Ministry of Education and Science of Ukraine

Odessa National Academy of Food Technologies

International Competition of Student Scientific Works

BLACK SEA SCIENCE 2021

Information Technology, Automation and Robotics

Proceedings

Odessa, ONAFT 2021

UDC 004.01/08

Black Sea Science 2021: Proceedings of the International Competition of Student Scientific Works. Information Technology, Automation and Robotics. / Odessa National Academy of Food Technologies; B.Yegorov, M. Mardar, S.Kotlyk (editors-in-chief.) [*et al.*]. – Odessa: ONAFT, 2021. – 526 p.

These materials of International Competition of Student Scientific Works «Black Sea Science 2021» contain the works of the contest participants in the section «Information technologies, automation and robotics» (not winners).

The author of the work is responsible for the accuracy of the information.

Odessa National Academy of Food Technologies, 2021

Organizing committee:

Prof. Bogdan Iegorov, D.Sc., Rector of Odessa National Academy of Food Technologies, Head of the Committee

Prof. Maryna Mardar, D.Sc., Vice-Rector for Scientific and Pedagogical Work and International Relations of Odessa National Academy of Food Technologies, Deputy Head of the Committee

Prof. Stefan Dragoev, D.Sc., Vice-Rector for Scientific Work and Business Partnerships of University of Food Technologies (Bulgaria)

Prof. Baurzhan Nurakhmetov, D.Sc., First Vice-Rector of Almaty Technological University (Kazakhstan)

Prof. Mircea Bernic, Dr. habil., Vice-Rector for Scientific Work of Technical University of Moldova (Moldova)

Prof. Jacek Wrobel, Dr. habil., Rector of West Pomeranian University of Technology (Poland)

Prof. Michael Zinigrad, D.Sc., Rector of Ariel University (Israel)

Dr. Mei Lehe, Ph.D., Vice-President of Ningbo Institute of Technology, Zhejiang University (China)

Prof. Plamen Kangalov, Ph.D., Vice-Rector for Academic Affairs of “Angel Kanchev” University of Ruse (Bulgaria)

Dr. Alexander Sychev, Ph.D., Assoc. Professor of Sukhoi State Technical University of Gomel (Belarus)

Dr. Hanna Lilishentseva, Ph.D., Assoc. Professor, Head of the Department of Merchandise of Foodstuff of Belarus State Economic University (Belarus)

Prof. Heinz Leuenberger, Ph.D., Professor of the Institute of Ecopreneurship of University of Applied Sciences and Arts (Switzerland)

Prof. Edward Pospiech, Dr. habil., Professor of the Institute of Meat Technology of Poznan University of Life Sciences (Poland)

Prof. Lali Elanidze, Ph.D., Professor of the Faculty of Agrarian Sciences of Iakob Gogebashvili Telavi State University (Georgia)

Dr. V. Kozhevnikova, Ph.D., Senior Lecturer of the Department of Hotel and Catering Business of Odessa National Academy of Food Technologies, Secretary of the Committee

The jury for the section «Information technologies, automation and robotics»

Head of the jury:

Sergii Kotlyk – Ph.D., Associate Professor, Director of the P.M. Platonov Educational-Scientific Institute of Computer Systems and Technologies "Industry 4.0" of Odessa National Academy of Food Technologies (Ukraine)

Members of the jury:

Piotr Artiemjew - Dr hab., Associate Professor in Decision Systems of the Faculty of Mathematics and Computer Science,University of Warmia and Mazury in Olsztyn (Poland)

Francisco Antonio Augusto – Dr., International Relations Manager of Higher Institute of Information and Communication Technologies (Angola)

Andrey Kuprijanov – Ph.D., Associate Professor of the Department of Software for Computers and Automated Systems of Belarusian National Technical University (Belarus)

Simon Milbert – Vice-President of Xtra Information Management, Inc. (USA)

Ivan Palov – D.Sc., Professor of University of Ruse "Angel Kanchev" (Bulgaria)

Degla Gérard Hugues – Communications and Training Manager of "MAPCOM solutions informatiques" company group (Benin)

Nugzar Kereselidze - Academic Doctor of Informatics (Computer Science), Associate Professor of the Department of Natural Sciences, Mathematics, Technology and Pharmacy, Sukhumi State University (Georgia)

