Aggregation Semantics for Link Validity: Technical Report

Léa Guizol, Madalina Croitoru, Michel Leclère

Aggregation Semantics for Link Validity: Technical Report

Léa Guizol, Madalina Croitoru and Michel Leclère

12th of April, 2013

Contents

1　Introduction

This paper is about partitioning a set of objects with respect to non-numerical criteria. We represent the partitioning problem as a complete graph partitioning problem in Section 2. Two partitioning semantics (global and local partitioning semantics), based on non-numerical criteria, are presented in Section 3 and 5. Partitioning problem algorithms for finding all best partition values are detailed for the global semantics in Section 4, and for local semantics in Section 5.

2　Graph Representation of a Partitionning Problem

2.1　Criterion

We would like to partition an object set $\mathbb{O}$ with respect to a criteria set $\mathbb{C} = \{C_1, ..., C_n\}$. Those criteria help to determine how much two objects look like each other. A criteria $C \in \mathbb{C}$ is a function that gives a comparison value for any couple of objects in $\mathbb{O}^2$. This comparison value is discrete and is in a totally ordered set $V = \{never\} \cup V_{far}^C \cup \{neutral\} \cup V_{close}^C \cup \{always\}$, such as:

- V_{far}^C and V_{close}^C are two totally ordered values sets $(<_C)$ of values giving a farness degree (respectively closeness degree) between objects. We denote $V_{far}^C = \{..., --, -\}$ (respectively $V_{close}^C = \{+, ++, ...\}$) the sets of farness (respectively closeness) values of criterion C such as $... <_C -- <_C -$ (respectively $+ <_C ++ <_C ...$).

- *never* and *always* means that objects are strictly different (respectively identical), so they are necessarily separated (respectively together).

- *neutral* means there is not enough information about the objects or the information is meaningless to decide for this criterion whether the objects should be together or not.

Definition 1 (Criterion structure) *A structure S_C of a criterion C is a couple $(V_C, <_C)$ such as :*

- *V_C is a set of comparison values composed of three subsets, V_{close}^C, V_{far}^C and $V_s = \{never, neutral, always\}$;*

- *$<_C$ is a totally ordered relation on V_C such as $\forall v_c \in V_{close}^C$, $\forall v_f \in V_{far}^C$ we have : $never <_C v_f <_C neutral <_C v_c <_C always$.*

2

V_s values are common for all criteria, but V_{close}^C and V_{far}^C are only for the criteria named C. We will denote abusively the $<_C$ relations $<$.

Definition 2 (Criterion) *Let $\mathbb{O}$, be an object set. A criterion C is a couple (e_C, S_C) such as:*

- *$S_C = (V_C, <_C)$ is a criterion structure, and*

- *e_C a binary operation, commutative on $\mathbb{O}$, which take its results in V_C, and such as $\forall o \in \mathbb{O} : e_C(o, o) = always$.*

$\mathcal{C}$ represents the set of all possible criteria, and $\mathbb{C} \subset \mathcal{C}$ the subset of criteria of interest to us. For each criterion $C \in \mathcal{C}$ and a set of objects $\mathbb{O}$, we define a criterion graph. This graph represents the objects of $\mathbb{O}$ by vertexes and the C comparison values by labels of edges.

Definition 3 (Criterion graph) *We represent a criterion C on an object set $\mathbb{O}$ by a complete criterion graph $G_C = (S, E_C)$ such as vertexes are the elements contained in $\mathbb{O}$ and edges are labeled by the comparison values defined by e_C.*

2.2 Graph Representation for Several Criteria

How to represent a criteria set $\mathbb{C}$ on an object set $\mathbb{O}$?

Definition 4 (Criteria (set) graph) *Let $\mathbb{C}$, be a criteria set on an objects set $\mathbb{O}$ such as every criterion $Ci \in \mathbb{C}$ is represented by a criterion graph $G_{Ci} = (S, E_{Ci})$. The multigraph $\mathbb{G}_\mathbb{C} = (S, \bigcup E_{Ci} \forall Ci \in \mathbb{C})$ is the criteria graph of $\mathbb{C}$, denoted $\mathbb{G}_\mathbb{C} = (S, E_\mathbb{C})$.*

We can now represent our problem as a graph partitioning problem. Let us define our global semantics for partitioning it in the next Section.

