1 研究目的、研究方法など

本研究計画調書は「小区分」の審査区分で審査されます。記述に当たっては、「科学研究費助成事業における審査及び評価に関する規程」（公募要領１８頁参照）を参考にすること。
本研究の目的と方法などについて、４頁以内で記述すること。
冒頭にその概要を簡潔にまとめて記述し、本文には、(1)本研究の学術的背景、研究課題の核心をなす学術的「問い」、(2)本研究の目的および学術的独自性と創造性、(3)本研究の着想に至った経緯や、関連する国内外の研究動向と本研究の位置づけ (4)本研究で何をどのように、どこまで明らかにしようとするのか、(5)本研究の目的を達成するための準備状況、について具体的かつ明確に記述すること。

（概要）

The aim the research proposal is to develop **advanced optimisation methods** to improve the performance and overall efficiency of (scientific) computer programs by addressing **data locality** on supercomputers, in computations with **sparse matrices**. The development of such advanced optimisation techniques is timely because the tools required to achieve a breakthrough in this field recently matured enough and became available, while the processor memory gap and data movement is still the primary bottleneck of compute utilisation in most applications. After the applicants preliminary work on **communication-avoiding matrix powers kernels** (CA-MPK), the “Diamond matrix powers kernel” (DMPK) algorithm [1] showed possibilities for applicability of **machine learning/deep learning** (ML/DL), which serves as an exciting entry point of the application of ML/DL in advanced optimisations.

（本文）

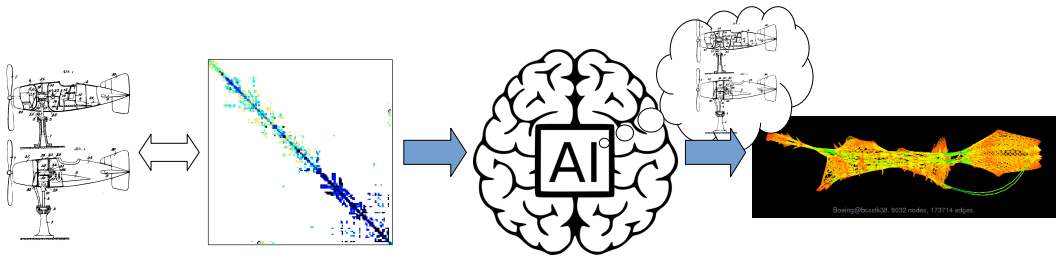


図 1: ML/DL (or AI) capturing the essence of the real world problem from a sparse matrix, producing advanced (“smarter”) optimised programs

The key scientific question at the core of the research proposal is the achievement of optimal utilisation of computer hardware. This is a complex problem because: (a) the overwhelming variety of the hardware makes it difficult to find a solution applicable for every device; (b) the different types of problems which are being solved by sparse matrices and the plethora of communication patterns they induce; and (c) the human factor of writing programs by scientists and domain experts who don’t necessarily specialise in writing code which utilises the given hardware efficiently. Hence, **automating such a task, to generate optimal code for any combination of hardware and (sparse) matrices could improve existing code bases as well es enable easier development of more efficient code in the future.**

The (technical) reason behind this problem is the gap between CPU and memory speeds. An effective way to alleviate this problem is spacial and temporal blocking, which is usually applicable to regular computation patterns (such as stencils), while for sparse matrices **communication avoiding kernels** can enable similarly efficient parallelism and data-locality. The partitions of the domain (corresponding to tiles) depend on the graph represented by the sparse matrix in question, and **ML/DL** is applicable as a more efficient graph partitioning algorithm.

This cooperation of sparse matrices, communication avoiding algorithms, ML/DL would open the path to a promising future research prospective on advanced opti-

【1　研究目的、研究方法など（つづき）】

misations.

Scientific background: The "processor memory gap" also referred to as the "memory wall" is well known since **data movement became the dominant factor in both energy consumption and performance in HPC** systems [2]. This is an area of active research and is being attacked from different sides including compiler level optimisations, polyhedral compilation and similar loop transformations, storing the input in novel data-structures or even using lossy compression methods which modify the results and behaviour of the program (within acceptable boundaries of course).

Most of the existing approaches are just slowly chipping away at the problem, but the tools, which constitute the necessary parts of a solution, have matured, and a breakthrough is due. Cracking the essence of optimisation is key, and would open a stream rich in results for both research and application, as well as a new prospective line of research.

