ISSN 1533-9211

DOI 10.5281/zenodo.7106602

A BLACK WIDOW OPTIMIZATION ALGORITHM FOR SOLVING KENKEN PROBLEM

ALAA MOKHTAR[1,*], HEGAZY ZAHER[2] and NAGLAA RAGAA[1]

[1]Department of Operations Research and Management, Faculty of Graduate Studies for Statistical Research, Cairo University, Cairo, Egypt.
[2]Department of Mathematical Statistics, Faculty of Graduate Studies for Statistical Research, Cairo University, Cairo, Egypt.

Abstract:

Kenken is the most famous kind of arithmetic and grid puzzle. Kenken is a grid puzzle like Sudoku. In Kenken, the main objective is to fill cells from 1 to n×n grid on the condition that the mathematical operation on every cage is achieved, so that the main constraint in the kenken problem is to produce a certain "target" number when combined using a specified mathematical operation {+, -, ×, ÷}, under this condition, every number appears once on every column and row. The grids range in size from 3×3 to 9×9. KenKen is a new puzzle phenomenon that's sweeping the world educational, and fun, KenKen is a great way to exercise your brain and sharpen your mathematical tools at the same time. In this paper, the proposed research used python programming to solve the kenken puzzle by using a heuristic search and a black widow optimization algorithm. The proposed algorithm depends on a heuristic search that used logic rules. A lot of puzzle cases cannot be solved, then the black widow optimization algorithm will be run. Obtained results show the efficiency that the proposed algorithm in grids ranges in size 5×5, 7×7, and 9×9 are more efficient than the old ones.

Keywords: Discrete Optimization, Kenken Puzzle, Mathematical Games Analysis, Black Widow Optimization Algorithm.

1. Introduction

A KenKen or Mathdoku is a puzzle math and logic similar to Sudoku. Mathdoku is a grid of n×n cells as Fig.1, aims to fill the column, row grid and assuring no number is repeated in any row and column. Kenken contains an additional feature where the grid is divided into "cages". Each cage has many adjacent cells. The top left corner of every cage has a target number and an arithmetic operation {+, -, ×, ÷}. The numbers in a cage must combine (in any order) to achieve the target number using the arithmetic operation. Fig.1 gives an example of a 4 × 4 Kenken puzzle taken from the KenKen website [1].

Among similar puzzles, KenKen is particularly interesting for mathematicians. Due to its mathematical constraints, it brings a different level of interest and challenge to the solver. It is considered a more challenging task to bring a mathematical model that can solve the puzzle, too. The research which has been done on KenKen is not much; whereas Sudoku has been studied extensively. Watkins shows how ideas from number theory may be used for solving KenKen [2]. Reiter et al. discuss how KenKen may be used for developing reasoning skills for different students' levels [3].

ISSN 1533-9211

DOI 10.5281/zenodo.7106602

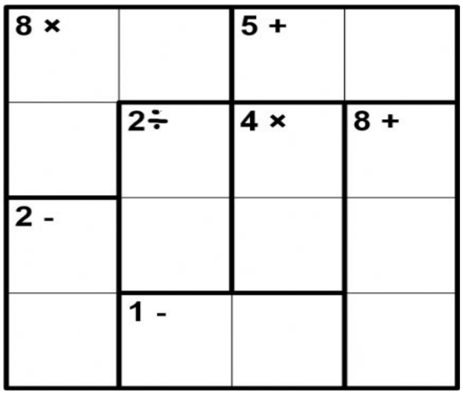

Figure 1: An example of 4×4 **KenKen.**

Solving The Kenken puzzle is not being discussed by many papers, however, some logical rules may be applied. There are Single square, naked pair, naked triple, evil twin, Hidden single, Killer combination, and X-wing. This paper will discuss the way to implement the rules so that computers may solve the problems generated by Will Shorts in Kenken Pro 2 game.

The paper is organized as follows: Section 2 gives the basic rules-based system in Kenken. Section 3 covers the black widow optimization algorithm. Section 4 gives the Black Widow Optimization Algorithm (BWOA) for modeling and solving the Kenken puzzle. The last section is a conclusion. An Appendix section offers code by Python to solve the Kenken puzzle.

