Mathematical Problems of Computer Science 45, 5–13, 2016.

Results of Performance Analysis of Advanced Inftheo New Package for R

Narek S. Pahlevanyan, Mariam E. Haroutunian

Gyumri State Pedagogical Institute
Institute for Informatics and Automation Problems of NAS RA
e-mail: narek@ravcap.com, armar@ipia.sci.am

Abstract

The authors have developed a new package "Advanced Inftheo" [1] for R to perform computations of complex information-theoretical results. For achieving higher computational speed, the package includes different types of parallelization. For evaluation of package performance various experiments were conducted. The analysis of these experiments are presented in this paper. Moreover, the advantages of Advacned Inftheo package compared with the existing Infotheo package are introduced.

Keywords: R language, R package, Multithreading, E-capacity bounds.

1. Introduction

The popularity of R has grown significantly in recent years. Created in 1993 as an alternative to proprietary statistical software, R is an integrated suite of software tools for data manipulation, calculation and graphical display. This open source language has become an important part of the IT arsenal for analysts and data scientists performing statistical analysis on big data. Currently, there are about 3 million R users worldwide who operate thousands of open sources packages within the R environment to increase productivity in such domains as spatial statistics, bioinformatics, financial market analysis and linear/nonlinear modeling, etc. Large companies are using R, too. For example, Facebook uses R for exploratory data analysis, big-data visualization, human resources and user behaviour analysis related to status updates and profile pictures.

Recent concerns in the financial sector has stimulated Oracle to support R, for instance, Oracle is now bundling it as part of its Big Data Appliance product. Google uses R for advertising effectiveness, economic forecasting, and big-data statistical modeling. Twitter uses R for data visualization and semantic clustering. It contains a number of built-in functions for organizing data, running calculations on the information and creating elegant graphical representations of data sets which are very useful for data scientists. R provides a lot of different techniques for time-series analysis, classification, clustering as well as graphical packages for creating high quality, and sophisticated, customized plots with very simple syntax. The facilities of R can be extended through user-created packages. Packages (also known as modules) are libraries developed in C++, that include specific functions set for

certain applications. R comes with core set of packages, in addition to that there are more than 5,800 various packages and 120,000 functions free available for download [2] .

We believe that R can be beneficial and helpful for calculations of complex formulas of Information Theory. In practice many information-theoretical results are difficult for computing because of the large volume of distributions. For example, the investigation of rate-reliability function [3] for various applications [4], [5], [6] is time consuming, and computational results are hard to obtain.

R already had a package for calculating various measures of Information Theory called Infotheo, but there was a need in creation of a new package for estimation and computation of more complicated formulas mentioned above.

To perform computations of complex information-theoretical results in Information Theory the authors have developed a new package for R, called Advanced Inftheo. It was developed in C++; allows computation of more complicated formulas of Information Theory, such as functionality for computation of the lower and upper bounds of rate-reliability function, etc. Functions inside Advanced Inftheo are parallelized. It allows three types of parallelization (CPU-based, GPU-based and cluster-based) to a consumer. Moreover, it has a core set of functions included in Infotheo package but with different technical implementation. The reason of reimplementation of the existing functions of Infotheo inside Advanced Inftheo package was the investigation results of Infotheo source code that revealed that the functions inside package are single threaded.

Technical solutions used inside Advanced Inftheo package for overcoming the existing limitations [7], [8] of R language as well as for solving the issues arising with multi-threaded approach are discussed in [1]. The Advanced Inftheo package experimentation results are published in [9]. Specifically, in [9] the authors have computed the lower and upper bounds of E-achievable secret key rate of the biometric generated secret key sharing system obtained in [6] for various distributions. Moreover, they provide graphical representations of the experimentation results to simplify the solutions in building of applications.

In this paper we present the results of performance analysis of Advanced Inftheo package functions for various parallelization modes. Moreover, we introduce the main differences of Advanced Inftheo package from the existing Infotheo package and provide performance analysis results that show advantages in computational speed for the same functions of 2 packages.

