Two Antagonistic Objectives For One Multi-Scale Graph Clustering Framework

Bruno Gaume,[1,3*] Ixandra Achitouv,[3] David Chavalarias[2,3*]

[1]Cognition, Langues, Langage, Ergonomie (CLLE, UMR 5263), CNRS, France
[2]Centre d'Analyse et de Mathématique Sociales (CAMS, UMR8557), CNRS, France
[3]Complex Systems Institute of Paris Île-de-France (ISC-PIF, UAR3611), CNRS, France
*To whom correspondence should be addressed: bruno.gaume@iscpif.fr, david@chavalarias.org

In the current state of knowledge, there is no consensus on an objective criterion for evaluating network communities as *cohesive sets of nodes* with the following two properties:

$\mathrm{P_{DC}}$: **Each community is *Densely Connected*;**

$\mathrm{P_{WC}}$: **Communities are *Weakly Connected* to each other.**

This makes it difficult to conduct comparative studies between dozens of graph clustering methods proposed over more than 20 years. To fill this gap:

■ **We propose a graph clustering framework by faithfully formalizing $\mathrm{P_{DC}}$ with $precision$ and $\mathrm{P_{WC}}$ with $recall$, which are two meaningful metrics, simple, well known and already widely used for many tasks in most sciences. The meaning of these metrics in the context of graph clustering is therefore easily interpretable by most users of real-world graphs.**

■ **We show that for most graphs, these two metrics are antagonistic, i.e. there is no solution that simultaneously maximizes $precision$ and $recall$. In other words, to select a clustering among the Pareto optimal solutions (clusterings such that no other clustering exist that both increases the $precision$ and the $recall$) we must first make a subjective compromise, according to our needs between the two properties $\mathrm{P_{DC}}$ and $\mathrm{P_{WC}}$.**

■ **We then show how to use this framework to compare, even without 'ground truth', the performances of five hitherto incommensurable state-of-the-art clustering methods, as well as that of a new family of clustering methods inspired by our approach.**

Introduction

From biology to social sciences, ecosystems or computer science, complex systems are defined as large sets of entities interacting in a decentralized ways. Graphs are one of the main conceptual structures for modeling them, where nodes represent the basic lowest-level entities and edges represent their interactions. Detecting communities in these graphs is a fundamental task for the study of the complex systems.

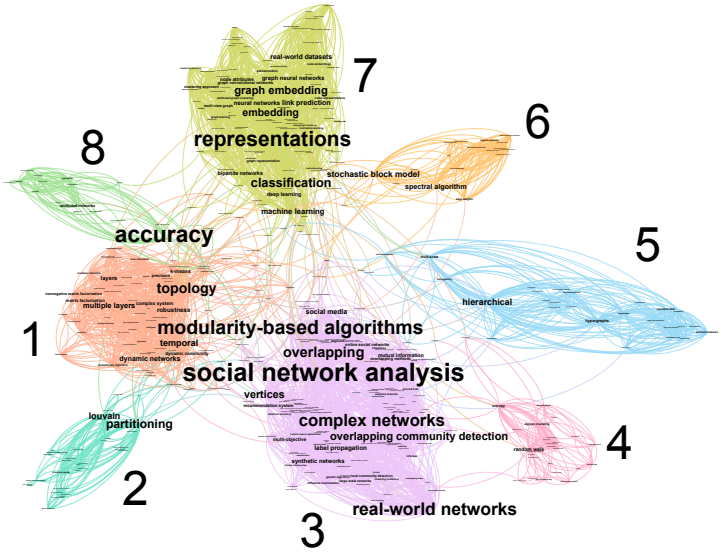


Figure 1: **Bibliometric analysis of 4,959 publications from the Web of Science Core Collection (extracted on October 7th 2024) for the query "graph clustering" OR "community detection"**. Size of terms is proportionnal to their occurrences in the corpus and weighted links represent the maximum of the two conditional probabilities of having a terms knowing the other in a document. Nodes of same color belong to the same subfield of research. Cluster 1 to 6 deal with variants of topological approaches to graph clustering (for temporal graphs, multi-layer graphs, hierachical graphs, etc.). Clusters 7 and 8 are more recent (emergence in the 2020s) and deal with graph embedding and attributed networks. This map has been made with the Gargan-Text libreware (*1*) and is available as interactive visualisation on http://maps.gargantext.org.

After 20 years of research, community detection in graphs is still considered as a challenging and *"not very well posed"* problem (*20*). A bibliometric analysis of 4,959 publications over the last 5 years from the Web of Science Core Collection on the topic "graph clustering" OR "community detection" (Fig. 1) reveals an intense activity with a fragmented landscape dominated by many different but exclusively topological approaches (*graph clusterings*: see for exemple (*3, 21, 31, 52*)).