For regular access patterns various tiling and loop transformations are performed to achieve data-locality, but for sparse marices, the memory access pattern depends on the pattern of non-zero elements in the matrix and becomes irregular and hard to predict or extrapolate. Communication avoiding algorithms split (the vertices of) the matrix into partitions and localise the computation to a single partition and/or using "halos" around the partition to trade the high cost of communication for redundant computation. The choice of these partitions determines the trajectory and long term performance of algorithms to a great extent, hence partitioning the matrix/graph is crucial. Most of the communication avoidance research only used the Metis graph partitioning library, so the application of more advanced graphical neural networks can widen the possibilities both in terms of results and further research perspective. **ML/DL methods have the potential to "learn" more essential characteristics of problems described by sparse matrices and to obtain genuine insight about the communication patterns and to provide the most efficient utilisation of the memory hierarchy (including remote access on clusters/supercomputers)** depicted in Figure 1. Using machine learning can also improve usability by delivering a more "end-to-end" type of solutions: for example automatically choosing the s parameter for s-step MPK algorithms.

Purpose, creativity and originality: The primary purpose of the proposed research is the optimisation of scientific programs. If successful, this would mean obtaining **faster results when performing scientific computations** by having better utilisation of supercomputers. This would also entail the reduction of power consumption, which at the scale of super computers is far from negligible, as well as potential new possibilities in various research fields due to the faster computations.

The most promising detail about the project is the **creative application of ML/DL** methods. General purpose graph partitioning algorithms, such as Metis, partition the graph based on the edge and node weights. In the DMPK algorithm [1], these weights are hand picked functions of the progress of each node. The matrices stem from real world problems, which implies some degree of regularity and structure, therefore ML/DL methods, such as graph neural networks, can be applied to extract the necessary information for a "smarter" partitioning (see Figure 1).

The originality stems from the proposed combination of sparse matrices, communication avoidance, data-locality and ML/DL has not been applied simultaneously in optimisation of scientific codes. Additionally, the development of original data-structures is also a possible direction and source of inspiration for future solutions.

Research development and trends: Data-locality related optimisations were central to the applicant PhD theses [3, 4] as well as the JSPS fellowship spent at The University of Tokyo [1]. The current project can be considered as the continuation of this work, hence it

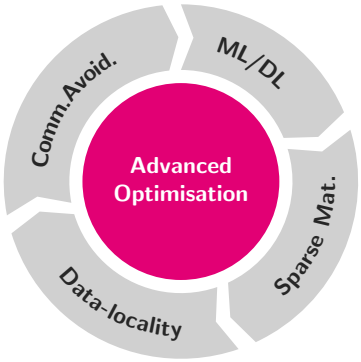
【1　研究目的、研究方法など（つづき）】

constitutes the research development leading to the conception of the present research proposal. Furthermore, the applicant gained experience in ML/DL which can be applied to the problem the proposal aims to solve. Additionally the goals of the proposal are in line with other related work conducted at Riken R-CCS, High Performance Artificial Intelligence Systems research team, the current place of employment of the applicant.

The importance of **sparse matrices** has become increasingly relevant. Their value is well established in scientific programs and recently AI/ML is also heading for similar direction [5]. The **communication avoiding algorithms** were also being revived [6] demonstrating their necessity.

About the **maturity of the required tools and libraries**: the proposed project involves multiple technologies such ML/DL models and compilers. The field of artificial intelligence has been going through major advances since AlexNet, including BERT which is referred as the "AlexNet moment of NLP" as well as advances in graphical neural networks. Other relevant advances are related to compilers and optimisations: LLVM is a compiler toolchain, which unlike previous more monolithic compilers, has a very flexible and modular design, which enables access to separate stages of the compilation, such as extracting the abstract syntax tree or modify optimisations, without needing to understand the entire codebase. This can be leveraged both to extract information from the source code at certain stages of the compilation as well as the potential to integrate the proposed advanced optimisation solutions into the toolchain. Recently developed TVM (`https://tvm.apache.org/`) and MLIR (`https://mlir.llvm.org/`) technologies are related efforts of developing and supporting better optimisations for programs.

What will be elucidated, how will it be pursued: The proposed research will elucidate **how to rearrange data to achieve data-locality when faced with arbitrary/general access patterns, such as the data access induced by SpMV kernels, and to achieve optimisations similar to tiling, using ML/DL techniques and/or custom data-structures.**



When (manually) optimising such programs (for a given hardware and/or for a given matrix) one is faced with many challenging decisions: how to partition/tile, what/when to recompute/communicate, how to determine the halo regions etc. Machine learning is an excellent candidate to use as a tool for answering these questions. The main challenge in the research will be finding the proper representations, both for the ML/DL model, as well as the representation of the data in the memory.