2. Description of Infotheo Package

This package implements various measures of Information Theory based on several entropy estimators. The package was developed by Patrick E. Meyer, at the moment latest available version is 1.2.0 (published on July 2014). Package includes the following primary functions available for usage [10]:

- entropy() - computes entropy, takes the dataset as input and computes the entropy in nats (base e).
- condentropy() - computes conditional entropy, takes two random vectors X and Y as input and returns the conditional entropy $H(X|Y)$ in nats.
- condinformation() - computes conditional mutual information, takes three random variables as input and computes the conditional mutual information in nats.

- mutinformation() - computes mutual information, takes two random variables as input and computes the mutual information in nats.
- multiinformation() - computes multiinformation, takes a dataset as input and computes the multiinformation (also called total correlation) among the random variables in the dataset. Returned value is in nats.
- interinformation() - computes interaction information, takes a dataset as input and computes the interaction information among the random variables in the dataset. This measure is also known as synergy or complementarity.
- discretize() - performs unsupervized data discretization, discretizes data using the equal frequencies or equal width binning algorithm.

From the descriptions of available functions inside Infotheo we reveal that this package does not provide options for computation of more complicated formulas rate-reliability function, error exponent and so on. Furthermore, analyzing the source code of Infotheo showed that all functions inside the package are single threaded, which can lead to loss of productivity and efficiency on big datasets.

3. Description of Advanced Inftheo Package

Advanced Inftheo package provides functionality for computation of various complex formulas of Information Theory as well as it implements the main functions of Infotheo package. The main feature of Advanced Inftheo package is that it is using multithreaded computations for faster performance [11]. It means that Advanced Inftheo can use the advantage of multiprocessor hardware. Moreover, it allows 3 types of parallelization:

- CPU-based parallelization,
- GPU-based parallelization,
- cluster-based parallelization (MPI technology).

Inside CPU-based parallelization of Advanced Inftheo package is the idea of multi-core systems, that are now very common, and the number of processors per chip is growing. There is a need for integration of R code into multi-core environments. This approach has the potential to speed up a numerically intensive code. To design a package to support a more abstract use of multi-core systems from package development point of view seems to be a very difficult task, because the interactive nature of R and the existing multi-core technology implies runtime compilation.

In terms of hardware, the parallel computing power of graphic processing units (GPUs) might provide significant performance benefits to users of Advanced Inftheo. NVidia CUDA technology is being used inside Advanced Inftheo package for GPU-based parallelization, CUDA offers a programming model that is designed to allow direct access to the specific graphics hardware, with the graphics hardware running a very high number of threads in parallel.

From user perspective, computer clusters are often unavailable, the physical costs (maintenance, etc.) are high, and effective use requires a various skill set (e.g, advanced user-level

system configuration, mastery of batch job submission, careful formulation of problems to effectively manage communication costs). Therefore, it is difficult to use this technology. In cluster-based parallelization type of Advanced Inftheo we consider that clusterized environment is already set up and R is properly installed there. The integration with cluster is done with OpenMPI library, which allows Advanced Inftheo package to take default settings (processors count, RAM limit etc) for performing computations.

In addition to functions described above for Infotheo package, Advanced Inftheo includes the following functions :

- setComputationMode() - allows selection of parallelization type (CPU, GPU or MPI) that's used in rest of functions.
- getComputationMode() - retrieves current parallelization type.
- setVerboseMode() - sets package in debug mode. In this mode debug messages will be shown on every step of computations.
- setThreadsCount() - sets threads count used in package functions. If this value is not set, the package will find and use the number of concurrent threads supported by hardware.
- getThreadsCount() - retrieves currently used threads count.
- setGPUCoresCount() - sets GPU cores count used in package functions. This value can't be bigger than the count of available physical multiprocessors.
- calcEntropy() - calculates entropy (in bits) of discrete random variable using probability distribution vector.
- calcJointEntropy() - calculates joint entropy of two discrete random variables using joint distribution matrix.
- calcRelativeEntropy() - calculates Kullback-Leibler divergence (information gain in bits).
- calcMutualInformation() - calculates mutual information of two discrete random variables using joint distribution matrix.
- calcConditionalEntropy() - calculates conditional entropy of two discrete random variables using joint distribution matrix.
- calcMarginalDistribution() - calculates marginal probability distribution of $n \times m$ sized probability matrix.
- calcLowerBoundOfECapacity() - calculates lower bound of E-capacity for provided reliability and distribution matrix, based on computation of minimum mutual information for all distributions in the condition of limited divergence.
- calcUpperBoundOfECapacity() - calculates upper bound of E-capacity for provided reliability and distribution matrix, based on computation of minimum mutual information for all distributions in the condition of limited divergence.

