Aggregation Semantics for Link Validity

Léa Guizol, Madalina Croitoru, and Michel Leclère

Abstract .

In this paper we address the problem of link repair in bibliographic knowledge bases. In the context of the SudocAD project, a decision support system (DSS) is being developed, aiming to assist librarians when adding new bibliographic records. The DSS makes the assumption that existing data in the system contains no linkage errors. We lift this assumption and detail a method that allows for link validation. Our method is based on two partitioning semantics which are formally introduced and evaluated on a sample of real data.

1 Introduction

Since 2001, ABES (French Bibliographic Agency for Higher Education) has been managing SUDOC[1] (University System of Documentation), a French collective catalog containing over 10 million bibliographic records. In addition to *bibliographic records* that describe the documents of the collections of the French university and higher education and research libraries, it contains nearly 2.4 million *authority records* that describe individual entities (or named entities) useful for the description of documents (persons, families, corporate bodies, events etc.). Bibliographic records contain *links* to authority records that identify individuals with respect to the document described.

A typical entry of a book, by a librarian, in SUDOC takes place as follows. The librarian enters the title of the book, ISBN, number of pages and so forth (referred later on as the attributes of the bibliographic record corresponding to the book in

LIRMM (University of Montpellier II & CNRS),
http://www.lirmm.fr/xml/en/0001-01.html
INRIA Sophia-Antipolis, France
http://www.inria.fr/en/

[1] http://en.abes.fr/Sudoc/The-Sudoc-catalog

question). Then (s)he needs to indicate the authors of the book. This is done by searching in the SUDOC base the authority record corresponding to that name. If several possibilities are returned by the system (e.g. homonyms), the librarian decides based on the bibliographic information associated to each candidate which one is most suitable for to choose as author of the book at hand. If none of the authors already in the base is suitable, then the librarian will create a new authority record in the system and link the book to this new record. The lack of distinguishing characteristics in the authority records and the lack of knowledge about the identity of the book's author imply that the librarian's decision is based on consultation of previous bibliographic records linked to each considered candidate. So any linkage error will entail new linkage errors.

In the SudocAd project [4] a decision support system was proposed to assist librarians choosing authority records. However, the SudocAd project relies on the assumption that the existing data in the SUDOC is clean, and namely:
• there are no distinct authority records describing one real world person,
• each contributor's name in a bibliographic record is linked to the "correct" authority record, and
• for each record, there is no big mistake in its attributes values.

In this work we lift the first two assumptions and aim to assess the quality of the data in the SUDOC. Preliminary work towards this goal was proposed in [5], where a general decision support system methodology was proposed to assess and repair such data. The method was based on partitioning contextual authorities (bibliographic records from the point of view of the authority record) according to various criteria. The global method, as represented in Figure 1, consists of:

1. Allowing the domain expert to enter an appellation (name and surname). The system returns the authority records in Sudoc data with an appellation syntactically close to the one entered by the expert. For each authority record a set of corresponding bibliographic records (the books written by the author in question) are also returned.
2. Constructing all contextual authorities for all returned authority records. A contextual authority is the union of an authority record with one of the bibliographic records pointing to it. Intuitively it corresponds to an author in the context of one written book.
3. Partitioning the set of contextual authorities according to a partitioning semantics and a set of criteria. The set of criteria all return symbolic values. The aim of the partitioning semantics is to obtain a partition of the set of contextual authorities which "makes sense" from the point of view of the set of criteria. The obtained partition can also be compared with the initial partition (where contextual authorities belonging to the same authority record are in the same class of partition). Such comparison could provide paths for eventual repairs of the SUDOC data set.

The contribution of the paper is addressing the third item above. Namely we address the research question of "*how to propose a partitioning semantics for a set of criteria returning symbolic values*". We propose two semantics (a local and a

global one) and evaluate them in terms of quality of returned partitions as well as algorithmic efficiency.

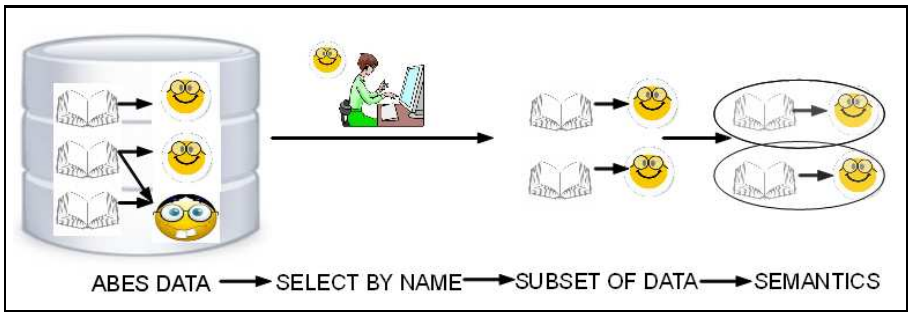

Fig. 1 First steps of the global approach

The paper is structured as follows. After presenting the SUDOC data as well as the different criteria currently implemented (Section 2), we present in Section 3 a motivating example showing the limitations of existing work. We then introduce two partitioning semantics (Section 4): while the first semantics can discard a criteria value overall on the dataset to be partitioned, the second semantics refines the first semantics by introducing the notion of local incoherence. We evaluate our approach on a sample of SUDOC data, then discuss its execution time in Section 5.

