UNDERSTANDING MULTIPLE REFLECTIONS IN TWO PLANE MIRRORS BY BUILDING COMPUTER SIMULATIONS

D. MARCIUC[1,2], C. MIRON[1,*]

[1]*University of Bucharest, Faculty of Physics, Măgurele, Str. Atomistilor nr. 2, Bucuresti, România,*

[2]*"Mihai Eminescu" National College of Satu Mare, Str. M. Eminescu nr.5, Satu Mare, Romania, E-mail: daly.marciuc@gmail.com*

[*]Corresponding author: cmiron_2001@yahoo.com

Abstract. In this article we investigate how students can build and use, in high-school Physics lessons, various interactive simulations aiming to a better understanding of the phenomenon of multiple reflections in two intersected plane mirrors. The applets achieved with *GeoGebra* allow changing the angle between the mirrors, the position of the source object and the position of the first reflexion point of a successively reflected ray of light. The applets provide a dynamic representation of the images of a movable object and also a representation of a reflected ray, highlighting the relation between the positions of the images and the extensions of a reflected ray. After discussing the positioning of the images of the point-like objects, we will address the same problem for some colourful polygonal shapes, resulting in kaleidoscopic figures. Finally, using *GeoGebra*, the students can simulate the movement of a point-like body that collides elastically against two walls forming an arbitrary angle, based on the analogy between the trajectory of the body and the path of a successively reflected ray in two plane mirrors. By building and using simulations students have the opportunity to make connections between their Math and Science knowledge, developing their skills in solving interdisciplinary problems.

Key words: Multiple reflections, GeoGebra, geometric optics, plane mirrors, simulations, interdisciplinary learning.

1. INTRODUCTION

In this article we show that the *GeoGebra* software represents a suitable tool to build simulations of the phenomenon of multiple reflections in two plane mirrors. By involving students in modelling-based learning activities [1, 2], we aim to eliminate some misconceptions regarding this phenomenon, while developing students' skills in applying and connecting their Math and Physics knowledge. In a note in 1902, Lloyd showed that the problem of determining the number of images formed in two intersected plane mirrors was incompletely solved in the textbooks at that time [3]. Only later, Brown [4], Kulkarny [5] and Liu [6] bring solutions for the general case, in different ways.

Over time, the practitioner teachers have concerned themselves with making more accessible the understanding of the phenomenon [7, 8]. Merrill, in the early years of using computers in teaching, tested the use of computer programs for the study of geometric optics [9]. The emergence of the dynamic geometry software has created the opportunity to develop applications that facilitate the understanding of the various concepts of Physics, including geometric optics [10, 11].

2. DYNAMIC REPRESENTATION OF THE IMAGES OF AN OBJECT IN TWO INTERSECTED PLANE MIRRORS

The first problem posed to students is determining the number of images of an object in two plane mirrors forming an arbitrary angle *u*. In general, textbooks address only the situation where the angle *u* formed by the mirrors has the measure equal to an integer fraction of a straight angle, that is 180º/*n*, where *n* is an integer. In this case, the number of images is equal to 2*n*-1. But what happens when the angle *u* does not have this special form? By using the *GeoGebra* software, the students can achieve a simulation of the phenomenon to help them discover the correct answer.

First, students can build a mathematical model of the optical system, which will be the basis of the computer simulation. We will represent the two mirrors by two segments, *OA* and *OB*, where the point *O* is the origin of the reference system, and the extremities *A* and *B* have the polar coordinates (*l*, *u*/2), and respectively (*l*, -*u*/2). In this reference system, a point object *S* placed between the two mirrors has the polar coordinates (*r*, *t*), with *t* being in the range (-*u/2*, *u/2*). To determine the number of images, we will take into account that the next image will appear only if the previous image is placed in the semi-plane in front of the mirror in which it would be reflected. By solving the obtained inequalities, students will find the mathematical expression of the number of the images, and entering the commands listed in Table 1 they will generate a *GeoGebra* simulation of the phenomenon.