For evaluation of Advanced Inftheo package performance some experiments were conducted: load testing, speed testing, configuration testing, isolation testing, etc.

4. Performance Analysis Results

Here we provide performance analysis of Advanced Inftheo package functions and give general recommendations for setting optimal thread count for various N (count of input arguments) values. Also we compare performance of common functions in both packages. Computations in Fig.1 have been performed on machine with medium parameters (Intel Core 2 Duo 2 x 2.00GHz, with 2.5GB RAM) with default hardware concurrency (in our case it was 2).

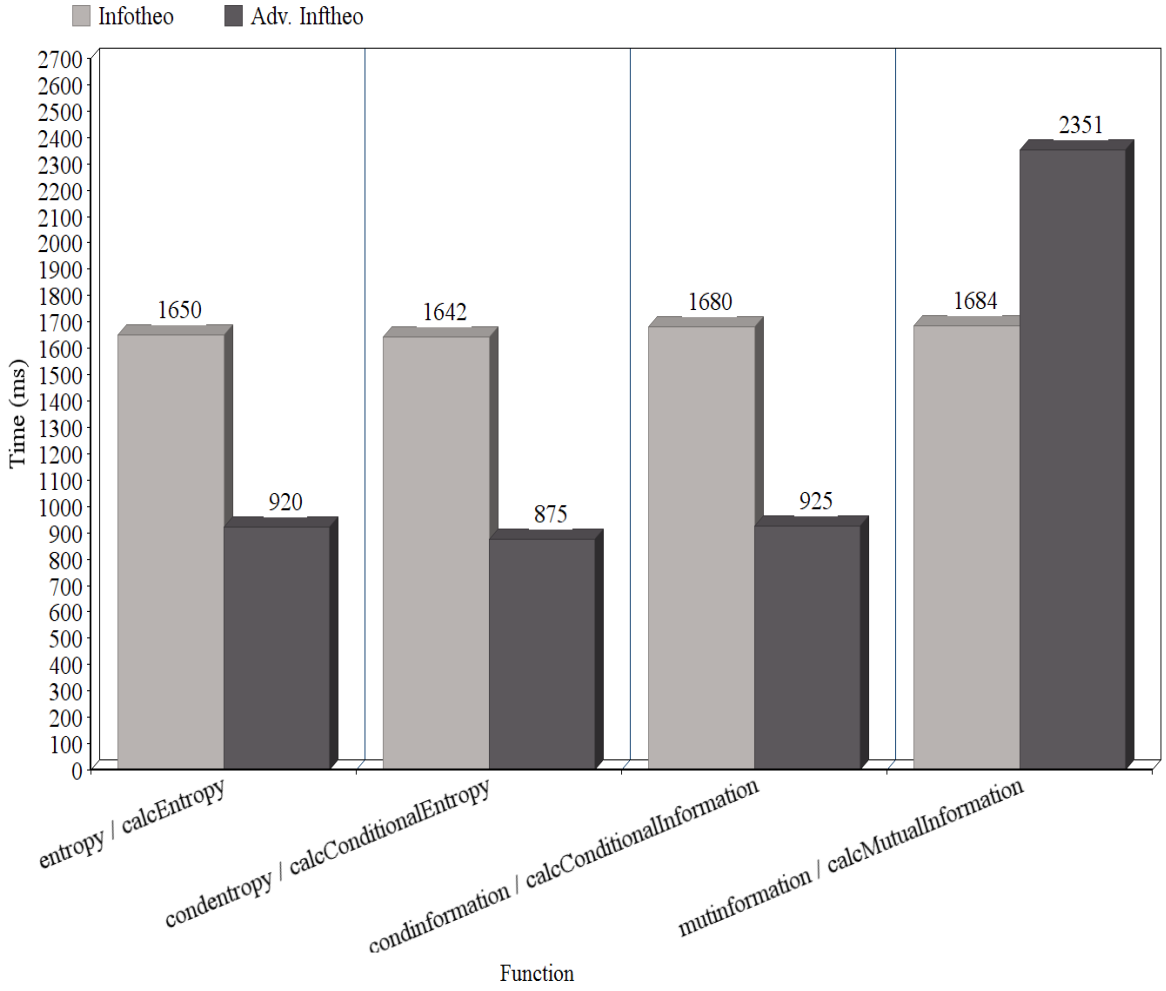

Fig. 1. Comparison of common functions when Advanced Inftheo is in CPU mode with thread count set to 2 and for N = 10^6.