2. Basic Rules-Based System in Kenken

In the Kenken puzzle, Certain logical rules are used for solving. They are Single square, naked pair, naked triple, evil twin, Hidden single, Killer combination, and X-wing. If a cage has a single cell, the single square rule is used. It means this cell has the same value as a target number. The naked pair-rule is used if there are two cells in a row or column with the same two possible values to fill the cells. This means that the other cells in this row or column respectively have no chance of having the same potential value as that of the two cells. Figure 2 illustrates the functioning of this naked pair rule. The green-labeled cells have exactly two possible values (1 or 7). These are a naked pair. Since 1 and 7 must occur in both cells, we can remove 1 and 7 of the red marked cells.

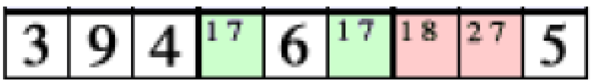

Figure 2: An example of how to detect Naked Pair Rule [4]

Naked triple works identical with naked pair, but the number of cells is three instead of two. It implies the number of possible values to fill the cells is also three. The evil twin is the easiest rule. If a cage consists of two cells and whenever one cell is solved, then the other cell is easy - it is simply the value is needed to make the workout out. For example, from Figure 3, once the cell in the bottom left corner is solved with the 4, then the square above it is the rule that

ISSN 1533-9211
DOI 10.5281/zenodo.7106602

can be generalized for larger areas. The last unsolved cell in an area is simply the value needed to make the sum work out.

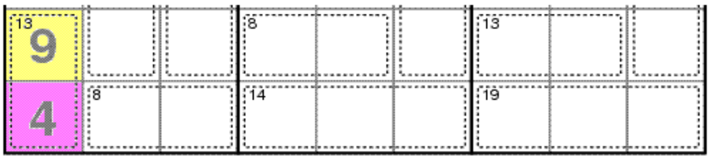

Figure 3: An example of Evil Twin Rule [4]

Hidden singles are a simple enough concept, but are often quite hard to spot. The definition of a hidden single is when a cell is the only one of its possible values an entire row, column. From figure 4, the possible values to solve the most left cell are 3, ,5, or 7. However, this row must have a “7” in it somewhere. It can be seen that only that cell has a “7” as one of its candidates. That is hidden single. That cell should be solved with “7”.

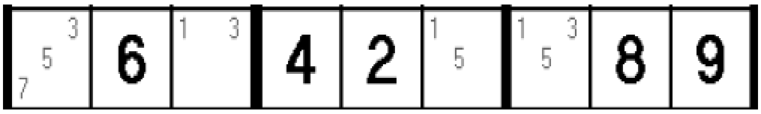

Figure 4: An example of Hidden Single Rule [4]

Killer combination is the most crucial rule. This rule is possibly applied if the cells covered in a cage are all in one row or column and the mathematical operation is in addition. There is a possible number of cells for this killer combination related to the cage size. For example, it means a possible number from 3 to 9 can be removed from those cells. If a cage with a target number is 23 then the only possible number for the cells are 6, 8, or 9. This killer combination for cage size 2 and any cage value is tabulated in Table 1. The table can be expanded to the other cage size. a clue of the target value is 3 in a cage of size 2 can only have 1 or 2 in it.

Table 1: Killer combination for cage size 2 [4]

Cage size	Cage value	Combination
2	3	1/2
2	4	1/3
2	17	8/9
2	16	7/9

X-wing rule stated that when there are only two possible cells for a value in every two different rows and these candidates lie also in the same columns, then all other candidates for this value in the columns can be eliminated. Figure 5 shows an example of this rule.

