ISSN 2186-7437

NII Shonan Meeting Report

No. 2013-10

Compact Data structures for Big Data

Kunihiko Sadakane
Wing-Kin Sung

September 27–30, 2013

National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-Ku, Tokyo, Japan

Compact Data structures for Big Data

Organizers:
Kunihiko Sadakane (National Institute of Informatics)
Wing-Kin Sung (National University of Singapore)

September 27–30, 2013

Big Data are structured and unstructured datasets whose size is in the order of billions or trillions. Because of their diversity and size, it is difficult to store, search and analyze them. This meeting therefore focuses on algorithms and data structures for efficient manipulation of Big Data. Especially, the meeting is devoted to compact data structures for managing Big Data.

Typical examples of big data are genomic sequences and gene expression data, Web and SNS data, sensor data in intelligent transport systems, etc. Traditional data structures do not scale to handle such data, and therefore we should design new data structures to handle them.

Although the amount of data explodes, the amount of the underlying information inside the data may not be exploded because it is observed that many big datasets are redundant. In the Web, many webpages were copies of others. In global positioning system (GPS), GPS position data change continuously, which can be compressed using differential encoding. In genomics, although different individuals have different genomes, the individual genomes have highly similarity. Therefore we can compress such data by identifying the similar parts. After the data is compressed, other issues are how to access and search them efficiently. Traditional data structures are not designed to handle compressed data and they may not manipulate Big Data well because the size of the data structures exceeds the limit of memory usage, or searching time increases due to their size. To handle these problems, researchers have worked on developing compact data structures. Such data structures are also called compressed or succinct data structures. They are much smaller than standard data structures, while keeping the same access time to data in theory. However actual performance of such compact data structures for storing Big Data is unknown or unsatisfactory.

The aim of this workshop is to bring together researchers active in the areas of compact data structures to exchange ideas for handling Big Data. We will discuss methods for compressing and storing Big Data. We will also discuss how to design time- and space-efficient data structures for them. Through discussion and sharing of knowledge, we hope to promote collaborations and further improve data structures for Big Data.

Participants

Rajeev Raman	University of Leicester
Venkatesh Raman	The Institute of Mathematical Sciences
Srinivasa Rao Satti	Seoul National University
Ian Munro	University of Waterloo
Moshe Lewenstein	Bar-Ilan University
Gonzalo Navarro	University of Chile
Francisco Claude	Universidad Diego Portales / Akori
Travis Gagie	University of Helsinki
Simon Gog	The University of Melbourne
Jesper Larsson	IT University of Copenhagen
Sebastiano Vigna	Università degli Studi di Milano
Giuseppe Ottaviano	University of Pisa
Ankur Gupta	Butler University
Hiroshi Sakamoto	Kyutech
Shirou Maruyama	Preferred Infrastructure
Tetsuo Shibuya	University of Tokyo
Taku Onodera	University of Tokyo
Takuya Akiba	University of Tokyo
Hiroki Arimura	Hokkaido University
Shin-ichi Minato	Hokkaido University
Takuya Kida	Hokkaido University
Shuhei Denzumi	Hokkaido University
Yasuo Tabei	Japan Science and Technology Agency
Koji Tsuda	AIST
Anish Shrestha	University of Tokyo
Martin Frith	CBRC, AIST
Takeaki Uno	NII
Alexander Bowe	NII
Atsushi Koike	NII
Wing-Kin Sung	National University of Singapore
Kunihiko Sadakane	NII

Schedule

September 27, 2013

9:00 - 10:30	Session 1
Ian Munro	Succinct data structures for representing equivalence classes
Rajeev Raman	Encodings for top-k and range selection
Moshe Lewenstein	Two Dimensional Range Minimum Queries and Fibonacci Lattices

11:00 - 12:00	Session 2
Sebastiano Vigna	Quasi-Succinct Indices
Simon Gog	Integer Alphabet-based Self-Indexes at Terabyte Scale

14:00 - 15:40	Session 3
Travis Gagie	An Alignment-Based Index for Genomic Datasets
Martin Frith	Bio-sequence similarity search with spaced suffix arrays and subset suffix arrays
Alexander Bowe	Succinct de Bruijn Graphs
Taku Onodera	Detecting Superbubbles in Assembly Graphs

