Graph databases as storage support for detection of star clusters in nearby galaxies

Martín Casatti[1], Analía Guzmán[1], María Alejandra Paz Menvielle[1]

Universidad Tecnológica Nacional - Facultad Regional Córdoba, Córdoba, Argentina,
mcasatti@frc.utn.edu.ar, aguzman@frc.utn.edu.ar,pazmalejandra@gmail.com

Abstract. The present work will explore the most outstanding features that a graph database should gather for the implementation of a system for structural patterns recognition in nearby star clusters. The parameters to be stored will be analyzed, according to the information revealed by various astronomical observation projects, as well as the most promising representation model, considering the purpose of storage. Some of the characteristics of existing databases will also be described, as well as their advantages and disadvantages when implementing the recognition system. It will be demonstrated that a representation in the form of graphs of the existing information in the astronomical repositories is not only possible and can help to the automatic processing but also can provide an effective mechanism for the implementation of pattern recognition algorithms and the consequent detection of stellar structures of interest.

Keywords: graph, database, modeling, pattern detection, astronomy, stellar clusters

1 Context

This report is part of the work aimed at developing a master's thesis for obtaining the title of Master in Information Systems Engineering, which seeks to determine the effectiveness of algorithms based on graphs for the detection of structural patterns in models of astronomical systems, specifically with regard to star clusters in nearby galaxies.

The aforementioned thesis is developed in the context of a research and development project that has been approved by the Secretaría de Investigación, Desarrollo y Posgrado, in the Universidad Tecnológica Nacional, which is carried out in the CIDS – Centro de Investigación, Desarrollo y Transferencia en Sistemas de Información.

At present there is a large amount of information from nearby galaxies due, in large part, to the fact that the Hubble Space Telescope (HST) has allowed obtaining data with high spatial resolution using several wide-field cameras (WFPC2, ACS) [12]. This access facilitates research linked to stellar groups, on different populations and histories of star formation.

It has long been recognized in the field of astrophysics that stellar clusters are important laboratories for research, since they contain statistically significant samples of stars of approximately the same age in a small space. On the other hand, the stellar groups existing in them provide valuable information for the understanding of the structure of the galaxy that contains them.

There is currently a huge amount of information obtained from continuous observation projects, mainly in survey mode[6, 15] such as:

- VVV[24] (https://vvvsurvey.org/)
- LSST[19] (https://www.lsst.org/)
- SDSS[7] (https://www.sdss.org/)
- Gaia-ESO[17] (https://www.gaia-eso.eu/)

All the mentioned examples require automatic mechanisms for the analysis of data.

The Data Minning (DM) algorithms, in particular those related to the automatic pattern recognition, are currently having an important revision and development[5, 2, 28] for their application on the data that arise from the big surveys.

In this work we intend to expose the capabilities of the graph databases to support algorithms for automatic pattern recognition on astronomical data. The aim is to determine if these techniques, from other fields of application, are transferable to the field of astronomy, with or without adjustments, or if the information from astronomical databases requires recognition algorithms designed specifically for them.

In a first stage, the way in which the preprocessing and data acquisition will be performed will be analyzed, to then carry out the stages of extraction of the fundamental characteristics and the grouping or classification to achieve the identification and parameterization of new stellar groups.

Regarding the preprocessing and acquisition of data, it will be described how the selected data model is and how, from the sample of data obtained from the optical instruments, it will be prepared to store it in a graph, in order to continue with the detection and recognition stages of related patterns.

2 Introduction

A pattern is an entity that can be given a name and that is represented by a set of measured properties and the relationships between them are represented in the so-called *feature vector*[32].

In the domain of astronomical data a pattern can be the average distances between stars of the same cluster, their spectral characteristics, their brightness curve, etc. The vector of characteristics would be formed, in this case, by physical, chemical or structural characteristics that relate the different elements.

There are works that have focused on improving the knowledge of our own Galaxy and the Magellanic Clouds, for example Baume et al. 2008[3], but currently there are several factors that significantly increase the number of objects to analyze, making an automatic method for data processing especially relevant.

The automatic recognition, description, classification and grouping of patterns are important activities in a great variety of scientific disciplines, such as biology, psychology, medicine, computer vision, artificial intelligence, remote sensing, etc. The main importance of pattern detection in the data is that you can infer causes for the grouping of the data and this is where the interest of the application of these techniques to the astronomical field lies.

Although currently there are research lines studying recognition methods based on vector support machines[9] the approach used in this research line focuses on the structural characteristics of the graphs[8, 25].