ISSN 1533-9211

DOI 10.5281/zenodo.7106602

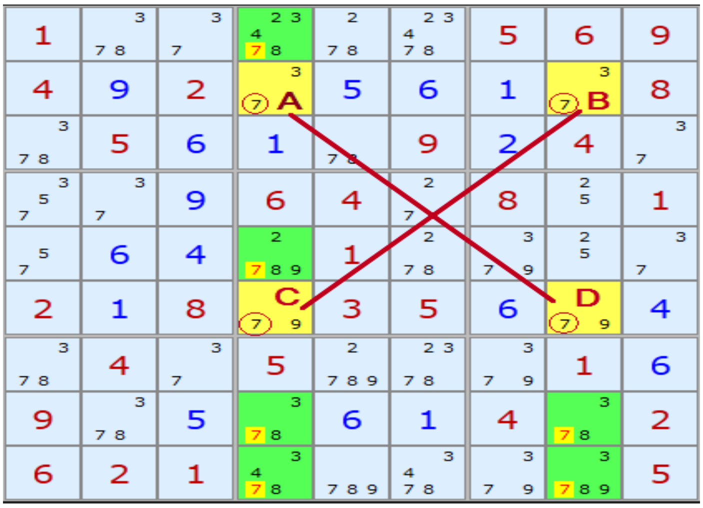

Figure 5: An example of X-Wing Rule [4]

If A turns out to be a **7** then it rules out a **7** at C as well as B. Because A and CD are 'locked' then D must be a **7**. So a 7 MUST be present at A, D or B, C. If this is the case, then any other **7**'s along the edge of our rectangle are redundant. We can remove the **7**'s marked in the green squares [4].

3. Black Widow Optimization (BWO) Algorithm

Fig. 6 shows the flowchart of BWO, the BWO algorithm each spider represents a potential solution starts with an initial population of spiders so, each spider represents the objective function Then, the new generation comes from reproducing initial spiders (i.e., in pairs). The female black widow eats male during or after mating Then she carries stored sperms in her sperm thecae and releases them into egg sacs [6, 7]. Mokhtar et al. (2021) said the new generation comes from the initial spiders. In this algorithm after mating the female black widow eats the male [8, 9].

Initial population

In BWO Algorithm, a black widow spider is the form of the solution. Every spider shows the values of the problem variables [10].

In a Near –dimensional optimization problem, a widow is an array of 1 × Nvar representing the solution of the problem. This array is defined as follows:

$$X_{N,d} = \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} \cdots\cdots x_{1,d} \\ x_{2,1} & x_{2,2} & x_{2,3} \cdots\cdots x_{2,d} \\ x_{N,1} & x_{N,2} & x_{N,3} \cdots\cdots x_{N,d} \end{bmatrix} \qquad (1)$$

$lb \leq X_i \leq ub$

Each of the variable values ($w1, w2, \ldots, w_{Nvar}$) is floating-point number. The fitness of widow is obtained by evaluation of fitness function f at a widow of (w1, w2, . . . , wNvar) So

ISSN 1533-9211
DOI 10.5281/zenodo.7106602

Fitness = f(widow) = $f(w_1, w_2, \ldots, w_{Nvar})$, (2)

To start the optimization algorithm, a candidate widow matrix of size Npop × Nvar is generated with an initial population of spiders. Then pairs of parents randomly are selected to perform the procreating step by mating, in which the male black widow is eaten by the female during or after that.