Table 1

GeoGebra commands for dynamically representing the multiple images of a point object in two mirrors intersected at an angle *u*

Command	Result
u=26º	Defining the angle formed by the two mirrors
O=(0,0); *l*=10; A=(*l* cos(-u/2), *l* sin(-u/2)); Segment[O,A] B=(*l* cos(u/2), *l* sin(u/2)); Segment[O,B]	Representing the two mirrors by the segments OA and OB
r=4; t=10º; S=(r cos(t), r sin(t))	Representing the object S
n_1=floor(1/2-t/u+180° /u) n_2=floor(1/2+t/u+180° /u)	Defining the number of images
S_i=Sequence[(r cos((-1)^i (t+i u)), r sin((-1)^i (t+i u))), i, 1, n_1] T_i=Sequence[(r cos((-1)^i (t-i u)), r sin ((-1)^i (t-i u))), i, 1, n_2]	Representing the images

Using the achieved simulation, students can see that if the angle *u* formed by the mirrors is not a whole fraction of a straight angle, then the number of images will change with the value of the angular coordinate *t* of the *S* object. They can note that the number of images changes when the *S* object crosses the extensions of the last mirror's images (Fig. 1). Then students can infer mathematically that the changes in the value of the numbers of the images occur when the angular coordinate of the object *S* crosses the limit values $\pm u$ (*f*-1/2), where we denote by *f*

the fractional part of the ratio $180°/u$ [12]. With our simulation, we can observe that if $f < 1/2$ the number of images formed is larger when the object is near one of the two mirrors. If $f > 1/2$ the number of images is higher when the object is in the central area, between the extensions of the last two images of the mirrors.
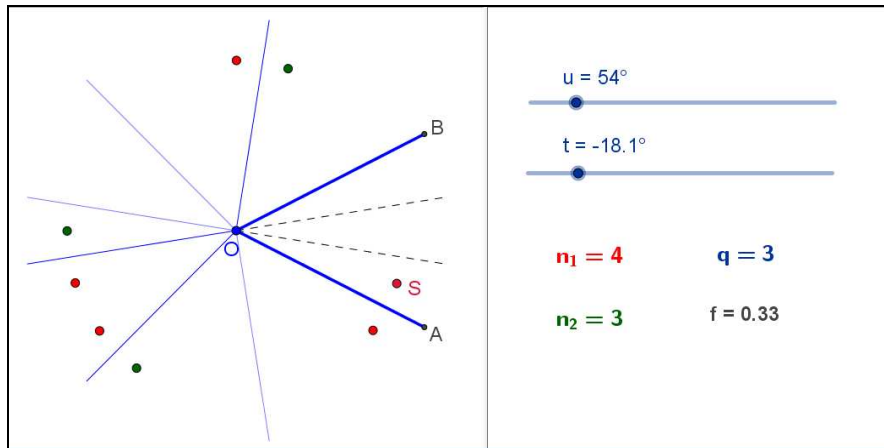

Fig. **1** – A dynamic representation with GeoGebra of the optical system. The full color version may be accessed at **http://www.infim.ro/rrp/**.

Figure 2 shows another simulation, obtained by extending the previous one. Here 35 red dots and 35 blue dots are represented, randomly distributed by the application, together with their 1330 images in two mirrors forming an 18 degrees angle. Table 2 lists the commands used to generate this kind of representation.
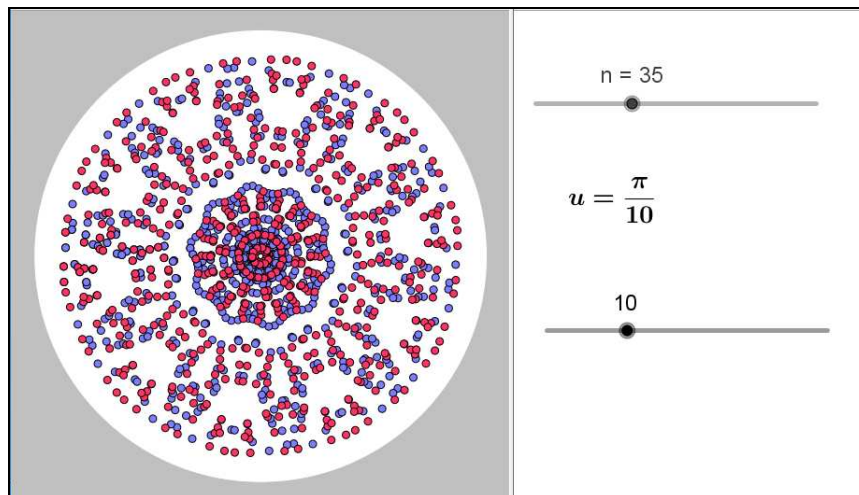

Fig. **2** – A simulation for the reflection of 70 differently coloured points in two intersected mirrors. The full color version may be accessed at **http://www.infim.ro/rrp/**.