2 SUDOC Data and Criteria

An authority record is used to represent a person in SUDOC. In addition to an identifier, it contains at least a set of names used to designate the person and, possibly, dates of birth/death, sex, nationality, titles and any comments in plain text. All the other information regarding his (her) contribution to some works (what (s)he wrote, what domains (s)he has contributed to etc.) are only available from the bibliographic records of the documents (s)he has (co-)authored. A bibliographic record is used to represent a particular document in SUDOC. Most information (such as title, publication date, language, domain) is reliable. The contributor information is added by searching the system for (person) authority records corresponding to each of the names indicated as contributing to the document.

We compute for each document a contextual description of each of its contributors. Such a description, denoted a *contextual authority*, will contain a set of selected information extracted from the bibliographic record and the reliable information about the contributor from the linked authority record. We select from the bibliographic record the title, the publication date, the domain, the language, the co-

contributors and the role. From the authority record, we consider that only names are reliable.

The contextual authorities will be then compared amongst each other in order to group together the contextual authorities which are similar. The comparison will be done according to various criteria. This process has to mirror the decision process of a domain expert when deciding if a contextual authority is close to another. This is the reason why in our project, one of the most important requirements is to consider symbolic criteria (i.e. the domain values of criteria considered are symbolic). The set of symbolic values for each criteria is also equipped with a total order.

As previously mentioned, given a set of contextual authority records we are interested to use the criteria (provided by the domain experts, the librarians) to "cluster" the authority records. The date of publication criteria, for instance, will provide a partitioning of the contextual authority records (publications close by date, far by date, very far by date etc.). We then want to combine the obtained partitions according to the different criteria and provide one or more overall partition(s) corresponding to the whole set of criteria.

In the reminder of the section we briefly describe the criteria considered and currently implemented in our system.

Let $\mathbb{O}$ be the objects set to partition. A criterion $C \in \mathbb{C}$ is a function that gives a comparison value for any pair of objects in $\mathbb{O}^2$. This comparison value is discrete and in a totally ordered set $V = \{never\} \cup V^C_{far} \cup \{neutral\} \cup V^C_{close} \cup \{always\}$. V^C_{far} and V^C_{close} are two totally ordered values sets.

Closeness values are denoted $+, ++, ...$ and farness values are denoted $-, --, ...$ such as : $always \geq ... \geq ++ \geq + \geq neutral \geq - \geq -- \geq ... \geq never$ (where $\geq$ stands for "is stronger" than).

The criteria we have currently implemented are: *domain*, *date*, *title*, *appellation*, *contributor* and *language*. The "domain" attribute is represented in SUDOC as a set of domain codes (see example in Table 1). The distance on domain codes and their aggregation function was provided by domain experts (omitted here for lack of space). The publication dates are compared using a distance based on the intervals between dates (intervals of 60, respectively 100 years). The titles are compared using a Levenstein adapted distance. The *contributor* criterion gives closeness values if there is common contributor(s) (without the contributor designed by the appellation). The *appellation* criterion is based on expert comparison function for names and surnames. The *language* criterion gives a farness comparison value if publications languages are distinct and none of them is English.

3 Motivating Example

As previously mentioned, partitioning semantics based on numerical values are not interesting for our problem since one of the requirements of the Decision Support System for the librarians is to use symbolic valued criteria (justified by the need of modelling human expert reasoning).

Even if the clustering methods such as those of [1] seemingly deal with symbolic values, the symbolic values are treated in a numerical manner. Let us consider a real world example from the SUDOC data and see how the Dedupalog semantics of [1] behaves.