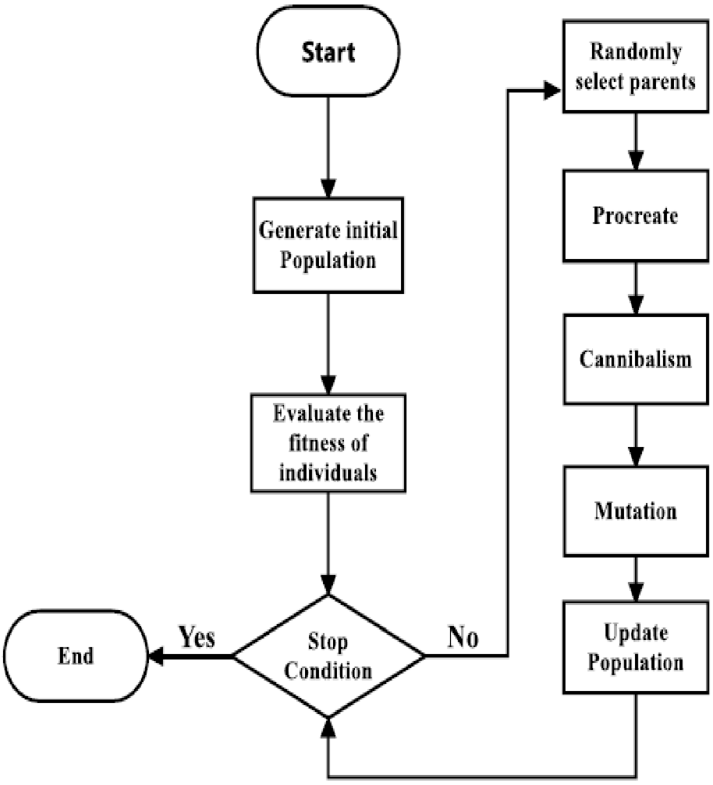

Fig 6: Flowchart of BWO algorithm [5]

Procreate

Parents are selected randomly in the procreation process to create a new generation an array known as Meu should also be generated to complete the further reproduction. Offspring y_1 and y_2 will be produced by taking α with the following equation in which w_1 and w_2 are parents [11-13].

$$A_1 = \beta \times w_1 + (1 - \beta) \times w_2 \quad (3)$$

$$A_2 = \beta \times w_2 + (1 - \beta) \times w_1 \quad (4)$$

Where A_1 and A_2 are the young spiders from reproduction and β is the random number between 0 and 1.

ISSN 1533-9211
DOI 10.5281/zenodo.7106602

Cannibalism

In sexual cannibalism the female black widow during or after mating eats his husband. The second one is called sibling cannibalism where the strong spiderlings eat their weaker siblings [14, 15].

Mutation

In mutation, a random of individually selected form population to try find a new solution. chosen solutions randomly maybe give a new area for solution. In mutation two-element exchange in the array randomly to get a new solution [16].

4. BWOA for Modelling and Solving Kenken Puzzle

Ken-ken puzzle of size n x n has n^2 cells. Cell located in a row, x and column, y is labeled as $C_{x,y=xn+y}$ and the value of that cell is $V(C_{x,y}) \in \{1,2,\ldots,n\}$ A i^{th} - cage labeled A_i is a set of cells, $A_i = \{C_{x,y}\}$ It correlates with one arithmetic operator $o_i \in \{+,-,\times,\div\}$ and one target number $H_i \in N$. This is the three rules in making the Ken-ken problem

$$|Ai| = 1 \rightarrow O_i = \phi,\ O_i \in \{-, \div\} \rightarrow |Ai| = 2 \text{ and } \forall C_{x,y} \rightarrow C_{x,y} \in \exists!\ Ai \qquad (5)$$

purpose of the puzzle is to find the value $V(C_{x,y})$ and meet the constraints:

$$|Ai| = 1 \wedge C_{x,y} \in A_i \rightarrow V(C_{x,y}) = H_i \qquad (5)$$

$$O_i \in \{-, \div\} \wedge A_i = \{C_{a,b}, C\ m, n\} \rightarrow \left|C_{x,y} C_{m,n}\right| = Hi \qquad (6)$$

$$O_i \in \{\div\} \wedge A_i = \{C_{a,b}, C\ m, n\} \rightarrow V(C_{x,y}) / V(C_{m,n}) = Hi \qquad (7)$$

$$O_i \in \{+\} \rightarrow \sum_{C_{x,y}} V(C_{x,y}) = Hi \qquad (8)$$

$$O_i \in \{\times\} \rightarrow \prod_{C_{x,y})} V(C_{x,y}) = Hi \qquad (9)$$

Rule-based search is started by assuming all unknown values of cells by all possible values to fill that cell without breaking the constraints, $V(C_{x,y}) = 1, 2, \ldots, n$ after the value of one cell has been determined, the possible values of a certa respectively were updated. For example, using the naked single rule is stated as equation 1, will cause all possible values for all cells in a row and a column is updated stated in equations 3 and 4. Applying naked pair is stated in equation 15 for row and 16 for the column. Other rules are also defined.

