The similarity-aware relational division database operator

André S. Gonzaga
Advisor: **Robson L. F. Cordeiro**

[1]Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo (USP)

asgonzaga@usp.br, robson@icmc.usp.br

Abstract. *This paper describes the motivation, contributions and impact of the MSc. dissertation that proposes the first Similarity-aware Division ($\hat{\div}$) database operator. The novel operator is naturally well suited to answer queries with an idea of "candidate elements and exigencies" to be performed on complex data from real applications of high-impact, such as in agriculture, genetics, industrial production, digital libraries and enterprise management.*

1. Introduction

In the Relational Algebra [Codd 1972], the operator of Division ($\div$) is an intuitive tool to write queries involving the concept of "for all", and thus, it is constantly required in many real applications. For example, it answers the queries as follows: (a) "*What products have **all** the requirements of the industrial quality control?*", (b) "*What students were approved in **all** Database-related courses?*", (c) "*What cities have **all** the requirements to produce a given type of crop?*". However, we demonstrate in this work that the Relational Division cannot support many of the needs common to modern applications, particularly, those that involve complex data analysis, such as processing images, audio, long texts, fingerprints, large graphs, and several other "nontraditional" data types. While investigating the problem, we found out that the main limitation is the existence of intrinsic comparisons of attribute values in the Relational Division, which, by definition, are always performed by *identity* ($=$), despite that in **most** cases complex data must be compared by *similarity*.

Let us use our example Query (c) with cities and crop requirements to exemplify the limitations of the division. Figures 1a, 1b and 1c illustrate a toy dataset for this query. Relation `CityRegions` describes three cities, i.e., the candidates for the crop production, each one represented by a set of regions identified by textual tags. For example, the city of Campinas contains regions with water, urban areas, silos and roads. Relation `Requirements` describes the needs to produce the crop. In this example, we assume that water, bare soil, silos and urban areas are required. The result of dividing `CityRegions` by `Requirements` is relation `Cities`. It contains the list of cities considered appropriate to produce the crop, that is, those cities that have a region tag *identical* ($=$) to each tag in `Requirements`. In this particular case, only the city of `São Carlos` satisfies all requirements.

The example in Figures 1a, 1b and 1c clearly indicates that the relational division is well suited to answer this kind of query; note, however, that it hardly depends on two facts: (i) only data in scalar domains exist in the relations, and; (ii) to compare these data by identity is appropriate. Unlike, a more realistic example at the bottom of Figure 1 illustrates a toy dataset in which the concept of division is also required, but the existing operator cannot handle. In this second case, relation `CityRegions` also describes three cities, i.e., the candidates to produce the crop, each one represented by

a set of satellite images taken from regions of the city. Note that attribute `Region` has now a complex data type (image) in relations `CityRegions` and `Requirements`, but the semantics of both the query and the data remain the same. For example, the city of Campinas still contains regions with water, urban areas, silos and roads, and we still look for cities with the requirements water, bare soil, silos and urban areas. *Only the data types were modified*. Despite this fact, the Relational Division is now unsuitable to validate the crop's needs, because it is virtually impossible to have any pair of *identical* ($=$) image tiles that come from distinct locations. In fact, it is imperative to compare the data by *similarity*, so to spot distinct – but similar – tiles of the same requirement, such as the tiles of water highlighted with double asterisks in Figures 1d and 1e. In our notation, we use the symbol $\hat{=}$ to refer to similarity comparison.

In this MSc work, we identified severe limitations on the usability of the Relational Division to process complex data, and tackled the problem by extending it into the new Similarity-aware Division ($\hat{\div}$) database operator. As opposed to the existing division, our new operator supports similarity-based attribute comparisons and it is naturally well suited to answer queries with an idea of "candidate elements and exigencies" to be performed on complex data from real applications of high-impact. For example, we demonstrated in case studies that the similarity-aware division has applications in genetics and agriculture, and we also discussed – see Chapter 7 of the Dissertation – how it may be helpful in digital library search, industrial quality control and even to identify prospective clients for enterprises. Besides designing and validating the operator in real data, we also formally defined the similarity-aware division and carefully designed two fast and scalable algorithms for it.