A complementary and fast growing subdomain since the 2020s is leveraging the recent advances of deep learning to classify a wide range of data structures, and deal with graphs through their representation in latent spaces (*11*) (see clusters 7 and 8 in Fig 1), taking into account not only the graph topology but also some nodes attributes (*attributed graph clusterings*: see for exemple (*4–6, 12, 15, 23, 51, 56*)).

There is however a Commonly Accepted Informal Definition (hereafter *CAID*) of what is expected from network communities with respect to the edges alone without taking into account other node characteristics (*5*) *"Generally, a community is usually regarded as a group of nodes that are closely connected internally, whereas the external links between different communities are sparse"*.

Despite this *CAID* there is still no consensus on an objective formal criterion for evaluating network communities. This prevents a constructive debate on the comparative advantages of the many proposed clustering methods. This difficulty has been known for a long time (*43, 55*) and has led some to conclude that the problem of community detection cannot be solved universally (*25, 55*).

In order to propose a theoretical framework taking into account only the graph topology to compare the accuracies of different network communities, we mathematically formalize this *CAID*. Our main point is that this *CAID* basically corresponds to the optimization of two antagonistic objectives, and that the proposed clustering methods can only be evaluated and compared if we first define the desired trade-off between these two antagonistic objectives.

A formal definition of community detection objectives

Communities as *Cohesive sets of nodes* are generally called *blocks, clusters* or *modules*, and we can break down this *CAID* into two distinct module properties:

$\mathbf{P_{DC}}$: Each module is *Densely Connected*;

$\mathbf{P_{WC}}$: Modules are *Weakly Connected* to each other.

These two properties put into synergy are in accordance with the common sense of what communities should materialize in the literature (*13, 38, 39, 54, 55*). Such modules are an essential feature of the graph structures extracted from real-world data (*8, 22, 35, 50, 57*) and are central to understanding a wide range of phenomena in complex systems.

We then propose a graph clustering framework (thereafter called $nPnB$) by faithfully formalizing these two properties with two meaningful metrics, $precision$ and $recall$ which are easily interpretable by most users of real-world graphs.

We show that generally, these two metrics are antagonistic, i.e. there is no solution that simultaneously maximizes $precision$ and $recall$. In other words, to select a clustering among the Pareto optimal solutions (clusterings such that no other clustering exist that both increases the $precision$ and the $recall$) we must first make a subjective compromise, according to our needs between $precision$ (each module is *densely connected*) and $recall$ (the modules are *weakly connected* between them).

We then show how to compare in this framework the performances of five state-of-the-art methods $Spectral\text{-}Graph\text{-}Clustering$, $Louvain$, $Infomap$, $Starling$, $Oslom$ and BEC, this latter being a new family of clusterings methods introduced in this paper. Until now, these methods were incommensurable without ground truth associated with the network.

We provide in Sect. 1 a quick state-of-the-art on the metrics intended to evaluate the quality of graph clusterings. In Sect. 2 we formalize the properties P_{DC} and P_{WC} with the two simple metrics, respectively $precision$ and $recall$. We show in section 2.3 that these two metrics are antagonistic, and we propose to use a third measure that integrates both of them and is parameterized by a trade-off between each of them. We shows in section 3 that this makes it possible to find overlapping or non-overlapping communities in real-world graphs. The identified clusters of the graph (sets of nodes) represent then some entities at higher scales of description (sets of basic entities) of the modeled system under this trade-off (*30, 32, 40, 46*). Since this trade-off allows to control the size of the meso-level entities under study, we call it *description scale*. In Sect. 4, we show how to compare the performances of different methods according to the chosen description scale, and in Sect. 5 we discuss the benefit of having a clear and simple definition of this description scale in a graph clustering framework.

1 Evaluating graph clusterings

For a set of vertices V, let's note $\mathcal{P}(V)$ the subsets of V and $\mathcal{P}_2(V) \subset \mathcal{P}(V)$ the pairs of elements from V. For $E \subset \mathcal{P}_2(V)$, $G = (V, E)$ defines an undirected graph on V. By definition, a set $\mathcal{C} \in \mathcal{P}(\mathcal{P}(V))$ such that $\mathcal{C} = \{C_i | C_i \subset V, C_i \neq \emptyset, i \in I\}$ is a clustering of G with clusters C_i if and only if $\bigcup_{i \in I} C_i = V$. It is a *partitional clustering* if clusters do not overlap ($\forall i \neq j \in I, C_i \cap C_j = \emptyset$), else it is an *overlapping clustering*.

