ARCHITECTURE CIVIL ENGINEERING ENVIRONMENT
The Silesian University of Technology No. 1/2024

doi: 10.2478/ACEE-2024-0007

SEMI-ANALYTICAL INTEGRATION OF THE ACCELEROGRAM AND EQUATIONS OF MOTION OF A SYSTEM SUBJECTED TO A PARASEISMIC SHOCK

Ryszard WALENTYŃSKI *

* Ass. Prof., DSc PhD Eng.; Silesian University of Technology, Faculty of Civil Engineering, ul. Akademicka 5, 44-100 Gliwice, Poland, e-mail: Ryszard.Walentynski@polsl.pl
ORCID: https://orcid.org/0000-0003-3530-682X
E-mail address: *Ryszard.Walentynski@polsl.pl*

Received: 28.04.2023; Revised: 26.05.2023; Accepted: 05.06.2023

Abstract
The task of analysing shocks in areas of paraseismic mining activities is one of the important engineering tasks. Acceleration measurements should be integrated to receive velocity and displacement functions. Sometimes, numerical algorithms produce wrong results. This paper proposes a semi-analytical approach to this problem. The algorithm has been implemented within the Mathematica system. Moreover, this method has been applied to integration of the equation of motion.

Keywords: **Accelerogram; Equation of motion; Paraseismic shock; Semi-analytical integration; *Mathematica*.**

1. INTRODUCTION

This contribution is an extension of a conference paper [1]. Motivation was an issue that my former student Kinga Zemła in her research of the master thesis. She compared results of velocity diagrams obtained from two numerical algorithms of accelerogram integration and they were different. The problem was related to the baseline correction described in the review paper [2] and cited in its source research papers.

Analysis of paraseismic shocks in mining activities areas is a difficult engineering task, and the reliability of the results is very important.

Usually, numerical integration algorithms are used to evaluate velocity and displacement functions [3, 4].

This paper presents an analytical approach to integration that is possible to apply thanks to the Mathematica system [5]. The procedure obtained is straightforward and surprisingly effective. It is presented as an alternative to numerical methods.

Moreover, the approach may be applied to integration of the equation of motion, and the crucial steps of this algorithm are also presented. The numerical approach to this problem is described in many books, for example [6, 7, 8].

2. SAMPLE ACCELEROGRAM

Figure 1 shows an accelerogram from the research mentioned in the Introduction, as an example of the method used. The entire recorded signal lasted 5 seconds, but the largest acceleration amplitudes were recorded in the subinterval shown on the right side of the figure. The set of recorded data within this time period consists of 327680 elements. Thus, the time distance between them is 0.000152588 s. The analysed signal is highly oscillating, and we cannot say that measurement points are very dense; see Fig. 2.

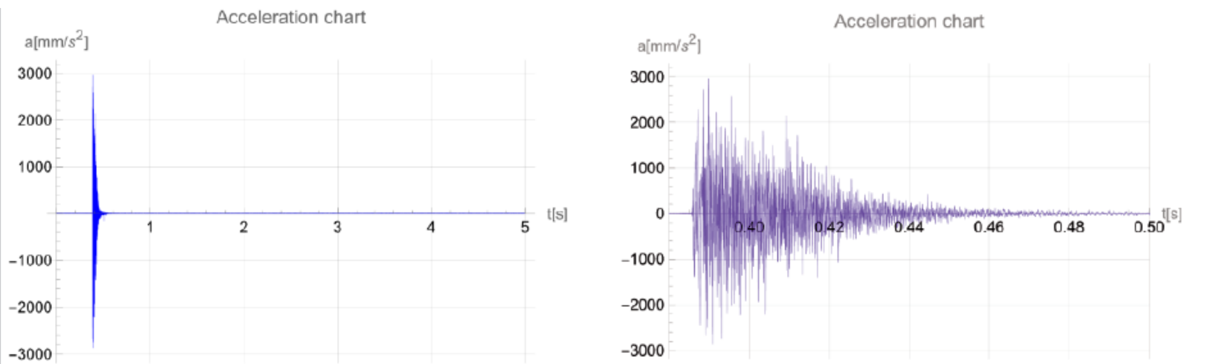

Figure 1.
Accelerogram in the entire measurement range and in the subrange of high accelerations

