Proc. International Conference on Space Optical Systems and Applications (ICSOS) 2012, 10-4, Ajaccio, Corsica, France, October 9-12 (2012)

Study on Error Coding Program for Implementation in SOTA

Hideki Takenaka*, Morio Toyoshima*, Yoshihisa Takayama*, Yoshisada Koyama*,

Maki Akioka*, Eiji Okamoto**, Takuma Kyo**

*National Institute of Information and Communications Technology,
Tokyo, Japan
take@nict.go.jp

**Nagoya Institute of Technology
Tokyo, Japan

***Abstract*—The National Institute of Information and Communications Technology (NICT) is developing a small optical transponder (SOTA) to be incorporated into the design of a 50-kg-class satellite. SOTA is equipped with a coding system. For launch on a microsatellite, SOTA requires a limited amount of resources for coding systems, and therefore, it is necessary to properly design the memory usage. We have realized a 75% reduction in the memory usage using unused bit areas of variables. In addition, we have successfully reduced the memory usage to 1% using the listing method by focusing on the number of 0s in the check matrix.**

Keywords-small optical transponder; SOTA; LDGM; memory reduction

I. INTRODUCTION

In recent years, the capacity for high-resolution image data and data in observation equipment has increased because of advances in satellite functions. In line with this trend, it is necessary to increase speeds of communication links between the ground and satellites. The communication satellite Wideband InterNetworking engineering test and Demonstration Satellite (WINDS) that uses radio frequency (RF) has realized communication speeds of up to 1.2 Gbps [1], and larger-capacity higher-speed communications will be required in the future. In its present form, the widely used RF communications are unable to achieve communication speeds of several dozens of gigabits per second because they are limited by the carrier wave frequency. However, the use of satellite optical communication technology is currently being considered as a solution to this issue. The frequency of lasers used in optical communication is several hundred Terahertz (1012) and they are capable of transmitting gigabit-class baseband signals. The signals transmitted through the ground-to-space atmosphere are affected by atmospheric fluctuations because of the variability of the atmosphere. Therefore, deterioration in communication quality occurs. One method used to control this deterioration in communication quality caused by atmospheric fluctuations is the coding of communication data.

The National Institute of Information and Communications Technology (NICT) is currently developing a small optical transponder (SOTA) for launch on a 50-kg-class satellite [2]. Coding programs that run on a PC cannot be directly transferred to a SOTA on a microsatellite because there are limited resources that can be allocated to the coding system. This paper shows a method for implementing the program in environments in which there are limited resources available, and the array compression method used in the low-density generator matrix (LDGM) for the implementation of coding functions on a 50-kg-class satellite [3].

II. SMALL OTPICAL TRANSPONDER

Fig. 1 shows a picture of the SOTA proto-flight model (PFM). The primary mission of the SOTA is to enable ground-to-satellite optical communication experiments. The SOTA is equipped with four lasers. TX1 and TX4 are communication lasers that operate at wavelengths of 980 nm and 1550 nm, respectively. TX2 and TX3 use 800-nm lasers to gather basic data for future quantum key distribution [2].

As shown in Table 1, a considerably limited amount of resources can be dedicated to the coding of SOTA, and it is necessary to ensure efficient memory usage.

TABLE I. RESOURCES OF SATELLITE AND CODING

Satellite Resources		
Size	50 × 50 × 50	cm^3
Power	100	W
Mass	50	Kg
Mass of SOTA	5.7	Kg
Speed rate	1/10	Mbps
Coding Resources		
CPU	SH4	
Working memory	12	MB

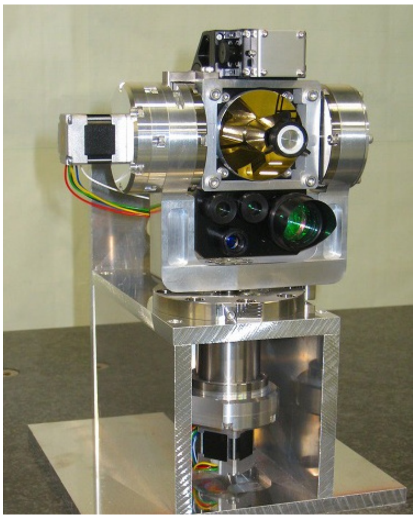

Figure 1. Photograph of a SOTA PFM.