The number of partitional clustering of a set of size $n = |V|$ is equal to the n^{th} Bell number, a sequence known to grow exponentially (*26*). Consequently, this definition tells us what a clustering is, but not what a *'good clustering'* is among the huge number of possible clustering. Therefore, users of real-world graphs need metrics to evaluate clusterings according to their expectations. The state-of-the-art (*13, 20, 58*) identifies more than 70 different metrics to evaluate the qualities of a clustering. These metrics fall into two categories:

Intrinsic metrics aiming at evaluating clustering in relation to a graph G alone according to some general principles (like the $Modularity$ of (*38, 39*) or the $Compressibility$ of (*47*)). We can also use these metrics to find clustering on G;

Extrinsic metrics aiming at evaluating clustering in relation to *a priori* known structures such as clusters of synthetic networks (*27*) or 'ground-truth' built by experts or from metadata (*43*) (like the $RandIndex$ of (*45*)).

Then in each category, one can find metrics for *partitional clustering* or *overlapping clustering*. This leads to four kinds of metrics: *intrinsic* or *extrinsic*, for clusters with or without overlapping (see Table 1).

	Intrinsic	**Extrinsic**
P	$\mathbf{\lvert M_{P,In}\rvert \approx 42}$ • Internal density; • Cut Ratio; • Modularity; • Map equation; ...	$\mathbf{\lvert M_{P,Ex}\rvert \approx 14}$ • precision; • recall; • Fscore; • RandIndex; • Entropy; ...
O	$\mathbf{\lvert M_{O,In}\rvert \approx 11}$ • Fuzzy Modularity; • Fitness Function; • Overlap. Modularity Density; ...	$\mathbf{\lvert M_{O,Ex}\rvert \approx 4}$ • Overlap. Normalized Mutual Information; • Generalized External Index; ...

Table 1: **Four species of metrics with some examples (see (*13*)).** **P** stands for partitional clustering; **O** for overlaping clustering.

It is worth noting that according to the definitions of these metrics: $M_{P,In} \cap M_{P,Ex} \cap M_{O,In} \cap M_{O,Ex} = \varnothing$, and even worse, in the state-of-the-art, each metric in $M_{P,In}$ has its own semantic and there is no consensus to designate one which could serve as an objective criterion for comparing the intrinsic qualities of two partitional clusterings. But then, how to compare two partitional clusterings on a graph without any associated ground truth?

2 Interpreting graph clusterings as edge classifiers

So far, no unified framework has been proposed to objectively evaluate in the same way, partitional and overlapping clusterings, both intrinsically and extrinsically (see (*13, 20, 58*)). To fill this gap, we propose to interpret graph clusterings as constrained binary classifiers of node pairs intended to find the edges of the graph.

2.1 Classical binary classification in experimental sciences

In all experimental sciences, binary classification is an important task which consists in separating a set of elements Ω into two complementary subsets $(A, \overline{A})$ where the elements of A are the *positive assignments* and those of $\overline{A}$ are the *negative assignments*.

Given two set Δ, Ω such $\Delta \subset \Omega$ and $(A, \overline{A})$ a binary classifier on Ω intended of modeling Δ, there are four basic combinations described in Table 2.

	Δ	$\overline{\Delta}$
A	Correct positive assignments True Positives: $\mathbf{TP}= A \cap \Delta$	Incorrect positive assignments False Positives: $\mathbf{FP} = A \cap \overline{\Delta}$
$\overline{A}$	Incorrect negative assignments False Negatives: $\mathbf{FN}= \overline{A} \cap \Delta$	Correct negative assignments True Negatives: $\mathbf{TN}= \overline{A} \cap \overline{\Delta}$

Table 2: **Four basic combinations:** A binary classifier $(A, \overline{A})$ intended of modeling a set $\Delta \subset \Omega$.

There are a large number of studies on measures based on these four basic outcomes TP, TN, FP, FN to evaluate the qualities binary classifiers $(A, \overline{A})$ intended of modeling various set of elements $\Delta \subset \Omega$, for exemple the two classical metrics $Precision((A, \overline{A}), \Delta) = \frac{|TP|}{|TP|+|FP|}$ and $Recall((A, \overline{A}), \Delta) = \frac{|TP|}{|TP|+|FN|}$. Of course, the studied elements $\Delta \subset \Omega$ vary according to the disciplines, and $Precision((\Delta, \overline{\Delta}), \Delta) = \frac{|TP|}{|TP|+|FP|} = Recall((\Delta, \overline{\Delta}), \Delta) = \frac{|TP|}{|TP|+|FN|} = 1$. That is to say that for any classical binary classification task, $Precision$ and $Recall$ are not *antagonistic* i.e. there exists one binary classifier $(\Delta, \overline{\Delta})$ such its $Precision = 1$ and its $Recall = 1$ at the same time.

However, it is often difficult to define a binary classification method met perfectly modeling $\Delta \subset \Omega$ such $met(\Omega, \boxplus) = (\Delta, \overline{\Delta})$ where $\boxplus$ is a set of various information about the elements of Ω. Most of the time $Precision(met(\Omega, \boxplus), \Delta) < 1$ and $Recall(met(\Omega, \boxplus), \Delta) < 1$. But then, how to compare two methods met_1 and met_2 such that $Precision(met_1(\Omega, \boxplus), \Delta) < Precision(met_2(\Omega, \boxplus), \Delta)$ and $Recall(met_2(\Omega, \boxplus), \Delta) < Recall(met_1(\Omega, \boxplus), \Delta)$ or vice versa.