Table 1 Example of real contextual authorities

Id	Title	Date	Domains	[...] Appellations
1	Le banquet	1868		"Platon"
2	Le banquet	2007		"Platon"
3	Letter to a Christian nation		[320,200]	"Harris, Sam"
4	Surat terbuka untuk bangsa kristen	2008	[200]	"Harris, Sam"
5	The philosophical basis of theism	1883	[100,200,150,100]	"Harris, Samuel"
6	Building pathology	2001	[720,690,690,690]	"Harris, Samuel Y."
7	Aluminium alloys 2002	2002	[540]	"Harris, Sam J."
8	Dispositifs GAA en technologie SON	2005	[620,620,530,620]	"Harrison, Samuel"

In Dedupalog, the criteria (denoted C) return a comparison value between two objects as follows : $C : \mathbb{O} \times \mathbb{O} \rightarrow \{close, far, always, never\}$. To decide whether two objects represent the same entity, the first step is to check if there is at least a criterion returning *always* or *never* [2]. If it is not the case, we simply count: $vote =$ (criteria returning *close*) $-$ (criteria returning far). If $vote \geq 0$, we consider the comparison value *close*, else far.

We are interested in a best partition on the object set $\mathbb{O}$. Semantically, two objects represent the same entity if and only if they are in a same partition class. A partition is valid if and only if there are no two objects with an *always* comparison value (respectively *never*) in distinct classes (respectively in a same class) of the partition. A partition P is a best partition if P is valid, and there is the fewest possible number of pairs of objects with a *close* comparison value (respectively far) in distinct classes (respectively in a same class) of the partition.

Let us consider the SUDOC data subset shown in Table 1. We consider the records set of "Harris, Sam" appellation (denoted $\mathbb{O}_s = \{3,4,5,6,7,8\}$). The expert-validated partition on $\mathbb{O}_s$ is $Ph_s = \{\{5\}, \{3,4\}, \{7\}, \{6\}, \{8\}\}$. The "domain" attribute is composed of a set of domains codes. Two objects are considered by *domain* criterion as *close* if they have at least a common domain, and far if not. Two objects are considered by *date* criterion as far if there is more than 59 years between their publication dates. Therefore, the *date* criterion returns that object 5 is far with all other objects. However, the *domain* criterion says than objects 3, 4 and 5 are pairwise *close* together because of common 200 domain code (= religion): 3, 4 and 5 should be in a same class. The *domain* criterion says than 6, 7 and 8 are pairwise far together and far from respectively 3, 4 and 5, so the only best partition is $\{\{5,3,4\}, \{7\}, \{6\}, \{8\}\}$. Unfortunately, this best partition is not the expert-validated partition. We claim that the reason for this unsatisfactory re-

[2] Dedupalog forbids the possibility of a criterion returning *always* and another returning *never* for the same pair of objects

sult is the way symbolic values are treated by such approaches: considering them as numerical.

In this paper, we propose two partitioning semantics that improve the state of art by allowing for:

- several levels of far and *close* values;
- non interference of *close* and far comparison values (for example, a *close* comparison value cannot erase a far comparison value).

The proposed semantics are detailed in the next Section.

4 Proposed Semantics

4.1 Partitioning

The set of contextual authorities is partitioned according to different criteria (at least a common name, closeness of domains, dates of publication, languages of publication etc.). The result is a partition of compared objects based on closeness criteria. Intuitively, objects in a same class are close from the point of view of the respective criteria, and far from objects in another class.

In Table 1, the contextual authority number 4 has been written 125 years after the contextual authority number 5. So, their authors should be different persons. With respect to the *date* criterion, these contextual authorities should then be in different classes of partitions. However, they have a common publication domain, the domain number 200 (= religion), so we could be tempted to put them in a same class with respect to the *domain* criterion. To decide whether these contextual authorities represent or not a same person, we should aggregate criteria, i.e. decide which value of which criteria is meaningless in this case. Once this is done, as explained in [5], we compare the aggregated partitions with the initial partition.

4.2 Preliminary Notions

In the following we solely consider valid partitions. A valid partition is a partition such that there are no two objects with an *always* comparison value (respectively *never*) in distinct classes (respectively in a same class) of the partition.

Of course, the ideal case consists of never having objects in the same class with a farness value, respectively objects in different classes with a closeness value. If such a partition does not exist, there are *incoherences* with respect to our criteria set. The incoherence (property of an objects subset such that they must be in a same class according to some criteria and must be in separated classes according to other criteria) notion is central to the definitions of the two semantics detailed in the Sections 4.3 and 4.4.

From the point of view of a single criterion at a time, we prefer to group objects linked by a higher closeness comparison value than a smaller one. Similarly, we prefer to separate objects with a greater farness value than objects with a lesser farness value. In the case of incoherences one has to make choices in order to satisfy closeness or farness values which, by definition, could lead to distinct partitions.