Etibar Seyidzade - Associate Professor of the Department of Computer and Information Technolohies, Baku Engineering University (Azerbaijan)

Vladimir Golenkov, D.Sc., Professor of the Department of Intelligent Information Technologies, Belarusian State University of Informatics and Radio Electronics (Belarus)

Zhanar Omirbekova - Ph.D., Associate Professor of the Department of Automation and Management, Satbayev University (Kazakhstan)

Ivan Palov - D.Sc., Professor of the Department of Power Supply and Electrical Equipment, University of Ruse "Angel Kanchev" (Bulgaria)

Siarhei Palavenia - Ph.D., Associate Professor, Head of the Department of Telecommunication Systems, Belarusian State Academy of Communications (Belarus)

Alexander Goloskokov - Ph.D., Professor of the Department of Software Engineering and Information Technology Management, National Technical University "Kharkiv Polytechnic Institute" (Ukraine)

Peter Nikolyuk - D.Sc., Professor of the Department of Computer Technology, Vasyl Stus Donetsk National University (Ukraine)

Vladimir Palagin - D.Sc., Professor, Head of the Department of Radio Engineering, Telecommunications and Robotics Systems, Cherkasy State Technological University (Ukraine)

Viktor Khobin – D.Sc., Professor, Head of the Department of Technological Processes Automation and Robotic Systems of Odessa National Academy of Food Technologies (Ukraine)

Valeriy Plotnikov – D.Sc., Professor, Head of the Department of Information Technology and Cybersecurity of Odessa National Academy of Food Technologies (Ukraine)

Sergii Artemenko – D.Sc., Professor, Head of the Department of Computer Engineering of Odessa National Academy of Food Technologies (Ukraine)

Fedir Trishyn - Ph.D., Associate Professor, Vice-Rector on Scientific and Educational Work, Odessa National Academy of Food Technologies (Ukraine)

Valerii Levinskyi – Ph.D., Associate Professor of the Department of Technological Processes Automation and Robotic Systems of Odessa National Academy of Food Technologies (Ukraine)

Viktor Yehorov – Ph.D., Supervisor of the Laboratory of Mechatronics and Robotics of Odessa National Academy of Food Technologies (Ukraine)

Pavlo Lomovtsev – Ph.D., Associate Professor of the Department of Information Technology and Cybersecurity of Odessa National Academy of Food Technologies (Ukraine)

Yurii Kornienko – Ph.D., Associate Professor of the Department of Information Technology and Cybersecurity of Odessa National Academy of Food Technologies (Ukraine)

Serhii Shestopalov – Ph.D., Associate Professor of the Department of Computer Engineering of Odessa National Academy of Food Technologies (Ukraine)

Anatoly Galiulin - Ph.D., Associate Professor, Acting Head of the Department of Electromechanics and Mechatronics, Odessa National Academy of Food Technologies (Ukraine)

Secretary of the jury:

Oksana Sokolova – Senior Lecturer of the Department of Information Technology and Cybersecurity of Odessa National Academy of Food Technologies (Ukraine)

1-4-seconds-in-android (access date: 03.03.2020).

8. How to record sound in Android with better quality and reduced noise. StackOverflow: website. URL: https://stackoverflow.com/questions/22282353/how-to-record-sound-in-android-with-better-quality-and-reduce-noise (appeal date: 15.03.2020).

9. Noise absorption in Android when recording microphone sound. Twigs Tech Tips: website. URL: https://twigstechtips.blogspot.com/2013/07/android-enable-noise-cancellation-in.html?m=1 (appeal date: 25.03.2020).

YOUNG'S PROBLEM AND ITS APPLICATION

Author: ***Kulesh Oleksandr***
Advisor: ***Rusnak Mykola***
Yuriy Fedkovych Chernivtsi National University (Ukraine)

Abstract. *Today, millions of computers, phones, tablets, and other home appliances work with mobile networks around the world. The problem of providing a large number of devices with high-quality, fast, stable and inexpensive communication is almost at the forefront. An important role in solving this problem is played by the task of optimal, in a certain sense, placement of transceiver stations - nodes of the mobile network.*

Keywords: *Young's Problem; placement of circles; covering the flat surface with circles; minimum radius; minimum amount of circles.*

I. INTRODUCTION

A circle is a part of a plane that lies inside a circle. In other words, this is the locus of points on the plane, the distance from which to a given point, called the center of the circle, does not exceed a given non-negative number R - called the radius of this circle. If the radius is zero, then the circle degenerates into point [1]. A point set is a finite set of points in a plane with fixed coordinates. Coverage circle - a circle of minimum radius, which includes all points of the population.

