Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 190 (2021) 701–705

Procedia Computer Science

www.elsevier.com/locate/procedia

2020 Annual International Conference on Brain-Inspired Cognitive Architectures for Artificial Intelligence: Eleventh Annual Meeting of the BICA Society

On the Euclidian Metric for Undirected Graphs and Exact Calculations

Victor A. Rusakov[a,*]

[a]*National Research Nuclear University MEPhI (Moscow Engineering Physics Institute), 31 Kashirskoe shosse Moscow 115409 Russia*

Abstract

The Euclidian metric gives better results when organizing a multi-agent interaction environment. The analytical basis for this metric is the matrix inverse to a simple matrix description of an undirected graph. In many applied tasks, the usual matrix inversion and real numbers in the framework of finite bit-depth calculations are quite sufficient. However, the unweighted undirected graph is a discrete object, and traditional metrics are able to support processing in the field of rational numbers. Here we show that the Euclidian metric has the same property. Moreover, the space with the dot product is much richer in possibilities in comparison to the spaces where only norms are introduced. Here these possibilities are at the heart of a simple algorithm for calculating the rational entries of the required inverse matrix. Also in the Euclidian space one can use the most important relationship between its elements - orthogonality. The results of numerical experiments are presented.

Keywords: graph metrics; dot product; the Euclidian graph metric; the Moore-Penrose pseudo inverse of the incidence matrix; the Gram matrix

* Corresponding author.
E-mail address: VARusakov@mephi.ru

1877-0509
10.1016/j.procs.2021.06.105

1. Introduction

Graphs are widely used as a model of real objects. In the analysis and synthesis of graphs a reasonable combination of heuristic and strict approaches is used. In practice, this entails the use of graph metrics. Usual shortest paths and minimal cuts are widely used as part of the various procedures for the analysis and synthesis of graphs. The Euclidian metric gives better results in such problems [1,2,3].

From now on a graph will be understood as a finite unweighted undirected connected graph without loops and multiple edges having k vertices and h edges.

Let X and Y be vector spaces with dimensions h and k correspondingly over the field of real numbers. The arbitrarily introduced *orientations* of the edges in the absence of limits to their capacity turns the incidence matrix A of an undirected graph having ±1 as non-zero elements in each column into a useful instrument for the *linear* transformation $Ax=y$ of the edge flows $x\in$ X into the vertex flows $y\in$ Y. From now on the incidence matrix A will be understood to be such a matrix.

Further «′» и «$^+$» symbolize a transposition and the Moore-Penrose pseudo inverse correspondingly, d_i stands for the degree of the i-th vertex, $d_{\max}=\max_{1\le i\le k} d_i$, $p=(d_{\max}+c_n)^{-1}$ where $0\le c_n<\infty$. In the matrix $P=(p_{uv})$ let the entry $p_{ij}=0$ when there is no edge (i,j) in the graph, $p_{ij}=p$ when there is the edge (i,j), $i\neq j$, and $p_{ii}=1-pd_i$. Matrix P is symmetrical even for an irregular graph. Its entries are the transition probabilities of the ergodic Markov chain. The flow representations of the metric are not dependent on the renormalizations making such a symmetry, although this sometimes results in a non-zero diagonal P [1,4,5,6].

The matrix $A^+=pA'Z$, where $Z=(I-P+F)^{-1}$ is the $k\times k$ positive definite [3] matrix, I is an identity matrix, and each entry of the matrix F is equal to k^{-1}, serves as the basis for obtaining the Euclidian metric characteristics of the graph [1,2,3,4,5,6].

The corresponding normalizations of X and Y lead to a quantitative rational description of ordinary shortest paths and minimal cuts [4,5,6]. The Euclidian *norm* formally means the use of the square root in similar descriptions of extreme (but now in a quadratic sense) flows in X and Y. Here, however, we note that all such quantities in their representations are not important in themselves. Actually, it is important to include them in the contexts of specific tasks in the right way.

Here and below let N(H) and R(H) mean the H kernel and the H image of an arbitrary matrix H, and θ_X and θ_Y mean the zero vectors of the spaces X и Y, respectively. Also let (u,w) mean the usual dot product of vectors u and w.

There is a graph class called a tree that is noteworthy for its simplicity. For any tree N(A)=θ_X, and for any pair of its vertices there is only *one* path between them. We will use the simplicity of trees here to avoid *explicit* matrix inversion. This means that when the matrix Z is obtained, the processing of integers will be performed each time over the *final* entries of the matrices Z of the *intermediate* graphs. But not over the fast-growing *intermediate* values when calculating the *final* matrix Z.

Of course, the bottleneck of bit-finite arithmetic isn't going anywhere. It manifests itself in the existence of graphs of a special type. We will focus on that in a separate section.

