

Probabilistic Modeling of Lossless Compression using Improved RLC Algorithm

Aisha Sattar Phulpoto
MS Scholar

The software engineering
department of MUET Jamshoro,
Pakistan

Sania Bhatti

Associate Professor
The software engineering
department of MUET Jamshoro,
Pakistan

Salahuddin Saddar

Assistant Professor
The software engineering
department of MUET Jamshoro,
Pakistan

ABSTRACT

Probabilistic model checking is a very famous formal verification technique to calculate accuracy evaluation while it is very complex and time-consuming to achieve via simulations these days. Probabilistic analysis has been applied successfully for a diverse range of application domains to examine the real-time attribute likewise, Performance and reliability, power management, biological process modeling, Communication, network, and multimedia protocols, security protocols, game theory, and many others but not applied in the domain of image compression techniques. The outstanding applicability of PRISM in a wider range of application domains motivated us to the analysis of compression algorithms using probabilistic techniques which are challenging and are still an open issue. For this, this paper has selected improved run-length coding IRLC, which is a lossless algorithm. This paper presents a label transition diagram for IRLC to understand the working of IRLC in terms of states and transitions and then developed the probabilistic discrete-time Markov chains (DTMC) of IRLC to perform its analysis by applying properties.

Keywords

Improved run length coding IRLC, discrete-time Markov chains (DTMC), Markov decision processes (MDP), continuous-time Markov chains (CTMC), PRISM, probabilistic modeling

1. INTRODUCTION

The research in the field of image compression is not a new one. The area of advanced image compression falls out into two different domains which are spatial and frequency. Application of compression on frequency domain will perform finest compression quantity as compared to counterparts of its spatial domain. [1]. Compression can be seen in a variety of network applications. All of the photos obtained from the web are compressed using the JPEG or GIF formats in general. However, when files are stored, some of them are uncompressed. Advanced image compression technology provides a way to store images with the least amount of memory. Image compression is the process to minimize the size of a graphics file without

Compromising the image's quality. By reducing the size of the file, it will allow more images to be stored in given memory space. It will also reduce the time that is required for an image to be downloaded or sent [2]. Image compression algorithms fall into two different categories like Lossy and lossless compression. Lossy compression techniques degrade the image quality due to the loss of some important image information such as Lossless compression Techniques will give the best quality image with the best compression result.

Although, researchers are effectively employing a range of probabilistic and hybrid model verification techniques. [3], such as PRISM, UPPAAL, Anquiro, Real-Time Maude, CaVI, APMC, AVISPA, and Slide. However, since PRISM's widespread use in several fields, this research focuses on probabilistic modeling of an image compression algorithm via the PRISM tool. Section II discusses PRISM Tool, Section III Presents a literature review section IV works on the IRLC algorithm. Section V presents IRLC construction via prism and Section VI details concluding observation.

2. PRISM MODELING TOOL

PRISM is an open-source tool that is used for probabilistic model checking and can run on the utmost of OSs. There is a website of PRISM [4] through which it can be easily downloaded and also provides many tutorials for guidance. It supports discrete-time Markov chains (DTMC), Markov decision processes (MDP), continuous-time Markov chains (CTMC) and probabilistic timed automata. The module sets represent the framework's components establish a PRISM model and every module includes moves and states. The states are representing the attainable setup of the system and transitions that change the states concerning time. PRISM model checking is done by using some properties in PCTL, LTL, and CSL. Costs and rewards are applied to the attributes to provide additional quantitative metrics. The two most important parts of system specification in the prism tool are: "Model" and "Properties". Prism works on several types of models and provides results of those models in either graph form or a log structure.

In PRISM, for storing effective data structure multi-terminal binary decision diagram (MTBDD) is used to represent vectors and matrixes. The main reason for using MTBDD is to minimize the storage space in a prism. This tool support three different computation engines such as symbolic engine, sparse matrices, arrays, and hybrid engine

3. LITERATURE REVIEW

This section first presents selected studies that employed PRISM during probabilistic experiments. Then lossless image compression algorithms are presented. The performance of a wide area of systems, inclusive of gossip protocols, team formation protocols, power-efficient algorithms, and FTTC has been effectively evaluated using probabilistic model checking via PRISM. This section is showing 4 diverse protocols from differing spaces analyzed and confirmed by PRISM. The main reason for this section is to illustrate that which protocols have been probabilistically analyzed in several application spaces.