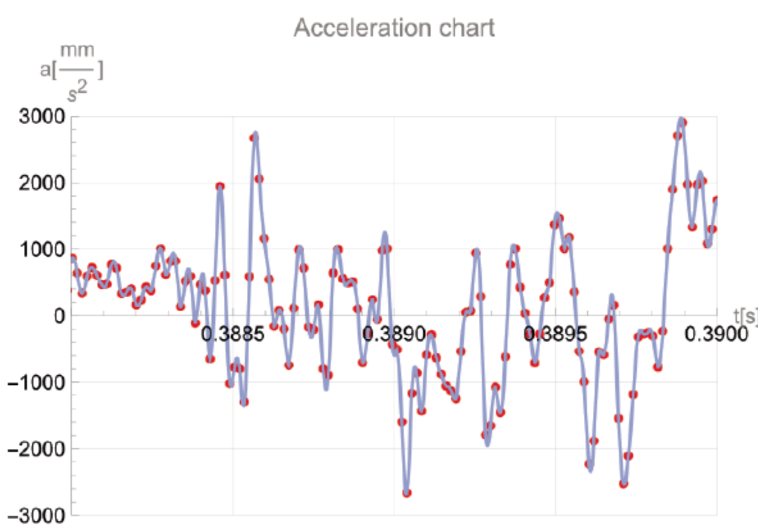

Figure 2.
Interpolation of measurement data in the range of extremal accelerations

Numerical integration of this signal in Mathematica is possible, but the only effective algorithm with the use of a built-in function requires the division of this time interval into 327679 subintervals. Computation took an unreasonably long time.

3. INTERPOLATION AND INTEGRATION

In the Mathematica system, we can interpolate a set of measurement points with a function. The argument of this function is the order of the interpolation function. This corresponds roughly to the degree of polynomial fit between the measurement points of the set ts.

```
a[i_] :=
 a[i] =
  Interpolation[ts, InterpolationOrder → i]
```

The interpolation with interpolation order equal to 6 is presented in Fig. 2. From this moment we can deal with the interpolation function almost like with the analytical one. This “almost” and the fact that it is based on numerical data explains the word “semi-analytical” in this title of the article.

The function of acceleration is highly oscillating. As has already been mentioned, numerical integration fails. It turns out that the analytical integration of this function is very fast.

We compute the velocity (speed) function as an integral:

```
v[i_][t_] := v[i][t] = ∫Evaluate[a[i][t]] ⅆt
```

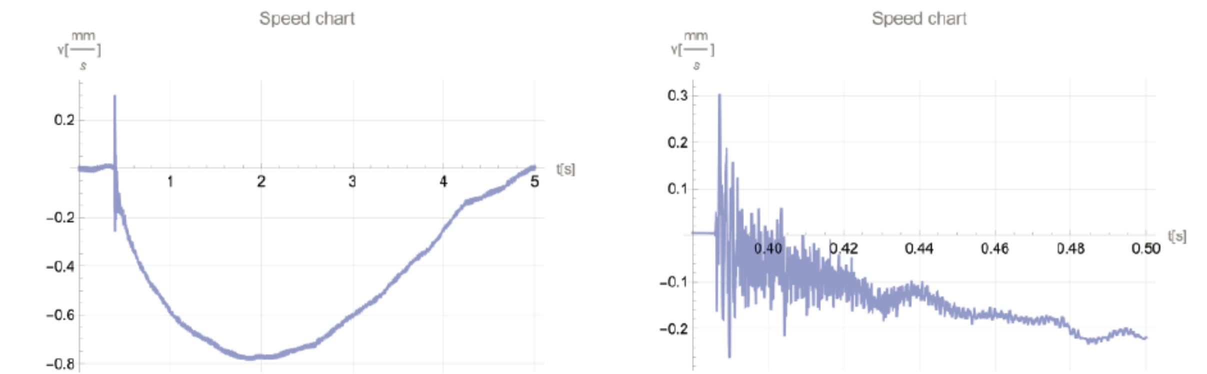

Figure 3.
Speed chart in the entire measurement range and in the subrange of high accelerations