3 Global Semantics of Best Partitioning

In this Section, we will talk about partitions of a criteria graph G (Definition 4). What is a good partition? This implies to define a semantics of best partitioning (in Section 3.1). Then, we will describes ours global semantics, for one or several criteria (Sections 3.2 and 3.3).

3.1 Common Partitioning Semantic Definition

Definition 5 (Best partitioning semantics) *Let $\mathbb{C}$ be a set of criteria on the object set $\mathbb{O}$; $\mathbb{P}$, the set of possible partitions on $\mathbb{O}$; and $\mathbb{VP}$, the set of possible partitions values. A semantic of best partitioning is composed of:*

- *a function that gives a value for a partition: $v : \mathbb{P}, \mathbb{C} \mapsto \mathbb{VP}$, and*

- *an order on $\mathbb{VP}$.*

For P, a partition on an object set $\mathbb{O}$ and C, a criterion on $\mathbb{O}$, we denote the value of P on $\mathbb{O}$ with respect to $C : v(P, C)$. When $\mathbb{O}$ and C are obvious, we denote it abusively $v(P)$.

Let vp_i, vp_j be two partition values. We denote "vp_i is preferred to vp_j'" $vp_i \succeq_{vp} vp_j$, and "vp_i is not comparable to vp_j" $vp_i \diamond_{vp} vp_j$ (means that $vp_i \succeq_{vp} vp_j$ and $vp_i \preceq_{vp} vp_j$ are both false).

When we would like to compare two partitions P_i, P_j on the same object set $\mathbb{O}$ with respect to a criterion C, we compare their values for C. Two partitions are equivalent ($P_i \approx_p P_j$) when they have the same value ($v(P_i) =_{vp} v(P_j)$). A partition ($P_i \succeq_p P_j$) is preferred to another one if and only if its value is preferred ($v(P_i) \succeq_{vp} v(P_j)$). Partitions are not comparable together ($P_i \diamond_p P_j$) when their value are ($v(P_i) \diamond_{vp} v(P_j)$).

Let us define our global partitioning semantics for a criterion graph. Once it is done, we will also define it for a criteria graph.

3.2 Global Partitioning Semantics for a Single Criterion

We are only interested in partitions such that the edge values between objects in the same class are never *never* and edge values between objects in distinct classes are never *always*. Those partitions are said valid (Definition 6).

For a partition P on graph G with s, a G vertex, we denote $class(s, P)$, the class of P such as $s \in class(s, P)$. When P is obvious, we abusively denote it $class(s)$.

Definition 6 (Valid Partition) *Let P be a partition on the criterion graph $G_C = (S, E)$. P is valid if and only if there is no edge (s_i, s_j) labeled always (respectively never) such as $class(s_i) \neq class(s_j)$ (respectively $class(s_i) = class(s_j)$).*

From now on, we will only be interested by valid partitions.

Definition 7 (Partition value with respect to a single criterion) *Let $G_C = (S, E)$, be the criterion graph of criterion C, and P a partition on G. In the case where P is not valid (Definition 6), then the value of P is invalid. If P is valid, the value of P is (v_p, v_n) with :*

- *$v_p = min(V_{close}^C \cup \{always\} | \forall$ edge (s_i, s_j) labeled with a value $\geq v_p$, $classe(s_i) = classe(s_j))$*

- *$v_n = max(V_{far}^C \cup \{never\} | \forall$ edge (s_i, s_j) labeled with a value $\leq v_n$, $classe(s_i) \neq classe(s_j))$*

We denote $v(P, C)$ the P partition value with respect to criterion C (abusively $v(P)$, if C is obvious). By extent, in cases with a criteria set $\mathbb{C}$, we denote $v(P, \mathbb{C})$ the P partition value with respect to criteria set $\mathbb{C}$ (abusively $v(P)$, if $\mathbb{C}$ is obvious).