At the end, numerical experiments are described.

2. Lemmas and algorithm

Let's consider two lemmas. They mark out those properties of matrices that will be used further.

Lemma1. Let the value p be the normalizing factor in the explicit representation $A^+=pA'Z$, $Z=(z_{ij})$ [1,2,3,4,5,6]. Then for $\forall\ i,j$ the value of $p(z_{ij}-k^{-1})$ does not change when p changes in its normalization range $0<p\le d_{\max}^{-1}$.

Proof. By [1,4,5], with the described changes in p, the quantities $pn_{sr}^{(t)}$ for $\forall\ s,r\neq t$ remain unchanged, where $n_{sr}^{(t)}$ is the s,r-th entry of the fundamental matrix N_t of the absorbing Markov chain. Multiply both sides of the well-known [7] equality $Z-F=(I-F)N^*(I-F)$ by p. Here N^* is $N_t, \forall\ t$, with the zero row and column inserted into the t-th places. Each entry on the right side of the resulting matrix equality is a linear combination of unchangeable values having constant coefficients, so that each entry of the matrix $p(Z-F)$ on the left is also unaltered. □

Lemma2. Let the i-th entry of the vector $y_i\in$ R(A) be equal to $1-k^{-1}$, and all its other entries equal to $-k^{-1}$. The matrix $p(Z–F)$ is the Gram matrix of the vectors A^+y_i, i=1,…,k.

Proof. $Z\xi = \xi$, where each entry of ξ is equal to 1 [7]. Further, $(A^+u, A^+w) = (u, pZw)$ for $\forall\ u,w\in$ R(A) by Lemma2 in [3]. Hence immediately $(A^+y_l, A^+y_q) = p(z_{lq}–k^{-1})$; l,q=1,…,k. □

The steps and components of the algorithm for calculating the rational matrix Z

1. A connected graph contains at least one spanning tree. Selecting *any* spanning tree from the graph presents no difficulties [8].
2. The defining property of the tree makes it possible to quickly and unambiguously find *integer* edge flow x_s that delivers a flow of magnitude 1 from the vertex s to *each* of the other vertices of the spanning tree, s = 1,…, k.
3. By Lemma1, the quantities $p(z_{lq}–k^{-1})$ do not change if we put $p = k^{-1}$. Further, the vertex flow y_s corresponding to the edge flow x_s of the previous item is k times larger than the similar vertex flow of Lemma2 above, s = 1,…, k. Hence, by Lemma2, we obtain a *rational* representation $z_{lq} = k^{-1}((x_l, x_q) +1)$; l,q = 1, ..., k of any entry of the matrix Z. This matrix corresponds to the spanning tree selected.
4. During the *reconstruction* of the original graph, you can *add* edge by edge to this spanning tree in *any* order. Each addition is accompanied by a correction of the entries of the current rational matrix Z [9,10]. The well-known [11] method with matrices of rank 1 is used.
5. The traditional Euclidian binary algorithm is used to search for the GCD in the usual way.

Remark. The original *connected* graph may differ from some graph with the same number of k vertices and a known rational (integer) matrix Z by the presence and/or absence of some edges. For example, for a complete graph $Z = I$ if $p = k^{-1}$. Instead of the spanning tree and its matrix Z, you can always use any other such reconstruction start point. As before, the reconstruction starts by adding all the missing edges. It ends with the removal of those edges that are present in the graph with the initially known matrix Z, but are absent in the original graph. In this case, the sequence of corrections Z is always feasible [9,10].

The edge flow x_a with a minimal Euclidian norm is $x_a = A^+y_a = pA'Zy_a$, $\forall\ y_a\in$ R(A). Apart from the obvious A', let all the other factors in the last product, that is, p, Z, and y_a, be rational. It is clear that the resulting quantitative descriptions of the entries of such an edge flow x_a are rational. For example, descriptions of entries of flows between any pairs of vertices used in reliability analysis [1,4] are rational. Or descriptions of entries of flows of colour t, t = 1,..., k, which are used in the analysis of the throughput [2,4]. In general, by $(A^+u, A^+w) = (u, pZw)$, $\forall\ u,w\in$ R(A), for rational u,w this dot product is rational. The formed objective functions $p(\mathrm{Sp}Z–1)$ of the synthesis [1,2] are also obviously rational.

Then we note the following. For $p = k^{-1}$, the entries of the symmetric matrix $I–P+F$ will be integer after the factor k^{-1} is taken out of the brackets. Its i-th diagonal entry is equal to d_i+1. Its u,v-th ($u\neq v$) entry is equal to 0 in the presence (!) of the edge (u,v) in the graph and 1 in its absence. Therefore, for inversion of such a matrix, one can use powerful universal approaches with all their advantages [12,13,14] and excesses for our task.