16:00 - 18:00	Session 4
	Discussions

September 28, 2013

9:00 - 11:00	Session 5
Srinivasa Rao Satti	Selection from Read-Only Memory with Limited Workspace
Venkatesh Raman	Improved Selection Algorithms for Integers in Read-only Memory and Restore Models
Ankur Gupta	Online Multiselection
Gonzalo Navarro	Document Retrieval on General Sequences

11:30 - 12:30	Session 6
Francisco Claude	Adaptive Data Structures for Permutations and Binary Relations
Giuseppe Ottaviano	Compressed tries and top-k string completion

14:00 - 16:20	Session 7
Anish Shrestha	New Challenges to Processing DNA Data from Modern-day Sequencers
Yasuo Tabei	Succinct data structures for scalable similarity search in Chem-Bioinformatics
Tetsuo Shibuya	Fast Indexing Method for Protein 3-D Structure Searching
Jesper Larsson	Encoding and modeling for set compression
Takuya Akiba	Fast Exact Shortest-Path Distance Queries on Large Networks by Pruned Landmark Labeling

16:20 - 18:00	Session 8
	Discussions

September 29, 2013

9:00 - 11:00	Session 9
Hiroshi Sakamoto	An application of stream compression
Shirou Maruyama	Fully-Online Grammar Compression
Hiroki Arimura	Faster Broad-Word Pattern Matching Algorithms for Regular Expressions and Trees
Takuya Kida	Data Compression using Variable-to-Fixed Length Codes

11:30 - 12:00	Session 10
Shin-ichi Minato	ZDD-Based Representation for Large-Scale Sparse Datasets and Z-Skip-Links for Fast Traversal

September 30, 2013

9:00 - 10:30	Session 11
Shuhei Denzumi	DenseZDD: A Fast and Compact Data Structure for Family of Sets & PathSeqBDD: A DAG Index based on Sequence BDD
Koji Tsuda	Enumeration Algorithms and Statistical Significance
Takeaki Uno	Similarity based Approach for Compression of Noisy Data

10:30 - 12:00	Session 12
	Discussions

Overview of Talks

Succinct Data Structures for Representing Equivalence Classes

Ian Munro, University of Waterloo

Given a partition of an n element set into equivalence classes, we consider time-space tradeoffs for representing it to support the query that asks whether two given elements are in the same equivalence class. This has various applications including for testing whether two vertices are in the same component in an undirected graph or in the same strongly connected component in a directed graph.

Encodings for Top-k and Range Selection

Rajeev Raman, University of Leicester

We study the problem of encoding the positions the top-k elements of an array $A[1..n]$ for a given parameter $1 \leq k \leq n$. Specifically, for any i and j, we wish create a data structure that reports the positions of the largest k elements in $A[i..j]$ in decreasing order, without accessing A at query time. This is a natural extension of the well-known encoding range-maxima query problem, where only the position of the maximum in $A[i..j]$ is sought, and finds applications in document retrieval and ranking. We give (sometimes tight) upper and lower bounds for this problem and some variants thereof.

Two Dimensional Range Minimum Queries and Fibonacci Lattices

Moshe Lewenstein, Bar-Ilan University

Given a matrix of size N, two dimensional range minimum queries (2D-RMQs) ask for the position of the minimum element in a rectangular range within the matrix. We study trade-offs between the query time and the additional space used by indexing data structures that support 2D-RMQs. Using a novel technique—the discrepancy properties of Fibonacci lattices—we give an indexing data structure for 2D-RMQs that uses $O(N/c)$ bits additional space with $O(c \log c (\log \log c)^2)$ query time, for any parameter c, $4 \leq c \leq N$. Also, when the entries of the input matrix are from $\{0, 1\}$, we show that the query time can be improved to $O(c \log c)$ with the same space usage.