An other way to express it is that v_p is the smallest value such as every edge (s_i, s_j) labeled with a bigger or equivalent value than v_p is always satisfied (as [1] meaning : v_p is a closeness or *always* value and $class(s_i) = class(s_j)$), and v_n is the biggest value such as every edge with a value smaller or equivalent to v_n is satisfied (meaning : v_p is a farness or *never* value and $class(s_i) \neq class(s_j)$).

Definition 8 (Partition values order for a single criterion) *Let (v_p, v_n), (v_p', v_n'), be two valid partitions values for a criterion graph G_C.*

- *(v_p, v_n) is (v_p', v_n') (denoted $(v_p, v_n) =_{vp} (v_p', v_n')$) if and only if $v_p = v_p'$ and $v_n = v_n'$*

- *(v_p, v_n) is better or equivalent to (v_p', v_n') (denoted $(v_p, v_n) \succeq_{vp} (v_p', v_n')$) if and only if $v_p \leq v_p'$ and $v_n \geq v_n'$*

- *in other cases, (v_p, v_n) and (v_p', v_n') are not comparable (denoted $(v_p, v_n) \diamond_{vp} (v_p', v_n')$).*

3.3 Global Partitioning Semantics for Several Criteria

In a context where we use several criteria, how do we determine what is a best partition? We take into consideration a particular relation between criteria: priority (denoted $\gg$). In the common case, criteria are independent (denoted $\diamond$).

Let A, B, be two independent criteria (denoted $A \diamond B$) defined on the same object set $\mathbb{O}$. To compare two partitions with respect to $\{A, B\}$, the

comparison of their values with respect to A is as important as the comparison of their values with respect to B. Let $\mathbb{C}$ be a criteria set so that criteria are all independent together (namely an independent criteria set). Then for all distinct $C_i, C_j \in \mathbb{C}$ we have $C_i \diamond C_j$. Let us see how to compare two partitions with respect to an independent criteria set in Definition 9.

Definition 9 (Partition values order for independant criteria) *Let P_1, P_2, be two partitions on $\mathbb{G}_{\mathbb{C}}$, a criteria graph so that $\mathbb{C}$ is an independent criteria set. Let C_i, be a criteria that belongs to $\mathbb{C}$. $v(P_1, \mathbb{C}) \succeq_{vp} v(P_2, \mathbb{C})$ if and only if :*

- $v(P_1, C_i) \succeq_{vp} v(P_2, C_i)$, and

- $v(P_1, \mathbb{C} - \{C_i\}) \succeq_{vp} v(P_2, \mathbb{C} - \{C_i\})$.

Values of partitions P_1 and P_2 are incomparable for $\mathbb{G}_{\mathbb{C}}$ in other cases.

In the next Section, we will see details and complexity of algorithms that give every best partitions values of an object set $\mathbb{O}$ according to a criteria set $\mathbb{C}$.

4 Problem for Global Semantics: to Find Every Best Partition Values

Let us see how to obtain best partitions values of an object set $\mathbb{O}$ with respect to one (Section 4.1) or several (Section 4.2) criteria.

4.1 With Respect to a Single Criterion

Let C be a criterion on an object set $\mathbb{O}$. In the aim to find the best partition values on $\mathbb{O}$ with respect to C, we have to find and to evaluate reference partitions (Definition 11) on $\mathbb{O}$ with respect to C for all closeness value $v_i \in V_{close}^{C} \cup \{always\}$. Let us see some definitions of special partitions.

Definition 10 (Thinner partition) *Let P_i, P_j, be two partitions on an object set $\mathbb{O}$. P_i is thinner 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 bigger than P_i partition.

Definition 11 (Reference partition for a criterion) *Let C be a criterion on an object set $\mathbb{O}$ and v_i a closeness value so that $v_i \in V_{close}^C \cup \{always\}$. The reference partition P_{ref} for C with respect to v_i, is the thinnest 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.