3. The sequence of graphs

The algorithm described in brief above allows one to immediately calculate the rational entries of the matrices Z without having to face rapidly growing values during intermediate *calculations* for each of such matrix. These matrices themselves correspond to some intermediate *graphs*. Thus, the problem of the finiteness of the number of bits can be formulated as follows. What should be the sequence of graphs with index k for which this constraint quickly becomes a bottleneck?

We consider an example of such a sequence below. For each of its graphs (we called it a *dandelion*), a flow with the minimum Euclidian norm is described that delivers one unit of flow from vertex s to vertex t. The beginning of the sequence is shown in Fig.1.

The largest part of the flow, that is, $p(n_{ss}^{(t)} - n_{st}^{(t)})$, immediately enters t from s, the smallest $f_{rt} = p(n_{sr}^{(t)} - n_{st}^{(t)})$ – from r. By definition, $n_{st}^{(t)} = 0$. The largest part of the flow is $pn_{ss}^{(t)} \approx 0.618$, and it practically does not change with increasing k. At the same time, f_{rt} decreases very quickly.

It can be shown [15] that u_k, the *inverse* of this smallest part of the flow, is a member of the following recurrent sequence $u_k = 3u_{k-1} - u_{k-2}$; $u_2 = 1$, $u_3 = 3$. The integer numerator of the smallest part is at least 1. Therefore, a rapid decrease in the smallest part of the flow with an increase in k means no less rapid increase in its (hence, the common) integer denominator.

It is not difficult to obtain the general form of the k-th term of this sequence using the well-known technique [16]. It looks like $5^{-1/2}(\alpha^{k-1} - \beta^{k-1})$, where $\alpha = 2^{-1}(3+5^{1/2})$ and $\beta = 2^{-1}(3-5^{1/2})$.

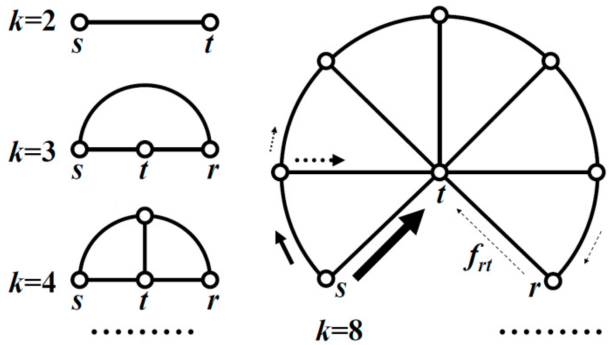

Fig.1. The sequence of graphs

4. Numerical Experiments

Numerical experiments were carried out to verify theoretical correspondences. Calculations were performed using CVF 6.6.c. The number of bits was limited to 64. Let's describe the experiments with *dandelions* here.

A graph is generated for a given number of k vertices. Its integer matrix Z is calculated together with a common integer denominator of its entries at $p=k^{-1}$. Overflow control is performed. The value f_{rt} of the flow along the edge (r,t) is calculated, see Fig.1.

Inverse values f_{rt}^{-1} are calculated using the described recurrent sequence.

The same values of the inverses to f_{rt} are calculated on the basis of an explicit representation in terms of the powers of the roots of the characteristic equation with subsequent transformation to the nearest integer. With this processing, starting from $k = 2$, discarding the term with a base less than 1 (that is, β^{k-1}) does not affect the result.

The described values for some values of k are collected in Table1.

Table1. Parameters of graphs

k	$n_{sr}^{(t)}$	$k \times$ denominator	f_{rt}^{-1}
2	–	–	1
3	–	–	3
4	1	8	8
8	8	3016	377
12	12	212532	17711
15	5	1589055	317811
18	18	102651966	5702887
20	20	781763380	39088169
22	11	2947057256	267914296
24	–	–	1836311903
27	–	–	32951280099
30	–	–	591286729879
35	–	–	72723460248141
40	–	–	8944394323791464

Fig.2 shows the change of f_{rt}^{-1} using a logarithmic scale along the vertical axis. Note $f_{rt}^{-1} = k\times$denominator / $n_{sr}^{(t)}$ by definition. Also note $n_{sr}^{(t)} - n_{st}^{(t)} = z_{sr} - z_{st} + z_{tt} - z_{tr}$ and, for example for k=22, we get 11= –133957137–0+133957148–0 in the numerator's values. So a small value of f_{rt} in terms of entries of Z simply means the proximity of their moduli with a difference in sign and an increase in the common denominator.