The value of a partition with respect to a criterion is given by a couple of values (v_p, v_n) such that v_p (p for "positive") is the smallest closeness or *always* value such that all objects pairs with a criterion value bigger or equivalent value than v_p are in the same class (denoted "satisfied" objects pairs in [2]). The bipolar condition on v_n (n for "negative") also applies: v_n is the biggest farness or *never* value such that all objects pairs with a criterion value smaller or equivalent value than v_n are in distinct classes.

The partition values are then used to order partitions. One partition is better than another if and only if its v_p value is smaller and its v_n value is bigger. We denote $v(P,C)$ the P partition value with respect to criterion C and $v(P,\mathbb{C})$ the P partition value with respect to criteria set $\mathbb{C}$. For several criteria, it is possible that there are several best partitions on an object set according to the partition order. Due to domain expert requirements we cannot employ a criterion preference order.

In the next subsections we present two partitioning semantics: a global semantics and a local semantics. The global semantics, in an incoherent case, will give the best partitions that respects a criterion value in the same manner overall. The local semantics tries to localise the incoherence sources and treat it separately.

4.3 Global Semantics

Let us consider the example in Table 1 and the two records sets of the "Harris, Sam" appellation (denoted $\mathbb{O}_s = \{3,4,5,6,7,8\}$) and "Platon" appellation ($\mathbb{O}_p = \{1,2\}$). The expert-validated partitions are respectively (for the two record sets) $Ph_p = \{\{1,2\}\}$ and $Ph_s = \{\{5\}, \{3,4\}, \{7\}, \{6\}, \{8\}\}$. We compute the best partition of the two separate sets.

Let us apply global semantics on $\mathbb{O}_s = \{3,4,5,6,7,8\}$. This objects set is not coherent with respect to our criteria. The Ph_s value is such that:

- $v(Ph_s, domains) = (++++, -)$,
- $v(Ph_s, date) = (always, -)$.

Ph_s has a best partition value on $\mathbb{O}_s$. However, partitions $P'_s = \{\{5,3,4\}, \{7\}, \{6\}, \{8\}\}$ with $(+,-)$ value for *domain* criterion and $(always, --)$ value for $date$ criterion is also a best partition. The plurality of best partitions values comes from incoherence between the $date$ and *domain* criteria.

Let us now apply global semantics on $\mathbb{O}_p = \{1,2\}$.The expert-validated partition is $Ph_p = \{\{1,2\}\}$. There is an incoherence in the $date$ criterion. Ph_p value is such that:

- $v(Ph_p, title) = (+, never)$,

- $v(Ph_p, date) = (always, never)$.

Ph_p is the only best partition possible on $\mathbb{O}_p$ because $\{\{1\},\{2\}\}$ is not valid with respect to $title$ criterion.

Let us now illustrate how global semantics will affect the whole set of objects by computing the best global partition on the union of records of the two appellations. We now apply global semantics on all our selected contextual authorities: $\mathbb{O} = \mathbb{O}_p \cup \mathbb{O}_s$. The expert-validated partition is $Ph_{ps} = \{\{1,2\},\{5\},\{3,4\},\{7\},\{6\},\{8\}\}$. We also encounter incoherences and this partition has the worst of Ph_p and Ph_s values for each criterion, in particular:

- $v(Ph_{ps}, title) = (+, never)$
- $v(Ph_{ps}, domains) = (+++, -)$
- $v(Ph_{ps}, date) = (always, never)$

Ph_{ps} has not the best partition value because we could improve partition value for *domain* criterion. This does not affect the *date* criterion value because it is already as bad as possible. For example, partition $P'_{ps} = \{\{1,2\},\{5,3,4\},\{7\},\{6\},\{8\}\}$ has a best value.

A way to fix this problem is to propose the local semantics detailed in the next Section 4.4. We complete this subsection by presenting the algorithms used to find all best partitions values according to global semantics. Please check [7] for more details of the algorithms presented in this paper.

4.3.1 Global Semantics Algorithm for One Criterion

The input data (SUDOC data) structure is represented internally by our system as a multiple complete graph $\mathbb{G}_\mathbb{C}$: the $\mathbb{G}_\mathbb{C}$ vertex set is $\mathbb{O}$ and $\mathbb{G}_\mathbb{C}$ edges are labelled by the comparison value between linked objects according to a specified criterion. We denote G_C the criterion graph of a single criterion C.

Let C be a criterion on an object set $\mathbb{O}$. In order to find the best partition values on $\mathbb{O}$ with respect to C, we have to find and to evaluate reference partitions on $\mathbb{O}$ with respect to C for all closeness values $v_i \in V^C_{close} \cup \{always\}$. To define a reference partition we need to use the notion of a refined partition as explained below.