Why do calculating and evaluating only reference partitions is enough to calculate and evaluate all the best partition values? The reference partition (Definition 11) $ref(v_i)$ such as $v(ref(v_i)) = (v_p, v_n)$ is the thinnest partition (Definition 10) with $v_p \leq v_i$, so it is the partition with the less possible farness edges such as both vertexes are inside the same class (edges inside a single class). These facts make it a partition with the best possible v_n value with respect to v_p value $\leq v_i$. So, we can find all best partitions values by calculating and evaluating only all the reference partitions' values. At best, other partitions will be as good as a reference partition is.

Best values are all the values of reference partitions.

A reference partition $ref(v_i)$ such as its value is $v(ref(v_i)) = (v_p, v_n)$ is a best partition if there is not an other reference partition $ref(v_i')$ with a better value. So, if $v_n = max(V_{far}^C \cup \{never\})$, v_n is the best possible, whatever v_p is, and $ref(v_i)$ is better than all other reference partition $ref(v_i')$ with $v_i' > v_i$.

To calculate a reference partition with respect to closeness value v_i for criterion C is to calculate connected components on G_C with considering only v_i' labeled edges such as $v_i' \geq v_i$. We can use Kruskal algorithm (complexity $\mathcal{O}(m \log n)$ for n vertexes and m edges). The connected component idea has been explored in [2] and [1] for ideal cases [1]. In the worst case, we have $k + 1 = |V_{close}^C \cup \{always\}|$ references partitions to find and evaluate. So, this algorithm has $\mathcal{O}((k + 1) * m \log n)$ complexity (see Algorithm 1).

4.2 For an Independant Criteria Set

Let $\mathbb{C}$ be an independent criteria set on an object set $\mathbb{O}$. We calculate, evaluate and compare reference partitions values to determine best partitions values with respect to $\mathbb{C}$ the same way we do for a single criterion (Section 4.1).

What is a reference partition for a criteria set?

---

[1] when there is a valid partition such that the comparison values between objects in the same class are never farness values and comparison values between objects in distinct classes are never closeness values.

Algorithm 1 BestsValuesForASingleCriteria

Require: C: criterion on an objects set $\mathbb{O}$; $G_C = (S, E_C)$: criterion graph
    of C on $\mathbb{O}$;
Ensure: set of best partitions values with respect to C on $\mathbb{O}$
 1: best partitions values set $bestV = \{\}$;
 2: **for all** value $v_i \in V_{close}^C \cup \{always\}$ in $<$ order **do**
 3:    Partition $P = ref(v_p)$;
 4:    Partition value $v = v(P)$;
 5:    **if** P is valid and $\nexists v' \in bestV | v' \succeq v$ **then**
 6:      add v to $bestV$;
 7:    **end if**
 8:    **if** $v(P) = (v_p', v_n')$ such as $v_n = max(V_{far}^C \cup \{never\})$ **then**
 9:      **return** $bestV$;
10:    **end if**
11: **end for**
12: **return** $bestV$;

Definition 12 (Closeness values set) *A closeness values set $\mathbb{VC}$ for a criteria set $\mathbb{C}$ is a set of $v_i | v_i \in V_{close}^{C_i} \cup \{always\}$ and $C_i \in \mathbb{C}\}$ with one and only one closeness value v_i for each criterion $C_i \in \mathbb{C}$.*

Definition 13 (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, $\forall C_i \in \mathbb{C}$, there are $v_i^1 \leq v_i^2$ with $v_i^1 \in (V_{close}^{C_i} \cup \{always\}) \cap \mathbb{VC}1$ and $v_i^2 \in (V_{close}^{C_i} \cup \{always\}) \cap \mathbb{VC}2$.*
    $\mathbb{VC}2$ is a descendant of $\mathbb{VC}1$.