$$\left|V(C_{x,y})\right| = 1 \wedge x \in P(C_{x,y}) \rightarrow V(C_{x,y}) = x \qquad (10)$$

$$(V(C_{x,y}) = x) \wedge (\forall a \in \{1, 2, \ldots, n\}) \rightarrow P(C_{x,y}) = P(C_{x,y}) - \{x\} \qquad (11)$$

$$(V(C_{x,y}) = x) \wedge (\forall n \in \{1, 2, \ldots, n\}) \rightarrow P(C_{x,y}) = P(C_{x,y}) - \{x\} \qquad (12)$$

$$\left|V(C_{x,y1})\right| = \left|V(C_{x,y1})\right| = 2 \wedge x \in P(C_{x,y1}) = P(C_{x,y2}) \rightarrow P(C_{x,n}) = P(C_{x,n}) - P(C_{x,y1}) \qquad (13)$$

$$\left|V(C_{x1,y})\right| = \left|V(C_{x,y1})\right| = 2 \wedge x \in P(C_{x1,y}) = P(C_{x1,y}) \rightarrow P(C_{x,n}) = P(C_{x,n}) - P(C_{x1,y}) \qquad (14)$$

The Seybold REPORT
ISSN 1533-9211
DOI 10.5281/zenodo.7106602

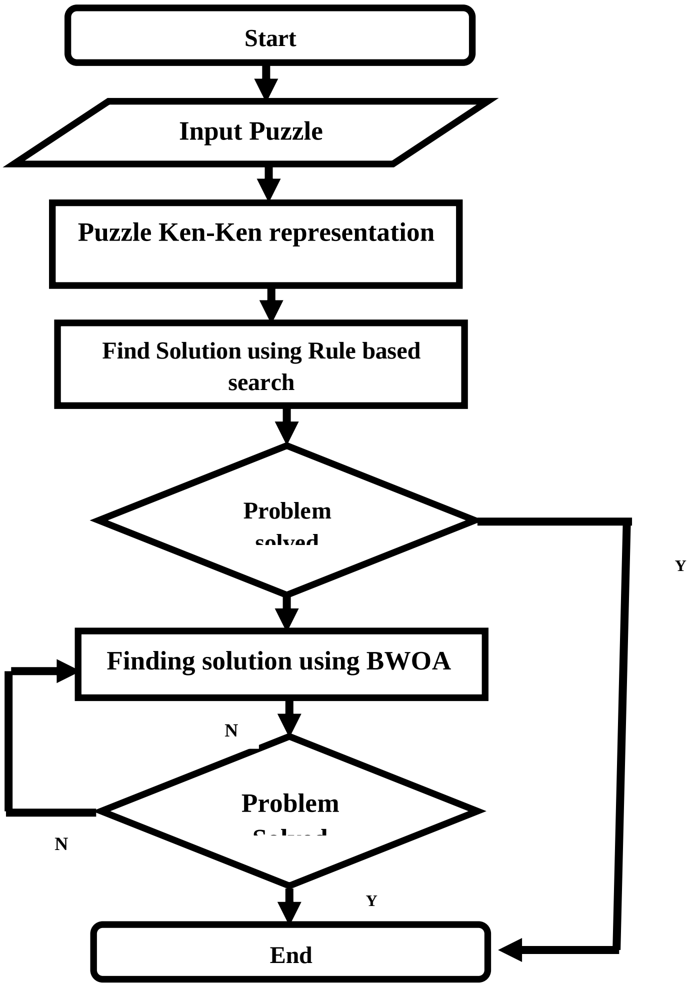

Figure 7: Flow chart of Solving Kenken Puzzle

BWOA is used while the puzzle still is not able to be solved after exercising all the rules repetitively. It started by encoding the chromosome. One chromosome consists of y number of segments; m ≤ n one segmenting cell is a set of genes that of unsolved cells in that segment.