Quasi-Succinct Indices

Sebastiano Vigna, Università degli Studi di Milano

Compressed inverted indices in use today are based on the idea of gap compression: documents pointers are stored in increasing order, and the gaps between successive document pointers are stored using suitable codes which represent smaller gaps using less bits. Additional data such as counts and positions is stored using similar techniques. A large body of research has been built in the last 30 years around gap compression, including theoretical modeling of the gap distribution, specialized instantaneous codes suitable for gap encoding, and ad hoc document reorderings which increase the efficiency of instantaneous codes. This paper proposes to represent an index using a different architecture based on quasi-succinct representation of monotone sequences. We show that, besides being theoretically elegant and simple, the new index provides expected constant-time operations and, in practice, significant performance improvements on conjunctive, phrasal and proximity queries.

Integer Alphabet-based Self-Indexes at Terabyte Scale

Simon Gog, The University of Melbourne

We show that succinct data structures can be engineered to support these terabyte-scale data sets. We present a new version of the succinct data structure library (SDSL) which provides the functionality to index large IR data sets. The library provides wavelet trees, compressed suffix arrays, FM-Indexes and compressed suffix trees which support both integer and byte alphabets. All data structures are carefully engineered to support space and time efficient construction and fast runtime performance.

An Alignment-Based Index for Genomic Datasets

Travis Gagie, University of Helsinki

With current hardware and software, a standard computer can now hold in RAM an index for approximate pattern matching on about half a dozen human genomes. Sequencing technologies have improved so quickly, however, that scientists will soon demand indexes for thousands of genomes. Whereas most researchers who have addressed this problem have proposed completely new kinds of indexes, we recently described a simple technique that scales standard indexes to work on more genomes. Our main idea was to filter the dataset with LZ77, build a standard index for the filtered file, and then create a hybrid of that standard index and an LZ77-based index. In this talk we describe how to our technique to use alignments instead of LZ77, in order to simplify and speed up both preprocessing and random access.

Bio-sequence Similarity Search with Spaced Suffix Arrays and Subset Suffix Arrays

Martin Frith, CBRC, AIST

For many decades, the main way of analyzing biological sequences has been by comparing and aligning them. This remains true today. Modern tasks include: comparing whole genomes; aligning bisulfite-converted DNA reads to a genome; aligning long, high-error sequences from single molecule sequencers; aligning ancient or degraded DNA; comparing metagenomic DNA to a protein database. Over the decades, statistically powerful alignment techniques have been developed. These include: log likelihood ratio scoring matrices, pair hidden Markov models, and probabilistic alignment. Unfortunately, these methods

are rarely used with modern deep sequencing data, perhaps because of a mistaken belief that they are too slow. This talk will demonstrate that they can be made fast enough, and that they offer great benefits. A statistical approach is especially useful in cases that benefit more from statistical modeling, such as: larger divergence between sequences, extreme AT-richness (e.g. DNA from Plasmodium or Dictyostelium), and bisulfite-converted DNA.

Succinct de Bruijn Graphs

Alexander Bowe, NII

We propose a new succinct de Bruijn graph representation. If the de Bruijn graph of k-mers in a DNA sequence of length N has m edges, it can be represented in $4m+o(m)$ bits. This is much smaller than existing ones. The numbers of outgoing and incoming edges of a node are computed in constant time, and the outgoing and incoming edge with given label are found in constant time and $O(k)$ time, respectively.

Detecting Superbubbles in Assembly Graphs

Taku Onodera, University of Tokyo

We introduce a new concept of a subgraph class called a superbubble for analyzing assembly graphs, and propose an efficient algorithm for detecting it. Most assembly algorithms utilize assembly graphs like the de Bruijn graph or the overlap graph constructed from reads. From these graphs, many assembly algorithms first detect simple local graph structures (motifs), such as tips and bubbles, mainly to find sequencing errors. These motifs are easy to detect, but they are sometimes too simple to deal with more complex errors. The superbubble is an extension of the bubble, which is also important for analyzing assembly graphs. Though superbubbles are much more complex than ordinary bubbles, we show that they can be efficiently enumerated. We propose an average-case linear time algorithm (i.e., $O(n+m)$ for a graph with n vertices and m edges) for graphs with a reasonable model, though the worst-case time complexity of our algorithm is quadratic (i.e., $O(n(n+m))$).

