mathematics and Computation*, **159**, 221-235. https://doi.org/10.1016/j.amc.2003.10.037

[38] Pourdarvish, A. (2006) A Reliable Symbolic Implementation of Algorithm for Calculating Adomian Polynomials. *Applied Mathematics and Computation*, **172**, 545-550. https://doi.org/10.1016/j.amc.2005.02.040

[39] Duan, J.S. and Guo, A.P. (2010) Reduced Polynomials and Their Generation in Adomian Decomposition Methods. *CMES-Computer Modeling in Engineering & Sciences*, **60**, 139-150.

[40] Duan, J.S. (2010) An Efficient Algorithm for the Multivariable Adomian Polynomials. *Applied Mathematics and Computation*, **217**, 2456-2467. https://doi.org/10.1016/j.amc.2010.07.046

[41] Hendi, F.A., Bakodah, H.O., Almazmumy, M. and Alzumi, H. (2012) A Simple Program for Solving Nonlinear Initial Value Problem Using Adomian Decomposition Method. *International Journal of Research and Reviews in Applied Sciences*, **12**, 397-406.

[42] Bairagi, M. (2023) A New Algorithm to Determine Adomian Polynomials for Nonlinear Polynomial Functions. *Asian Research Journal of Mathematics*, **19**, 1-12. https://doi.org/10.2139/ssrn.4419957

[43] Fatoorehchi, H. and Abolghasemi, H. (2011) On Calculation of Adomian Polynomials by MATLAB. *Journal of Applied Computer Science & Mathematics*, **5**, 85-88.