III. IMPLEMENTING A CODING PROGRAM ON SOTA

The main issue that arises when implementing the coding program on SOTA is the size of the working memory, and only 12 MB of memory is available when SOTA executes the program.

The program was created on a PC using the int type for numerical arrays. For instance, to generate LDGM codes using the parameters in Table 2, as much as 2 MB of working memory it would be required for only a check matrix if the memory is obtained with the int type.

TABLE II. CODE PARAMETERS

Parameters	Values
Code	LDGM
Galois field	GF (28)
Code length	1000 (symbol)
Code rate	0.5

It was estimated that if the number of symbols increased with this set of parameters, the working memory usage would exceed 12 MB only with a check array when the number of symbols exceeded approximately 2500.

Therefore, we focused on the fact that the maximum value of a symbol can be obtained from the Galois field index and on the unused area of the int type in Fig. 2.

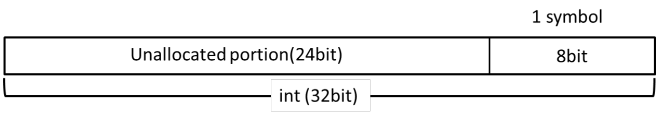

Figure 2. Type of int.

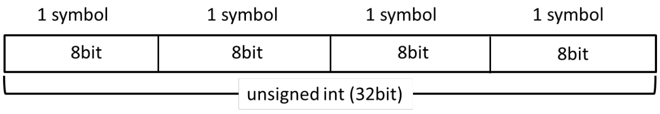

Figure 3. Type of economy using unsigned int.

If the Galois field index is 8, 8 bits are used in a symbol, and therefore, the 24-bit unused area can be obtained with the int type. Other symbols are inputted into the unused area, allowing four symbols to be stored in an unsigned int type, as shown in Fig. 3. Fig. 4 shows the comparison with the type of INT and type of economy using unsigned int. This allows a 75% reduction in the memory usage when the parameters in Table 2 are employed.

However, the processing time required by this method is higher then that required by ordinary array references because it uses functions to refer to arrays.

IV. SHRINKING A CHECK MATRIX

LDGM generates a check matrix when coding, the size of which varies with parameters. For instance, to generate codes using the previous parameters in Table 2, the size of an array would be column: 500, row: 1000, and the values present in the array would be 0–255. From the contents of this large array, it was clear that the value 0 accounted for a significant portion, as shown in Fig. 5. Therefore, we focused on further reducing the memory usage by omitting this value 0.

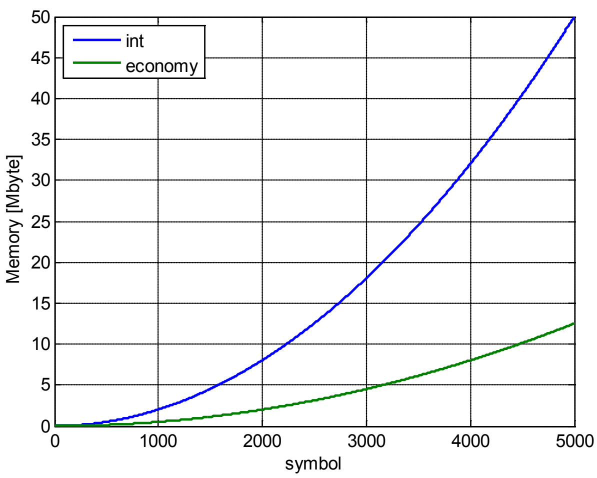

Figure 4. Memory size of check matrix.

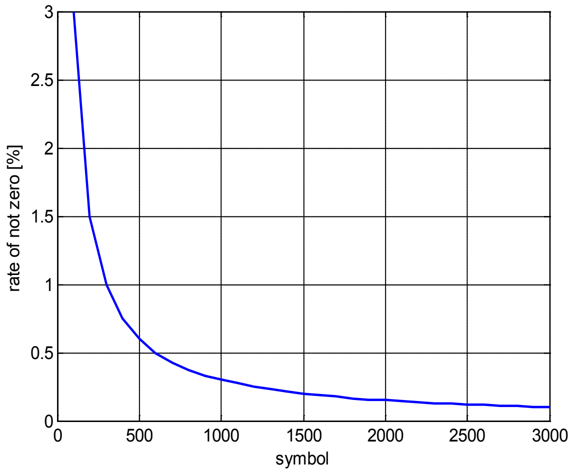

Figure 5. Rate of the check matrix if not zero.