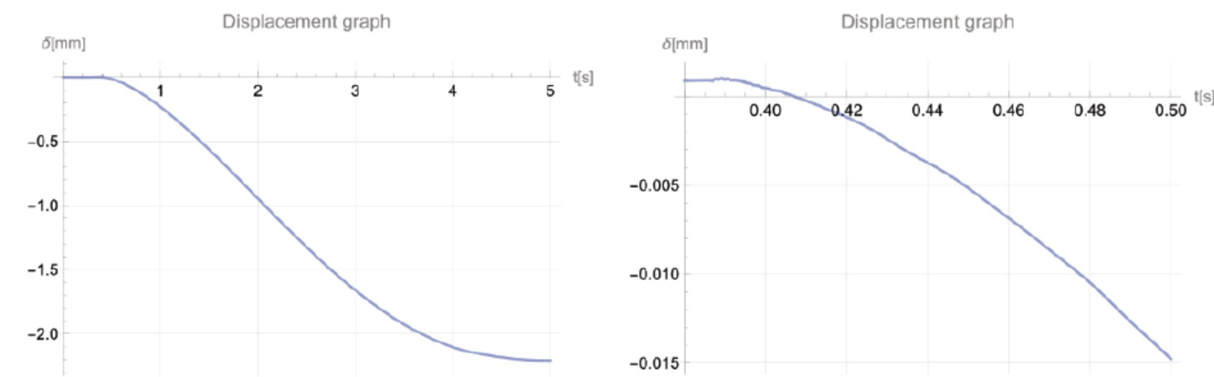

Figure 4.
Displacement graph in the entire measurement range and in the subrange of high accelerations

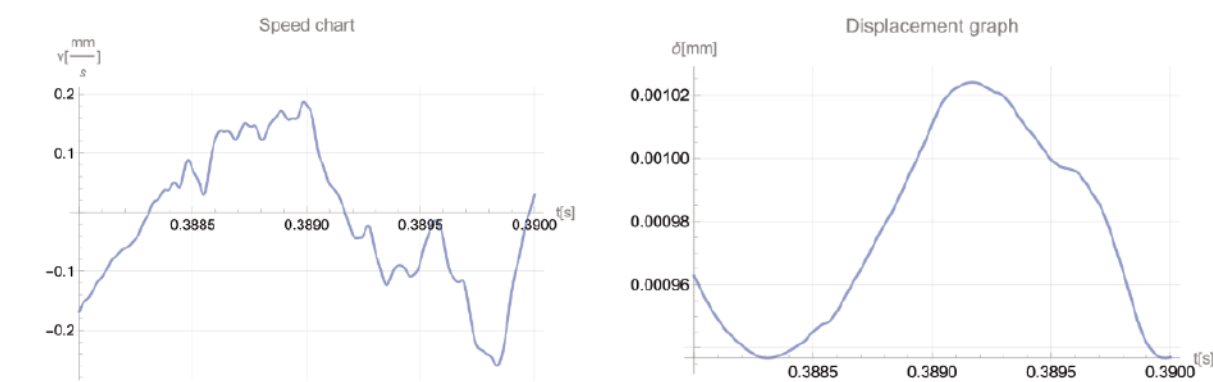

Figure 5.
Velocity and displacement diagrams of displacements in the range as in Fig. 2

Figure 3 shows that the highest velocity values are reached around the 2 s of measurement, when the vibrations seem to stop. The speed value is about three times higher than at the moment of the greatest accelerations. Further vibrations with small accelerations cause the speed to drop to zero in the fifth second.

By integrating the semi-analytical velocity function, we obtain the displacement function.

```
δ[i_][t_] := δ[i][t] = ∫Evaluate[v[i][t]] ⅆt
```

The graphs shown in Fig. 4 may suggest that the displacement function is quite smooth. However, a "microscopic" close-up (Fig. 5) to the time interval as in Fig. 2 shows the actual course of the function after successive integrations.