The following are fundamental concepts related to the construction of graphs:

Node: A node is a *information entity* different from any other entity in the model. All nodes represent a single set of information which is indivisible and independent of any other information represented. Larger units can be represented only as groups of related nodes. For the purposes of this project, it is necessary that each node can be uniquely and differently identified from all other nodes in the network.

Link: A link is used to establish a relationship between two nodes of the model. A link can only connect two nodes. One called *origin* from which the link exits and one called *destination* to which it arrives. A relationship between a source node and two destination nodes requires two links, both with the same origin, but with different destinations. A node can be the source of several links, as well as the destination of several links[31, 4].

2.1 Patterns in Graphs

The recognition or detection of patterns within graphs seeks to detect a subgraph (pattern) in a graph (objective). We must consider that this search for coincidences can be broken down into two parts:

1. A structural match, where the nodes and relations of the pattern make up an existing structure in the objective graph.
2. A match at the level of elements, where the nodes and relations, at the level of their particular attributes, have the same values as in the structure found in the objective graph.

Many times the search for these two matches is executed separately to optimize the algorithms or reduce the search space[14].

In the domain under study, the detection of a subgraph (pattern) will be performed on the graphs generated from astronomical information provided by continuous observation files (survey) such as those found in the VVV Survey project, the Large Scale Telescope (LST) project or Hubble Space Telescope (HST) project, the latter being the origin of data that will be used in this and subsequent works.

2.2 Metrics in graphs

A tool widely used to describe graphs and often used to start the analysis of existing patterns in them, is the calculation of local or global metrics[31], which allow to characterize the objective graph or the base graph. The metrics can be divided into two large groups:

- Static Metrics: When they are calculated on a static graph at a given point in time. They focus mainly on the structural characteristics of it.
- Dynamic metrics: They take into account the temporal dimension of the changes that occur on the graph. They are more focused on the variations between two instants of time, rather than on the characteristics of the graph in each one of those instants.

Another approach to the analysis of the metrics is to analyze on which components of the graph the measurements are made. From this point of view there are different perspectives, the most common being:

- Network metrics (or global): Metrics that take as a reference the complete graph, with all the nodes and arcs that comprise it.
- Node metrics (or local): Are those that take as reference a node or subset of nodes to perform the calculations.

Here are some common metrics:

Global metrics: Centrality: This metric tries to determine which node or nodes occupy a central location in the network, being equidistant from the other nodes.

Connection: It seeks to establish the degree to which the nodes of a graph are connected with all the other nodes of the same graph. You can find, by applying this metric, strongly connected or weakly connected components.

Number of Components: In a graph that is not completely connected, it indicates the number of related subgraphs that are part of the graph. A component is a set of connected nodes that are part of the main graph.

Size of the giant component: It measures the number of nodes that the connected component has that is greater than all the other components of the graph. In a connected graph the size of the giant component is equal to the total number of nodes.

Shortest/longest route: Express the minimum/maximum length (in arcs) between two given nodes.

Local metrics: Connectivity: Expresses the number of connections a specific node has. It can be expressed as 'degree', if it does not take into account the direction of the arcs that impinge or leave the node, or as 'input degree' or 'output degree' when only taking into account incoming or outgoing arcs, respectively .

Centrality: It is a metric, associated with a node in a graph, which determines its relative importance within it, and can be divided into:

- Centrality of degree: Number of connections with other nodes
- Centrality of proximity: Indicates how close one unit is to the network of others.
- Intermediation centrality: Indicates whether a unit is within some of the shortest routes between two nodes in the network.

2.3 Pattern recognition design

The main objective of a automatic pattern recognition system is to discover the underlying nature of a phenomenon or object, describing and selecting the fundamental characteristics that allow them to be classified in a certain category[29, 16].

Automatic pattern recognition systems allow addressing problems in computer science, engineering and other scientific disciplines [13, 23], therefore the design of each stage requires joint analysis criteria to validate the results[21, 20].

After analyzing different ways of designing a pattern recognition system, we consider three phases[1]:

1. Acquisition and preprocessing of data.
2. Feature extraction.
3. Decision making or grouping.

In the data acquisition and pre-processing phase, the database infrastructure will be prepared in order to continue with the following phases.

For the next two phases, feature extraction and decision making or grouping the most relevant parameters that make up stellar groups of interest will be considered, working in collaboration with experts from the Instituto Astrofísico de La Plata, Buenos Aires, Argentina.

3 Methods and results

The boom of the current graph databases gives us important possibilities when it comes to modeling a domain for which the relational databases have no direct application.