3.1.1 Gossip protocol[5] is a group of communication protocols that is analyzed by using the prism tool. Each system node sends information to a large number of peer

nodes regularly. The performance of gossip protocols is both probabilistic and nondeterministic. Nondeterministic in case of distributed network activities for a discrete node which arises asynchronously and probabilistic because of a random selection of nodes and then exchanges information to those nodes. The gossip protocol is modeled using Markov decision processes (MDP), and the protocol's best and worst-case performance is studied.

3.1.2 The performance of the team formation protocol[6] is analyzed and verified by the prism tool. Five agents have been constructed in both DTMC and MDP models, with 4 distinct network topologies: completely linked, network, ring, and a star with one isolated agent. Furthermore, the stochastic two-player game (STPG) is used to validate the characteristics of multi-agent systems on various network topologies with cooperative and hostile agents. The organization's predicted performance and the organization's optimal resource distribution among agents are calculated.

3.1.3 Power Efficient Algorithm for Data Gathering (PEADG) [7] is proposed and evaluated via PRISM. PEADG's goal is to extend the lifespan of data-gathering wireless sensor networks. To prevent the problem of state space explosion, Probabilistic Automata (PAs) are first built and mapped to the discrete-time Markov chains DTMC module. The original technique uses several modules, but the improved approach uses a single module that creates a module with just reachable state space, not potential state space. As a result, the advanced technology can cope with 226 sensor nodes instead of the previous approach's 26 sensor nodes.

3.1.4 Fault-tolerant target tracking protocol[3] is modeled via prism with the enhanced version of the continuous-time Markov chains CTMC model in the wireless sensor network. To help symmetry Reduction during modeling, PRISM is selected for probabilistic modeling of the FTTT protocol. The number of sensor nodes joining the cluster, estimated message count, count of sensor nodes overlapping, tracking probability, percentage of fault tolerance, and the energy consumption is all evaluated using the FTTT protocol. In PRISM (GRIP) models, PRISM-symm and generic representations are used. Experiments with modeling confirmed that PRISM-symm produces better results than PRISM and GRIP.

3.1.5 Software Adaptation for an Unmanned Undersea Vehicle [8]: The work in this paper is done on Probabilistic Representation of Intent Commitments to Ensure Software Survival (PRINCESS). They develop PRINCESS to enable software that adopts both hardware changes and changing environments. The model they construct is DTMC Model whose state includes both the UUV's current mission position and the battery level. They successfully demonstrated adaptation to updated and degraded sensors, system failures, changes in the environment, and new architecture in PRINCESS.

3.1.6 A Heuristics-Based Incremental Probabilistic Model Checking at Runtime [9]: In this paper heuristics-based incremental probabilistic model checking at runtime was done. This paper's main contribution is to explore the concepts of existing real-time probabilistic model checking and to outline the limits of different verification methods, before applying the concept of incremental verification to the probabilistic model checking process. It emphasizes the DTMC model and reachability, which incorporates the Crowds and BRP models. The results of the research show that the method proposed in this research

can cut the time it takes to verify a system using normal runtime probabilistic model checking by more than 40%.

3.2 PERFORMANCE OF IMAGE COMPRESSION ALGORITHM:

Compression approaches that apply the encoding procedure, such as spatial domain methods and transform domain techniques, can be effective in images compression. The encoding procedures are identical to those used with grayscale pictures. Compression-based studies with different performance merit analyzed the performance of various compression techniques with the best compression ratio.

3.2.1 Efficient Run-length Coding (E-RLC) [10] is a lossless image compression technique used with different grayscale images. In this technique size of the image is divided into the different nub of block size and applied the efficient way Of RLC to get the high quality of reconstructed images. Other lossless image compression algorithms do not perform. The proposed E-RLC achieves high-quality encoding results in the Lena and Cameraman images, with compression ratios ranging from 1.79 to 160.