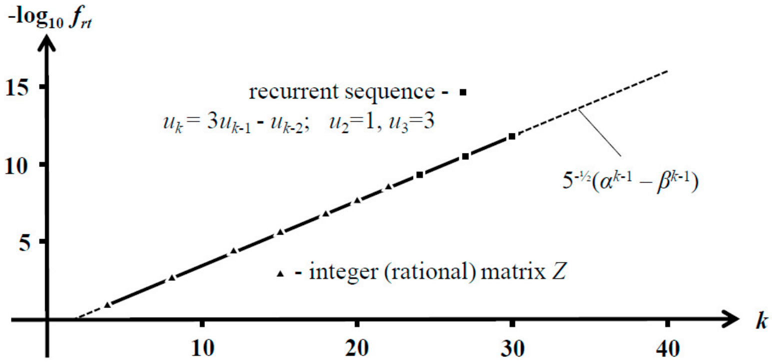

Fig.2. The description of the graph sequence

5. Conclusion

Like traditional graph metrics, the more powerful Euclidian metric supports processing in the field of rational numbers. An algorithm for calculating the rational entries of the explicit representation of the Moore-Penrose pseudo inverse of the incidence matrix of an undirected graph is described. The sequence of graphs is described for which a finite bit-depth quickly becomes a bottleneck. Numerical experiments have been carried out, which have confirmed the theoretical propositions.

References

[1] Rusakov V.A. (2020) "Using Metrics in the Analysis and Synthesis of Reliable Graphs", Samsonovich, A.V. (Ed.). BICA 2019. Advances in Intelligent Systems and Computing, vol. 948. Cham, Switzerland: Springer, pp 438-448. DOI: 10.1007/978-3-030-25719-4_58

[2] Rusakov V.A. (2021) "Using Metrics in the Throughput Analysis and Synthesis of Undirected Graphs", Antipova T. (Ed.): ICIS 2020, Lecture Notes in Networks and Systems, vol. 136. Cham, Switzerland: Springer, pp. 277–287. DOI: 10.1007/978-3-030-49264-9_25

[3] Rusakov V.A. (2021) "On the Regularity of the Bias of Throughput Estimates on Traffic Averaging", Samsonovich A.V. et al. (Eds.). BICA 2020. Advances in Intelligent Systems and Computing, vol. 1310. Cham, Switzerland: Springer, pp 425–434. DOI: 10.1007/978-3-030-65596-9_51

[4] Rusakov V.A. (1979) *Analysis and Synthesis of Computer Network Structures. Part 1. Analysis*, Moscow Engineering Physics Institute Report: VNTI Centre No Б796153, p 122

[5] Rusakov V.A. (2020) "On Markov Chains and Some Matrices and Metrics for Undirected Graphs", Antipova T. (Ed.). ICIS 2019. Lecture Notes in Networks and Systems, vol. 78. Cham, Switzerland: Springer, pp 340-348. DOI: 10.1007/978-3-030-22493-6_30

[6] Rusakov V.A. (2018) "Matrices, shortest paths, minimal cuts and Euclidian metric for undirected graphs", Procedia Computer Science, **145** (2018), pp 444-447. DOI: 10.1016/j.procs.2018.11.104

[7] Kemeny J. and Snell J. (1960) *Finite Markov Chains*. University series in undergraduate mathematics, Princeton NJ: Van Nostrand, p 210

[8] Christofides N. (1975) *Graph theory: an algorithmic approach*, London-New York: Academic Press, p 400

[9] Rusakov V.A. (1978) "Implementation of the methodology for analysis and synthesis of computer network structures using Markov chains", Engineering-mathematical methods in physics and cybernetics, Issue 7, Moscow: Atomizdat, pp 41-45

[10] Rusakov V.A. (2011) "Reconstruction of the Euclidean metric of an undirected graph by metrics of components", Natural and technical sciences **2** (**52**), Moscow: Sputnik+, pp 22-24

[11] Faddeev D.K., Faddeeva V.N. (1963) *Computational methods of linear algebra*, Moscow-Leningrad: Fizmatgiz, p 734

[12] Dixon John D. (1982) "Exact solution of linear equations using p-adic expansions", Numer. Math. **40**, pp 137-141

[13] Wan Zhendong (2006) "An algorithm to solve integer linear systems exactly using numerical methods", Journal of Symbolic Computation, **41** (2006), pp 621–632. DOI:10.1016/j.jsc.2005.11.001

[14] Saunders B.D., Wood D.H., Youse B.S. (2011) "Numeric-symbolic exact rational linear system solver", *ISSAC'11*, ACM, 2011

[15] Rusakov V.A. (1999) "On the accuracy of the representation of mutually inverse matrices for the description of undirected graphs", Proceedings of the VIII International Scientific and Technical Seminar, Moscow-Alushta: MAI, pp 131-132

[16] Markushevich A.I. (1950) *Recurrent sequences*, Moscow-Leningrad: GITTL, p 52