By replacing the points with some differently coloured triangles, we obtain various kaleidoscopic images, as seen in Figure 3. The five different coloured triangles generating this kaleidoscopic image can be moved using the control points displayed after ticking the box labelled *Control Points*. Each triangle has two control points: one that allows us to rotate the triangle and another that allows us to translate it. The generated image, captured in Figure 3, has seven axes of symmetry represented by the mirrors and their images. The students can prove that the images of each triangle can be obtained by applying rotations of angles multiple of 2*u* to the original triangle and its primary image in one of the two mirrors, clockwise and respectively anticlockwise.

Table 2

GeoGebra commands for representing few randomly distributed points and their images in two mirrors forming an arbitrary angle u

Command	Result
n=30	Defining the number of randomly placed points
dist=Sequence[RandomBetween[1, 700] / 100, i, 1, n]	Generating the list containing the distances from origin to the randomly placed points
angl=Sequence[(-u) / 2 + i u / (n + 1), i, 1, n]	Creating the list of angular coordinates of the points
Sequence[(Element[dist, i] cos(Element[angl, i]), Element[dist, i] sin(Element[angl, i])), i, 1, n]	The list of points that will be reflected in the two mirrors
nS=Sequence[floor((u / 2 + Element[angl, i] + π) / u), i, 1, n] nT=Sequence[floor((u / 2 - Element[argm, i] + π) / u), i, 1, n]	The lists containing the number of images for each point
equence[Sequence[(Element[dist, j] cos((-1)^(i + 1) (i u - Element[angl, j])), Element[dist, j] sin((-1)^(i + 1) (i u - Element[angl, j]))), i, 1, Element[ns, j]], j, 1, n]	Representing the images S_i of the points
Sequence[Sequence[(Element[dist, j] cos((-1)^i (i u + Element[angl, j])), Element[dist, j] sin((-1)^i (i u + Element[angl, j]))), i, 1, Element[nt, j]], j, 1, n]	Representing the images T_i of the points

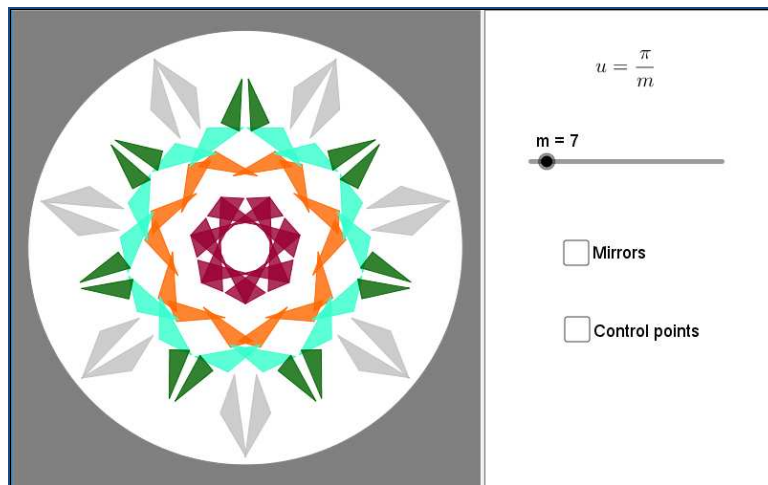Fig. **3** – A kaleidoscopic image obtained with GeoGebra. The full color version may be accessed at
http://www.infim.ro/rrp/.

3. DYNAMIC REPRESENTATION OF A MULTIPLY REFLECTED RAY

To understand how multiple images are formed, it is useful to represent a light ray reflected successively by the two mirrors. We can easily determine the number and position of the successive reflection points by applying the so-called *unfolding* procedure [8]. If A_1, the first point of reflection of the light ray, is located on the *OB* mirror, we construct in the semi-plane delimited by the line *OB* which does not contain point *A*, a few adjacent angles with the measure equal to *u* (Fig. 4). The total number of reflection points of a light beam starting from point *S* and reflected in point A_1 is equal to the number of points in which the SA_1 line intersects the sides of the angles with measure *u* previously constructed. By an elementary geometric reasoning [12-13] students can calculate the number of reflection points.

Based on the method of unfolding, we achieve a *GeoGebra* simulation to represent the path followed by a multiply reflected ray (Fig. 4). The obtained applet allows changing the angle *u* between the mirrors as well as the positions of the source object *S* and of the first point of incidence A_1. The commands used to create the applet are listed in Table 3.