We can check the quality of integration by differentiation. So, if we differentiate the velocity function and subtract the acceleration function from it,

```
Abs[∂{t} v[6][t] - a[6][t]]
```

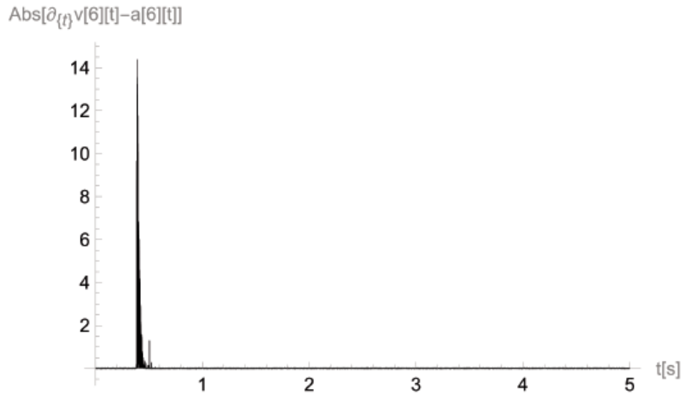

Figure 6.
Absolute error of interpolation

we get what can be seen in Fig. 6, that the error is negligible since even at points where acceleration is equal to 3000 mm/s^2 the error is not bigger than 15 mm/s2 (0.5%). When we integrate the error function and divide it by an integral of an absolute acceleration value, we obtain an average error that is equal to 0.2%.

4. INTEGRATION OF EQUATIONS OF MOTION

The above method can be used to integrate equations of motion [3, 4]. We will show it in the example of a dynamical system with one degree of freedom excited by the analysed acceleration function.

Mathematica can provide a general solution to this problem using a function to analytically solve differential equations.

```
DSolve[{q''[t] + 2 c q'[t] + ω^2 q[t] == a[t],
q[0] == 0, q'[0] == 0}, q[t], t]
/. {K[i_] → τ}
```

The result of such a generally formulated task is given by the program in the following form.

$$\left\{\left\{q[t]\to -e^{t\left(-c+\sqrt{c^2-\omega^2}\right)}\int_1^0 \frac{e^{2c\tau+\tau\left(-c-\sqrt{c^2-\omega^2}\right)}\sqrt{c^2-\omega^2}\,a[\tau]}{2\,(c-\omega)\,(c+\omega)}\,d\tau + e^{t\left(-c+\sqrt{c^2-\omega^2}\right)}\int_1^t \frac{e^{2c\tau+\tau\left(-c-\sqrt{c^2-\omega^2}\right)}\sqrt{c^2-\omega^2}\,a[\tau]}{2\,(c-\omega)\,(c+\omega)}\,d\tau - e^{t\left(-c-\sqrt{c^2-\omega^2}\right)}\int_1^0 -\frac{e^{2c\tau+\tau\left(-c+\sqrt{c^2-\omega^2}\right)}\sqrt{c^2-\omega^2}\,a[\tau]}{2\,(c-\omega)\,(c+\omega)}\,d\tau + e^{t\left(-c-\sqrt{c^2-\omega^2}\right)}\int_1^t -\frac{e^{2c\tau+\tau\left(-c+\sqrt{c^2-\omega^2}\right)}\sqrt{c^2-\omega^2}\,a[\tau]}{2\,(c-\omega)\,(c+\omega)}\,d\tau\right\}\right\}$$

This expression – as beings endowed with real intelligence, not artificial, still superior to machines – we can simplify to the form:

$$q[t\_] := e^{-t\left(c-\sqrt{c^2-\omega^2}\right)}\int \frac{e^{\tau\left(c-\sqrt{c^2-\omega^2}\right)}\,a[\tau]}{2\sqrt{c^2-\omega^2}}\,d\tau - e^{-t\left(c+\sqrt{c^2-\omega^2}\right)}\int \frac{e^{\tau\left(c+\sqrt{c^2-\omega^2}\right)}\,a[\tau]}{2\sqrt{c^2-\omega^2}}\,d\tau$$

which in traditional notation can be denoted as:

$$q(t) := e^{-t\left(c-\sqrt{c^2-\omega^2}\right)}\int \frac{e^{\tau\left(c-\sqrt{c^2-\omega^2}\right)}a(\tau)}{2\sqrt{c^2-\omega^2}}\,d\tau - e^{-t\left(c+\sqrt{c^2-\omega^2}\right)}\int \frac{e^{\tau\left(c+\sqrt{c^2-\omega^2}\right)}a(\tau)}{2\sqrt{c^2-\omega^2}}\,d\tau$$

This solution satisfies the initial conditions that, at time t = 0, both the displacement and the velocity are equal to zero. This is done taking into account the assumption that in the launching moment of motion registration the system do not move. It makes it possible to omit the discussion on constants of integration.