ISSN 1533-9211

DOI 10.5281/zenodo.7106602

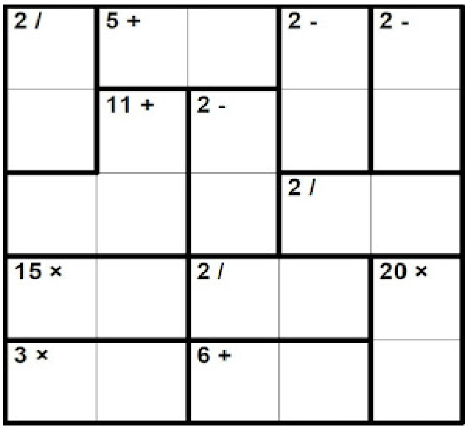

Figure 8: An example of Kenken puzzle 5x5

A Segment represents a row or a column. Segments are ordered in spiders. An example of one spiders from Kenken puzzle in Figure 7 is 33 32 | 27 28 22 | 0 3 5 2 1 | 11 5 9 7 9 | 12 14 15 16 13.

The objective function will be calculated after the generation value of the gene on the chromosome has been done. Value for widow j^{th} on a widow was represented by $y_{i,}$ y_j is set 0 i^{th} f widow is not solved, otherwise is set 1. For spiders with several widows k, the fitness function is shown by the equation below. Consequently, the solution of the puzzle is to find out the widow with the fitness value is 1.

$$X_j = \begin{cases} 0 & yj = 0 \\ 1 & yj \neq 0 \end{cases} \tag{15}$$

$$\text{Fitness} = f(\text{widow}) = f\,(w_1, w_2, \ldots, x_{Nvar}), \tag{16}$$

Step 1: Initialization

Each individual represents a row-matrix 1×n where n is the number of observations, each widow contains an integer [1, K].

Step 2: Procreate

According to equations 3 and 4, we get offspring using parents to procreate the young spiders from the reproduction process. The reproduction process is carried out for d/2 times [17].

Before: 33 32 | 27 28 22 | 0 3 5 2 1 | 11 5 9 7 9 | 12 14 15 16 13.

After: 33 22 | 27 4 22 | 9 6 5 28 1 | 11 2 9 9 7 | 12 3 15 6 1.

Step 3: Cannibalism

Sibling cannibalism is when the strong spiderlings eat their weaker siblings. In this algorithm, we set a cannibalism rating (CR) according to which the number of survivors is determined [18].

ISSN 1533-9211
DOI 10.5281/zenodo.7106602

Step 4: Mutation

For each individual, the mutation operator is implemented as follows, first select two columns randomly from i_{th} individual and then generate two new columns [19]. Exchange mutation is used to get another possible solution. The mutation is done between widow in the same segment. The spider below is an example of a mutation process between two widow on the same segment.

Before: 33 32 | **27** 28 **22** | 0 3 5 2 1 | **11** 5 9 7 **9** | 12 14 15 16 13.

After: 33 32 | **22** 28 **27** | 0 3 5 2 1 | **9** 5 9 7 **11** | 12 14 15 16 13.

5. Experiment Results

Two types of experiments were conducted. They are to know how fast the system can solve the puzzle according to the size and the level of the difficulty of the puzzle. Size variations of Kenken puzzle experimented 5 × 5, 7 × 7, and 9 × 9 with a variety of levels of difficulty easy, medium, hard, and expert. For each category, 10 sample puzzles are used. Some of the puzzles can be solved only by rule-based whereas the other should be with BWOA. Table 2 showed the percentage of puzzles that can be solved only by using a rule-based search.

Table 2: Percentage of problems that can be solved using rule base search

	Results of GA [4]				**Results of Our BWOA**			
Size	Easy	Medium	Hard	Expert	Easy	Medium	Hard	Expert
5 × 5	100%	80%	50%	0%	100%	92%	68%	0%
7 × 7	100%	60%	0%	0%	100%	79%	44%	0%
9 × 9	100%	90%	0%	0%	100%	94%	53%	0%

The average time (in milliseconds) of 10 attempts for a certain puzzle per unit cell can be seen below the figure.

