Revista de Matemática: Teoría y Aplicaciones 2015 **22**(1) : 21–30
cimpa – ucr issn: 1409-2433 (Print), 2215-3373 (Online)

SPECIAL SOFTWARE FOR COMPUTING THE SPECIAL FUNCTIONS OF WAVE CATASTROPHES

SOFTWARE ESPECIAL PARA CALCULAR LAS FUNCIONES ESPECIALES DE CATÁSTROFES DE OLAS

Andrew S. Kryukovsky[*] Dmitry S. Lukin[†]
Sergey V. Rogachev[‡]

Received: 16/Dec/2013; Revised: 3/Oct/2014;
Accepted: 17/Oct/2014

[*] Faculty of Information Systems and Computer Technology, Russian New University, Radio st., 22, 105005 Moscow, Russia. E-mail: kryukovsky@rambler.ru

[†] Moscow Institute of Physics and Technology (State University), Institutskiy lane 9, 141700 Dolgoprudny, Moscow Region, Russia. E-mail: lukin@mail.mipt.ru

[‡] Misma dirección que/*same address as* A.S. Kryukovsky. E-mail: rogachevsergei@gmail.com

Abstract

The method of ordinary differential equations in the context of calculating the special functions of wave catastrophes is considered. Complementary numerical methods and algorithms are described. The paper shows approaches to accelerate such calculations using capabilities of modern computing systems. Methods for calculating the special functions of wave catastrophes are considered in the framework of parallel computing and distributed systems. The paper covers the development process of special software for calculating of special functions, questions of portability, extensibility and interoperability.

Keywords: ODE method; special functions of wave catastrophes; numerical methods; algorithms; parallel computing; distributed computing.

Resumen

Se considera el método de ecuaciones diferenciales ordinarias ordinarias en el contexto de calcular funciones especiales de catástrofes de olas. Se describen métodos y algoritmos numéricos complementarios. El artículo muestra enfoques para acelerar tales cálculos usando capacidades modernas de sistemas de cálculo. Se consideran métodos para calcular funciones especiales de catástrofes de olas en el marco de computación en paralelo y sistemas distribuidos. El artículo cubre el proceso de desarrollo de software especial para calcular funciones especiales, así como asuntos de portabilidad, extensibilidad e interoperabilidad.

Palabras clave: métodos de EDO; funciones especiales de catástrofes de olas; métodos numéricos; algoritmos; computación en paralelo; computación distribuida.

Mathematics Subject Classification: 78A99.

1 Introduction

The theory of wave catastrophes is an important field, which has a direct practical importance in areas such as radiolocation, formation of directional beams in plasma, transfer of data over fiber-optic lines, etc. Methods of the theory of wave catastrophes allow to go beyond the boundaries of the known asymptotic methods in the study of electromagnetic wave propagation.

This article describes approaches to numerical calculation of special functions using modern computing systems. The software product based on these methods is planned to be embedded to the information system ”Wave catastrophes in radio physics, acoustics, and quantum mechanics.”

2 ODE method

Known methods of the theory of wave catastrophes (for example, the contour method, summing the Taylor series) are difficult to apply for multiparameter catastrophes as their complexity increases with the number of parameters [3, 5]. The method of ordinary differential equations is free from these disadvantages [4].

ODEs for a rather small number of catastrophes are derived by hand [7], but, in theory, this can be done automatically. The algorithm can be implemented quite simply using one of the modern packages of symbolic computation (Maxima or Wolfram Mathematica, for example).

Definition 1 *A vector consisting of a wave catastrophe special function and a part of its first direvatives is a fundamental vector of this special function (for example, fundamental vector of A_3 catastrophe is (1)):*

$$\vec{W} = (V, V^1, V^2), \gamma = 3. \tag{1}$$

The other derivatives of a special function are uniquely expressed in terms of the fundamental vector's components with the help of linear algebraic relations resulting from the canonical system of differential equations.

Argues that for calculation of the fundamental vector's components a system of ordinary differential equations can be derived from the system of canonical equations. The resulting system can be easy calculated by numerical methods like Runge-Kutta or Kutta-Merson.