Definition 14 (Reference partition for 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 thinnest partition such as $v(P_{ref}) = \{v(P_{ref}, C_i) \forall C_i \in \mathbb{C}\}$ with $\forall C_i$ criterion: $v(P_{ref}, C_i) = (v_p, v_n) | v_p \leq v_i \in \mathbb{VC}$.*

This algorithm 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 12) for $\mathbb{C}$. We calculate reference partition (Definition 14) for each closeness values set $\mathbb{VC}$: this is done by searching for connected components with Kruskal algorithm (complexity $\mathcal{O}(m \log n)$) on $\mathbb{G}_\mathbb{C}$ with take care only of v_p labeled 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 keep only 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 13) 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 $k = max(|V^{C_i}| \ \forall C_i \in \mathbb{C})$. So, this algorithm has $\mathcal{O}((k+1)^c * m \log n)$ as complexity (see Algorithm 2).

Algorithm 2 BestPartitionsValuesForIndependnantCriteriaSet

Require: $\mathbb{C}$, independent 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** $bestV$;

We have seen algorithms to compute all best partitions values for global semantics. Let us see the local semantics in the next Section.

5   Local Semantics

Our problems are the following:

- to define incoherences in a criteria graph (Section 5.1);

- to decide which incoherences must be processed together (Section 5.2);

- to define what is the coherent part of the criteria graph (Section 5.3);

- to define what is a best partition, with respect to incoherent parts and the coherent part of the graph (Section 5.4).

Let us see them one by one, then, we will see what is the interest of local semantics over the global one in Section 5.5.

5.1   Incoherences in a Criteria Graph

Definition 15 (Incoherent cycle) *Let $\mathbb{C}$ be a criteria set on an objects set $\mathbb{O}$, and the criteria graph $\mathbb{G}_{\mathbb{C}}$. A cycle $Cycle = \{o_1, ..., o_n\}$ in $\mathbb{G}_{\mathbb{C}}$ is incoherent if and only if:*

- *it is a simple cycle (does not repeat vertexes),*

- $\forall i \in [1, ..., n-1], \exists C \in \mathbb{C} | e_C(o_i, o_{i+1}) \in V_{close}^C \cup \{always\}$, *and*

- $\exists C \in \mathbb{C} | e_C(o_n, o_1) \in V_{far}^C \cup \{never\}$.

Note: A cycle $Cycle = \{o_1, o_2\}$ such as $\exists C_i \in \mathbb{C} | e_{C_i}(o_1, o_2) \in V_{close}^C \cup \{always\}$ et $\exists C_j \in \mathbb{C} | e_{C_j}(o_2, o_1) \in V_{far}^C \cup \{never\}$ is incoherent.

Definition 16 (Incoherent edge) *Let $\mathbb{G}_{\mathbb{C}}$ be a criteria graph. An edge is incoherent if and only if it belongs to an incoherent cycle of $\mathbb{G}_{\mathbb{C}}$.*
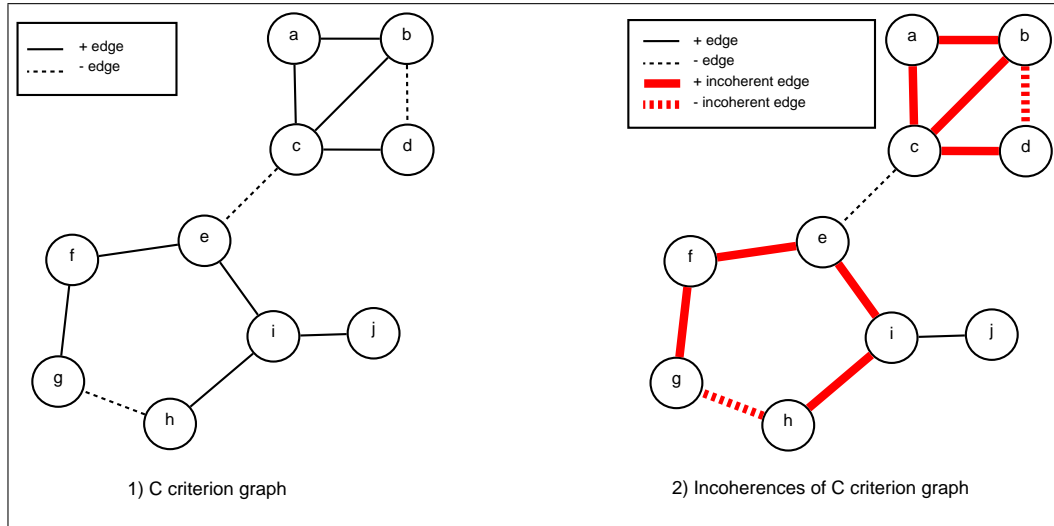