2. Basic Concepts and Related Work

In Relational Algebra [Codd 1972], the Division ($\div$) allows simple and intuitive representations for queries with the concept of "*for all*". In fact, it is the **only** algebraic operator that directly corresponds to the Universal Quantification ($\forall$) from the Relational Calculus [Codd 1972]. The division is expressed by $\mathsf{T}_1\ [\mathsf{L}_1 \div \mathsf{L}_2]\ \mathsf{T}_2 = \mathsf{T}_R$. In the equation, T_1, T_2 and T_R are relations that refer to the dividend, the divisor and the quotient, respectively. L_1 and L_2 are lists of attributes from T_1 and T_2, in that order. Both lists must have the same number of attributes, and each attribute in L_1 must be union-compatible with its counterpart in L_2. The quotient relation T_R has all the attributes of T_1 except for those ones listed in L_1. That is, the schema of T_R is given by the relative complement $\overline{\mathsf{L}_1}$ of list L_1 with respect to the schema of T_1, i.e., $Sch(\mathsf{T}_R) = \overline{\mathsf{L}_1} = Sch(\mathsf{T}_1) - \mathsf{L}_1$. The instance of T_R is the subset of $\pi_{\left(\overline{\mathsf{L}_1}\right)}(\mathsf{T}_1)$ with the largest possible cardinality, such that $\mathsf{T}_R \times \mathsf{T}_2 \subseteq \mathsf{T}_1$.

Many researchers have been proposing strategies to support similarity comparison in Relational Database Management Systems — RDBMS [Silva et al. 2015, Pola et al. 2015, Budíková et al. 2012, Belohlavek and Vychodil 2010], commonly by extending Relational Operators. The vast majority of them focuses on the Selection [Silva et al. 2013] in which similarity awareness is achieved by means of range queries, nearest neighbors queries, and their many variants. Recent works also focus Grouping and Aggregation [Tang et al. 2016] and the set-based operators [Al Marri et al. 2016]. However, to the best of our knowledge, no one focuses on the Division.[1]. This MSc. work tackles the problem by presenting the **first** Similarity-

[1]Note that there exist works focused on relaxing the division by means of fuzzy logic, but they cannot