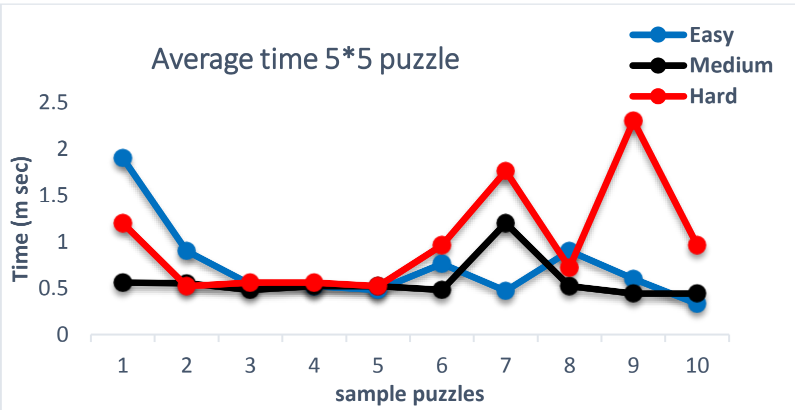

Figure 9: Average time to each cell in millisecond (puzzle 5x5)

The Seybold REPORT

ISSN 1533-9211

DOI 10.5281/zenodo.7106602

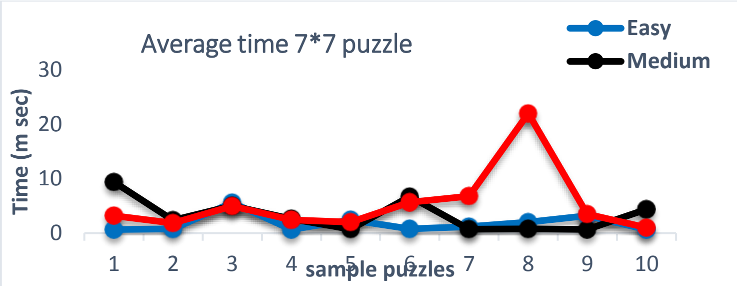

Figure 10. Average time to each cell in millisecond (puzzle 7x7)

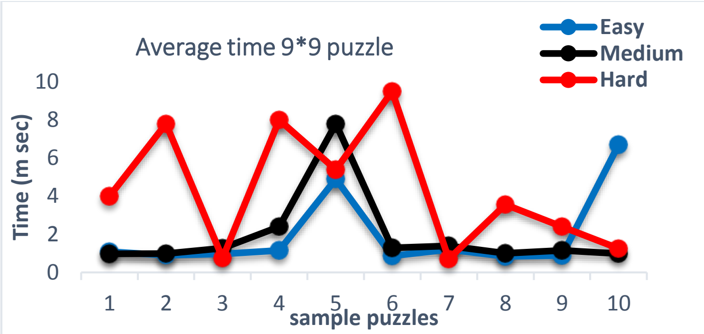

Figure 11. Average time to each cell in millisecond (puzzle 9x9)

Analyzing the data with F-test, proved that completion time was independent of puzzle size for any level of difficulty. F calculation for level difficulty easy, medium, hard and expert is 3.1, 0.65, 0.65, and 0.84 whereas F table is 4.68. It means that for any type of level of difficulty, the average time to solve each cell is independent of the size of the puzzle.

6. Conclusion

This paper proposes a black widow optimization algorithm for solving the Kenken puzzle. The proposed black widow optimization algorithm has higher performance in finding the solution of the Kenken puzzle. Obtained results show the efficiency that the proposed algorithm in grids range in size 5×5, 7×7, and 9×9 with a variation of the level of difficulty easy, medium, and hard are more efficient than the old ones. Kenken puzzle level of difficulty expert cannot be done only by using single square and X-wing rules. It needs to apply other sophisticated logic rules.

REFERENCES

1) The KenKen Website. http://www.kenkenpuzzle.com/Seen 7/1/2022.