3.2.2 Hybrid DCT – DWT Techniques with Huffman Encoding[11]: This study offers a DCT, DWT, and Huffman method hybrid compression for image transmission over a ZigBee network. The input image is compared to BMP, JPG, and PNG, which are three distinct types of file formats. For each format, the DCT and DWT compression techniques are used, and the result shows that the compressed image's size may be reduced by 96.92 % and 99.8%, respectively, from its original image, resulting in a high-quality image. In comparison to previous research, the data obtained indicate a higher PSNR value As a result, a hybrid compression approach is presented to achieve higher input image compression while reducing image data storage space.

3.2.3 Modified SVD (MSVD) [12]: The research includes image compression using the SVD (Singular Value Decomposition) technique is applied on different images with a different type of format and resolution. A unique technique for preprocessing the SVD is provided, which decreases the computing cost when compared to the SVD (in terms of time or clock cycles). The suggested approach, called Modified SVD (MSVD), is compared to the standard SVD and other SVD variations depending on various factors.

3.2.4 An improved lossless image compression algorithm based on Huffman coding [13]: This paper presents an enhanced lossless image compression algorithm that combines linear prediction, integer wavelet transform (IWT) with output coefficients processing, and Huffman coding. The key contribution of this paper is a new hybrid transform that uses a new prediction template and IWT coefficient processing. This technique improves state-of-the-art algorithms in three different image sets, according to the experimental results. Compression ratios have increased by at least 6.22 percent, reaching 72.36 percent. This technique is better suited to compressing images with complex textures and greater resolutions at a reasonable pace.

3.2.5 Improve run-length coding (IRL-C) [14] the suggested I-RLC model begins by obtaining the original image and converting it to a matrix format. The large-size matrix is subdivided into many non-overlapping small-size block matrices. To achieve the compression ratio, smaller size blocks are used. This technique is repeated for the whole image block size. The suggested I-RLC algorithm is tested on a variety of image formats to see which technique produces

the best results and quality. I-RLC method is applied to the sub-block starting from 256x256 while reducing the block size up to 4 x 4. MATLAB Version 2013a is used to evaluate and implement the suggested I-RLC algorithm using various parameters such as MSE, SNR, and PSNR.

4. WORKING OF (IRL-C) ALGORITHM:

The proposed technique disposes an input image and divides that image into a small size of the non-overlapping number of sub-blocks. I-RLC algorithm makes the sequential number of the first block from the matrix of the sub-block. By removing a simple form of spatial redundancy the compression can be achieved. The (IRLC) technique uses the method of mapped pair which has two pairs, first for gray level and second for the run-length that shows how many of them are on the run [14].

Take a look at the sub-image matrix below as an example. The grey scale value sub-image is displayed in a matrix format as seen below. It might or might not be repeating symbols in the sub-image of greyscale value. If the repeating symbols are repeated in sequential order, the compression ratio will be satisfactory; otherwise, it will be the worst.

$$\text{Sub image} = \begin{bmatrix} 2 & 1 & 2 & 2 & 2 & 2 \\ 2 & 1 & 3 & 3 & 3 & 3 \\ 2 & 2 & 4 & 4 & 4 & 5 \\ 2 & 2 & 2 & 2 & 2 & 4 \\ 3 & 3 & 1 & 1 & 1 & 1 \end{bmatrix}$$

As every grayscale value has 8-bit integers, the sub-image has 30 elements and takes up 30 bytes as the original block size. From the beginning to the finish, the above block matrix is rearranged into a row-wise sequential number:

2 1 2 2 2 2 1 3 3 3 3 2 2 4 4 4 5 2 2 2 2 2 4 3 3 2 2 2

By removing a simple form of spatial redundancy, compression is achieved. The following mapped to pair is used in enhanced run-length coding (IRLC).

Mapped pair = (c1, l1), (c2, l2), . . . (Cn, ln)
(Gray level, run length)

The mapped pair is encircled by producing pairs of values, the first of which represents the grey level and the second of which represents the number of them on the run. Where ci is the pixel intensity and li is the number of pixels in a row with the same intensity.

(2,1), (1,1), (2,5), (1,1), (3,4), (2,2), (4,3), (5,1), (2,5),
(4, 1), (3, 2), (2, 4).

Rearrange the obtained values in sequential order are follows.