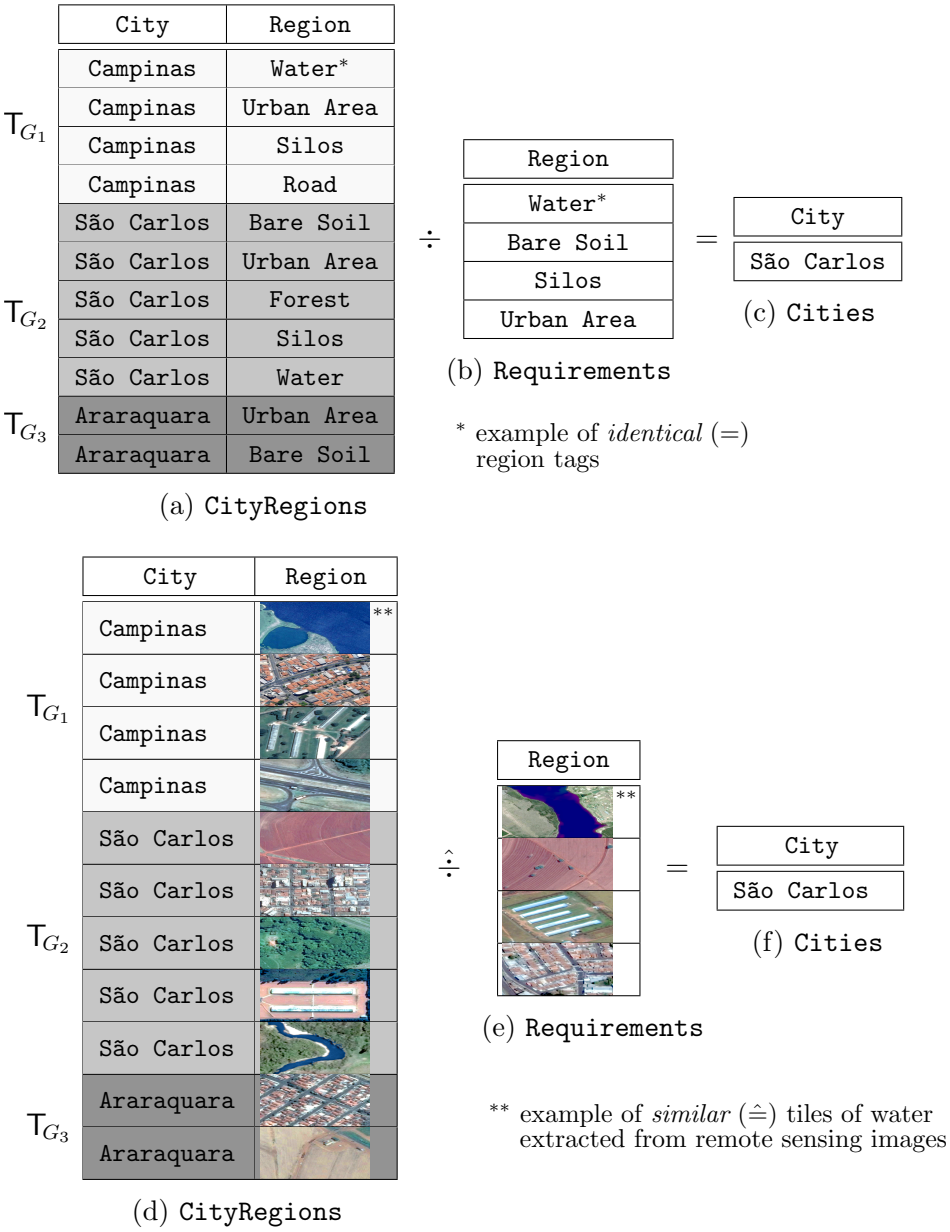

Figure 1. Example of the division used to select cities well suited to produce a particular type of crop. Top: textual tags are compared by *identity* (=). Bottom: tiles extracted from remote sensing images are compared by *similarity* (≙). Best viewed in color.

aware Division (÷̂) database operator.

3. Main contributions of this MSc. Work

3.1. Operator Design and Usability

We identified severe limitations on the usability of the Relational Division to process complex data, and extended it into a new operator to tackle the problem. To make it possible, we identified and studied attribute comparisons that are intrinsically performed

be applied to complex data – see page 32 in the Dissertation for details.

by the original operator of division, and found out that two categories of comparisons must be extended to develop a similarity-aware division: *intra* and *inter-relation* comparisons – see details in the Dissertation (Section 5.1.1). The former was then re-engineered to find candidates for the quotient T_R by grouping together the tuples of the dividend T_1 with **similar** values in the attributes of list $\overline{L_1}$, while we improved the latter to populate T_R with the candidates that have at least one tuple **similar** to each tuple of the divisor T_2, taking into account only the attributes of L_1 and L_2.

IMPACT: This MSc. work introduces a new branch of research by demonstrating that similarity comparison applied to the Division ($\div$) database operator turns it into a valuable tool to process complex data coming from modern applications; our Similarity-aware Division ($\hat{\div}$) is naturally well suited to answer queries with an idea of "candidate elements and exigencies" to be performed on these data. In fact, we show how to use it to support applications in five distinct areas: agriculture (see Section 6.0.1 in the Dissertation), genetics (Section 6.0.2), industrial quality control (Section 7.0.1), digital library search (Section 7.0.2) and prospective client identification in enterprises (Section 7.0.3). Note that none of these applications would benefit from the original division.