Selection from Read-Only Memory with Limited Workspace

Srinivasa Rao Satti, Seoul National University

In the classic selection problem the task is to find the kth smallest of N elements. We study the complexity of this problem on a space-bounded random-access machine: The input is given in a read-only array and the capacity of workspace is limited. We prove that the linear-time prune-and-search algorithm—presented in most textbooks on algorithms—can be adjusted to use $O(N)$ bits instead of $\Theta(N)$ words of extra space. Prior to our work, the best known algorithm by Frederickson could perform the task with $O(N)$ bits of extra space in $O(N \log^* N)$ time. In particular, our result separates the space-restricted random-access model and the multi-pass streaming model (since we can bypass the $\Omega(N \log^* N)$ lower bound known for the latter model).

Improved Selection Algorithms for Integers in Read-only Memory and Restore Models

Venkatesh Raman, The Institute of Mathematical Sciences

We consider the classical selection and sorting problems in a model where the initial permutation of the input has to be restored after completing the computation. While the requirement of the restoration is stringent compared to the classical versions of the problems, this model is more relaxed than a read-only memory where the input elements are not allowed to be moved within the input array. We first show that for a sequence of n integers, selection (finding the median or more generally the k-th smallest element for a given k) can be done in $O(n)$ time using $O(lgn)$ words of extra space in this model. In contrast, no linear-time selection algorithm is known which uses polylogarithmic space in the read-only memory model.

Online Multiselection

Ankur Gupta, Butler University

We introduce a new online algorithm for the multiselection problem which performs a sequence of selection queries on a given unsorted array. We show that our online algorithm is 1-competitive in terms of data comparisons. In particular, we match the bounds (up to lower order terms) from the optimal offline algorithm proposed by Kaligosi et al. [ICALP 2005]. We provide experimental results comparing online and offline algorithms. These experiments show that our online algorithms require fewer comparisons than the best-known offline algorithms. Interestingly, our experiments suggest that our optimal online algorithm (when used to sort the array) requires fewer comparisons than both quicksort and mergesort.

Document Retrieval on General Sequences

Gonzalo Navarro, University of Chile

Document retrieval is one of the best established information retrieval activities since the sixties, pervading all search engines. Its aim is to obtain, from a collection of text documents, those most relevant to a pattern query. Current technology is mostly oriented to "natural language" text collections, where inverted indices are the preferred solution. As successful as this paradigm has been, it fails to properly handle some East Asian languages and other scenarios where the "natural language" assumptions do not hold. In this survey we cover the recent research in extending the document retrieval techniques to a broader class of sequence collections, which has applications bioinformatics, data and Web mining, chemoinformatics, software engineering, multimedia information retrieval, and many others. We focus on the algorithmic aspects of the techniques, uncovering a rich world of relations between document retrieval challenges and fundamental problems on trees, strings, range queries, discrete geometry, and others.

Adaptive Data Structures for Permutations and Binary Relations

Francisco Claude, Universidad Diego Portales / Akori

We present new data structures for representing binary relations in an adaptive way, that is, for certain classes of inputs we achieve space below the general information theoretic lower bound, while achieving reasonable space complexities in the worst case. Our approach is derived from a geometric data structure [Arroyuelo et al., TCS 2011]. When used for representing permutations, it converges to a previously known adaptive representation [Barbay and Navarro, STACS 2009]. However, this new way of approaching the problem shows that we can support range searching in the adaptive representation. We extend this approach to representing binary relations, where no other adaptive representations using this chain decomposition have been proposed.

Compressed Tries and Top-k String Completion

Giuseppe Ottaviano, University of Pisa

In this talk we will discuss space-efficient representations of string sets. After a brief review of succinct tree representations, we will show how to use path decomposition trees of the compacted trie of a string set to obtain a space-efficient and fast string set representation. We will then show how to adapt the technique to scored string sets, where each string is assigned a score, in order to support top-k completion queries, that is, given a query string p, return the k highest-scored strings that are prefixed by p.

New Challenges to Processing DNA Data from Modern-day Sequencers

Anish Shrestha, University of Tokyo

Aligning short DNA sequences ("reads") to a reference genome is usually one of the first steps in processing DNA data from modern-day sequencers. This task is likely to get more complicated because of several developments: (1) reads are getting longer, whereas most of the widely-used methods today are customized for short reads, (2) while sequencers are capable of longer reads, they still remain extremely error-prone, (3) multiple reference genomes are available for some species, and it is desirable to incorporate this extra information during alignment, (4) sequencer throughput is increasing, (5) there are increasingly many species whose reference genomes are not available.