2 1 1 1 2 5 1 1 3 4 2 2 4 3 5 1 2 5 4 1 3 2 2 4.

Now the total elements are 24 so the compression ratio is obtained as 1.25 from [30/24]. The sub-block size ranges from 4 to 256, with 4 being the smallest and 256 being the largest.



Fig 1: block diagram of IRL-C

5. IRL-C MODEL CONSTRUCTION AND ANALYSIS VIA PRISM

Sharpness is valued more than tone reproduction when it comes to image quality. Image quality is a subjective term. Measuring can be done directly or indirectly. To determine picture quality, the suggested work was implemented for various block sizes and the compression ratio of the image was assessed and size of the model, building time of model, model checking time, and storage of model of IRL-C algorithm. Then developed the probabilistic model (DTMC)

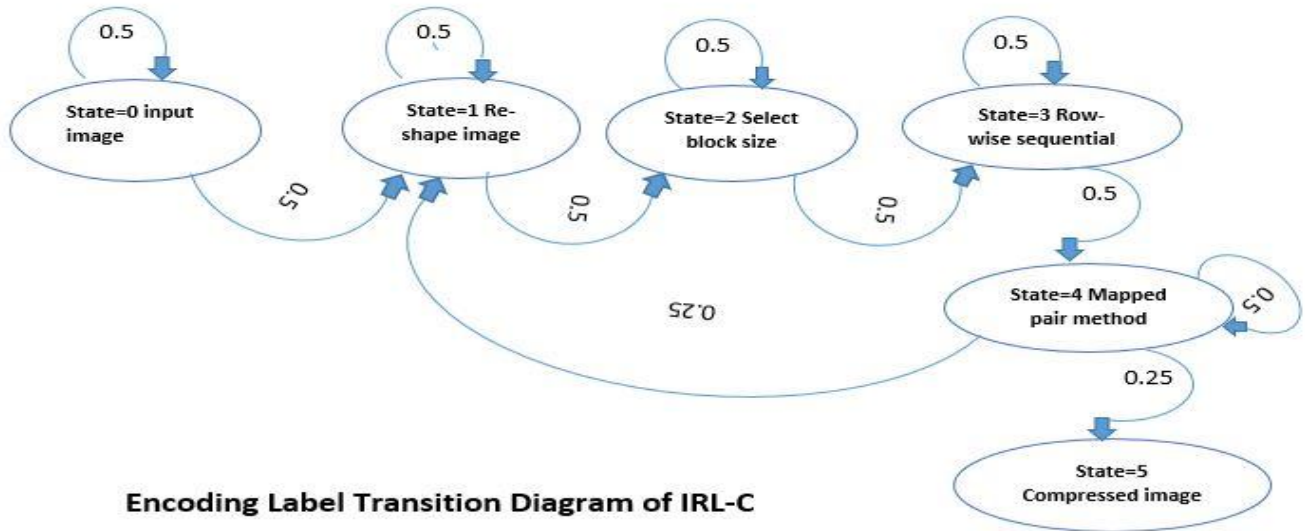
of IRLC (Encoding) applied to images. The IRLC of image compression involves several steps constructed via DTMC a non-deterministic model. Some properties and reward structures are used for the analysis of the model structure and behavior [15].

5.1 Label transition diagram (LTS):

is a triplet Form of states actions and transition. Firstly, this paper presents a label transition diagram of the IRL-C algorithm shown in fig.2. LTS of IRL-C involves six states with a probability of 0.5 to move next state. At state 0 model get an

original image then move to next state 1 which re-shapes the original with 256 and then at state 2 select a block size from 4 by 4 up to 256. Then state 3 gets the first block elements and makes the row-wise sequential so that the mapped pair method

can be applied at state 4. From the state 4 models again move to state 1 this process is repeated until the last block. Then at state 5, get a compressed image.



Encoding Label Transition Diagram of IRL-C

Fig 2: Label Transition Diagram of IRL-C Algorithm

Table 1. image1 of size 260728.

Block size	Compressed image size	Compression ratio
256*256	162558	1.60
128*128	72458	3.59
64*64	36432	7.15
32*32	3245	80.34

Table 1 presents the reconstructed grayscale image of size 260728. The grayscale image with the best compression rate was evaluated with block sizes ranging from 32 to 256. In the image block size of 256, the higher compressed process of the image can be computed as 1.60.