For example, consider the A_3 catastrophe (2):

$$V_{A_3}(\vec{\lambda}) = \int_{-\infty}^{+\infty} exp(i(x^3 + \lambda_2 x^2 + \lambda_1 x))\, dx. \tag{2}$$

The following system of canonical equations corresponds to the catastrophe described above [4]:

$$\begin{aligned} (\lambda_1 - 4\frac{\partial^2}{\partial\lambda_2\partial\lambda_1} - 2i\lambda_2\frac{\partial}{\partial\lambda_1})I = 0, \\ (\frac{\partial^2}{\partial\lambda_1^2} - i\frac{\partial}{\partial\lambda_2})I = 0. \end{aligned} \tag{3}$$

The system of ODEs (4) can be derived from the system of canonical equations [4, 2]:

$$\begin{gathered}
\frac{dV}{d\lambda_1} = V^1, \frac{dV}{d\lambda_2} = V^2, \frac{dV^1}{d\lambda_1} = iV^2, \\
\frac{dV^2}{d\lambda_1} = \frac{1}{4}(\lambda_1 V - 2i\lambda_2 V^1) \equiv U_{21}, \frac{dV^1}{d\lambda_2} = U_{21}, \\
\frac{dV^2}{d\lambda_2} = -\frac{i}{4}(V + \lambda_1 V^1 + 2\lambda_2 V^2).
\end{gathered} \tag{4}$$

The set of initial conditions (5) supplements the system:

$$\begin{gathered}
V(0) = \frac{1}{2}\Gamma(\frac{1}{4}) exp(\frac{i\pi}{8}), \\
V^1(0) = 0, \\
V^2(0) = \frac{i}{2}\Gamma(\frac{3}{4}) exp(\frac{i3\pi}{8}).
\end{gathered} \tag{5}$$

The method of deriving a system of ODEs can be described by the following steps [4, 6]:

- First of all, it is necessary to select the starting point. It may be $\lambda_2 = 0$ and $\lambda_2 = 0$, for example (for A_3).
- After the selection of the starting point for a future system of ODEs it is needed to determine the components of the vector $\vec{W}$ and its length. This can be done by drawing up a table, each row of which corresponds to the function itself or one of its first derivatives, and the columns correspond to the derivatives with respect to each parameter (the table corresponding to the A_3 catastrophe (Table 1)).
- All components of the table are derivable from the system of canonical equations. The table also helps to determine the amount of fundamental vector's components.

3 Numerical methods and algorithms

The software solution uses the Kutta-Merson method to calculate special functions with different sets of parameters. This is a method with fourth-order accuracy and step autocorrection [8].

	$\frac{\partial}{\partial\lambda_1}$	$\frac{\partial}{\partial\lambda_2}$
V	(I, I)	(I, II)
V_1	(II, I)	(II, II)
V_2	(III, I)	(III, II)

Table 1: Table corresponding to the A_3 catastrophe.

.

Consider a system of the form:

$$\dot{x} = f(t, x). \tag{6}$$

To integrate the system following formulas can be applied:

$$\begin{aligned} x_{n+1} = x_n + \frac{1}{2}(k_1 + 4k_4 + k_5) + O(h^5), \\ k_1 = \frac{1}{3}hf(t_n, x_n), \\ k_2 = \frac{1}{3}hf(t_n + \frac{1}{3}h, y_n + k_1), \\ k_3 = \frac{1}{3}hf(t_n + \frac{1}{3}h, y_n + \frac{1}{2}k_1 + \frac{1}{2}k_2), \\ k_4 = \frac{1}{3}hf(t_n + \frac{1}{2}h, y_n + \frac{3}{8}k_1 + \frac{9}{8}k_3), \\ k_5 = \frac{1}{3}hf(t_n + h, y_n + \frac{3}{2}k_1 - \frac{9}{2}k_3 + 6k_4). \end{aligned} \tag{7}$$

The local truncation error is expressed by the following formula:

$$\delta \sim k_1 - \frac{9}{2}k_3 + 4k_4 - \frac{1}{2}k_5. \tag{8}$$

There are several criteria to determine how to change the integration step. In the current implementation the step is multiplied by 2 if the local truncation error is less than $\frac{5}{32}E$ (the specified accuracy of calculation). If the local truncation error exceeds $5E$ the step must be devided by 2. In the other cases the step is not changed.

Studied catastrophes are complex multi-variable systems; to calculate their special functions and visualize the result as a 3D plot it is needed to choose parameters which will be changed discretely, other parameters are assigned to some fixed values.

```
Data: Initial parameters of SODE
Result: Set of SODE solutions
initialization;
for FirsParam := MinFirstParam ... MaxFirstParam,
FirstParamStep do
    for SecondParam := MinSecondParam ... MaxSecondParam,
    SecondParamStep do
        prepare the initial vector;
        ResultVector := KuttaMersonMethod(InitialVector);
        save the resulting vector for future use;
    end
end
```

Algorithm 1: Sequential special wave function calculation.