Jung's Problem is to construct a covering circle for a given point set. By construction we mean the indication of the coordinates of the center of the circle and its radius (diameter).

In this work an attempt is made to adapt a well-known method of constructing a spanning circle of a point set on a plane of minimal diameter to solve such problems [2]. An algorithm was built, a program was developed and the results of its work were tested. Based on them, new tasks could be solved, which have important practical applications.

The developed software product, as a result of the tests showed its full efficiency. The product meets all functional requirements, interface requirements.

The software was implemented using WebStorm 2019.3.1 on HTML, CSS and JavaScript.

II. LITERATURE ANALYSIS

2.1. Problem statement №1

A point set is defined on the plane, the elements of which are set by mobile consumers who need a signal of at least a certain level. Anywhere in the plane, you can install a repeater signal of any power network (range). There is a law of attenuation of the repeater signal depending on the distance to it, and the price of the repeater, depending on its power. It is necessary to choose a repeater of the lowest cost (power), which would provide each consumer with a signal of the desired level and indicate its location.

2.2. Problem statement №2

If we consider **the problem №2** from a practical point of view, then placing one global signal transmitter is expensive, energy-intensive, and the radiation will be too strong in the area closest to it. Therefore, of course, we can consider another problem - covering the point set on the plane with a minimum number of circles with given radii, or covering the point set on the plane with a given number of circles with minimum radii. In terms of the problem №1, this means minimizing the number of repeaters of a given power to provide each signal consumer with at least a certain level, or choose the power of the repeater for this.

2.3. Where to begin

From [2], p. 122-130, we can summarize that from all the specified points by the enumeration method, you can select three points, the circle standing on which will also include all other points, and the radius will be minimal. They can be divided into three types, of which the further development of algorithms will follow:

- **acute-angled**, all angles are less than 90 degrees, - all three points will belong to a circle;
- **obtuse**, one angle is more than 90 degrees, - two sharp peaks will lie on the circle, and the large side between them will be the diameter;
- **right-angled** triangles can be attributed to the second type, since the point at the vertex of the right angle will lie on the circumscribing circle, and its hypotenuse will be the diameter.

Thus, in the first case, the coordinates of the center of the circle can be found from the coordinates of these points by the formulas (1) and (2)

$$x_O = -\frac{1}{2}\frac{y_A(x_B^2 + y_B^2 - x_C^2 - y_C^2) + y_B(x_C^2 + y_C^2 - x_A^2 - y_A^2) + y_C(x_A^2 + y_A^2 - x_B^2 - y_B^2)}{x_A(y_B - y_C) + x_B(y_C - y_A) + x_C(y_A - y_B)}$$

$$y_O = \frac{1}{2}\frac{x_A(x_B^2+y_B^2-x_C^2-y_C^2)+x_B(x_C^2+y_C^2-x_A^2-y_A^2)+x_C(x_A^2+y_A^2-x_B^2-y_B^2)}{x_A(y_B-y_C)+x_B(y_C-y_A)+x_C(y_A-y_B)}$$ [3]

In the second and third cases, the third point is not needed to calculate the radius and center of the circle. Formulas for calculating the coordinates of the midpoint of a segment on the plane (3) and (4)

$$x_c = \frac{x_a+x_b}{2} \qquad y_c = \frac{y_a+y_b}{2} \qquad [4]$$

The number of all combinations of three points for the set *n* – C_n^3 =*n*(n-1)*(n-2)/3*, the number of combinations of two points, to find the largest radius – C_n^2 =*n*(n-1)/2*. Therefore in order to save computational resources and time, it is better to start sorting out all combinations of two points.