Fig 3 shows the x is the original image of size 260728 which is reconstructed with different block sizes. Y4 is a compressed image of size 162558 with block size 256 which gives a compression ratio of 1.60, y3 is a compressed image of size 72458 with block size 128 which gives a compression ratio of 3.95, y2 is a compressed image of size 36432 with block size 64 which gives a compression ratio of 7.15 and y4 is a compressed image of size 3245 with block size 32 which gives a compression ratio 80.34.

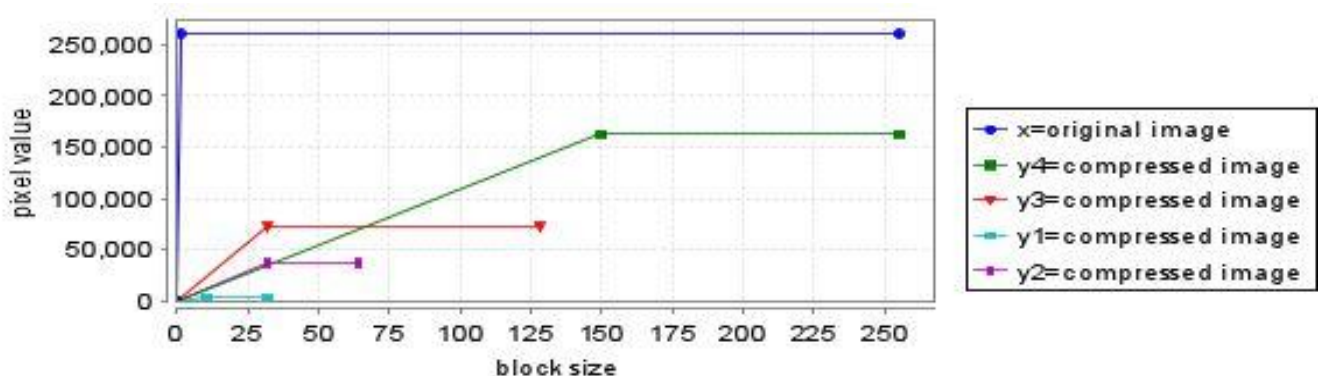


Fig 3 Graph of Image1

In this research, image2 of the original image of 128*128. Pixel values the total size of the image is 16384 and the block size of the model starts with 8 is taken. After constructing the model obtained compressed image size 8192 as shown in

Fig.4.x-axis shows time and y shows the pixel value of the original image size 16384 and compressed image size 8192. The time for model construction is 8.036 seconds and storage of the model is TOTAL: [28.0 KB].