Definition 1 (Refined partition). Let P_i, P_j, be two partitions on an object set $\mathbb{O}$. P_i is more refined than P_j if and only if $\forall\, c_i$ class $\in P_i\ \exists c_j$ class $\in P_j | c_i \subseteq c_j$.

P_j partition is said to be less refined than P_i partition.

Definition 2 (Reference partition for a criterion). Let C be a criterion on an object set $\mathbb{O}$ and v_i a closeness value so that $v_i \in V^C_{close} \cup \{always\}$. The reference partition P_{ref} for C with respect to v_i, is the most refined partition P such as $v(P,C) = (v_p, v_n)$ and $v_p \leq v_i$.

We denote $ref(v_i)$ the reference partition for a criterion C with respect to closeness value v_i.

Evaluating the reference partitions is enough to calculate and evaluate all the best partition values. Since the reference partition with respect to a closeness value v_i for criterion C is the most refined partition with $v_p \leq v_i$, it is the partition with the less possible farness edges such as both vertexes are inside the same class (edges inside a single class). This makes it a partition with the best possible v_n value with respect to the v_p such that it is $\leq v_i$.

To calculate a reference partition with respect to a closeness value v_i for criterion C comes to calculating connected components on G_C considering only v'_i labelled edges such as $v'_i \geq v_i$. We can then simply use Kruskal's algorithm (complexity $\mathcal{O}(m \log n)$ for n vertexes and m edges). Please note that the connected component idea has been already explored in [6] and [2]. However the authors do not consider incoherence problems or even more levels of farness and closeness values. In the worst case (when there is no valid partitions with a v_n value such as $v_n = max(V^C_{far} \cup \{never\})$) we have $k+1 = |V^C_{close} \cup \{always\}|$ references partitions to find and evaluate. The complexity of the global semantics for one criterion algorithm is $\mathcal{O}((k+1) * m \log n)$, and it is depicted below (Algorithm 1).

Algorithm 1 BestsValuesForASingleCriteria

Require: C: criterion on an objects set $\mathbb{O}$; G_C: criterion graph of C on $\mathbb{O}$;
Ensure: set of best partitions values with respect to C on $\mathbb{O}$

```
best partitions values set bestV = {};
for all value v_i ∈ V^C_close ∪ {always} in < order do
    Partition P= ref(v_p);
    Partition value v = v(P);
    if P is valid and ∄v' ∈ bestV | v' ⪰ v then
        add v to bestV;
    end if
    if v(P) = (v'_p, v'_n) such as v_n = max(V^C_far ∪ {never}) then
        return bestV;
    end if
end for
return bestV;
```

4.3.2 Global Semantics Algorithm for Several Criteria

Let us now consider the global semantics when there are more than one criteria to consider. We will first need three notions: closeness values set, ascendant closeness values set and reference partition. The reference partition, as explained above, is the actual test to be performed by the algorithm. The cardinality of the closeness value set represents the number of tests that the algorithm will need to perform in the worse case. Finally, the ascendant closeness values notion will allow us to skip some tests, and optimise the algorithm.

Definition 3 (Closeness values set). A closeness values set $\mathbb{VC}$ for a criteria set $\mathbb{C}$ is a set of v_i such as $v_i \in V_{close}^{C_i} \cup \{always\}$ and $C_i \in \mathbb{C}$, with one and only one closeness value v_i for each criterion $C_i \in \mathbb{C}$.

Definition 4 (Ascendant closeness values set). Let $\mathbb{VC}1$ and $\mathbb{VC}2$ be two closeness values sets for the same criteria set $\mathbb{C}$. $\mathbb{VC}1$ is an ascendant of $\mathbb{VC}2$ if and only if $\mathbb{VC}1$ has a best or equivalent (smaller) closeness value than $\mathbb{VC}2$ for each criterion in $\mathbb{C}$.

$\mathbb{VC}2$ is a descendant of $\mathbb{VC}1$.

Definition 5 (Reference partition with respect to a criteria set). Let $\mathbb{C}$ be a criteria set on an object set $\mathbb{O}$, and $\mathbb{VC}$ a closeness values set for $\mathbb{C}$. The reference partition P_{ref} for $\mathbb{C}$ with respect to $\mathbb{VC}$ (denoted $ref(\mathbb{VC})$) is the most refined (please see definition 1) partition such as $v(P_{ref}) = \{v(P_{ref}, C_i) \forall C_i \in \mathbb{C}\}$ with $\forall C_i$ criterion: $v(P_{ref}, C_i) = (v_p, v_n) | v_p \leq v_i \in \mathbb{VC}$.

The global semantics algorithm for several criteria is an extension of the one for one criteria (Algorithm 1). The best partition values are also reference partitions values, so we calculate, evaluate and compare them.