III. METHODS OF RESEARCH

Before working methods and algorithms, we need to check the number of points in total. If their number is two or more, the methods of finding and constructing will work, in the case of one, the global circle will be this point, and the circle with a fixed radius will fall into it.

3.1. Method for solving the problem №1

We will find the largest distnce, draw a circle on it and check if all the points belong to it. If yes, the task of building a global circle is solved - the center lies in the middle of this segment and the radius is equal to half of it.

Otherwise, we will need to iterate over n-2 points to find the third and build a circle on them according to formulas (1) and (2), the first two points were left from an attempt to construct a circle by two points, since the largest segment will always be included in final triangle.

It follows from this that the solution to the main Young's problem can be found in a minimum of *n * (n-1) / 2* iterations and a maximum in *n *(n-1)/2 + (n-2)*.

3.2. Method for solving the problem №2

The fixed / given radius imposes an additional condition that makes it necessary to find a method for placing the minimum number of circles to describe all points - this is the problem of covering a set [5].

Predetermined radius creates two possible situations: the radius is equal to or greater than the global, which we found in the past method, and a radius less than the global.

In the first case, after comparing the radii, we will place the center of the circle with a fixed radius in the same place as the center of the global one.

In the second, we must place them so that the rules of the greedy algorithm [6] are fulfilled: at each step we will place one circle that covers the maximum number of not yet covered points until their number is equal to zero.

Since the possible positions of the placement of centers on a flat surface are infinite, we will start from the same methods for finding a circle by two and three points. The time to find all circles is approximately equal to $\sum_{i=1}^{m}\left(\frac{n_i(n_i-1)}{2}+n_i-2\right)$, where n_i is the number of points not yet covered at each step, m is the number of all steps.

3.3. Software product that implements algorithms

Now our task is to bring the methods to life by writing algorithms that will execute them.

To visually demonstrate the performance of the algorithms, we decided to write in JavaScript and upload it as a website with a convenient interface to the online platform GitHub: https://6mahdapihka9.github.io/Coverage_Task_v2/.

First of all, we will write the software modules that we will operate on. The "Dot" class will have two coordinates, the coverage property and information about the circle that dot was covered by.

```
class Dot{
  constructor(_x, _y) {
    if (_x !== undefined && _y !== undefined) {
      this.x = _x;
      this.y = _y;
    } else {
      this.x = 0;
      this.y = 0;
    }
    this.covered = false;
    this.coveredBy = undefined;
  }
}
```

The class "Dist" includes two dots, and the already calculated value of the distance.

```
class Dist {
  constructor(_v, _d1, _d2) {
    this.value = _v;
    this.dot1 = _d1;
    this.dot2 = _d2;
  }
}
```

The class "Circle" has coordinates and the value of the number of covered points, so later determine which circle with a fixed radius to build.

```
class Circle {
  constructor(_x, _y) {
    this.x = _x;
    this.y = _y;
    this.dotsCovered = 0;
  }
}
```

Create array *d* with all dots in it.

We start the method of building a global circle with a double nested loop, which will go through all possible distances and find the largest:

```
let m = 0, n = 0, max = 0;
let newDist;
for (let i = 0; i < d.length-1; i++){
      for (let j = i+1; j < d.length; j++) {
```

```
                        lenX = Math.abs(d[i].x - d[j].x);
                        lenY = Math.abs(d[i].y - d[j].y);
                        newDist = new Dist( Math.sqrt(Math.pow(lenX, 2) + Math.pow(lenY,2)) , d[i],
d[j]);
                        if (max < dist[k].value) {
                                max = dist[k].value;
                                m = i;
                                n = j;
                        }
                }
        }
```

m and *n* – numbers of points in the array, the distance between which is the greatest.