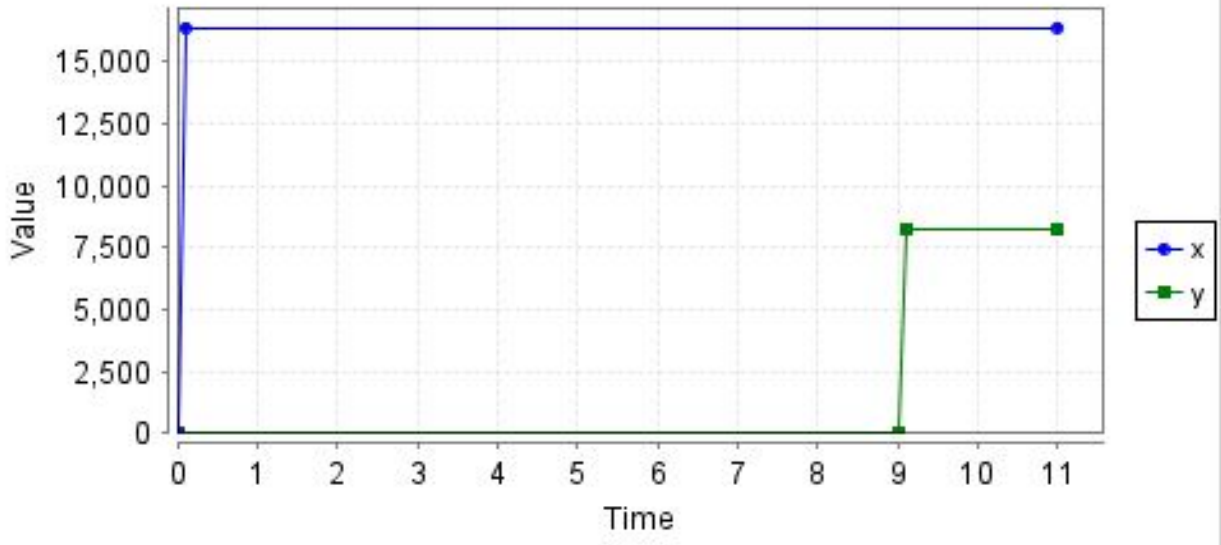


Fig 4:Graph of Image2 Original Image over-Compressed Image

5.2 Pmax=? [F (S=5 & Y=compressed image)]

The model takes 0.06 seconds to execute with 112 (1 initial) state, 208 transitions, and completes 11 iterations. This property shows us that the probability of getting compressed

Image y at the final state gives the maximum probability of 1.0. The result of this property is shown in figure 5 where the y-axis is showing a compressed image pixel value concerning time in the x-axis.

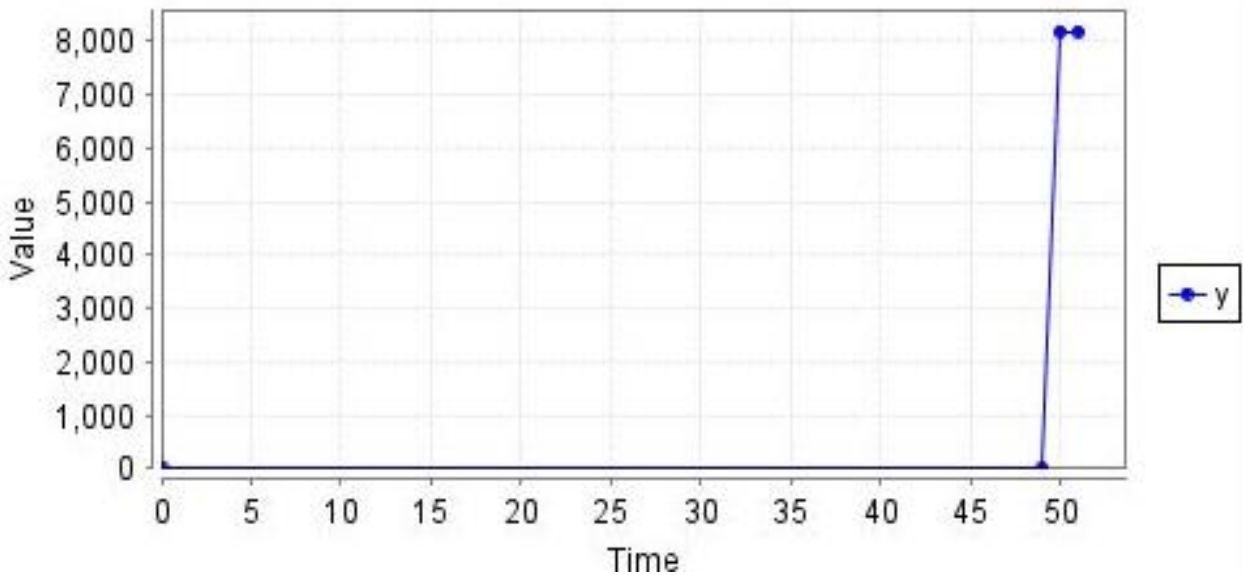


Fig.5 probability of y-compressed image (image2)

5.3 R {"compression_ratio"}=? [C<=1]

A reward structure called "compression_ratio" is added to the model to get the compression ratio of the original imageOver-compressedimage. The better compression rate for an

Image1 with a block size of 256x256 is 1.60 shown in fig 6. And the expected compression_ratio of image1 is 2.0 shown in fig.7 where the x-axis shows the block size and the y-axis shows the compression rate.