Also, in order to be able to compare the performance of different methods despite these difficulties, for many years and up to today, many studies propose various strategies in order to be able to trade simultaneously with these four basic outcomes TP, TN, FP, FN. As shown in (*10, 16, 41, 44, 48, 49, 53, 60*), using various trade-off on these four basic outcomes is most often a good way to compare different methods.

2.2 nPnB: binary classifier of nodes Pairs by nodes Blocks

Definition: Let an undirected graph $G = (V, E)$. A $nPnB$ is a *binary classifier of* **n***odes* **P***airs by* **n***odes* **B***locks*. Instead of providing $(A, \overline{A})$ two complementary sets of nodes pairs as $Positive\ Pairs$ and $Negative\ Pairs$, a $nPnB$ classifier has to provide its predictions in the form of nodes blocks $\{C_i | C_i \subset V, C_i \neq \emptyset, i \in I\} = \mathcal{C}$, a clustering such that $\{x, y\}$ is a $Positive\ Pair$ if and only if $\exists C_i \in \mathcal{C}$ such $\{x, y\} \in C_i$, otherwise $\{x, y\}$ is a $Negative\ Pair$. We will note $nPnB^P$ clusterings for which blocks are not overlapping and $nPnB^O$ those allowing overlap.

In the $nPnB$ framework, given an undirected graph $G = (V, E)$, the set of the elements to be divided into two groups by a $nPnB$ classifier is $\Omega = \mathcal{P}_2(V)$, the subset to be modelized is defined by $\Delta = E \subset \mathcal{P}_2(V)$, the information about an element $\{x, y\} \in \mathcal{P}_2(V)$ is $[G]_{x,y}$ where $[G]$ is the adjacency matrix[1] of the graph G.

This can be formalized by defining $\widehat{\mathcal{C}} = \big(U(\mathcal{C}),\ \Xi(\mathcal{C})\big)$ the derived graph from a clustering $\mathcal{C}$ using the following two functions:

$$U : \mathcal{P}(\mathcal{P}(V)) \longrightarrow \mathcal{P}(V),\ U(\mathcal{C}) = \bigcup_{C_i \in \mathcal{C}} C_i \tag{1}$$

$$\Xi : \mathcal{P}(\mathcal{P}(V)) \longrightarrow \mathcal{P}_2(V),\ \Xi(\mathcal{C}) = \bigcup_{C_i \in \mathcal{C}} \mathcal{P}_2(C_i) \tag{2}$$

These functions satisfy the following properties:

- If $\mathcal{C}$ is a clustering of a graph $G = (V, E)$ then $U(\mathcal{C}) = V$;
- $\forall\ E \subset \mathcal{P}_2(V),\ \Xi(E) = E$;
- $\forall\ \mathcal{C} \subset \mathcal{P}(\mathcal{P}(V)),\ \Xi(\Xi(\mathcal{C})) = \Xi(\mathcal{C})$.

For any clustering $\mathcal{C}$ on a graph $G = (V, E)$ we can then compute the two classical metrics in diagnostic binary classification which assess the capacities of the graph $\widehat{\mathcal{C}} = \big(U(\mathcal{C}),\ \Xi(\mathcal{C})\big)$ to detect the edges of the graph $G = (V, E)$ (see Fig. 2):

$$\textbf{Precision: } P(\widehat{\mathcal{C}}, G) = \frac{|TP|}{|TP| + |FP|} = \frac{|\Xi(\mathcal{C}) \cap E|}{|\Xi(\mathcal{C})|} \tag{3}$$

This is the probability that an edge drawn at random in $\Xi(\mathcal{C})$ (edges of $\widehat{\mathcal{C}}$), actually belongs to E (edges of G);

[1] In this article, we limit ourselves to using only information of the graph alone, without taking into account all other possible characteristics of the nodes (which is the case of its ajacency matrix).

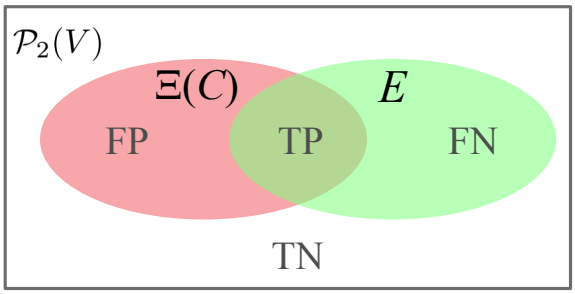
$$\textbf{Recall: } R(\widehat{\mathcal{C}}, G) = P(G, \widehat{\mathcal{C}}) = \frac{|TP|}{|TP| + |FN|} = \frac{|\Xi(\mathcal{C}) \cap E|}{|E|} \tag{4}$$

This is the probability that an edge drawn at random in E (edges of G), belongs to $\Xi(\mathcal{C})$ (edges of $\widehat{\mathcal{C}}$).