The possibility of including in the designed schema information of various types, without penalizing for them the search capabilities or the representativeness of the information, is one of the most favorable characteristics of the graph-based model, which makes it especially recommendable in environments where there is no established scheme or is not stable or suffers from frequent variations. Examples of this are design prototypes, databases for concept testing and storage for data mining[26].

In terms of expressiveness, the graph databases reduce the impedance difference between the analysis model and the final implementation, a problem that has plagued the various database models for many years.

3.1 Representation factors

In order to have relevant information of the chosen domain and be able to detect patterns, it is necessary to have a graph base that represents the information of the star clusters with the greatest possible accuracy.

Currently, at least four fundamental factors must be taken into account when designing a system of knowledge representation in any given domain[30]:

- **Representational Adecuation:** Ability to represent all kinds of knowledge that are necessary in the domain.
- **Inferential Adecuation:** Ability to manipulate representation structures in such a way that they generate new structures that correspond to new knowledge inferred from the previous ones.
- **Inferential Efficiency:** Ability of the system to incorporate additional information to the representation structure, called meta-knowledge, which can be used to focus the attention of the inference mechanisms in order to optimize the computations.
- **Acquisition Efficiency:** Ability to easily incorporate new information. Ideally the system itself should be able to control the acquisition of new information and its subsequent representation.

These factors were taken into account during the process of designing the structures of the database, in such a way that the results can be maximized while computing operations are kept efficient.

3.2 The model of tagged graphs

The approach proposed in the present work is based on the tagged graph model, existing in a multitude of graph database products, both commercial and Open Source.

A *tagged properties graph* is composed of nodes, relationships, properties and labels, as can be seen in Figure 1.

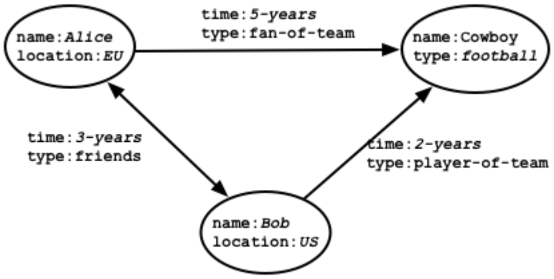

Fig. 1. Property Graph Model

- The **nodes** contain **properties**. You can think of nodes as documents that group together a set of properties that define the characteristics of the document to which they belong.
- The **nodes** can be marked with one or more **labels**. Labels group nodes by similar characteristics or indicate the roles each plays in the data model.
- The **links** connect nodes to each other and define the structure of a graph. A link has an direction, a source node and a destination node.
- Like nodes, links can also have properties that add semantic value to the relationship.
- In all cases, the properties and their values can be used to restrict the results of searches performed in a graph.

Based on the theoretical model of tagged graphs discussed above, an implementation model was designed that will provide support to the data on which the recognition algorithms will act.

The central element to be considered in the implementation model are the **nodes**. The nodes, in this area, model individual readings obtained by the observation instrument. Each node becomes equivalent, with this approach, to a record of the information exchange files. This process will be detailed in the section 3.3.

A unique identifier will be generated for each node to simplify the queries and subsequent recovery of the information associated with them. There are some properties inherent to each of the observations that will be stored in the corresponding nodes, such as the coordinates of that observation, the section of the sky that is being surveyed and the settings of the observation instrument. All these characteristics are necessary to reconstruct the original observation conditions from the data.

The **labels**, once the general data of each node is imported, will play a fundamental role since they are responsible for the storage of all the additional information that will be used in the searches.

Among the possible values to be stored are the various readings taken by the instruments and optical processors of the telescopes from which the information is obtained, including readings in the optical spectrum as well as in the infrared or ultraviolet (identified with certain particular wavelengths), labels relating to brightness, rotation speed, nomenclature, etc. will also be incorporated.

In the graph model the relations between the nodes are of crucial importance for the paths and are fundamental for the detection of patterns. But in the field under study it must be recognized that there is no single characteristic that links two stars and their associated readings.

There are many parameters that can indicate relations, such as the distance between the stars, the luminosity, the size, the different spectral readings, their rotation period, etc. As it is impracticable to establish relationships for all possible combinations of characteristics, the implementation will allow generating those relationships on demand.

That is, as a preliminary step to the analysis and detection of patterns, the user must indicate which feature or combination of them he wants to use to generate the relationships between the stars. Then the indicated relationships will be generated and the analysis will be made based on the resulting graph with the structure determined according to the characteristics indicated by the user.