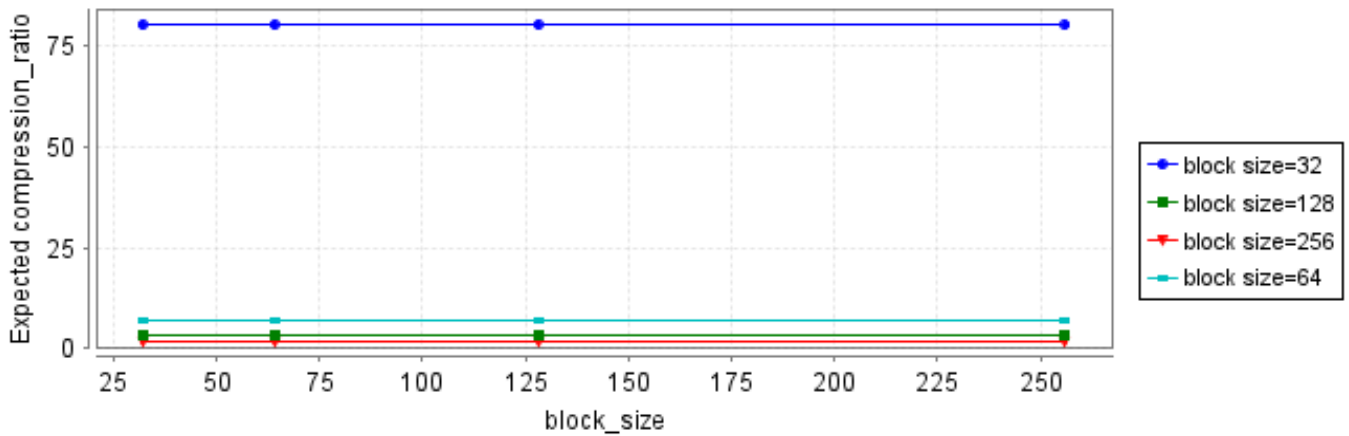


Fig 6 graph for a compression ratio of imagel

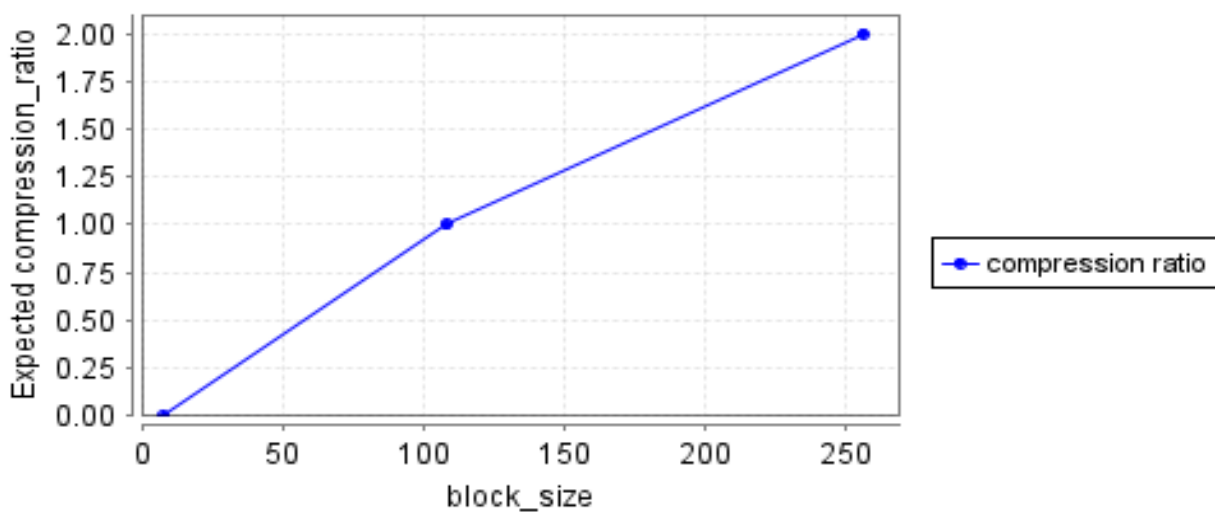


Fig 7: Compression Ratio of Imag2

6. CONCLUSION & FUTUREWORK

The rise of probabilistic model checking in several fields, as well as the adaptability of PRISM in a wide range of applications, prompted us to develop a PRISM model of an image and analyze the image compression algorithm with some parameters via prism modeling tool. This paper presents some case studies of different protocols with their implementation in PRISM and also presents some case studies of image compression algorithms with different tools. We probabilistically analyzed the performance of IRLC in terms of compression ratio and also measured the size of the model, building time of model, model checking time, and storage of the model. The suggested I-RLC is simple to implement and implement quickly. The presented experimental results outperform all other strategies in terms of compression ratio. Grayscale images are well-suited to this image compression technology. Different Lossy compression transformations can be applied to improve this approach. Various transformations of grayscale and color images can be used in future work.