From Fig. 1 we can see that in CPU mode with 2 threads Advanced Inftheo is in average 1.85 times faster than Infotheo. We will have different results for smaller N if we use GPU mode with higher number of cores. Fig. 2 shows that when Advanced Inftheo is in GPU mode and the lower N is used we have downgrade in computational speed compared with Infotheo.

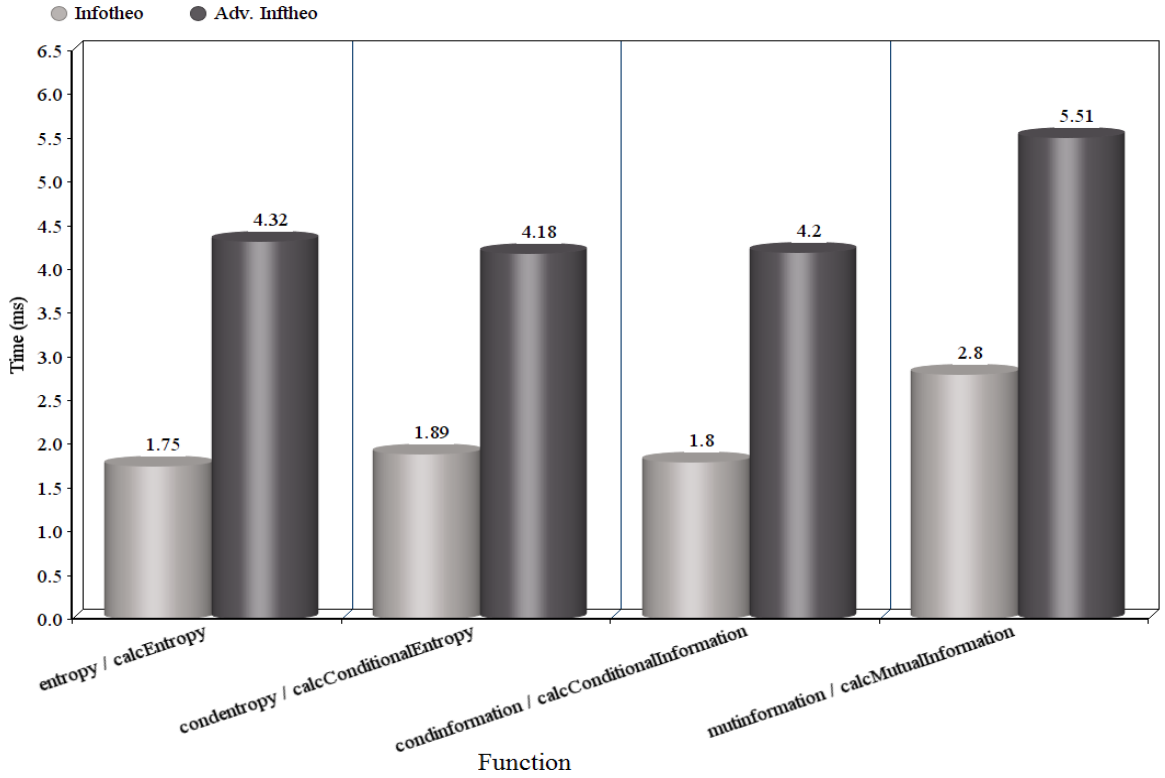

Fig. 2. Comparison of common functions when Advanced Inftheo is in GPU mode with core count set to 32 and for N = 500.

Next figures (Fig. 3 - Fig. 5) are demonstrating performance of calcUpperBoundOfECapacity() function for various N and for various modes of Advanced Inftheo.