First, we find all closeness values set (Definition 3) for $\mathbb{C}$. We compute the reference partition (Definition 5) for each closeness values set $\mathbb{VC}$ by searching for connected components with Kruskal's algorithm (complexity $\mathcal{O}(m \log n)$) on $\mathbb{G}_{\mathbb{C}}$ considering only v_p labelled edges such as $v_p \geq v_i | v_i \in (V_{close}^{C_i} \cup \{always\}) \cap \mathbb{VC}$ and $C_i \in \mathbb{C}$. We then evaluate reference partition values and only keep best ones.

If a reference partition $ref(\mathbb{VC})$ has $v(P, C_i) = (v_p, v_n)$ for each criteria $C_i \in \mathbb{C}$ such as $v_n = max(V_{far}^{C_i} \cup \{never\})$, then reference partitions $ref(\mathbb{VC}')$ with $\mathbb{VC}'$ descendants (Definition 4) of $\mathbb{VC}$ have a worse or same value than $ref(\mathbb{VC})$, so we do not need to evaluate them.

For a criteria set $\mathbb{C}$ of c criteria, we have to calculate and evaluate $|V_{close}^{C_1} \cup \{always\}| * ... * |V_{close}^{C_c} \cup \{always\}|$ reference partitions in the worst case, namely $(k+1)^c$ reference partitions with $k = max(|V_{close}^{C_i}| \ \forall C_i \in \mathbb{C})$. So, this algorithm has $\mathcal{O}((k+1)^c * m \log n)$ as complexity (see Algorithm 2).

4.4 Local Semantics

Local semantics do not consider incoherence for the whole treated object set (denoted $\mathbb{O}$) but only for pairs of objects that cause incoherence. The pairs of objects that cause incoherence represent the objects that are to be put in the same class by some criteria and kept separate according to others (e.g. objects 4 and 5 described in Table 1 : *date* criterion returns than they must be in distinct classes but *domain* criterion returns than they must be in a same class).

We consider in $\mathbb{O}$ the objects in incoherent parts. A minimal incoherent subset $\mathbb{I}$ of $\mathbb{O}$ is a subset of $\mathbb{O}$ such that:

Algorithm 2 BestPartitionsValuesForCriteriaSet

Require: $\mathbb{C}$, a criteria set on an objects set $\mathbb{O}$; $\mathbb{G}_\mathbb{C}$ criteria graph of $\mathbb{C}$
Ensure: set of best partitions values with respect to $\mathbb{C}$ on $\mathbb{O}$.
1: best partitions values set $bestV = \{\}$;
2: set of closeness values set to test $toTest = \{\mathbb{VP} | \mathbb{VP}$, closeness values set for $\mathbb{C}\}$;
3: **while** $toTest \neq \{\}$ **do**
4: pick up $\mathbb{VP}$ from $toTest$ such as $\mathbb{VP}$ has no ascendant in $toTest$;
5: Partition $P = ref(\mathbb{VP})$;
6: Partition value $v = v(P, \mathbb{C})$;
7: **if** P is valid and $\nexists v' \in bestV | v' \succeq v$ **then**
8: add v to $bestV$;
9: **end if**
10: **if** $\forall C_i \in \mathbb{C}, v(P, C_i) = (v_p, v_n) | v_n = max(V_{far}^{C_i} \cup \{never\})$ **then**
11: remove all descendants of $\mathbb{VP}$ from $toTest$;
12: **end if**
13: **end while**
14: **return** $bestV$;

- it contains a pair of objects that causes incoherences,
- there are no closeness comparison values between an object of $\mathbb{O} \setminus \mathbb{I}$ and an object in $\mathbb{I}$, and
- there is no subset of $\mathbb{I}$ which is a minimal incoherent subset of $\mathbb{O}$.

In the previous examples, we saw than 4 and 5 contains an incoherence. However, $\{4,5\}$ is not a incoherent subset of $\mathbb{O}_s = \{3,4,5,6,7,8\}$ because 3 is linked by a closeness value (*domain* criterion) to 4 and 5. $\{3,4,5\}$ is an incoherent subset of $\mathbb{O}_s$.

The coherent part contains all $\mathbb{O}$ objects but considers that the comparison value for every pair of objects that are occurring in the same minimal incoherent subset is $neutral$. We denote the incoherent part of a $\mathbb{G}_\mathbb{C}$ criteria graph according to the set $\mathbb{IP}$ of incoherent parts of $\mathbb{G}_\mathbb{C}$: $coherentPart(\mathbb{G}_\mathbb{C}, \mathbb{IP})$.