To reduce the memory, values other than 0 are listed and values that are not included in the list are recognized as 0.

This method consists of three arrays, i.e., Array P that indicates positions, Array V that stores values, and Array B that indicates the positions of blocks. It first runs horizontally, and if a value other than 0 is found, the product of the horizontal and vertical values of the position is put into Array P, and the value itself is put into Array V. This value in Array P does not determine the position uniquely by itself; therefore, Array B is used. In addition, Array B is used to obtain the vertical position of the array and to increase the array reference speed. Blocking is a list of records of the starting points of the divisions, which divides a check matrix into groups of a specified number of values. Fig. 6 shows an example of a case that uses a listing method.

The reference procedure of this listing method determines the block value to which the vertical and horizontal values are referred. It refers to the target block sequentially from its beginning. If the desired vertical or horizontal value is within the block, it returns the value, and if not, 0 is returned.

This listing method allows for a reduction in memory usage to approximately one hundredth of the value that can be realized using an int array.

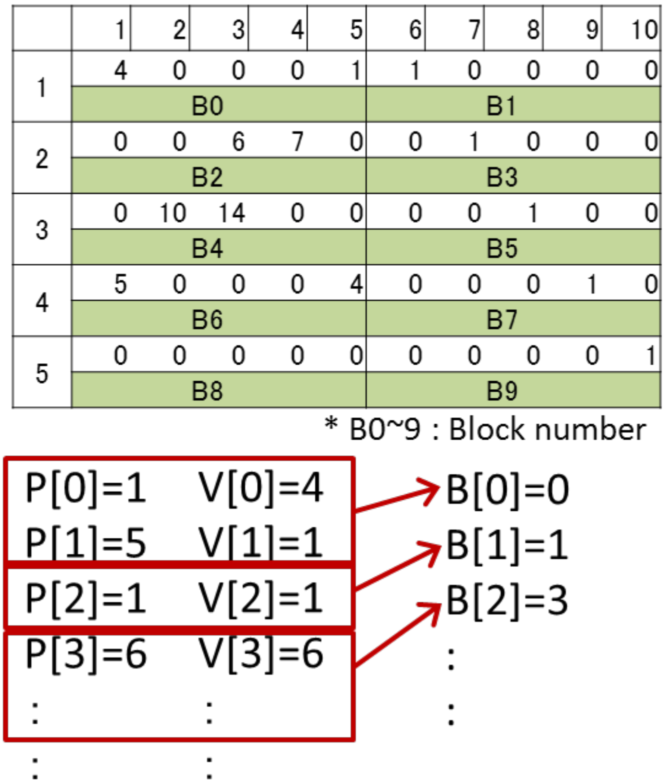

Figure 6. Example of a listing method.

V. CONCLUSIONS

This paper reported the implementation of a coding program on SOTA. The coding system of SOTA was limited the amount of memory. Therefore it was necessary to reduce the amount of memory used in coding. We were able to decrease the memory usage by 75% using the unused bit areas of variables. In addition, we successfully reduced the memory by 1% using the listing method, with a focus on the number of 0s in the check matrix. However, it should be noted that these methods involving the shrinking of arrays incur a longer time to be referred to an array than that required by ordinary arrays.

REFERENCES

[1] T. Takahashi "On-board Baseband Switch for Wideband InterNetworking engineering test and Demonstration Satellite," IEIC Technical Report, vol. 91, no. 9, pp. 772-726, 2008.

[2] H. Takenaka, M. Toyoshima, Y. Takayama, Y. Koyama, and M. Akioka," Experiment plan for a small optical transponder onboard a 50 kg-class small satellite," ICSOS, pp. 113-116, May 2011.

[3] V. Roca and C. Neumann, "Design, Evaluation and Comparison of Four Large Block FEC Codecs, LDPC, LDGM, LDGM Staircase and LDGM Triangle, plus a Reed-Solomon Small Block FEC Codec," INRIA Research Report RR-5225, June 2004.