It is necessary to explain why definite integrals have been replaced by indefinite ones. It is possible since the definite integral is equal to the difference of antiderivatives of the integral at both ends of the integration interval and the antiderivative of the interpolated function at the moment of time equal to 0 is equal to 0, too.

In this case, the interpolation procedure requires an additional step. The integrands in the above formula are a mixture of analytical functions and interpolated acceleration.

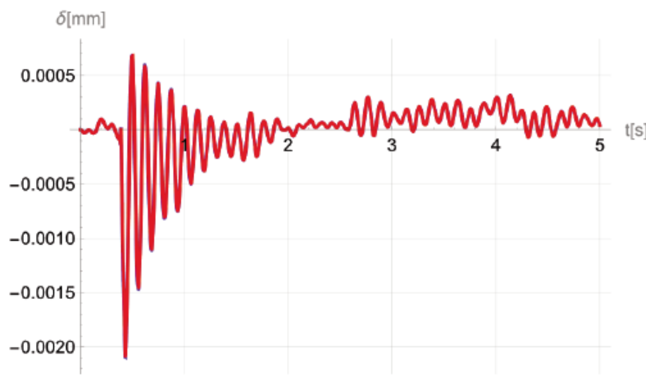

Figure 7.
Result of integration of the equation of motion

$$\mathbf{f1} = \tau \mapsto \frac{e^{\tau\left(c-\sqrt{c^2-\omega^2}\right)}\, a[6][\tau]}{2\sqrt{c^2-\omega^2}};$$

$$\mathbf{f2} = \tau \mapsto \frac{e^{\tau\left(c+\sqrt{c^2-\omega^2}\right)}\, a[6][\tau]}{2\sqrt{c^2-\omega^2}};$$

Such mixtures cannot be integrated analytically.

Let us set the cyclic frequency and dumping parameter equal to

$\omega = 50;\ c = 2$

To make it possible to integrate these functions analytically, we have to reinterpolate them:

$$\mathbf{cf1} = \mathrm{Interpolation}\left[\mathrm{Table}\left[\{t, \mathrm{f1}[t]\}, \left\{t, 0, 5, \frac{1}{327679 \times 2}\right\}\right], \mathrm{InterpolationOrder} \to 6\right];$$

$$\mathbf{cf2} = \mathrm{Interpolation}\left[\mathrm{Table}\left[\{t, \mathrm{f2}[t]\}, \left\{t, 0, 5, \frac{1}{327679 \times 2}\right\}\right], \mathrm{InterpolationOrder} \to 6\right];$$

Now, the calculations are straightforward. Input in the form.

$$y[t\_] = e^{-t\left(c-\sqrt{c^2-\omega^2}\right)} \int \mathrm{cf1}[t]\, dt - e^{-t\left(c+\sqrt{c^2-\omega^2}\right)} \int \mathrm{cf2}[t]\, dt$$

produces the following output:

$$-e^{-\left((2+8\,i\,\sqrt{39})\,t\right)}$$

```
InterpolatingFunction[ Domain: {{0., 5.}} Output: scalar ][t]
Data not in notebook. Store now
```

$$e^{-\left((2-8\,i\,\sqrt{39})\,t\right)}$$

```
InterpolatingFunction[ Domain: {{0., 5.}} Output: scalar ][t]
Data not in notebook. Store now
```

This result is presented in Fig. 7

It could be added that the general solution of a differential equation of motion can be presented in the equivalent form:

$$q[t\_] := \frac{1}{e^{c\,t}}\left(\mathrm{Cos}\left[\sqrt{\omega^2-c^2}\, t\right]\int -\frac{e^{c\,\tau}\, a[\tau]\, \mathrm{Sin}\left[\sqrt{\omega^2-c^2}\, \tau\right]}{\sqrt{\omega^2-c^2}}\, d\tau + \mathrm{Sin}\left[\sqrt{\omega^2-c^2}\, t\right]\int \frac{e^{c\,\tau}\, a[\tau]\, \mathrm{Cos}\left[\sqrt{\omega^2-c^2}\, \tau\right]}{\sqrt{\omega^2-c^2}}\, d\tau\right)$$