A partition on $\mathbb{O}$ is better than another partition if it has a best value for the coherent part and for each incoherent part. The values of each (in)coherent part are determined by global semantics.

Let us consider an example and apply local semantics on all the selected contextual authorities in Table 1: $\mathbb{O} = \mathbb{O}_p \cup \mathbb{O}_s$. The expert-validated partition is $Ph_{ps} = \{\{1,2\}, \{5\}, \{3,4\}, \{7\}, \{6\}, \{8\}\}$. There are two minimal incoherent subsets : $\{5,3,4\}$ and $\{1,2\}$. For $\{5,3,4\}$ we have a best partition as for $\{1,2\}$. Since in this semantics the incoherent subsets are considered independently one from the other, one of the best values for Ph_{ps} on the whole subset is equal to the value of the domain expert validated partition.

4.4.1 Local Semantics Algorithm for Several Criteria

To find all best partitions values on a criteria graph according to local semantics, we first need to identify incoherent (and coherent) parts with a connected compo-

nents algorithm (complexity $\mathcal{O}(m \log n)$ [3]). Then, for the coherent part and each of the at most $n/2$ incoherent parts [4], we execute the algorithm of global semantics (of complexity $\mathcal{O}((k+1)^c * m \log n)$. So, the complexity in the worst case of the local semantics algorithm for several criteria is: $\mathcal{O}(n * (k+1)^c * m \log n)$ (please see Algorithm 3).

Algorithm 3 BestPartitionsValuesForLocalSemantics

Require: $\mathbb{C}$, a criteria set on an objects set $\mathbb{O}$; $\mathbb{G}_{\mathbb{C}}$ criteria graph of $\mathbb{C}$
Ensure: set of best partitions values with respect to $\mathbb{C}$ on $\mathbb{O}$.
1: best partitions values set $bestV = \{\}$;
2: Partition $P_a = ref(\mathbb{VP})$;
3: set of graphs: $Gparts = \{\}$;
4: **for** each incoherent class $\mathbb{I} \in P_a$ **do**
5: add $incoherentPart(\mathbb{G}_{\mathbb{C}}, \mathbb{I})$ to $Gparts$;
6: **end for**
7: add $coherentPart(\mathbb{G}_{\mathbb{C}}, Gparts)$ to $Gparts$;
8: apply algorithm 2 on each graph in $Gparts$;
9: $bestV = \{$ best partition for $\mathbb{G}_{\mathbb{C}}$: best partition for each graph in $Gparts\}$;
10: **return** $bestV$;

5 Evaluation

We have experimented the algorithms on 133 SUDOC data subsets related to 133 random (on a list of common names and surnames) appellation. For each appellation, we select the associated SUDOC data subset as follows:

- each authority record which has a close appellation is selected;
- for each selected authority record, linked bibliographic records (up to 100 upper limit) are selected;
- for each link between a selected bibliographic record and a selected authority record, we construct a contextual authority.

We measure the execution time in nanoseconds on each 133 Sudoc data subsets selected for algorithms 2 and 3. The 133 appellations generated between 1 and 349 contextual authorities each. The number of criteria we considered is 6 with a 84 closeness value sets (between 0 and 6 values per criterion).

We used a Intel(R) Core(TM) i7-2600 CPU 3.40 GHz based PC with 4GB of RAM running Windows 7 64 Bit with a Java 1.6 implementation. The execution times are shown on figure 2 and 3.

As seen in Figure 2 the execution time for global semantic algorithm is fast (less than one second even for 349 contextual authorities). This is an acceptable result