The two properties P_{DC} and P_{WC} are then faithfully formalized by these two metrics $precision$ and $recall$ in the sense that:

$\mathbf{P_{DC}}$: The more each module is densely connected, the higher the $precision$ and vice versa;

$\mathbf{P_{WC}}$: The less the modules are connected to each other, the higher the $recall$ and vice versa.



$\mathbf{TP} = \Xi(\mathcal{C}) \bigcap E$ is the set of True Positives;

$\mathbf{TN} = \overline{\Xi(\mathcal{C})} \bigcap \overline{E}$ is the set of True Negatives;

$\mathbf{FP} = \Xi(\mathcal{C}) \bigcap \overline{E}$ is the set of False Postives;

$\mathbf{FN} = \overline{\Xi(\mathcal{C})} \bigcap E$ is the set of False Negatives;

Figure 2: **Venn diagram of the capacities of the graph $\widehat{\mathcal{C}} = \big(U(\mathcal{C}),\ \Xi(\mathcal{C})\big)$ to detect the edges of the graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$.** Interpreting graph clustering as binary classifier of node Pairs by node Blocks with to formula 2.

Moreover, these two definitions make sense for all clusterings $\mathcal{C}$ with or without overlaps because $\Xi(\mathcal{C})$ makes sense for all clusterings, and for any clustering on a graph $G = (V, E)$, $precision$ and $recall$ satisfy the following property:

$$\begin{gathered} P(\widehat{\mathcal{C}}, G) = 1 \ \& \ R(\widehat{\mathcal{C}}, G) = 1 \\ \Updownarrow \\ U(\mathcal{C}) = V \ \& \ \Xi(\mathcal{C}) = E \\ \Updownarrow \\ \widehat{\mathcal{C}} = G \end{gathered}$$

Because the two metrics $precision$ and $recall$ are such $R(\widehat{\mathcal{C}}, G) = P(G, \widehat{\mathcal{C}})$, they are symmetrical to each other, and we will see in next section that they are moreover antagonistic.

2.3 Graph clustering as a task with two antagonistic objectives

The set of the edges E and the set of maximal cliques $\mathcal{C}_{mc}$ on the graph $G = (V, E)$ are such that $\widehat{E} = \widehat{\mathcal{C}_{mc}} = G$ ($precision = 1$ and $recall = 1$, they are maximal). That is to say that for any graph $G = (V, E)$ there exists at least one clustering $\mathcal{C}$ such $P(\widehat{\mathcal{C}}, G) = R(\widehat{\mathcal{C}}, G) = 1$.

When $E \neq \varnothing$, by removing overlaps between maximal cliques, it is always possible to build a partitional clustering $\mathcal{C}$ from $\mathcal{C}_{mc}$ such that $P(\widehat{\mathcal{C}}, G) = 1$; and it is also straightforward to say that the partitional clustering $\{V\}$ is such that $R(\widehat{\{V\}}, G) = 1$. Consequently, partitional clustering such that either $P(\widehat{\mathcal{C}}, G) = 1$ or $R(\widehat{\mathcal{C}}, G) = 1$ always exist. But unless $G = (V, E)$ is reduced to a set of unconnected cliques, the following proposition implies that there is no partitional clustering $\mathcal{C}$ such $P(\widehat{\mathcal{C}}, G) = R(\widehat{\mathcal{C}}, G) = 1$. The two metrics $precision$ and $recall$ are thus generally $antagonistic$.

Proposition: Let an undirected graph $G = (V, E)$ and a partitional clustering $\mathcal{C} = \{C_i | C_i \subset V, C_i \neq \emptyset, i \in I\}$ such $\bigcup_{i \in I} C_i = V,$ and $\forall i \neq j \in I, C_i \cap C_j = \varnothing$. Then:

$$\textbf{[A]}\ P(\widehat{\mathcal{C}}, G) = R(\widehat{\mathcal{C}}, G) = 1$$
$$\Downarrow$$
[B] The graph $G = (V, E)$ is reduced to a set of unconnected cliques.

Proof :

(1) $P(\widehat{\mathcal{C}}, G) = 1 \iff \forall C_i \in \mathcal{C}, C_i$ is a clique of G (by the formulas 3 and 2);

(2) $R(\widehat{\mathcal{C}}, G) = 1 \iff \forall i \neq j \in I, \nexists \{x, y\} \in E$ such $x \in C_i$ and $y \in C_j$ (because $\forall i \neq j \in I, C_i \cap C_j = \varnothing$ and by the formulas 4 and 2);

$\big($**(1)** and **(2)**$\big) \Longrightarrow \big($**[A]** $\Longrightarrow$ **[B]**$\big)$ ■

Since $precision$ and $recall$ are generally $antagonistic$, finding sets of non overlapping clusters on networks can be envisioned as a non trivial bi-objective task in the $nPnB^P$ framework.