Table 3

GeoGebra commands for representing a ray reflected in two mirrors intersected at an angle *u*

Command	Result
u=53°	Defining the angle formed by the two mirrors
O=(0,0); l=10; A=(l cos(u / 2), l sin(u / 2));	Representing the two mirrors by

|	
B= (l cos(-u / 2), l sin(-u / 2)); f=Segment[O,A]; g=Segment[O,B]	the segments OA and OB
r=4; t=0.04; S=(r cos(t), r sin(t))	Representing the source S
A_1=Point(g)	Representation of the first reflection point on the mirror g = OB
q=ceil(180°/u); list1=Sequence[(l cos(u / 2 - i u), l sin(u / 2 - i u)), i, 1, q] list2=Sequence[Ray[O, Element[list1, i]], i, 1, q]	Representing the sides of adjacent angles of measure u in the semi-plane delimited by OB that does not contain the point A.
h=Ray[S,A_1]	Representing the ray SA_1
i_1=Angle[B, A_1,S]	Defining the first angle of incidence
q1=ceil((180° - i_1) / u)	Determining the total number of reflection points
list3=Sequence[Intersect[h, Element[list2, i]], i, 1, q1]	Determining the intersection points of SA_1 ray with the sides of the adjacent angles with measure u
list4=Sequence[Circle[O, Segment[O, Point[Circle[O, Element[list3, i]]]]], i, 1, q1]	Determining the distances from point O to the reflexion points
list5=Sequence[Intersect[Element[list4, i], Ray[O, B]], i, 1, q1, 2]	Determining the position of the reflection points on OB mirror
list6= Sequence[Intersect[Element[list4, i], Ray[O, A]], i, 2, q1, 2]	Determining the position of the reflection points on OA mirror
a=Length[list5]; b=Length[list6]	Determining the number of reflection points
c=Min[a,b]; d=Min[a-1,b] ray1=Sequence[Segment[Element[list5, i], Element[list6, i]], i, 1, c] ray2=Sequence[Segment[Element[list6, i], Element[list5, i + 1]], i, 1, d]	Representing the multiply reflected ray
i= PerpendicularLine[Element[list5, a], g] j= PerpendicularLine[Element[list6, b], f] P= If[a □ b, Reflect[Element[list5, a], j], Reflect[Element[list6, b], i]] k=If[a □ b, Ray[Element[list6, b], P], Ray[Element[list5, a], P]]	Representing the last portion of the reflected ray
S'=Reflect[S, i]; Ray[A_1, S']	Representing the last portion of reflected ray for the case when A_1 is the only point of reflection
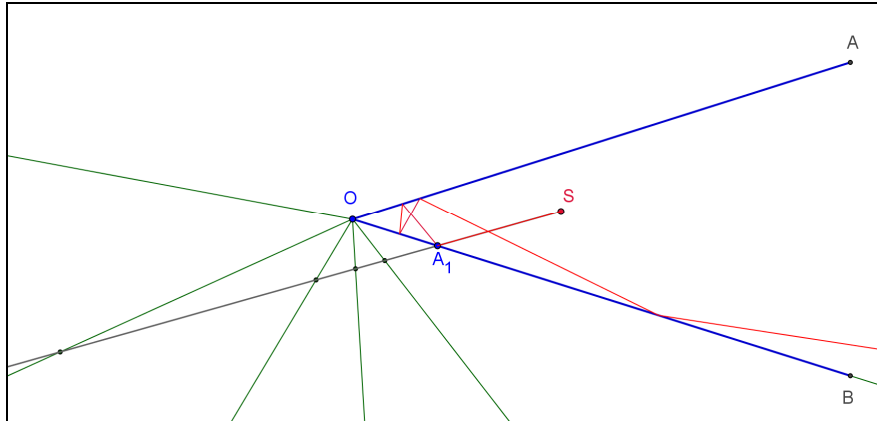
Fig. **4** – Simulating a multiply reflected ray based on the unfolding method. The full color version may be accessed at **http://www.infim.ro/rrp/**.