The generic sequential algorithm for calculation special functions of wave catastrophes can be expressed with the simple enough pseudocode (Algorithm 1).

Usually it is enough to integrate the system from 0.0 to 1.0.

It is easy to note that the system can be integrated in the parallel environment by the devision of the set containing descrete parameters' values to a number of subsets. Integration in these parameter spaces can be performed independently and the results can be merged after the calculation. Thus, the algorithm showed above fits the parallel solution, and does not require any change. On the contrary, the set of input values must be prepared, and each thread in the system must get a separate part of this set.

The performance gain can be calculated using the formula (9). Assume N is a number of independent calculating units (cores, processors, network nodes, etc), time of calculation is proportional to $\frac{1}{N}$:

$$T_{parallel} = C\frac{T_{sequential}}{N}. \tag{9}$$

4 Portability and extensibility

One of the main requirements to the software developed in this work is portability and this requirement is refracted in some design solutions assumed in the project. C++ is used as a main programming language: it helps to develop very portable solutions which can be built for many modern architectures like x86, ARM, MIPS, etc. Graphical interface and

OpenGL graphics are drown through the FLTK library (very portable and tiny GUI toolkit), network communication and multithreading are based on small self-developed libraries (MS Windows, GNU/Linux and many other Unixes are supported).

The software uses a common intermediate representation of float point numbers to communicate between network nodes. Thus, these nodes can use different internal representations and work together in one computing network.

To simplify development of extensions the software supports a special domain specific language. ODEs of wave catastrophes can be described in the language and used to calculate these catastrophes. A part of this functionality is still under development.

5 Examples of visualization

The equation of the catastrophe A_3 is already described above (2), Figure 1 shows its module.

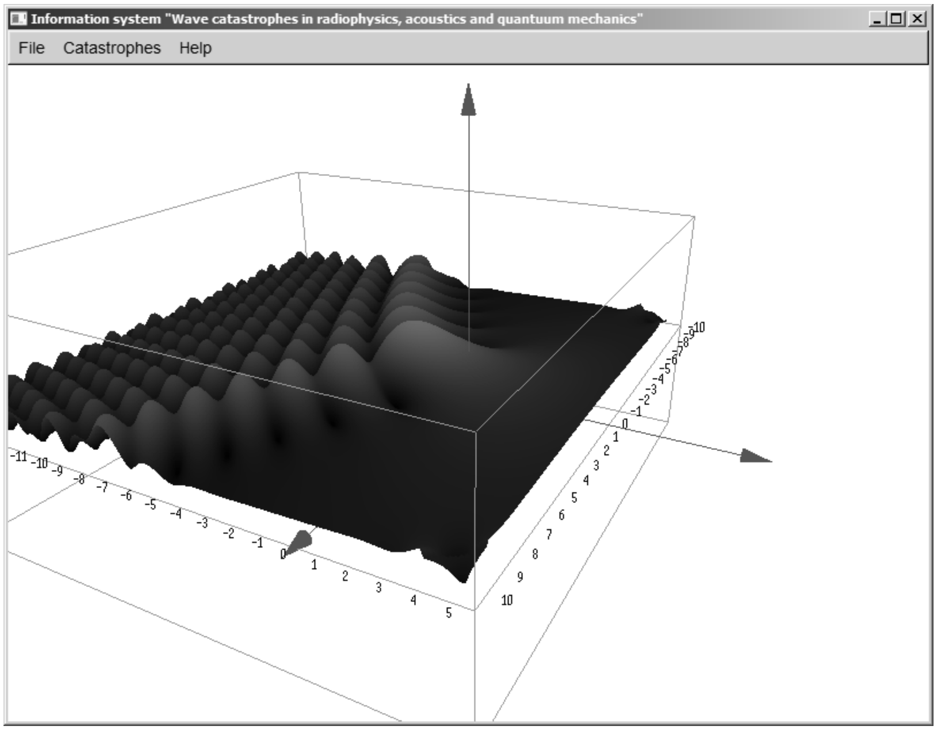

Figure 1: A_3, module. $\lambda_1 = -15 \ldots + 15$, $\lambda_2 = -10 \ldots + 10$.