Succinct Data Structures for Scalable Similarity Search in ChemBioinformatics

Yasuo Tabei, Japan Science and Technology Agency

Analyzing functional interactions between small compounds and proteins is indispensable in genomic drug discovery. Since rich information on various compound-protein interactions is available in recent molecular databases, strong demands for making best use of such databases require to invent powerful methods to help us find new functional compound-protein pairs on a large scale. We present succinct data structures that efficiently performs similarity search in databases for chemical compounds and compound-protein pairs with respect to both binary fingerprints and real-valued properties. Our methods achieve both time and space efficiency by developing the data structures called succinct multibit trees and interval-splitting trees, which efficiently prune the useless portions of search space. We experimentally test our methods on the ability to retrieve similar compounds and compound-protein pairs for a query from large databases with over 30 million compounds and 200 million compound-protein pairs and show that our methods perform significantly better than other possible approaches.

Fast Indexing Method for Protein 3-D Structure Searching

Tetsuo Shibuya, University of Tokyo

Searching for protein structure-function relationships using three-dimensional (3D) structural coordinates represents a fundamental approach for determining the function of proteins with unknown functions. In this talk, we introduce a new indexing algorithm for for fast protein 3-D structure similarity queries. The new method runs in $O(m + N/m^{0.5})$ time, after $O(N \log N)$ preprocessing, where N is the database size and m is the query length. It is about 2 to 50 times faster than the previous practically best-known $O(N)$ algorithm, which was also proposed by us, even if we include the preprocessing time. It is almost 20-1000 times faster than the naive comparison algorithm, according to experiments on a huge SCOP database.

Encoding and Modeling for Set Compression

Jesper Larsson, IT University of Copenhagen

This talk addresses the compressing unordered sets of distinct items. Specifically, some available options for defining a context for probabilistic coding is discussed. Existing methods for compressing sets of integers in a fixed range (or equivalently, fixed-length bitstrings) are recapitulated, and a novel method that allows use of statistics for individual elements described.

Fast Exact Shortest-Path Distance Queries on Large Networks by Pruned Landmark Labeling

Takuya Akiba, University of Tokyo

We propose a new exact method for shortest-path distance queries on large-scale networks. Our method precomputes distance labels for vertices by performing a breadth-first search from every vertex. Seemingly too obvious and too inefficient at first glance, the key ingredient introduced here is pruning during breadth-first searches. While we can still answer the correct distance for any pair of vertices from the labels, it surprisingly reduces the search space and sizes of labels. Moreover, we show that we can perform 32 or 64 breadth-first searches simultaneously exploiting bitwise operations. We experimentally demonstrate that the combination of these two techniques is efficient and robust on various kinds of large-scale real-world networks. In particular, our method can handle social networks and web graphs with hundreds of millions of edges, which are two orders of magnitude larger than the limits of previous exact methods, with comparable query time to those of previous methods.

An Application of Stream Compression

Hiroshi Sakamoto, Kyutech

Our research group propose the framework of stream compression for speeding-up network migration using grammar compression. We design algorithm and prove its upper/lower bound expressiveness.

Fully-Online Grammar Compression

Shirou Maruyama, Preferred Infrastructure

We present a fully-online algorithm for constructing straight-line programs (SLPs). A naive array representation of an SLP with n variables on an alphabet of size σ requires $2n\lg(n+\sigma)$ bits. As already shown in [Tabei et al., CPM13], in offline setting, this size can be reduced to $n\lg(n+\sigma)+2n+o(n)$, which is asymptotically equal to the information-theoretic lower bound. Our algorithm achieves the same size in online setting, i.e., characters of an input string are given one by one to update the current SLP.

Faster Broad-Word Pattern Matching Algorithms for Regular Expressions and Trees

Hiroki Arimura, Hokkaido University

In this talk, we present the recent progress of bit-parallel pattern matching algorithms for regular expressions and trees. We focus on a variant of bit-parallel approaches called broad-word pattern matching, which uses Boolean and arithmetic operations on w-bit registers based on parallelism inside a computer word. We show efficient algorithms for the regular expression matching problem for expressions with small height, and variations of the tree matching and inclusion problems under many-to-one matching.