3.2. Formal Definition and Novel Algorithms

In Chapter 5 of the Dissertation we formally define the new Similarity-aware Division ($\hat{\div}$) operator and present two fast and scalable algorithms for it. The first approach takes advantage of index structures to speed-up the queries; the second one uses a full table scan for the cases when appropriate indexes are not available. To evaluate the efficiency of our algorithms, we performed experiments in synthetic data with up to millions of tuples; our algorithms presented either *linear* or *sub-linear* scalability tendencies in every single experiment. Theoretical time complexity analyses were also performed and corroborate this result. Finally, we demonstrate that our index-based algorithm can also optimize the original division in RDBMS – details are in Chapter 4 of the Dissertation.

IMPACT: The formal definition of the similarity-aware division is compatible with the Relational Algebra and can coexist with the traditional operators. It also maintains the same elemental properties of the original relational division, being in essence the opposite operation of the cartesian product – see details in Section 2.2 and Definition 5.9 of the Dissertation. Thus, it can be included in any commercial RDBMS by means of one of our algorithms. We also demonstrate theoretically and experimentally that the algorithms can handle extensive sets of data with (sub-)linear scalability, which is essential for the practical use of the new operator.

3.3. Case Studies and Semantic Validation

To validate our proposals, we performed case studies on the support of agriculture and genetics through semi-automatic complex data analysis. First, the similarity-aware division was used to accurately identify Brazilian cities well suited to produce a particular type of crop, based on the analysis of geopositioned remote sensing images. The same setting of our motivational example from Figures 1d, 1e and 1f was used; see details in Section 6.0.1 of the Dissertation. In the second case study, our proposed operator accurately identified animals that are the few top-quality milk producers, among 4.1 thousand animals, by only analyzing their genetic conditions represented by Single Nucleotide Polymorphisms (SNPs). See Section 6.0.2 of the Dissertation for details.

Additionally, Chapter 7 of the Dissertation provides conceptual evidences in support of the similarity-aware division's generality and usability, by describing how it can be helpful in three other applications: **(1) Automatic Quality Control in Industry:** to deploy an automatic quality control system to work in real-time using only pictures taken from the products in the production line; **(2) Digital Library Search:** to search documents that include a set of terms of interest, i.e., individual words or expressions, as well as documents with terms alike to them, and; **(3) Prospective Client Identification in Enterprises:** to automatically identify auction-like Request for Tender procedures – in Portuguese known as *licitações públicas* – for which a given enterprise can make a bid, as well as to estimate how large is the potential competition for it, by only analyzing textual product descriptions.

IMPACT: We demonstrated the usability and generality of the similarity-aware division by: (i) performing case studies in agriculture and genetics, and; (ii) describing how to use it to support applications in other three areas. Note that very little human intervention was necessary in both case studies performed: a single small example image per requirement, e.g., the images in Figure 1e; nothing else, was enough to accurately evaluate Brazilian cities for the crop production, and; a single outstanding animal given as example, among 4.1 thousand animals, allowed us to spot the few top-quality milk producers for selective breeding. Since few training data is required and our algorithms are fast and scalable, we argue that the similarity-aware division is potentially useful to analyze very large amounts of complex data, even in real-time.

Finally, let us summarize four well established facts in complex data analysis: (1) today, many applications manage complex data, such as images, fingerprints, DNA sequences, audio, large graphs, etc. (2) these data **must** be compared by similarity, instead of identity (=); (3) traditional and complex data are commonly modeled in the same way in a relational database, just like we do in Figure 1, and; (4) the division is the simplest and most intuitive way to represent queries with the concept of "for all" (Universal Quantification $\forall$). Note that our new operator **naturally** fits into this context; whenever the input of **any** for-all-based query has at least one complex attribute, a similarity-aware division should be performed.