This approach has been used in a timely manner in the papers published by Coutinho et. al 2016 [11], an example of which is presented in Figure 2.

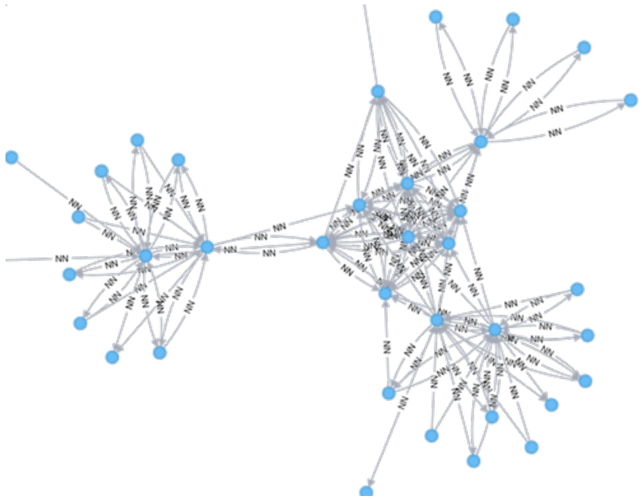

Fig. 2. Galaxies as labeled graph. Coutinho et. al

3.3 Sources and pre-processing of data

The aforementioned astronomical survey projects have large repositories of publicly accessible information, which are going to be used as sources of information to load the analysis database.

The *de facto* standard for the exchange of astronomical information is based on the FITS file format (Flexible Image Transport System[18]) which will be used as a basis for all data import routines.

ONgDB, the database used in this work, has the possibility of importing data files in a massive way, which is very necessary because each one of the source files generally have volumes of approximately half a million records. For this, it uses comma separated files with a particular format.

It is necessary to pre-process the information so that it is in an adequate format for its massive import into the database. To this end, a Python language development is proposed, which provides the AstroPy [27] libraries that are suitable for the reading and parsing of FITS files and allows the CSV files to be managed with ease. The complete procedure is described in Figure 3.

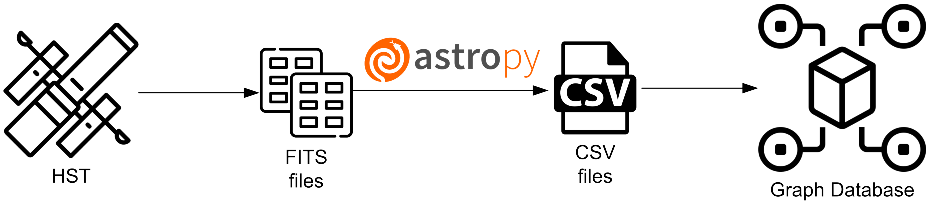

Fig. 3. Data pre-processing and import flow

3.4 Selected storage

For the implementation of the data model we opted for the use of ONgDB (https://www.graphfoundation.org/projects/ongdb/), a completely Open Source alternative to Neo4J (https://neo4j.com/), which is perhaps the most widespread commercial graph database.

We chose this product because it has some very valuable characteristics for the project:

- Natively represents the characteristics of the labeled graph model mentioned above
- Based on a commercial product of proven quality and updated technology
- It has a non-restrictive license that allows it to be used freely and free of charge in educational institutions and as part of research projects
- It has no restrictions regarding advanced features such as
  - ACID Transactions
  - Replication
  - Monitoring

4 Conclusions and future work

In view of the results obtained during the elaboration of this work, it is considered that the graph databases are a valid mechanism for the storage of astronomical information, providing a flexible scheme but that at the same time can query efficiently large volumes of data.

The labeled graphs allow to accurately reflect the astronomical data, also allowing the growth and adaptation of the structures according to the needs that arise from the data files, a task that is difficult in the case of using relational databases.

In subsequent stages during the evolution of the thesis plan, it is expected to optimize the mechanisms for importing astronomical information and provide users with some intuitive tools to perform both the data import tasks and the definition of the criteria for the generation of relationships demand.

An end user query system will be developed to indicate the parameters to be used in the searches and a module will be implemented that obtains some relevant statistical indicators according to the stored data.

Subsequently, some pattern models existing in other disciplines will be analyzed and tests will be implemented to use these patterns in an astronomical environment. The purpose of this activity is to determine if there are pre-existing patterns that have application in an area for which they have not been designed. Of special interest are the small-world network algorithms[22] and the social network algorithms[10].