[3] with n vertexes and m edges

[4] since an incoherent part contains at least two edges between two vertexes

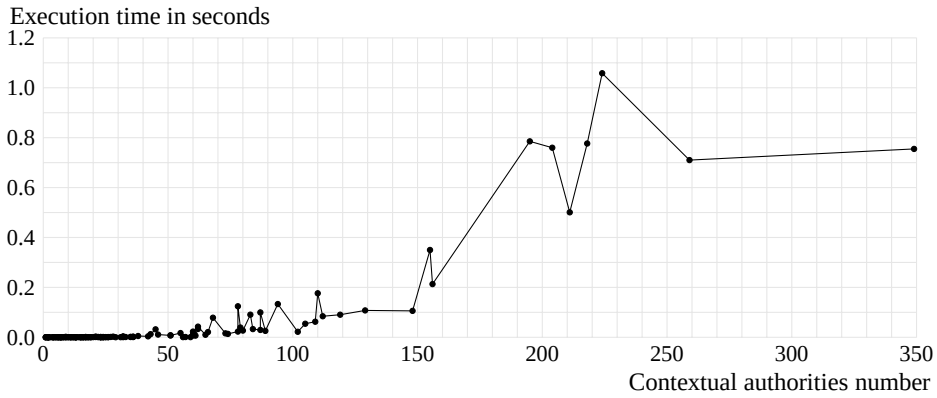

Fig. 2 Execution time for global semantics algorithm

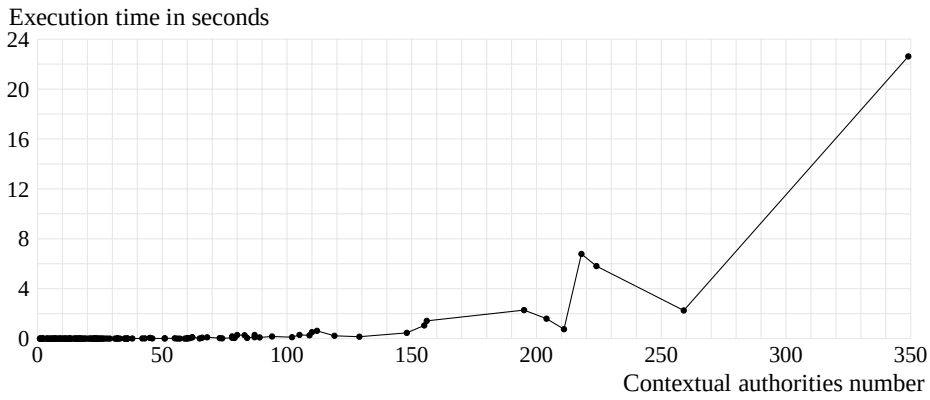

Fig. 3 Execution time for local semantics algorithm

since, according to the domain experts, it is very rare for an appellation to have more than 350 books authored. However, we notice that the execution time is irregular. This is due to the fact that since we have too many conflicting comparison values (causing incoherences) we are in the worst case scenario thus augmenting the execution time. Please note that this case is not necessarily dependent on the number of contextual authorities considered. We are currently investigating the relation between conflicting values and number of contextual authorities in the SUDOC by means of sampling.

In Figure 3 the execution time for local semantics algorithm is depicted. Even if the local semantics returns better qualitative results, the execution time is much longer than the global semantics algorithm. The results are not acceptable for a large number of contextual authorities with a lot of incoherence (as seen in the case of the 349 records with a time of 22 seconds) but we hope to be able to better understand

the SUDOC data in order to show that such cases are extremely rare. Such analysis of the SUDOC data, as mentioned before, constitutes current ongoing work.

6 Conclusion

In this paper we presented two partitioning semantics based on non-numerically valued criteria. The partitioning semantics were introduced due to a main feature of our system and namely that we want to keep the symbolic values of the criteria as much as possible (as opposed to aggregation techniques that reduce them to numerical values for manipulation). We explained the need of such semantics in the case of our application and explained how the two semantics yield different results on a real world example. We also shown than those semantics are scalable on most of real Sudoc subsets selected randomly.

Similar to conditional preferences [3], we need to decide which criterion value to improve over another depending on context. Links between conditional preferences and the presented partitioning semantics have to be explored in a future work.

Acknowledgements This work has been supported by the Agence Nationale de la Recherche (grant ANR-12-CORD-0012).

References

1. Arvind Arasu, Christopher Ré, and Dan Suciu. Large-scale deduplication with constraints using dedupalog. In *Proceedings of the 25th International Conference on Data Engineering (ICDE)*, pages 952–963, 2009.
2. Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. In *MACHINE LEARNING*, pages 238–247, 2002.
3. Craig Boutilier, Ronen I. Brafman, Carmel Domshlak, Holger H. Hoos, and David Poole. Cp-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. *Journal of Artificial Intelligence Research*, 21:135–191, 2004.
4. Michel Chein, Michel Leclère, and Yann Nicolas. SudocAD: A Knowledge-Based System for Object Identification. Technical report, LIRMM, INRIA Sophia Antipolis, December 2012.
5. Madalina Croitoru, Léa Guizol, and Michel Leclère. On Link Validity in Bibliographic Knowledge Bases. In *IPMU'2012: 14th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, volume Advances on Computational Intelligence, pages 380–389, Catania, Italie, July 2012. Springer.
6. A. Guénoche. Partitions optimisées selon différents critères: évaluation et comparaison. *Mathématiques et sciences humaines. Mathematics and social sciences*, (161), 2003.
7. Léa Guizol, Madalina Croitoru, and Michel Leclère. Aggregation semantics for link validity: technical report. Technical report, LIRMM, INRIA Sophia Antipolis, http://www.lirmm.fr/~guizol/AggregationSemanticsforLinkValidity-RR.pdf, 2013.