7. REFERENCES

[1] Sayeed, Khalid. Introduction to data compression. Morgan Kaufmann, 2017
 [2] Hussein, A. H., S. Sh Mahmud, and R. J. Mohammed. "Image compression using proposed enhanced run-length

encoding algorithm." *Ibn AL-Haitham Journal for Pure and Applied Science* 24, no. 1 (2017).

[3] Bhatti, Sania, MohsinMemon, and SheerazMemon. "Evaluating FTTT Protocol via PRISM, PRISM-symm, and GRIP." *International Journal of Computer Theory and Engineering* 9, no. 3 (2017): 162.
 [4] The PRISM website. [Online]. Available <https://www.prismmodelchecker.org/>
 [5] Kwiatkowska, Marta, Gethin Norman, and David Parker. "Analysis of a gossip protocol in PRISM." *ACM SIGMETRICS Performance Evaluation Review* 36, no. 3 (2008): 17-22.
 [6] Chen, Taolue, Marta Kwiatkowska, David Parker, and AistisSimaitis. "Verifying team formation protocols with probabilistic model checking." In *International Workshop on Computational Logic in Multi-Agent Systems*, pp. 190-207. Springer, Berlin, Heidelberg, 2011.
 [7] He, Kang, Hongli Yang, Yachao Feng, Yuan Liu, and ZongyanQiu. "Performance analysis of data gathering protocol using PRISM." In *2012 IEEE 17th International*.
 [8] Pfeffer, Avi, Curt Wu, Gerald Fry, Kenny Lu, Steve Marotta, Mike Reposa, Yuan Shi et al. "Software

- Adaptation for an Unmanned Undersea Vehicle." *IEEE Software* 36, no. 2 (2019): 91-96.
- [9] Liu, Yang, and Chao He. "A Heuristics-Based Incremental Probabilistic Model Checking at Runtime." In *2020 IEEE 11th International Conference on Software Engineering and Service Science (ICSESS)*, pp. 355-358. IEEE, 2020.
- [10] Babu, S. Anantha, and Eswaran Perumal. "Efficient Approach of Run Length Coding Technique Using Lossless Grayscale Image Compression (E-RLC)." In *2018 3rd International Conference on Inventive Computation Technologies (ICICT)*, pp. 680-686. IEEE, 2018.
- [11] Hussin, Mumtaz Anwar, Farhana Ahmad Poad, and Ariffuddin Joret. "A Comparative Study on the Performance of DWT and Huffman Compression Technique on a 2D Signal." *Journal of Electronic Voltage and Application* 2, no. 1 (2021): 11-19.
- [12] Prasantha, Dr. HS. "NOVEL APPROACH FOR IMAGE COMPRESSION USING MODIFIED SVD." *International Journal of Creative Research Thoughts (IJCRT)* 8, no. 8 (2020):2234-2243.
- [13] Liu, Xiaoxiao, Ping an, Yilei Chen, and Xinpeng Huang. "An improved lossless image compression algorithm based on Huffman coding." *Multimedia Tools and Applications* (2021): 1-15.
- [14] Babu, S. Anantha, P. Eswaran, and C. Senthil Kumar. "Lossless compression algorithm using improved RLC for the grayscale image." *Arabian Journal for Science and Engineering* 41, no. 8 (2016): 3061-3070.
- [15] Jose, Jais, Neha Gautam, Mohit Tiwari, Tripti Tiwari, Arjun Suresh, VinuSundararaj, and M. R. Rejeesh. "An image quality enhancement scheme employing adolescent identity search algorithm in the NSST domain for multimodal medical image fusion." *Biomedical Signal Processing and Control* 66 (2021):10248.