4. Conclusion

In this MSc work we identified severe limitations on the usability of the Relational Division to process complex data, and tackled the problem by extending it into the **first** Similarity-aware Division ($\hat{\div}$) database operator. As opposed to the existing division, our new operator is naturally well suited to answer queries with an idea of "candidate elements and exigencies" to be performed on complex data from real applications of high-impact, such as the numerous modern applications that process images, genetic data, audio, long texts, fingerprints, and several other "non-traditional" data types that must be compared by similarity. We formally defined the new operator to allow its use in queries together with the existing relational operators, and carefully designed two fast and scalable algorithms for it. Case studies were performed to demonstrate that the similarity-aware division can support genetics and agriculture, and we also described how it may be helpful in digital library search, industrial quality control and to spot prospective clients for enterprises.

In summary, this MSc. work introduced a new branch of research with focus on inserting the similarity-aware division into the environment of a commercial RDBMS, with

query optimization and the like, as well as discovering new applications that can benefit from it. In fact, it already inspired another MSc. work in development at ICMC/USP, whose initial results include a case study that validades our example with prospective client identification [Vasconcelos et al. 2018]. For all its contributions and potential to impact on real world critical problems, for opening the door to tackle interesting future work, we believe that this work is a singular, outstanding contender for this year award.

Main publications of this MSc. work: The core of this work generated three main papers – (1) Gonzaga, A. S., Cordeiro, R. L. F.; A New Division Operator to Handle Complex Objects in Very Large Relational Datasets. In **EDBT**, 2017: p. 474-477 (International Conference – **Qualis CC A1**); (2) Gonzaga, A. S., Cordeiro, R. L. F.; The Similarity-aware Relational Division Database Operator. ACM SAC, 2017: p. 913-914 (International Conference – **Qualis CC A1**), and (3) Gonzaga, A. S., Cordeiro, R. L. F.; Fast and Scalable Relational Division on Database Systems. SBBD, 2016: p. 169-174. Also, an improved and extended version of the article published at EDBT 2017 is currently in the second round of revision for publication at the Elsevier Information Systems Journal (**Qualis CC A2**). Finally, one paper at ICEIS 2016 (International Conference – **Qualis CC B2**) was also developed in the MSc. work, focused on a distinct topic.

References

Al Marri, W. J., Malluhi, Q., Ouzzani, M., Tang, M., and Aref, W. G. (2016). The similarity-aware relational database set operators. *Inf. Syst.*, 59(C):79–93.

Belohlavek, R. and Vychodil, V. (2010). Query systems in similarity-based databases: Logical foundations, expressive power, and completeness. In *ACM SAC*, pages 1648–1655, New York, NY, USA.

Budíková, P., Batko, M., and Zezula, P. (2012). Query language for complex similarity queries. In *ADBIS*, pages 85–98.

Codd, E. F. (1972). Relational completeness of data base sublanguages. *In: R. Rustin (ed.): Database Systems: 65-98, Prentice Hall and IBM Research Report RJ 987, San Jose, California*.

Pola, I. R., Cordeiro, R. L., Traina Jr, C., and Traina, A. J. (2015). Similarity sets: A new concept of sets to seamlessly handle similarity in database management systems. In *Inf. Syst. 52:*, pages 130–148.

Silva, Y., Aref, W., Larson, P.-A., Pearson, S., and Ali, M. (2013). Similarity queries: their conceptual evaluation, transformations, and processing. *The VLDB Journal*, 22(3):395–420.

Silva, Y. N., Pearson, S. S., Chon, J., and Roberts, R. (2015). Similarity joins: Their implementation and interactions with other database operators. *Inf. Syst.*, 52:149–162.

Tang, M., Tahboub, R. Y., Aref, W. G., Atallah, M. J., Malluhi, Q. M., Ouzzani, M., and Silva, Y. N. (2016). Similarity group-by operators for multi-dimensional relational data. *IEEE Trans. Knowl. Data Eng.*, 28(2):510–523.

Vasconcelos, G. Q., Zabot, G. F., de Lima, D. M., Rodrigues, J. F. J., Traina, C. J., dos S. Kaster, D., and Cordeiro, R. L. F. (2018). Tender-sims: Similarity retrieval system for public tenders. In *ICEIS, pages 143-150*, Funchal, Portugal.