To illustrate the antagonism of the two metrics $precision$ and $recall$ in $nPnB^P$ framework, we consider a toy graph G_{toy}, all its possible partitions and their respective $precision$ and $recall$ scores (cf. Fig. 3). Some of these partitions are optimal, *i.e.* they have the properties that no other partition exist that both increases the $precision$ and the $recall$. This special set of partitions is called the *Pareto front* – or Pareto optimal solutions, noted $Pos(Gtoy)$.

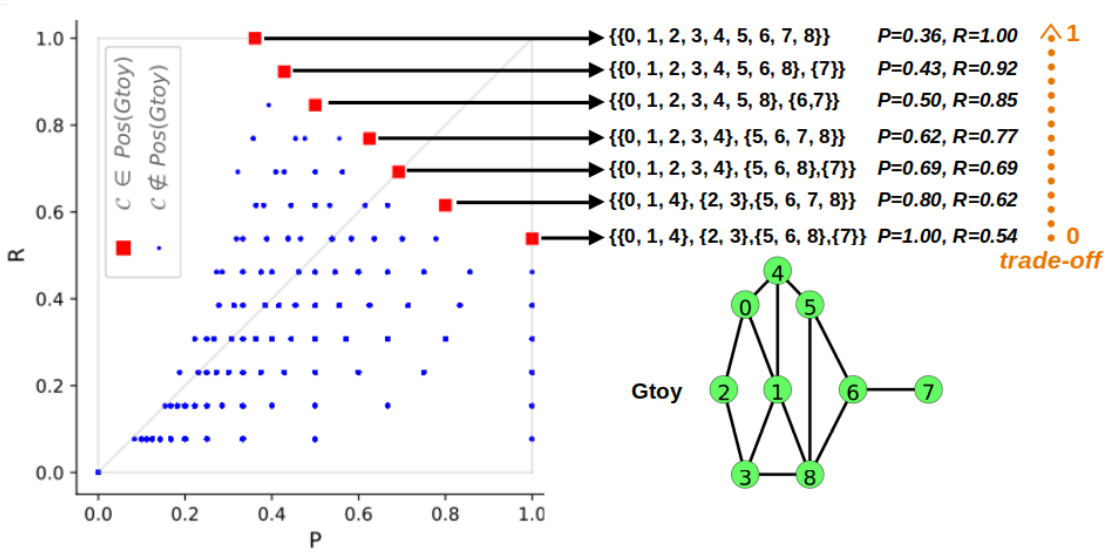


Figure 3: **The set of all the $21,147$ partitional clustering of Gtoy in the precision (P) $\times$ recall (R) space.** The Pareto front is highlighted in red.

With the graph $Gtoy$ it is clear that $precision$ and $recall$ are $antagonistic$ (i.e. $\nexists \mathcal{C} \in nPnB^P$ such $P(\widehat{\mathcal{C}}, Gtoy) = R(\widehat{\mathcal{C}}, Gtoy) = 1$), therefore to select an optimal partition clustering from the Pareto front, we need to specify our priorities in terms of $precision = \frac{|TP|}{|TP|+|FP|}$ faithfully formalizing P_{DC} and $recall = \frac{|TP|}{|TP|+|FN|}$ faithfully formalizing P_{WC}. This decision will depend on the needs of the modeler, from which she will define what constitutes a "good clustering". It's only once the trade-off between $precision$ and $recall$ has been made that the modeler can assess the performances of different clustering methods.

In the state-of-the-art there are numerous measures combining TP, TN, FP and FN (For example $Jaccard = \frac{TP}{TP+FP+FN}$ proposed in (*24*) or $RandIndex = \frac{TP+TN}{TP+TN+FP+FN} = \frac{TP+TN}{\binom{|V|}{2}}$ proposed in (*45*) or $MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$ proposed in (*34*)).

The best known, the most used to evaluate binary classifier, with a hands s on the trade-off between $precision$ and $recall$ is the F-score function: $\frac{(1+f(s)^2).|TP|}{(1+f(s)^2).|TP|+f(s)^2.|FN|+|FP|} =$

$$F_s(P, R) = \frac{(1 + f(s)^2).(P.R)}{R + f(s)^2.P} \tag{5}$$

With $s \in [0, 1]$, $f(s) = tan(\frac{\pi.s}{2})$ and $F_1(P, R) = R$.

For $s = 0.5$, $precision$ and $recall$ are of the same importance, for $s = 0$, only the $precision$ counts, whereas for $s = 1$, only the $recall$ counts.