Data Compression using Variable-to-Fixed Length Codes

Takuya Kida, Hokkaido University

A VF code is an encoding scheme that uses a fixed-length code, and thus, one can easily access the compressed data. However, conventional VF codes usually have an inferior compression ratio to that of variable-length codes. We propose a new VF coding method that applies a fixed-length code to the set of rules extracted by the Re-Pair algorithm. We also present several experimental results to show that the proposed coding achieves comparable performance to well-known compression tools based on variable-length codes.

ZDD-Based Representation for Large-Scale Sparse Datasets and Z-Skip-Links for Fast Traversal

Shin-ichi Minato, Hokkaido University

This talk gives a brief explanation of the data structures called BDD (Binary Decision Diagram) and ZDD (Zero-suppressed BDD), for efficiently manipulating large-scale sparse datasets. We also discuss "Z-SkipLink," an additional data structure for fast traversal of ZDDs.

DenseZDD: A Fast and Compact Data Structure for Family of Sets & PathSeqBDD: A DAG Index based on Sequence BDD

Shuhei Denzumi, Hokkaido University

Zero-suppressed Binary Decision Diagram (ZDD) is a data structure to manipulate families of sets. Current ZDDs require a huge amount of memory and membership operations are slow. We introduce DenseZDD, a compressed index for a static ZDD. Our technique not only indexes set families compactly but also executes fast membership operations. Sequence Binary Decision Diagram (SeqBDD) is a data structure to manipulate sets of strings. There have been many indexes for linear texts. However, indexes for directed acyclic graphs (DAGs) have not been studied enough. We introduce PathSeqBDD, a complete inverted file for a DAG based on SeqBDD. Our technique allows us to retrieve frequencies and occurrences of patterns for given DAGs.

Enumeration Algorithms and Statistical Significance

Koji Tsuda, AIST

More than three transcription factors often work together to enable cells to respond to various signals. The detection of combinatorial regulation by multiple transcription factors, however, is not only computationally nontrivial but also extremely unlikely because of multiple testing correction. The exponential growth in the number of tests forces us to set a strict limit on the maximum arity. Here, we propose an efficient branch-and-bound algorithm called the "limitless arity multiple-testing procedure" (LAMP) to count the exact number of testable combinations and calibrate the Bonferroni factor to the smallest possible value. LAMP lists significant combinations without any limit, whereas the family-wise error rate is rigorously controlled under the threshold. In the human breast cancer transcriptome, LAMP discovered statistically significant combinations of as many as eight binding motifs. This method may contribute to uncover pathways regulated in a coordinated fashion and find hidden associations in heterogeneous data.

Similarity based Approach for Compression of Noisy Data

Takeaki Uno, NII

We approach genome sequence compression from the use of similarity. The noisy data such as genome sequence does not accept usual compression techniques, such as word frequency and run length. However, those data often contain long or short similarity much inside. So, we extract these similarity and represent the data by difference from similar ones. By the computational experiments for genome strings, we show that our method is very efficient compared to the existing best method.

Discussion

On September 27th, 2013, we have a 2-hour discussion session. We partition our people into three groups, every group discuss three topics on compact data structures for big data. The topics are: (1) Theory aspect on compact data structure, (2) Practical implementation aspect on compact data structure, and (3) Applications on compact data structure. After that, the whole team came together and summarized the finding.

From the feedback of the participants, people tends to work on one single topic in the past and they got to know better the relationship among the three topics. Below, we summarize the points we discussed.

(1) Theory aspect: We discussed what are the new problems on compact data-structures. It seems that we don't have some good data-structures for compact set. It may be an interesting problem to work on in the future. People also asked if there is any compact way to store prime numbers.

(2) Practical aspect: We discussed how to deliver the practical implementations of different compact data-structures. It may be good if we can make a standard package for people to download do use. Another issue is on how to give credit to practical implementations. Currently, there is little conferences for practical implementations.

(3) Application aspect: Compact data-structure finds a lot of applications in bioinformatics. For example, BWA and Bowtie successfully applied BWT (or FM-index) to align short reads on the reference genome. However, there is not enough communication between application people and theory people. More discussions are needed to build the connection.