Next step – build a circle by two points and check if all the points fall into it:

```
tempX = (d[m].x + d[n].x)/2;
tempY = (d[m].y + d[n].y)/2;
tempR = Math.sqrt(Math.pow(d[m].x - tempX, 2) + Math.pow(d[m].y - tempY, 2));
enoughOne = true;
for (let i = 0; i < d.length; i++)
        if ((Math.pow(d[i].x-tempX, 2) + Math.pow(d[i].y-tempY, 2)) >
Math.round(tempR*tempR) + 0.0001) {
                enoughOne = false;
                break;
        }
Cx = tempX;
Cy = tempY;
R = tempR;
```

If during the check some points are not included, select the third anchor point with index *l* and draw a circle around them.

```
Cx = -(d[n].y * (d[m].x * d[m].x + d[m].y * d[m].y - d[l].x * d[l].x - d[l].y * d[l].y) + d[m].y
* (d[l].x * d[l].x + d[l].y * d[l].y - d[n].x * d[n].x - d[n].y * d[n].y)  + d[l].y * (d[n].x * d[n].x +
d[n].y * d[n].y - d[m].x * d[m].x - d[m].y * d[m].y)) / (2 * (d[n].x * (d[m].y - d[l].y) + d[m].x *
(d[l].y - d[n].y) + d[l].x * (d[n].y - d[m].y)));
Cy = (d[n].x * (d[m].x * d[m].x + d[m].y * d[m].y - d[l].x * d[l].x - d[l].y * d[l].y) + d[m].x
* (d[l].x * d[l].x + d[l].y * d[l].y - d[n].x * d[n].x - d[n].y * d[n].y) + d[l].x * (d[n].x * d[n].x +
d[n].y * d[n].y - d[m].x * d[m].x - d[m].y * d[m].y)) / (2 * (d[n].x * (d[m].y - d[l].y) + d[m].x *
(d[l].y - d[n].y) + d[l].x * (d[n].y - d[m].y)));
R = Math.sqrt(Math.pow(d[n].x - Cx, 2) + Math.pow(d[n].y - Cy, 2));
```

To construct circles of a fixed radius, the same algorithms were used with an additional counter of covered points to select the optimal circle at each iteration.

We will not consider the rest of the program details of the interface type and information input / output functions in this work, since we are not talking about them.

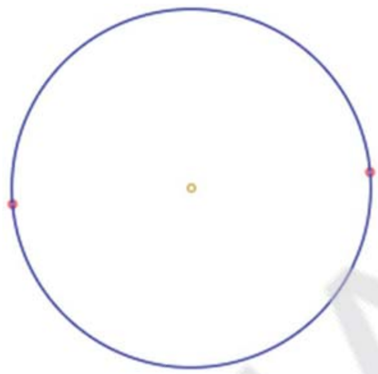
IV. RESULTS

The developed software has the following structure:

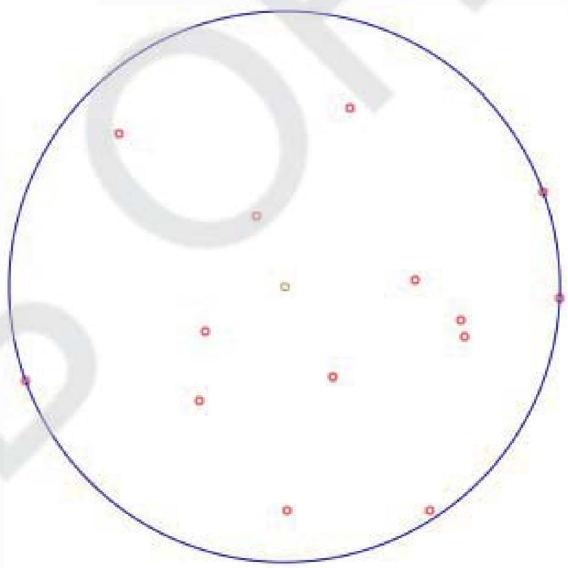
- classes Dot (point), Dist (segment), Circle (circle), which are the main objects with which the algorithms work;
- modules that process user actions;

- algorithms that find and build optimal circles;
- auxiliary modules (implement functions for convenient work with arrays and others).