Consider the graphical representation of the corner catastrophe A_1^4 (see Figure 2):

$$V_{A_1^4}(\lambda_1,\lambda_2,a)=\int_0^{+\infty}\int_0^{+\infty} exp(i(k_1z^2+ayz+k_2y^2+\lambda_1 y+\lambda_2 z))dy\,dz. \tag{10}$$

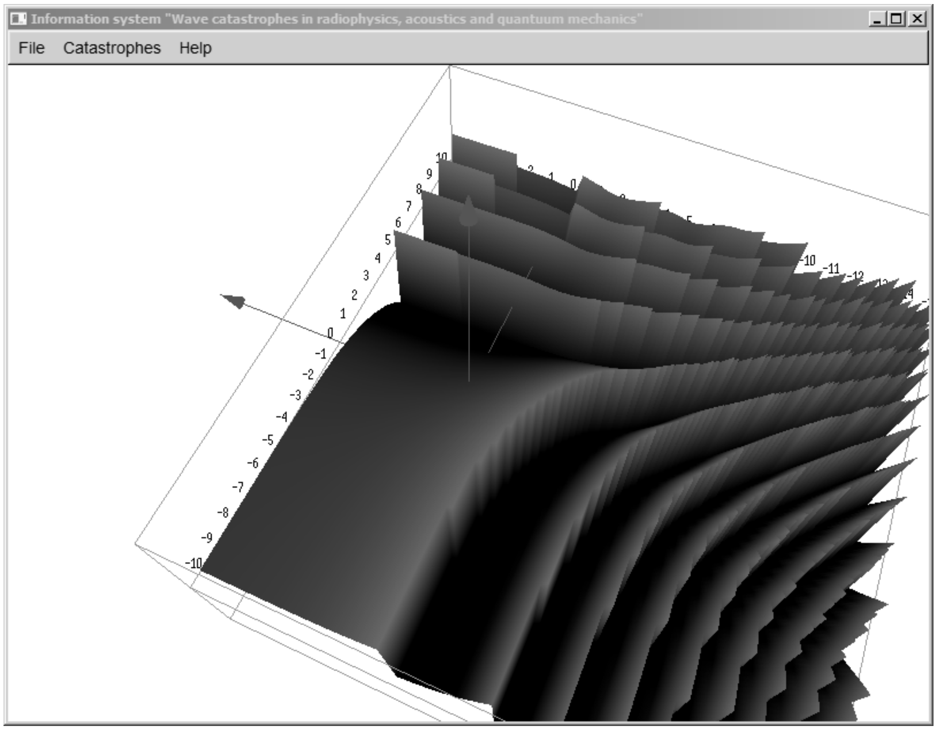
Information system "Wave catastrophes in radiophysics, acoustics and quantuum mechanics"
File
Catastrophes
Help
10
9
8
7
6
5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-7
-8
-9
-10
2
1
0
-10
-11
-12

Figure 2: A_1^4, phase. $\lambda_1=-10\ldots+10$, $\lambda_2=-15\ldots+5$, $\beta_1=-1$, $\beta_2=1$, $\alpha=2cos(\frac{\pi}{4})$.

The edge catastrophe $K_{4,2}$ described by the equation 11 is shown on figures 3 and 4:

$$\begin{aligned} &V_{K_{4,2}}(\lambda_1,\ldots,\lambda_4,\alpha)=\\ &\quad=\int_0^{+\infty}dz\int_{-\infty}^{+\infty}dx(i(z^2+\alpha x^2z+x^4+\lambda_1x+\lambda_2x^2+\lambda_3z+\lambda_4zx)). \end{aligned} \tag{11}$$

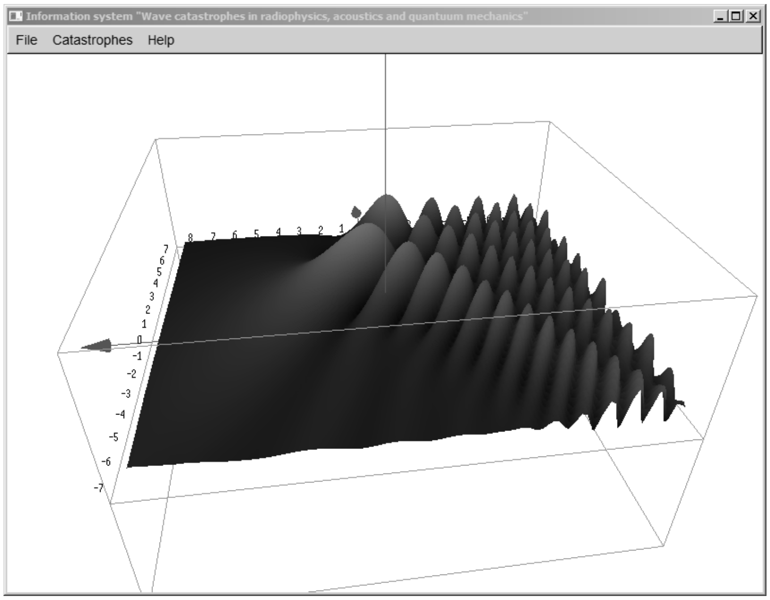
Information system "Wave catastrophes in radiophysics, acoustics and quantuum mechanics"
File
Catastrophes
Help
8
7
6
5
4
3
2
1
7
6
5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-7