- For $s \in \ ]0,\ 0.5[$: In order to improve $F_s(P, R)$, $precision$ need to be higher with a greater

number of smaller and denser modules;

- For $s \in \ ]0.5,\ 1[$: In order to improve $F_s(P, R)$, $recall$ need to be higher with a fewer number of bigger but less dense modules.

Thereby, the trade-off s between $precision$ and $recall$ can be used to adjust the desired granularity of the clustering, *i.e.* the desired *'description scale'*. The modules of the graphs then represent the observed entities at *'description scale'* s (*30, 32, 40, 46*).

2.4 nPnB as Unified Graph Clustering Framework

In the state-of-the-art $M_{P,In} \cap M_{P,Ex} \cap M_{O,In} \cap M_{O,Ex} = \varnothing$. But interpreting graph clustering as $nPnB$, and using the Ξ function to define the four metrics TP, TN, FP, FN makes it possible to use these four metrics for clustering with and without overlaps, intrinsically against the original graph $G = (V, E)$ (as in Fig. 2) or extrinsicaly against a ground-truth $\mathcal{C}_G^{ref}$, by replacing E, the edges of the graph G, by $\Xi(\mathcal{C}_G^{ref})$, the edges of the graph $\widehat{\mathcal{C}_G^{ref}}$: $\mathbf{TP}= \Xi(\mathcal{C}) \bigcap \Xi(\mathcal{C}_G^{ref})$; $\mathbf{TN}= \overline{\Xi(\mathcal{C})} \bigcap \overline{\Xi(\mathcal{C}_G^{ref})}$; $\mathbf{FP}= \Xi(\mathcal{C}) \bigcap \overline{\Xi(\mathcal{C}_G^{ref})}$; $\mathbf{FN}= \overline{\Xi(\mathcal{C})} \bigcap \Xi(\mathcal{C}_G^{ref})$.

Any metric $\maltese$ defined as a function of these four metrics is now such that $\maltese \in M_{P,In} \cap M_{P,Ex} \cap M_{O,In} \cap M_{O,Ex}$ (for exemple P, R, F_s, $Jaccard$, $RandIndex$ and MCC). This is the key change of perspective brought about by this article. Therefore we can now objectively evaluate the quality of any clustering $\mathcal{C}$ in the 2-dimensional space $precision \times recall$ with the metrics P, R and in a 1-dimensional space with the metric F_s once the description scale s has been chosen:

- **Intrinsically:** $\big(P(\widehat{\mathcal{C}}, G), R(\widehat{\mathcal{C}}, G)\big)$ and $F_s(P(\widehat{\mathcal{C}}, G), R(\widehat{\mathcal{C}}, G))$;
- **Extrinsically:** $\big(P(\widehat{\mathcal{C}}, \widehat{\mathcal{C}_G^{ref}}), R(\widehat{\mathcal{C}}, \widehat{\mathcal{C}_G^{ref}})\big)$ and $F_s(P(\widehat{\mathcal{C}}, \widehat{\mathcal{C}_G^{ref}}), R(\widehat{\mathcal{C}}, \widehat{\mathcal{C}_G^{ref}}))$.

Moreover, these metrics are very simple, well known and already widely used for many tasks in most sciences (*9, 17*). They are the natural building blocks of a framework that would allow modelers to compare clustering methods according to their needs with respect to both properties P_{DC} (faithfully formalized by $precision$) and P_{WC} (faithfully formalized by $recall$).

3 New clustering methods based on F_s optimization

$nPnB$ framework can help to compare clustering methods. Can it also be used to define new clustering methods? We demonstrated that in $nPnB$ framework $F_s \in M_{P,In} \cap M_{P,Ex} \cap M_{O,In} \cap M_{O,Ex}$. In particular, $F_s \in M_{P,In} \cap M_{O,In}$ and can now be used as intrinsic metric to be

optimized to find graph clusterings with or without overlaps, which we will now illustrate by presenting the BEC algorithm (Binary Edges Classifier).

Since the set of partitional clusterings of a graph G is finite, the set of the optimal clusterings $Pos(G)$ is finite and $Pos(G) \neq \varnothing$ (like $Pos(Gtoy)$ in Fig. 3). Then given an undirected and unweighted graph $G = (V, E)$ and a desired description scale $s_p \in [0, 1]$ there exist at least one clustering $\mathcal{C}^* \in Pos(G)$ such that $F_{s_p}(P(\widehat{\mathcal{C}^*}, G), R(\widehat{\mathcal{C}^*}, G))$ is maximal.

Finding such optimal clustering by examining all the partitional clusterings of a graph $G = (V, E)$ is however often untracktable since the number of partionings grows exponentially in function of $|V|$. Therefore we will use a heuristic that can find in a reasonable amount of time a clustering $\mathcal{C}$ that tentatively optimises $F_{s_p}(P(\widehat{\mathcal{C}}, G), R(\widehat{\mathcal{C}}, G))$.