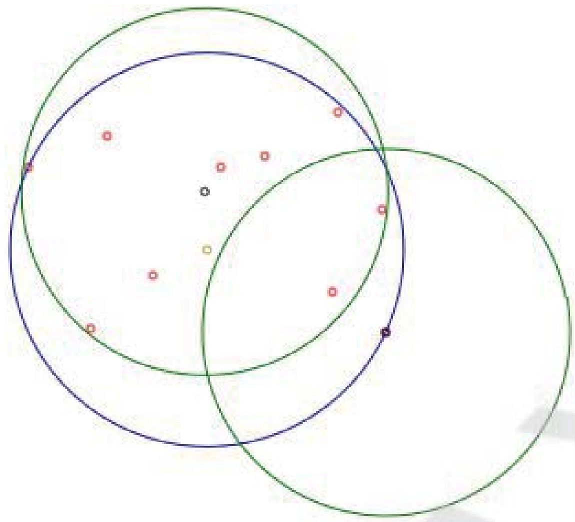
4.1. Software product testing



Img. 1. Test of two dots:
red dots – should be covered, brown – the center of the circle, blue – the global circle



Img. 2. Test of more dots



Img. 3. Test two algorithms at once:
green – circles of minimum radius

V. CONCLUSIONS

In the course of work on the practical part, software was developed to solve problems on covering a flat surface with circle of optimal radius and circles of fixed radius.

We used the following algorithms to design the software:

1. Algorithm for constructing a general range (3.1);
2. Algorithm for constructing circles with a given radius (3.2).

Thus, as a result of the conducted researches algorithms and their program realization for the decision of Jung's problem on the plane with various criteria of optimality are developed. The user has all the necessary information to design new and set up existing mobile networks.

Also in the developed product are realized:

- input of coordinates;
- saving coordinates;
- adding / deleting an image;
- cancellation of actions;
- delete everything;
- input of radius
- construction of a circle and / or circles;
- scaling;
- moving the image;
- output / hiding statistics;
- saving statistics.

The software product is developed in the programming language HTML, CSS, JavaScript. The product allows you to enter coordinates, both manually, on the canvas, and from a file, enter the radius with which the circles will cover the plane, create and work with files containing coordinates and information about the circles that cover them. It is also possible to add images to the canvas, for the convenience of coordinate points, scaling, moving the image, displaying statistics on the screen.

Checking the operation of the software product was carried out by testing different sequences of actions. No errors were detected during testing. The program works correctly.

The developed software product meets the requirements and has demonstrated its efficiency.

Regarding the prospect of continuing research, we note the need to solve such problems on surfaces with a given relief (maps), taking into account the presence of obstacles of natural and artificial nature. Separately highlight the spread of research on three-dimensional space - the task is to construct a sphere of minimum diameter, which covers a given set.

VI. REFERENCES

1. Круг (28 December 2020). In Wikipedia https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D1%83%D0%B3

2. Rademacher G., Teplitz O. (1962). Числа и фигуры. Опыты математического мышления (10th ed.) http://ilib.mccme.ru/djvu/bib-mat-kr/num-fig.htm

3. Описанная окружность (23 June 2020). In Wikipedia https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B8%D1%81%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BE%D0%BA%D1%80%D1%83%D0%B6%D0%BD%D0%BE%D1%81%D1%82%D1%8C

4. Середина отрезка (16 October 2020). In Wikipedia https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%80%D0%B5%D0%B4%D0%B8%D0%BD%D0%B0_%D0%BE%D1%82%D1%80%D0%B5%D0%B7%D0%BA%D0%B0

5. Задача о покрытии множества (17 October 2020). In Wikipedia https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%B4%D0%B0%D1%87%D0%B0_%D0%BE_%D0%BF%D0%BE%D0%BA%D1%80%D1%8B%D1%82%D0%B8%D0%B8_%D0%BC%D0%BD%D0%BE%D0%B6%D0%B5%D1%81%D1%82%D0%B2%D0%B0

6. Жадный алгоритм (30 December 2020). In Wikipedia https://ru.wikipedia.org/wiki/%D0%96%D0%B0%D0%B4%D0%BD%D1%8B%D0%B9_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC

TABLE OF CONTENTS

International Competition of Student Scientific Works

BLACK SEA SCIENCE 2021

Information Technology, Automation and Robotics

Proceedings

Odessa National Academy of Food Technologies

The collection includes student works of the participants of the competition, which were not included in the number of prize-winners. The texts of the competitive works are published in the form in which they were submitted by the authors. The authors of the articles are responsible for the content and form of submission of the material.

Responsible for the issue: Sergii Kotlyk

Computer typesetting and layout: Oksana Sokolova

Odessa 2021