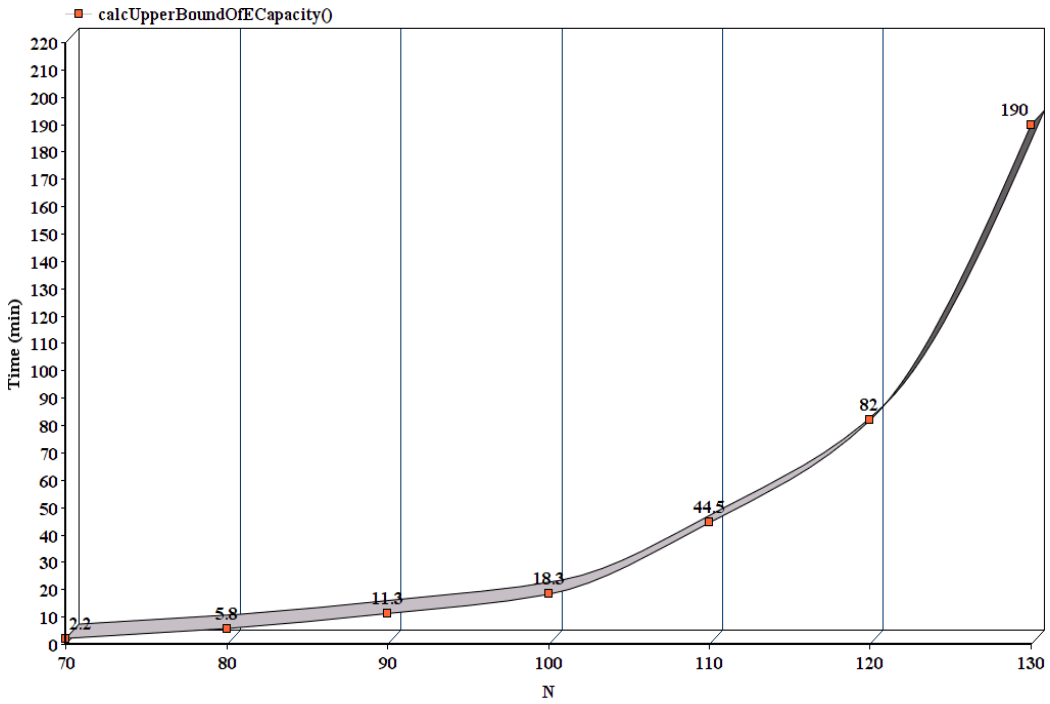

Fig. 3. Performance of calcUpperBoundOfECapacity() function for various N when Advanced Inftheo is in CPU mode and threads count is set to 8.