in traditional notation:

$$q(t) := \frac{1}{e^{c\,t}}\left(-\cos\left(\sqrt{\omega^2-c^2}\, t\right)\int \frac{e^{c\,\tau} a(\tau)\sin\left(\sqrt{\omega^2-c^2}\,\tau\right)}{\sqrt{\omega^2-c^2}}\, d\tau + \sin\left(\sqrt{\omega^2-c^2}\, t\right)\int \frac{e^{c\,\tau} a(\tau)\cos\left(\sqrt{\omega^2-c^2}\,\tau\right)}{\sqrt{\omega^2-c^2}}\, d\tau\right)$$

This form seems to be more suitable when c < ω. In this case, all the terms in the expression above are real numbers. In the previous case, they are complex numbers, but the final result is a real function. It is not a problem for *Mathematica* to deal with such functions.

Both forms of equations exactly satisfy the differential equation and initial conditions. However, the solution was compared with numerical approaches. The numerical solution is less precise and requires significant dense steps, so the computation time is comparable. This comparison will be shown in the following contributions.

The numerical problem may occur with both formulas mentioned above if we are close to critical dumping c ≈ ω. In the case of critical dumping c = ω we have the following solution:

```
q[t_] := e^(-t ω) (t ∫e^(τ ω) a[τ] dτ - ∫e^(τ ω) τ a[τ] dτ)
```

in traditional notation:

$$q(t) := e^{-t\,\omega}\left(\int e^{\tau\,\omega}\,a(\tau)\,d\tau - \int e^{\tau\,\omega}\,\tau\,a(\tau)\,d\tau\right)$$

Numerical problems connected with this special case can be overridden with higher precision of computation, since Mathematica can do calculations with an arbitrary precision.

5. CONCLUSIONS

The presented approach can be an alternative and verification tool for numerical algorithms embedded in accelerometer and system software for numerical calculations in the near future. This statement requires further comparison with contemporary professional numerical algorithms. Such comparisons will be made in further papers. The high speed of semi-analytical calculations and the verifiability of the results speak to their use in the analysis of paraseismic shocks in mining areas.

REFERENCES

[1] Walentyński R. (2023). Półanalityczne całkowanie akcelerogramu i równań ruchu układu poddanego wstrząsowi parasejsmicznemu (Semi-analytical integration of the accelerogram and equations of motion of a system subjected to a paraseismic shock), presented at 68 Scientific Krynica Conference KILiW PAN i KN PZITB, 24-28 September 2023, in Orlik-Kożdoń B., Jasiński M., Pokorska-Silva I. (eds.): Wyzwania budownictwa na terenach górniczych, pogórniczych i zdegradowanych (Construction challenges in mining, post-mining and degraded areas). Book of abstracts, 2023, Silesian University of Technology, 317–320, Gliwice, Poland.

[2] Guorui H. & Lu T. (2016). Review on baseline correction of strong-motion accelerogram, *East Asian Science Technology and Society an International Journal, 36*(3), 309–314.
DOI: 10.11648/j.ijsts.20150306.16.

[3] Chmielewski T., Zembaty Z. (2015). Podstawy dynamiki budowli (Fundamentals of building dynamics). Warszawa: Arkady.

[4] Gromysz K. (2016). Dynamika budowli. Obliczenia układów prętowych o masach skupionych (Building dynamics. Calculations of rod systems with concentrated masses). Warszawa: PWN.

[5] Wolfram Research, Inc. (2022). Mathematica, Version 13.3, Champaign, IL, https://www.wolfram.com/mathematica

[6] Esfandiari R. S. (2017). Numerical Methods for Engineers and Scientists Using MATLAB®. Boca Raton: CRC Press.

[7] Dechaumphai P. &, Wansophark N. (2020). Numerical Methods in Science and Engineering: Theories with MATLAB, Mathematica, Fortran, C and Python Programs. Oxford: Alpha Science International Ltd..

[8] Dubin D. (2003). Numerical and Analytical Methods for Scientists and Engineers Using Mathematica. Hoboken: John Wiley & Sons, Inc..