Figure 1: Incoherences in a graph

Example 1 (Incoherent cycle and edges) *Let $\mathbb{C} = \{C\}$ be a criteria set on $\mathbb{O}$, represented by $\mathbb{G}_\mathbb{C}$ graph on Figure 1 part 1. We can identify the following incoherent cycles:*

- $Cycle_1 = \{h, i, e, f, g\}$

- $Cycle_2 = \{b, c, d\}$

- $Cycle_3 = \{b, a, c, d\}$

The incoherent edges are deduced from incoherent cycles and represented Figure 1 part 2.

5.2 Incoherent Edges Group

Let us present an incoherent edge group definition based on connected components of a graph.

Definition 17 (Incoherent edges group) *Let $\mathbb{G}_\mathbb{C}$ be a criteria graph for the objects set $\mathbb{O}$. Let AI_i be a subset of $\mathbb{G}_\mathbb{C}$ edges. AI_i is an incoherent edge group if and only if:*

- *AI_i is the set of edges of a connected component for $\mathbb{G}_\mathbb{C}$ only taking into account closeness edges and always labelled edges.*

- *AI_i contains at least an incoherent edge.*

Example 2 (Incoherent edges group) *Let us follow the Example 1 represented Figure 1. The set of incoherent edges represented Figure 1 part 2 is $AI = \{(a, b), (a, c), (b, c), (b, d), (c, d), (e, f), (f, g), (g, h), (h, i), (i, e)\}$. We deduce two incoherent edges groups:*

- *$AI_1 = \{(a, b), (a, c), (b, c), (b, d), (c, d)\}$, and*

- *$AI_2 = \{(e, f), (f, g), (g, h), (h, i), (i, e), (i, j)\}$.*

5.3 Coherent Edges Group

A coherent edge group of a graph $\mathbb{G}_\mathbb{C} = (V, E)$ is the set of all edges in E such as they are not also in an incoherent edge group (Definition 17).

Definition 18 (Coherent edges group) *Let $\mathbb{G}_\mathbb{C}$ be a criteria graph for objects set $\mathbb{O}$. We denote A the $\mathbb{G}_\mathbb{C}$ edges set and $AI = \{AI_1, ...AI_i\}$ the set of all incoherent edges groups of $\mathbb{G}_\mathbb{C}$. AC is the coherent edge group of $\mathbb{G}_\mathbb{C}$ means $AC = A \backslash AI$.*

5.4 Locality Notion, Semantics Adaptation and Computational Complexity

This semantics deals with every incoherent edge group separately from others edges groups (Definitions 17 and 18).

Definition 19 (Best partition with local semantics) *Let $\mathbb{G} = (V, E)$, be a criterion graph. Let AI_j, be the incoherent edges group j of $\mathbb{G}$ (Definition 17). A best partition on $\mathbb{G}$ is a partition such as it has:*

- *a best value for $\mathbb{G}c = (V, AC)$, and*

- *a best value for each $\mathbb{G}i_j = (V, AI_j)$, with AI_j, incoherent edge group of $\mathbb{G}$.*

$\mathbb{G}c$ is the coherent part of $\mathbb{G}$, and the $\mathbb{G}i_j$ are its incoherent parts.