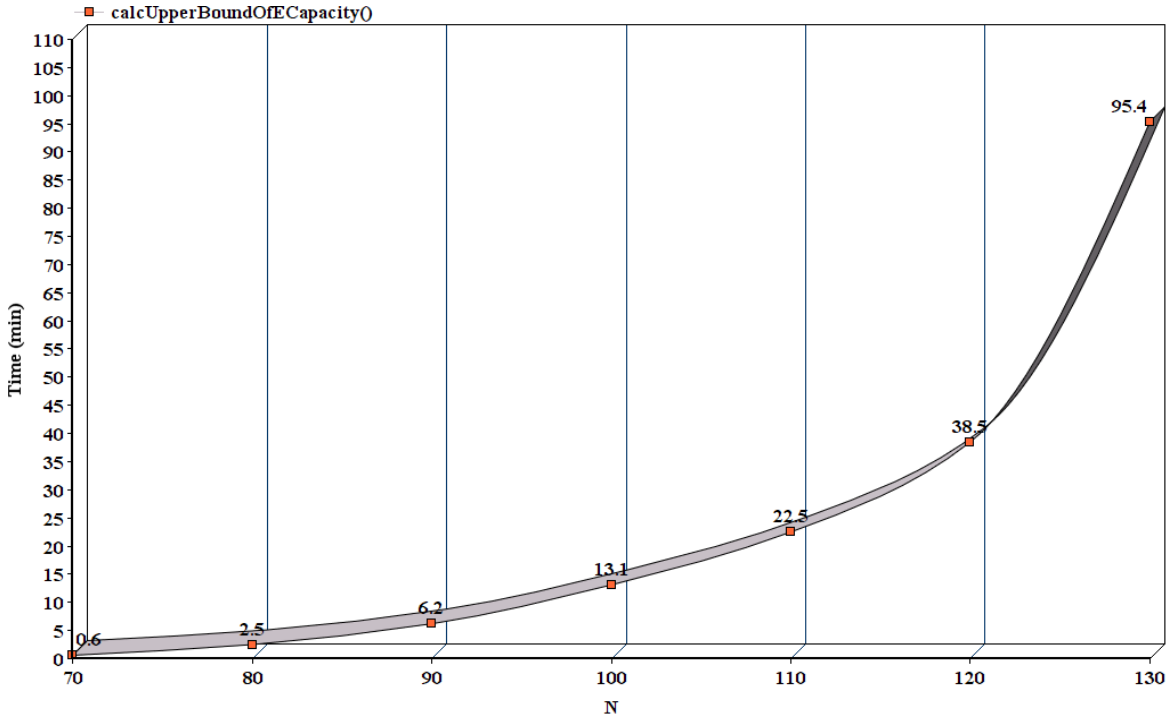

Fig. 4. Performance of calcUpperBoundOfECapacity() function for various N when Advanced Inftheo is in GPU mode with core count set to 32.

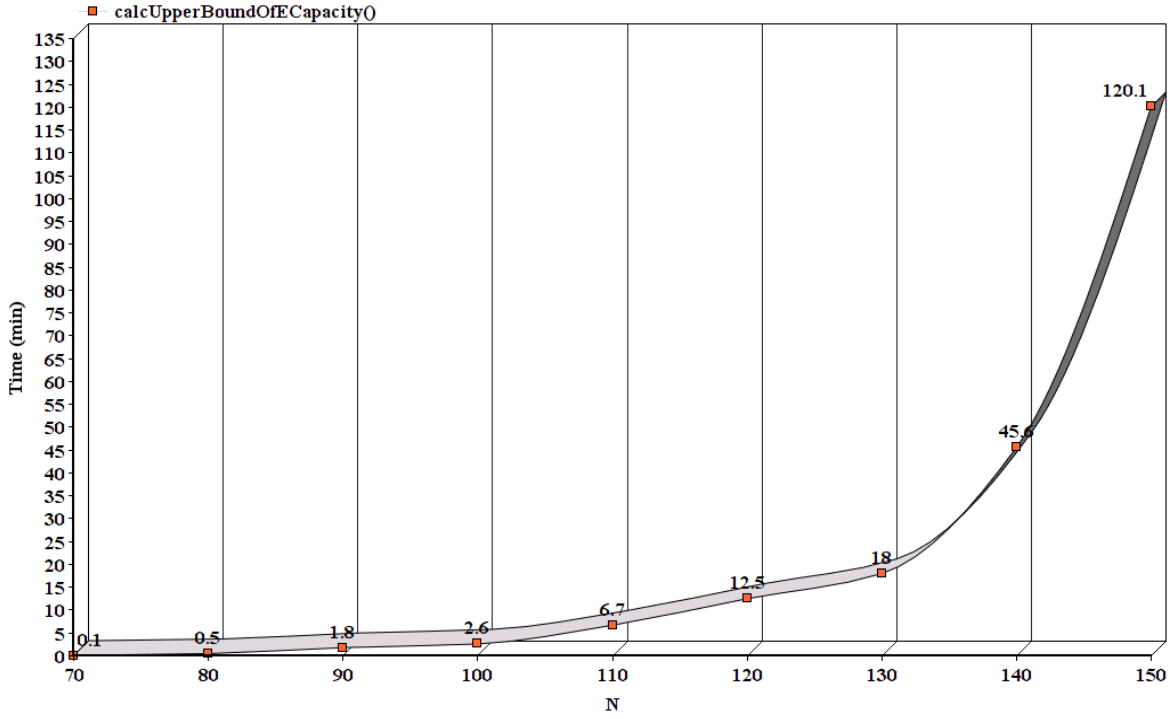

Fig. 5. Performance of calcUpperBoundOfECapacity() function for various N when Advanced Inftheo is in MPI mode with usage of 40 processors.

In GPU mode tests were performed on NVidia GeForce GT 440 graphical adapter. Experiments in MPI mode (cluster-based parallelization) are depicted in Fig. 5, those tests were been executed on ArmCluster with usage of 40 processors.

From Fig. 4 and Fig. 5 we can see that we will have time gain if we use Advanced Inftheo in GPU or MPI modes. The thread count for GPU mode should be selected wisely, otherwise downgrade in performance is imminent. Our other experiments with GPU mode showed that the number of threads should be set as multiplier of GPU multiprocessors warp size for better performance.