2) J. Watkins, Triangular Numbers, Gaussian Integers, and KenKen. The College Mathematics Journal, (43), 37-42, 2012. http://dx.doi.org/10.4169/college.math.j.43.1.037.

3) H. Reiter, J. Thornton, G.P. Vennebush, Using KenKen to Build Reasoning Skills. Mathematics Teacher, 107, 341-347, 2013.

ISSN 1533-9211
DOI 10.5281/zenodo.7106602

4) O. Johanna, S. Lukas, K. V. I. Saputra, Solving and modeling Ken-ken Puzzle by Using Hybrid Genetic Algorithm, The First International Conference on Engineering and Technology Development (ICETD), 2012.

5) V. Hayyolalam, A. A. P. Kazem, Black widow optimization algorithm: a novel meta-heuristic approach for solving engineering optimization problems. Engineering Applications of Artificial Intelligence, 87, 103249, 2020.

6) M. Chlond, Classroom Exercises in IP Modeling: Sudoku and the Log Pile, INFORMS Transactions on Education, 5, 77-79, 2005. http://dx.doi.org/10.1287/ited.5.2.77.

7) A.C. Bartlett, T.P. Chartier, A.N. Langville, T.D Rankin, An Integer Programming Model for the Sudoku Problem. The Journal of Online Mathematics and Its Applications, 8 2008. http://www.maa.org/external_archive/joma/Volume8/Bartlett/index.html.

8) E. Mostafa, A. Mokhtar, H. Zaher, N. Ragaa, A Black Widow Optimization Algorithm Using Chaos Reproduction Operator, Design Engineering, 4751-4763, 2021. https://thedesignengineering.com/index.php/DE/article/view/5426.

9) Gaskett, A.C., Spider sex pheromones: Emission, reception, structures, and functions. Biol. Rev, 2007.

10) B. Barbara, G. Wallace, Darwin, and the theory of natural selection. J. Hist. Biol. 1 (2), 261–323, 1968.

11) E. Eid, H. M. Zawbaa, and A. E. Hassanien, Binary grey wolf optimization approaches for feature selection. Neurocomputing 172, 371-381, 2016.

12) Birkhead, Tim R., Muller, Anders Pape, Sperm Competition and Sexual Selection. Academic Press, 1998.

13) Garb, Jessica E., Gonzlez, Alda, Gillespie, Rosemary G. The black widow spider genus latrodectus (araneae: theridiidae): Phylogeny, biogeography, and invasion history. Mol. Phylogenetic Evol. 31 (3), 1127–1142, Retrieved, 2004, http: //www.sciencedirect.com/science/article/pii/S1055790303003968.

14) K. Premkumar, M. Vishnupriya, T. S., Babu, B. V. Manikandan and A. Z. Kouzani, Black Widow Optimization-Based Optimal PI-Controlled Wind Turbine Emulator, Sustainability, 12, 2020. doi:10.3390/su122410357.

15) Berning, W Aric. et al, Sexual cannibalism is associated with female behavioral type, hunger state and increased hatching success. Anim. Behav. 84 (3), 2012, 715-721, http://www.sciencedirect.com/science/article/pii/67667/.

16) Gage, Matthew J.G., Evolution: Sex and cannibalism in redback spiders. Curr. Biol. 15 (16), R630–R632, Retrieved, 2005. http://www.sciencedirect.com/science/ article/pii/S0960982205008626.

17) Fox, L.R., Cannibalism in natural populations. Annu. Rev. Ecol. Syst. 6 (1), 87-106. http://dx.doi.org/10.1146/annurev.es.06.110175.000511, 1975.

18) Forster, L.M. The behavioural ecology of latrodectus hasselti (thorell), the australian redback spider (araneae: theridiidae): A review. Rec. West. Aust. Mus. Suppl. 52, 13-24, 1995.

19) H. Zaher, A. Mokhtar, N. Ragaa, E. Mostafa, A Black Widow Optimization Algorithm for Clustering Problems, Journal of University of Shanghai for Science and Technology, pp. 82-93, 2021. https://jusst.org/a-black-widow-optimization-algorithm-for-clustering-problems/.