Example 3 (Best partition with locality) *Let us take back the Example 2 and represent the graph $\mathbb{G} = (V, E)$ on Figure 2 part 1. A best partition is one with a best value on each separated subgraphs "Coherent part", "Incoherent part 1" and "Incoherent part 2". So, we can privilege some comparison values for "Incoherent part 1" and not the same comparison value for "Incoherent part 2". These parts of $\mathbb{G}$ are represented on Figure 2 part 2.*
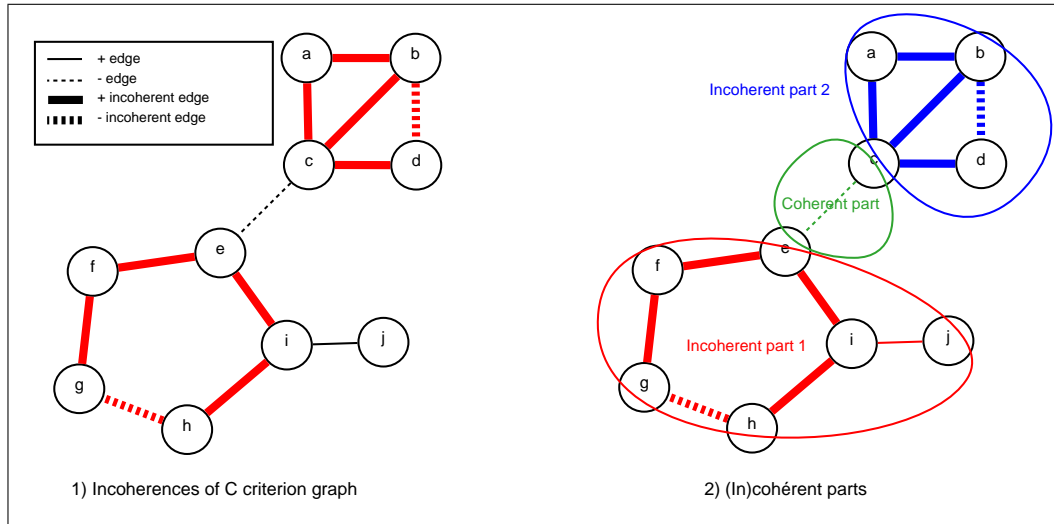

Figure 2: (In)coherent parts of C criterion graph

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 components algorithm (complexity $\mathcal{O}(m \log n)$ [2]). Then, for the

---

[2]with n vertexes and m edges

coherent part and each of the at most $n/2$ incoherent part [3], we execute algorithm of global semantics (complexity $\mathcal{O}((k+1)^c * m \log n)$ [4] Section 3.3). So, the complexity in the worst case is: $\mathcal{O}(n * (k+1)^c * m \log n)$.

5.5 Local Semantics Interest

The interest of this semantic approach is the following one. Let $\mathbb{G}^a = (V_a, E_a)$ and $\mathbb{G}^b = (V_b, E_b)$ be two criteria graph on two distinct objects set totally independent (each comparison between an object of V_a and an object of V_b has a value in $v \in V_{far} \cup \{neutral, never\}$) with respect to considered criteria. To find the best partitions on $\mathbb{G}^a$ or $\mathbb{G}^b$ or on $\mathbb{G}^{a \cup b}$ (the criteria graph on $\{V_a \cup V_b\}$ objects set) has same results with this semantics. In fact, if $\mathbb{G}^a$ and $\mathbb{G}^b$ have not a single incoherent edge, the results are equal for global and local semantics. It will be also the same for local semantics when there are incoherences because they will be treated separately (because there is not a closeness or *always* labelled edge between a vertex from V_b and a vertex from V_a). However, it is not always the case for global semantics when there are incoherences.

6 Conclusion

In this technical report we represented the partitioning problem as a complete graph partitioning problem. More precisely we defined a complete multi criteria graph such that the nodes represent the objects to partition and the multi edges between two nodes are labeled with the comparison value of each of the criterion between two objects. We described two partitioning semantics of these graphs and presented algorithms to find all best partition values according to these partitioning semantics.

References

[1] Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. In *MACHINE LEARNING*, pages 238–247, 2002.

[2] A. Guénoche. Partitions optimisées selon différents critères: évaluation et comparaison. *Mathématiques et sciences humaines. Mathematics and social sciences*, (161), 2003.

---

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

[4]with, c, criteria number and $k = max(|V^{C_i}| \; \forall C_i \in \mathbb{C})$.