The proposed research will distil all such decisions and knowledge to solve these problems by providing a path to a general solution: to be able to make the correct decisions for any given matrix, running code on any given hardware, with automatic optimisation, autotuning and code generation in mind.

The elusive goal of the singularity where HPC improves AI and AI improves HPC ad infinitum will probably remain fiction, however the applicant is committed to realise the **prospect of an AI system which has some "understanding" of computer programs**. In line with this sentiment, other related research will also be pursued, such as learning custom data-structures and algorithms (such as [3]) and cooperation with other similar research done at RCCS on stencil computations.

The core of the proposed research will follow the usually steps for developing

【1　研究目的、研究方法など（つづき）】

ML/DL models: 1. create a dataset; 2. design a model; 3. train and optimise the model. The starting point for the first step (dataset) is the collection of sparse matrices found in the SuiteSparse Matrix Collection (formerly the University of Florida Sparse Matrix Collection, available at `https://sparse.tamu.edu/`), the second step involves the implementation of the model in a standard framework like PyTorch and its integration into the DMPK algorithm [1] algorithm, while the third step of training will be done on the Fugaku and ABCI supercomputers and the optimisations will cover incremental improvements the model, finding better hyperparameters etc. However, the repeated iteration of these three steps will be considered in a wide sense: it will not focus exclusively on the narrow problem of finding the optimal partition for DMPK algorithm, an eye will be kept on possible solutions to advanced ML/DL based optimisation, with the possibility of deeper comprehension of (scientific) codes by ML/DL models.

Preparation status: The proposal is a continuation of different aspects of the applicant's previous research. The main direction is the expansion of the **existing research on CA-MPK** [1], developed during the JSPS fellowship, in synergy with approaches inspired by the applicant's work during PhD and the ML/DL experience the applicant obtained in recent years.

A natural continuation of [1] is the integration of the DMPK algorithm described in the paper into a more comprehensive numerical library like the Ginkgo project or PETSc. Since DMPK was developed with good coding practices in mind, these milestones are reachable withing a few months and the more relevant task of the project of developing an advanced ML/DL algorithm instead of the Metis library can begin.

The applicant is **working on a similar project** (as part of his current employment) with a slightly different take on advanced optimisations. Both projects aim to achieve a breakthrough in optimising scientific codes by combining multiple techniques such as ML/DL and modern compilers tools: the current proposal focuses on spars matrices, while the other focuses on loop transformations. Nevertheless, both projects will share a great deal of infrastructure, such as a **framework to train ML/DL models for code optimisation** (similar to the autotuning framework described in the first Figure in [7]), which will be integrated in existing tools and made available for public use in an effort to aid the progress towards advanced optimisation.

参考文献

[1] Emil Vatai and Utsav Singhal and Reiji Suda, Diamond Matrix Powers Kernels, Proc. of the Int. Conf. on High Perf. Comp. in Asia-Pacific Region, HPCAsia2020 102–113, 2020.

[2] Didem Unat et al., Trends in Data Locality Abstractions for HPC Systems, IEEE Transactions on Parallel and Distributed Systems, 28(10):3007-3020, 2017.

[3] Antal Járai and Emil Vatai, Cache optimized linear sieve, Acta Univsitatis Sapientiae. Informatica, 3(2):205–223, 2011.

[4] Emil Vatai, Inverse sieve, Annales Univ. Sci. Budapest, Sect. Comp, 41:355-360, 2013.

[5] Torsten Hoefler et al., Sparsity in Deep Learning: Pruning and growth for efficient inference and training in neural networks, arXiv, 2021.

[6] Yasuhiro Idomura et al., Application of a Preconditioned Chebyshev Basis Communication-Avoiding Conjugate Gradient Method to a Multiphase Thermal-Hydraulic CFD Code, SCFA, 2018, 257–273.

[7] Prasanna Balaprakash et al., Autotuning in High-Performance Computing Applications, Proceedings of the IEEE, 106(11):2068-2083, 2018.

２　応募者の研究遂行能力及び研究環境

応募者の研究計画の実行可能性を示すため、(1)これまでの研究活動、(2)研究環境（研究遂行に必要な研究施設・設備・研究資料等を含む）について２頁以内で記述すること。
「(1)これまでの研究活動」の記述には、研究活動を中断していた期間がある場合にはその説明などを含めてもよい。

The applicant has been working on algorithms addressing data-locality since his PhD and is in a **unique position of having experience in the components required for the proposed project**: sparse matrices; communication avoiding and tiling algorithms; high performance tuning; machine learning; low level optimisations; strong foundation in mathematics; basics knowledge of compilers.