References

1. Alonso Romero, L. & Calonge Cano, T. Redes neuronales y reconocimiento de patrones (2001).
2. Ball, N. M. & Brunner, R. J. Data mining and machine learning in astronomy. *International Journal of Modern Physics D* **19,** 1049–1106 (2010).
3. Baume, G. *et al.* Basic parameters of three star clusters in the Small Magellanic Cloud: Kron 11, Kron 63 and NGC 121. *Monthly Notices of the Royal Astronomical Society* **390,** 1683–1690 (2008).
4. Bondy, J. A. & Murty, U. S. R. *Graph theory with applications* (Citeseer, 1976).
5. Borne, K. D. Astroinformatics: a 21st century approach to astronomy. *arXiv preprint arXiv:0909.3892* (2009).
6. Borne, K. D. in *Next Generation of Data Mining* 114–137 (Chapman and Hall/CRC, 2008).
7. Bundy, K. *et al.* Overview of the SDSS-IV MaNGA survey: mapping nearby galaxies at Apache Point observatory. *The Astrophysical Journal* **798,** 7 (2014).
8. Bunke, H. & Allermann, G. Inexact graph matching for structural pattern recognition. *Pattern Recognition Letters* **1,** 245–253 (1983).
9. Burges, C. J. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery* **2,** 121–167 (1998).
10. Carrington, P. J., Scott, J. & Wasserman, S. *Models and methods in social network analysis* (Cambridge university press, 2005).
11. Coutinho, B. *et al.* The network behind the cosmic web. *arXiv preprint arXiv:1604.03236* (2016).
12. Dalcanton, J. J. *et al.* The ACS nearby galaxy survey treasury. *The Astrophysical Journal Supplement Series* **183,** 67 (2009).
13. Devijver, P. A. & Kittler, J. *Pattern recognition theory and applications* (Springer Science & Business Media, 2012).
14. Fan, W. *Graph pattern matching revised for social network analysis* in *Proceedings of the 15th International Conference on Database Theory* (2012), 8–21.
15. Frinchaboy, P. M. *et al.* in *Star Clusters in the Era of Large Surveys* 31–38 (Springer, 2012).
16. Fukunaga, K. *Introduction to statistical pattern recognition* (Elsevier, 2013).
17. Gilmore, G. *et al.* The Gaia-ESO public spectroscopic survey. *The Messenger* **147,** 25–31 (2012).

18. Hanisch, R. J. *et al.* Definition of the flexible image transport system (FITS). *Astronomy & Astrophysics* **376,** 359–380 (2001).
19. Ivezić, Ž. *et al.* Astrometry with digital sky surveys: from SDSS to LSST. *Proceedings of the International Astronomical Union* **3,** 537–543 (2007).
20. Kim, H. Y. & Giacomantone, J. O. *A new technique to obtain clear statistical parametric map by applying anisotropic diffusion to fMRI* in *Image Processing, 2005. ICIP 2005. IEEE International Conference on* **3** (2005), III–724.
21. Kim, H. Y., Giacomantone, J. & Cho, Z. H. Robust anisotropic diffusion to produce enhanced statistical parametric map from noisy fMRI. *Computer Vision and Image Understanding* **99,** 435–452 (2005).
22. Kleinberg, J. M. Navigation in a small world. *Nature* **406,** 845 (2000).
23. Meyer-Baese, A. & Meyer-Baese, A. *Pattern recognition for medical imaging* (Academic Press, 2004).
24. Minniti, D. *et al.* VISTA Variables in the Via Lactea (VVV): The public ESO near-IR variability survey of the Milky Way. *New Astronomy* **15,** 433–443 (2010).
25. Pavlidis, T. *Structural pattern recognition* (Springer, 2013).
26. Robinson, I., Webber, J. & Eifrem, E. *Graph databases: new opportunities for connected data* (" O'Reilly Media, Inc.", 2015).
27. Robitaille, T. P. *et al.* Astropy: A community Python package for astronomy. *Astronomy & Astrophysics* **558,** A33 (2013).
28. Schmeja, S. Identifying star clusters in a field: A comparison of different algorithms. *Astronomische Nachrichten* **332,** 172–184 (2011).
29. V., B., H, B. & A., F. *Data Science and Classification* (Springer, 2006).
30. Van Harmelen, F., Lifschitz, V. & Porter, B. *Handbook of knowledge representation* (Elsevier, 2008).
31. Van Steen, M. Graph theory and complex networks. *An introduction* **144** (2010).
32. Watanabe, S. *Pattern recognition: human and mechanical* (John Wiley & Sons, Inc., 1985).