4. A GEOGEBRA DEMONSTRATION FOR THE BELONGING OF IMAGES TO THE EXTENSIONS OF A MULTIPLY REFLECTED RAY

The two simulations can be combined to highlight that the images are placed on the extension of the segments representing the successively reflected ray. Figure 5 shows the configuration obtained for the case when the angle u formed by the mirrors has the measure of 8° and the angular coordinate of the object S is 0.7°. The *GeoGebra* application allows visualizing details in this representation by pressing the *Zoom In* button. The first point of incidence, A_1, can be moved with the mouse, and so new configurations will be generated.
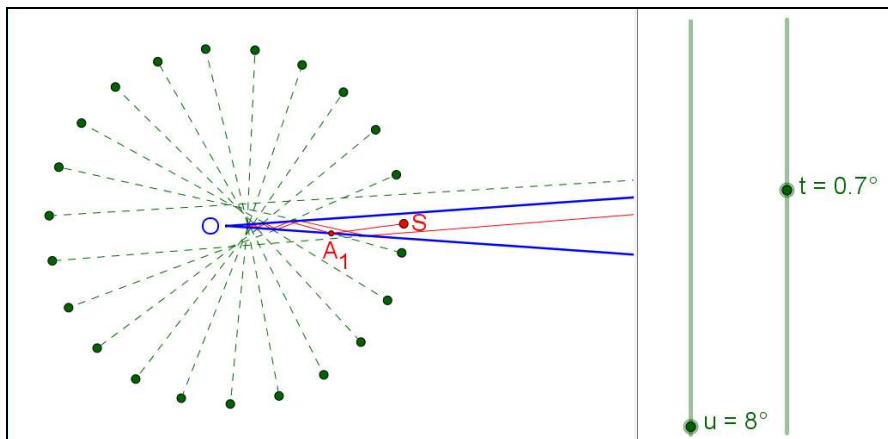

Fig. **5** –GeoGebra simulation showing the relation between the images and the extensions of a multiply reflected ray. The full color version may be accessed at **http://www.infim.ro/rrp/**.

The two sliders in the right-hand panel of the application (Fig. 5) allow changing the angle between the mirrors and changing the position of the *S* object. At any change, the application recalculates the values of the dependent variables and updates the images positions and also the reflected ray path. Table 4 lists the *GeoGebra* commands added to represent the segments that link the object's images to the reflection points.

Table 4

GeoGebra commands for representing segments determined by images and reflection points

Command	Result
imd= Sequence[(r cos(-(i u + t)), r sin(-(i u + t))), i, 1, q1, 2] Sequence[Segment[Element[list5, i], Element[imd, i]], i, 1, a]	Representing segments joining the reflection points on the OB mirror with the images placed in the semi-plane bounded by OB line and not containing A point
imu= Sequence[(r cos(i u + t), r sin(i u + t)), i, 2, q1, 2] Sequence[Segment[Element[list6, i], Element[imu, i]], i, 1, b]	Representing segments joining the reflection points on the OA mirror with the images placed in the semi-plane bounded by OA line and not containing B point

5. AN ANALOGOUS SIMULATION FOR SUCCESSIVE COLLISIONS

The achieved simulations gives us the opportunity to discuss with our students the analogy between the reflected ray path and the trajectory of a point-like object that starts from point S, collides elastically the *OB* wall at point A_1, then collides the *OA* wall, then the *OB* wall again, etc. An animation to simulate the movement of this point can also be achieved with *GeoGebra*. We create a variable *v*, for defining the speed of the object, and a variable t_c, representing the time elapsed since the beginning of the movement. We initialize t_c variable with 0, setting on the automatic increment of its value for running the animation. We represent the point *M*, which runs on the semi-line $h = (SA_1$ with constant speed *v*, then we will determine the corresponding position of *M* on the real trajectory, obtained by the above-mentioned method of *unfolding*. Table 5 lists the *GeoGebra* commands added to achieve the animation.

Table 5

GeoGebra commands for simulating the elastic collisions with two walls forming an angle *u*

Command	Result
t_c=0; v=1; dist=t_c*v; c= Circle[S, dist]; M= Intersect[c, h]	Determining the position of the object on half-line h=(SA$_1$, representing the unfolded trajectory

ang1= Sequence[Angle[M, O, Element[list3, i]], i, 1, q1, 2] sl1= Sequence[Ray[O, (r cos((-u) / 2 + Element[ang1, i]), r sin((-u) / 2 + Element[ang1, i]))], i, 1, a] Sequence[Intersect[Element[ray1, i], Element[sl1, i]], i, 1, c]	Representing the moving point on the $A_{2k-1}A_{2k}$ segments
ang2=Sequence[Angle[M, O, Element[list3, i]], i, 2, q1, 2] sl2=Sequence[Ray[O, (r cos(u / 2 - Element[ang2, i]), r sin(u / 2 - Element[ang2, i]))], i, 1, a] Sequence[Intersect[Element[ray2, i], Element[sl2, i]], i, 1, d]	Representing the moving point on the $A_{2k}A_{2k+1}$ segments
C= If[a □ b, Intersect[k, Element[sl2, c]], Intersect[k, Element[sl1, a]]]; M'=Reflect[M, g]	Representing motion on the last portion of the trajectory