Hitherto research activities

The two algorithms during the applicant's PhD addressed the data-locality in computational number theory, more concretely in sieving algorithms. Sieving algorithms, such as the sieve of Eratosthenes (SOE) for finding primes or the general field number sieve (GNFS) for factorising large integers become very inefficient with respect to cache hierarchy, because the lack of data-locality. The reason for this, is that the operation in the inner most loop of a sieving algorithm, the operation of "sieving with a prime p" needs to do a minimal amount of compute (setting a bit for SOE, adding a float for GNFS) at effectively random locations in memory because sieving is done at every p-th element, which are far apart for large primes.

The COLS [2] algorithm uses an advanced data-structures (and the property of the operation executed when sieving) to rearrange the memory accesses to make it almost entirely contiguous, which also introduced natural opportunities for parallelism. The inverse sieve [3] tackles the problem of data-locality by applying a very simple and fast compression to the sieve table of SOE, which can be applied when the table becomes sufficiently sparse. This research contributed in finding the largest known primes, called Cunningham chain of length 3 of the first kind [5].

This was followed by a hiatus in research and publishing because the applicant was concentrating on teaching and developing materials for courses, including two courses of discrete mathematics (in English) for international students majoring in computer science. This was required by the university in Hungary because of the shortage of lecturers.

The next project, DMPK [1] was done as part of a JSPS fellowship. The DMPK algorithm is a communication avoiding algorithm, which rearranges memory accesses, to achieve data-locality similar to temporal tiling in parallel stencil computations.

Since before the fellowship, the applicant has learned and been active in teaching and implementing various developing applications of ML/DL methods (no published papers, activities and talks in Machine Learning Tokyo `https://mltokyo.ai/` a the non-profit organisation with the goal of democratising machine learning). Since 2020, the applicant has worked on NLP (`https://github.com/vecto-ai/langmo`), simulations and the prospect of future hardware [4] as well as benchmarking (`benchmarker-git`).

1. "Diamond Matrix Powers Kernels", Emil Vatai and Utsav Singhal and Reiji Suda, Proceedings of the International Conference on High Performance Computing in Asia-Pacific Region **HPCAsia2020**, 102–113 (2020).

2. "Cache optimized linear sieve", Antal Járai and Emil Vatai, Acta Univsitatis Sapientiae. Informatica **3(2)**, 205–223 (2011).

【2　応募者の研究遂行能力及び研究環境（つづき）】

3. “Inverse sieve”, Emil Vatai, Annales Univ. Sci. Budapest, Sect. Comp **41**, 355-360 (2013).

4. “Matrix engines for high performance computing: A paragon of performance or grasping at straws?”, Jens Domke and Emil Vatai and Aleksandr Drozd and Peng Chen and Yosuke Oyama and Lingqi Zhang and Shweta Salaria and Daichi Mukunoki and Artur Podobas and Mohamed Wahib and Satoshi Matsuoka, 2021 IEEE International Parallel and Distributed Processing Symposium **IPDPS**, 1056–1065 (2021).

5. “The largest known Cunningham chain of length 3 of the first kind”, Farkas, Gábor and Gévay, Gábor E and Járai, Antal and Vatai, Emil, Studia Universitatis Babes-Bolyai Mathematica **59(4)**, 457–462 (2014).

Research environments

The research environments relevant to the conduct of the proposed research:

- The applicant did his PhD in the Department of Computer Algebra, Faculty of Informatics, **Eötvös Loránd University**, in Budapest, Hungary under the mentorship of professor Antal JÁRAI. Since the beginning of his PhD, the applicant was teaching at the university, primarily discrete mathematics (from 2012 this also included courses for international students) as well as an introductory course to compilers.

- The JSPS scholarship was conducted in **The University of Tokyo**, lab of professor Reiji SUDA.

- From 2020 to the present, the applicant is working in the High Performance Artificial Intelligence Research team of **RIKEN Center for Computational Science**.

３　人権の保護及び法令等の遵守への対応（公募要領４頁参照）

本研究を遂行するに当たって、相手方の同意・協力を必要とする研究、個人情報の取扱いの配慮を必要とする研究、生命倫理・安全対策に対する取組を必要とする研究など指針・法令等（国際共同研究を行う国・地域の指針・法令等を含む）に基づく手続が必要な研究が含まれている場合、講じる対策と措置を、１頁以内で記述すること。
個人情報を伴うアンケート調査・インタビュー調査・行動調査（個人履歴・映像を含む）、提供を受けた試料の使用、ヒト遺伝子解析研究、遺伝子組換え実験、動物実験など、研究機関内外の倫理委員会等における承認手続が必要となる調査・研究・実験などが対象となります。
該当しない場合には、その旨記述すること。

N/A (not applicable).