Figure 3: $K_{4,2}$, module. $\lambda_1 = -8 \ldots + 8$, $\lambda_2 = -7 \ldots + 7$, $\lambda_3 = -2$, $\lambda_4 = -2$, $\alpha = -1$.

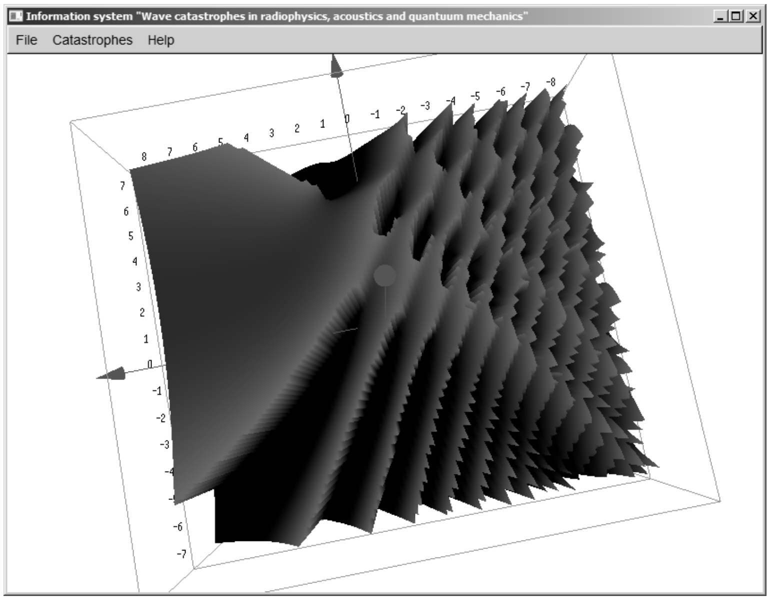
Information system "Wave catastrophes in radiophysics, acoustics and quantuum mechanics"
File
Catastrophes
Help
8
7
6
5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-7
-8
7
6
5
4
3
2
1
0
-1
-2
-3
-4
-6
-7

Figure 4: $K_{4,2}$, phase. $\lambda_1 = -8 \ldots + 8$, $\lambda_2 = -7 \ldots + 7$, $\lambda_3 = -2$, $\lambda_4 = -2$, $\alpha = -1$.

References

[1] Arnold, V.I. (1992) *Catastrophe Theory*, 3rd edition. Springer, Berlin.

[2] Dorokhina, T.V.; Kryukovsky, A.S.; Malyshenko, A.B. (2008) "Development of numerical algorithms for the calculation and visualization of wave catastrophes", *RosNOU Bulletin. Management, Computing and Informatics* **3** Russian New University, Moscow: 25–47.

[3] Ipatov, E.B.; Kryukovsky, A.S.; Lukin, D.S.; Palkin, E.A. (1986) "Edge catastrophes and asymptotics", *Reports of the Academy of Sciences*, USSR, **291**(4): 823–827.

[4] Kryukovsky, A.S. (1992) "Method of ordinary differential equations for computing the special functions of wave catastrophes (SWC)", *Diffraction and Propagation of Electromagnetic Waves: Interagency Collection of Scientific Papers*, Moscow: MIPT: 29–48.

[5] Kryukovsky, A.S.; Lukin, D.S.; Palkin, E.A. (1995) "Uniform asymptotics and corner catastrophes", *Reports of the Russian Academy of Sciences* **341**(4): 456–459.

[6] Kryukovsky, A.S. (2013) "The uniform asymptotic theory of edge and corner wave catastrophes", Monograph, Russian New University, Moscow.

[7] Kryukovsky, A.S.; Lukin, D.S. (2003) "The theory of calculating the reference focal and diffraction electromagnetic fields on the basis of special functions of wave catastrophes", *Technology and Electronics* **48**(8), Moscow (in Russian): 912–921.

[8] Shampine, L.F. (2005) "Error estimation and control for ODEs", *Journal of Scientific Computing* **25**(1): 3–16.