6. CONCLUSIONS

Students' involvement in building simulations for the multiple reflections in two plane mirrors helps them understand this phenomenon in depth and distinguish the various possible situations. Using a dynamic geometry software, such as *GeoGebra*, the students can easily check the correctness of the mathematical model they built. Validation of the mathematical model can be done experimentally, using the device made up of two plane mirrors joined by a hinge. Also, to understand the *unfolding* method that we used to represent a multiply reflected ray, we can print the image presented in Figure 4 on a transparent foil. By successively folding the foil along the sides of the six congruent angles, we can visualize the path of the multiply reflected ray. After bending, the semi-line SA_1 drawn on the transparent foil takes the shape of the reflected ray. On this material model the students can observe obeying of the reflection law, for each of the five reflection points of the considered example.

By highlighting the analogy between the path of a multiply reflected ray and the trajectory of a point-like object that collides successively against two intersected walls, we can make the transition to study, by building computer simulation, the issues in other chapters of Physics [14-15]. The *GeoGebra* software can be used successfully for modelling some mechanics phenomena, so the initiation of students by modelling various optical phenomena represents a suitable starting point. One of the important goals of physics education is to foster the students' understanding of physical concepts by the classic experiment approach [16-17] or by simulations with various software [18-21].

REFERENCES

1. M. Wells, D. Hestenes, G. Swackhamer, G. *Am. J. Phys.*, **63**(7), 606-619, (1995).
2. D. Hestenes. Notes for a Modeling Theory of Science, Cognition and Instruction. *Proceedings of the 2006 GIREP conference: Modeling in physics and physics education*, **31**, 27, (2006).
3. M.G. Lloyd. *Science*, **16**(399), 316-317, (1902).
4. F.L. Brown. *Am. J. Phys.*, **13**(5), 278-280, (1945).
5. V.M. Kulkarni. *Am. J. Phys.*, **28**(4), 317-318, (1960).
6. C.H. Liu. *Am. J. Phys.*, **30**(5), 380-381, (1962).
7. Greenslade Jr, T.B. *The Physics Teacher*, **20**(1), 29-33, (1982).
8. H.E. Evans. *Physics Teacher*, **34**(5), 314-315, (1996).
9. J.R. Merrill. *Am. J. Phys.*, **39**(4), 442-446, (1971).
10. Y. Rodríguez, A. Santana, L.M. Mendoza. *Phys. Educ.*, **48**(5), 621-628, (2013).
11. D. Marciuc, C. Miron, P. Csereoka. Understanding of Optics Phenomena by Building Mathematical and Computational Models. *Proceedings of Conference: 11th International Scientific Conference on eLearning and Software for Education (eLSE),* **3**, 503-510, (2015).
12. D. Marciuc, C. Miron. Using VPython programming for interdisciplinary learning and problem solving. *Proceedings of Conference 13th International Scientific Conference eLearning and Software for Education; Bucharest (eLSE),* **2**, 106-113, (2017).
13. D. Marciuc, P. Csereoka, P. Using multi-touch technologies for developing scientific and mathematical competencies. *Proceedings of 10th International Scientific Conference on eLearning and Software for Education (eLSE)*, **2**, 318-325, (2014).
14. D. Marciuc, C. Miron, E.S. Barna, *Rom. Rep. Phys*. **68**(4), 1603–1620, (2016).
15. D. Marciuc, C. Miron, E.S. Barna, *Rom. Rep. Phys*. **68**(3), 1296–1311, (2016).
16. N. Micescu, M.V. Rusu, C. Berlic, *Rom. Rep. Phys*., **65**(4), 1547-1556, (2013).
17. N. Micescu, E.S. Barna, C. Berlic, M.V. Rusu, *Rom. Rep. Phys*., **65**(2), 563-577, (2013).
18. I.Grigore, C. Miron, E.S. Barna, *Rom. Rep. Phys*. **69**(1), Article Number: 901 (2017).
19. I. Grigore, C. Miron, E.S. Barna, *Rom. Rep. Phys*. **68**(2), 891–904, (2016).
20. M. Oprea, C. Miron, *Rom. Rep. Phys*. **68**(2), 905–917, (2016).
21. C. Kuncser, A. Kuncser, G. Maftei, S. Antohe, *Rom. Rep. Phys*. **64**(4), 1119–1130, (2012).