5. Conclusion

Parallelism inside Advanced Inftheo can give huge time gain in computations of information-theoretical results for practical applications. Different modes inside Advanced Inftheo package will provide various performance results, to achieve higher performance results the thread count in CPU and GPU modes should be calculated precisley based on computational problem. For computation of non-complex formulas (such as entropy, mutual information etc.) when $N < 10^3$ we recommend the usage of smaller threads count (2-4). For computation of complex formulas (such as calculation of upper bound of E-capacity for provided reliability and distribution matrix) GPU and MPI modes can give big time gain compared with CPU mode, thread count for GPU mode should be set as multiply of multiprocessor warp size.

References

[1] N. Pahlevanyan and M. Haroutunian, "Technical solutions of developing Advanced Inftheo new module for R," in *Proceedings of the 10th International Conference on Computer Science and Information Technologies*, Yerevan, Armenia, 2015, pp. 306–309.

[2] W. N. Venables, D. M. Smith and the R Core Team, "An introduction to R," *version 3.1.1*, pp. 51–77, 2014.

[3] E. A. Haroutunian, M. E. Haroutunian and A. N. Harutyunyan, "Reliability criteria in information theory and in statistical hypothesis testing," *Foundations and Trends in Communications and Information Theory*, vol. 4, no. 2-3, 2008.

[4] T. Ignatenko and F. M. Willems, "Biometric security from an information-theoretical perspective," *Foundations and Trends in Communications and Information Theory*, vol. 7, no. 2-3, 2012.

[5] R. Ahlswede and I. Csiszar, "Common randomness in information theory and cryptography. i secret sharing," *IEEE Transactions on Information Theory*, vol. 39, no. 4, pp. 1121–1132, 1993.

[6] M. Haroutunian and N. Pahlevanyan, "Information theoretical analysis of biometric secret key sharing model," *Transactions of IIAP of NAS of RA, Mathematical Problems of Computer Science*, vol.42, pp. 17-27, 2014.

[7] R Development Core Team, "R: A language and environment for statistical computing," *R Foundation for Statistical Computing*, 2011.

[8] D. Smith, "The R ecosystem," useR! conference, Coventry, United Kingdom, August 2011.

[9] M. Haroutunian and N. Pahlevanyan, "Experimentation of Advanced Inftheo module for R on the example of biometric generated secret key sharing system," *International Journal Information Content and Processing*, vol. 2, no. 1, pp. 62–70, 2015.

[10] P. E. Meyer, "Package Infotheo," *The Comprehensive R Archive Network*, pp. 1–12, July 2014.
[11] C. Hughes and T. Hughes, *Professional Multicore Programming: Design and Implementation for C++ Developers*. Birmingham, UK, UK: Wrox Press Ltd., 2008.

Submitted 12.09.2015, accepted 26.01.2016

R-ի նոր Advanced Inftheo փաթեթի արտադրողականության վերլուծության արդյունքները

Ն. Փահլևանյան, Մ. Հարությունյան

Ամփոփում

Ինֆորմացիայի տեսության բարդ բանաձևերի հաշվման համար մշակվել է նոր փաթեթ Advanced Inftheo [1] R լեզվի համար: Հաշվողական մեծ արագության հասնելու համար փաթեթն իր մեջ ներառում է տարբեր տիպի զուգահեռացման մեխանիզմներ: Փաթեթի արտադրողականության գնահատման համար կատարվել են տարբեր փորձեր: Այդ փորձերի վերլուծությունը ներկայացված է այս հոդվածում: Քննարկված են նաև Advanced Inftheo փաթեթի առավելությունները գոյություն ունեցող Infotheo փաթեթի համեմատությամբ:

Результаты анализа производительности нового пакета Advanced Inftheo для R

Н. Пайлеванян, М. Арутюнян

Аннотация

Для вычислений результатов сложных функций теории информации разработан новый пакет Advanced Inftheo [1] для языка R. В модуле для достижений высокой эффективности встроены разные механизмы параллелизаций. Проведены различные эксперементы для оценки производительности пакета. В данной статье приводится анализ этих экспериментов. Кроме того, представлены преимущества пакета Advanced Inftheo в сравнении с сущетвующим модулем Infotheo.