Thread Pool Implementation in C

Jialin Li and Edward Hu

***Abstract*—This is a short report on our thread pool performance.**

I. Introduction

THREAD is a basic unit of CPU utilization, it consists of program counter, register set, and its own stack. Compared to processes, threads are much more lighweight since virtual memory space remains the same during a thread switch, which is not true for process switch. However, even threads sometimes pose overhead that will affect the performance of the program. Initializing and cleaning threads can pose a certain amount of overhead. For an parallel application that creates and destroys a large number of threads that each only runs for a short time, the thread management mechanism can create unnecessary resource waste. Thread pool can be useful in this type of scenario as it reduces the complexity of thread management and the overhead involved in thread creation and cleaning.

II. Motivation

We want to be able to implement a thread pool that is convinient to use. We aim to create a provide an interface that is easy for programmers to use. Thus we minimize the number of functions exposed to the programmers to maintain simplicity. We also want the performance of the thread pool to be reasonably more efficient than normal thread-based version. Thus we create several test cases to compare the performances in different implementations. We mainly focused on testing the performance difference in thread-based implementation and Goroutine-based version for certain types of jobs. Finally, we simply want to learn how to implement a thread pool and explore the drawbacks along with possible improvmetns.

III. Goals and Expectation

The report can be written in LaTeX or Microsoft Word, but LaTeX is definitely preferred. Its appearance should be as close to this document as possible to achieve consistency in the proceedings.

References should be cited as numbers, and should be ordered by their appearance (example: "... as shown in [1], ..."). Only references that are actually cited can be listed in the references section. The references' format should be evident from the examples in this text.

References should be of academic character and should be published and accessible. Your advisor can answer your questions regarding literature research. You must cite all used sources. Examples of good references include text books and scientific journals or conference proceedings. If possible, citing internet pages should be avoided. In particular, Wikipedia is *not* an appropriate reference in academic reports. Avoiding references in languages other than English is recommended.

Figures and tables should be labeled and numbered, such as in Table I and Fig. 1.

TABLE I
Simulation Parameters

Information message length	$k = 16000$ bit
Radio segment size	$b = 160$ bit
Rate of component codes	$R_{cc} = 1/3$
Polynomial of component encoders	$[1, 33/37, 25/37]_8$

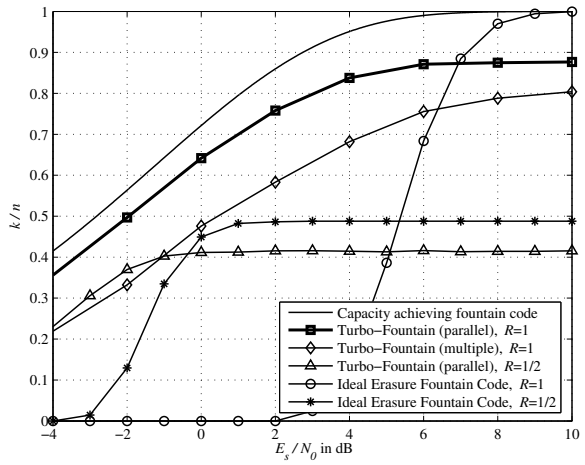

Fig. 1. Simulation results on the AWGN channel. Average throughput k/n vs E_s/N_0.

IV. Filling this page

Gallia est omnis divisa in partes tres, quarum unam incolunt Belgae, aliam Aquitani, tertiam qui ipsorum lingua Celtae, nostra Galli appellantur. Gallos ab Aquitanis Garumna flumen, a Belgis Matrona et Sequana dividit. Horum omnium fortissimi sunt Belgae, propterea quod a cultu atque humanitate provinciae longissime absunt, minimeque ad eos mercatores saepe commeant atque ea quae ad effeminandos animos pertinent important, proximique sunt Germanis, qui trans Rhenum incolunt, quibuscum continenter bellum gerunt. Qua de causa Helvetii quoque reliquos Gallos virtute praecedunt, quod fere cotidianis proeliis cum Germanis contendunt, cum aut suis finibus eos prohibent aut ipsi in eorum finibus bellum gerunt. Eorum una, pars, quam Gallos obtinere dictum est, initium capit a flumine Rhodano, continetur Garumna flumine, Oceano, finibus Belgarum, attingit etiam ab Sequanis et Helvetiis flumen Rhenum, vergit ad septentriones. Belgae

ab extremis Galliae finibus oriuntur, pertinent ad inferiorem partem fluminis Rheni, spectant in septentrionem et orientem solem.

V. Conclusion

This section summarizes the paper.

References

[1] J. Hagenauer, E. Offer, and L. Papke. Iterative decoding of binary block and convolutional codes. *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 429-445, Mar. 1996.

[2] T. Mayer, H. Jenkac, and J. Hagenauer. Turbo base-station cooperation for intercell interference cancellation. *IEEE Int. Conf. Commun. (ICC)*, Istanbul, Turkey, pp. 356–361, June 2006.

[3] J. G. Proakis. *Digital Communications*. McGraw-Hill Book Co., New York, USA, 3rd edition, 1995.

[4] F. R. Kschischang. Giving a talk: Guidelines for the Preparation and Presentation of Technical Seminars. http://www.comm.toronto.edu/frank/guide/guide.pdf.

[5] IEEE Transactions LaTeXand Microsoft Word Style Files. http://www.ieee.org/web/publications/authors/transjnl/index.html