The trivial clustering $\mathcal{C} = \{\{i\} |\ i \in V\}$ where each vertice is assigned to its own cluster is a partitional clustering. Then $\forall s \in [0, 1]$ its F_s score $F_s(P(\widehat{\mathcal{C}}, G), R(\widehat{\mathcal{C}}, G)) = 0$ since its $recall$ $R(\widehat{\mathcal{C}}, G) = 0$.

We can then improve this trivial clustering by an agglomeration process that reviews each edge of G only once and merges the clusters of their vertices if this operation does not decrease $F_{s_p}(P(\widehat{\mathcal{C}}, G), R(\widehat{\mathcal{C}}, G))$ (*cf.* SI Algorithm 1: BEC^{s_p} for pseudo-code).

The order in which edges are traversed is essential. The proposed algorithm involves choosing an ordering function on E derived from a similarity measure $Sim(G, x, y)$ on the vertices of G. Edges $\{x, y\} \in E$ are then reviewed by descending order of $Sim(G, x, y)$.

The quality of the process depends on the choice of the similarity measure, and there is no guarantee of obtaining an optimal clustering $\mathcal{C}^*$ such $F_{s_p}(P(\widehat{\mathcal{C}^*}, G), R(\widehat{\mathcal{C}^*}, G))$ is maximal. However, few trials are sufficient to find similarity measures such that the associated partitional clustering outperforms state-of-the-art partitional clustering methods in the $nPnB$ framework.

We tested 84 state-of-the-art similarities (*37*), with many real-world graphs, and benchmarks proposed in (*18, 19*), (*38*) and (*27*). One of the best scalable metrics was $CosP$ which has been chosen in the subsequent application (*cf.* SI A.1).

This approach can be generalized to define families of overlapping clustering $BEC^{s_p}_{s_o}$ where s_p defines the desired scale of description and s_o defines the desired amount of overlapping that we will call *stickiness scale* (*cf.* SI A.2 and Algorithm 2: $BEC^{s_p}_{s_o}$ for pseudo-code).

4 Clusterings comparison in the nPnB framwork

In the following, we will distinguish the use of Eq. 5 as the function F_s to be optimized for the family BEC^s using the variable name s, and its use as the function F_σ in the context of the selection of the description scale using the variable name σ to evaluate the various clustering methods. Note that the metric $F_{\sigma=0.5}$ gives equal importance to $precision$ and $recall$ (Eq. 5);

and can be interpreted as a 'middle point of view'. It has both homogeneity and completeness, two fundamental properties for metrics intending compare clusterings (*2*). On contrary, $precision$ has only homogeneity property –it is the archetypal metric of homogeneity– and $recall$ has only completness property –it is the archetypal metric of completness.

4.1 Clusterings comparison on a standard real-world graph

Let's now illustrate clustering methods comparison in the $nPnB$ framework with a standard real-world graph $G_{em} = (V_{em}, E_{em})$ frequently used for benchmarking (*59*). We compare the BEC clustering family to several hitherto incommensurable state-of-the-art algorithms: $Louvain$ (*7*), $Infomap$ (*47*) , $Starling$ (*21*), $Oslom$ (*28,29*) and $Spectral\text{-}Graph\text{-}Clustering$ (*33*) – for which we consider several resolutions.

G_{em} describes e-mail data from a large research institution composed of a set V_{em} of employees. This is a standard benchmark with $|V_{em}| = 1,005$, $|E_{em}| = 16,064$. The graph contains an undirected edge $\{i, j\}$ if employee i and employee j have exchanged at least one e-mail either way. The dataset[2] on which G_{em} is build, also contains the list of the 42 departments of the research institute that are often considered as a 'ground-truth' partition $\mathcal{C}_{Dep}$ on G_{em}.

We add to this clusterings comparison the Oracle method met_{Dep} returning the 'ground-truth' partition itself ($\mathcal{C} = \mathcal{C}_{Dep} \in \mathcal{P}(\mathcal{P}(V))$) and the omniscient overlapping clustering method met_E returning the edges of graph itself ($\mathcal{C} = E \in \mathcal{P}(\mathcal{P}(V))$).

For BEC clustering, we consider two families: the partitional clustering family BEC^{s_p}, returning partitional clustering based on the optimization of F_{s_p} with scale of description s_p and the overlapping clustering $BEC^{s_p}_{s_o}$, returning overlapping clustering by gluantly extending the clusters produced by BEC^{s_p} through the optimization of F_{s_o} with stickiness scale s_o.

Spectral Graph Clustering (SGC, (*33*)) methods require to specify the number κ of clusters. Our comparison includes the SGC partitional clustering for $\kappa = 24$ (SGC_{24}) and $\kappa = 54$ (SGC_{54}). We select these two values at scale $\sigma = 0.5$ (which is a natural entry point to compare clustering given the two properties of *homogeneity* and *completeness* of $F_{\sigma=0.5}$) because (i) SGC_{24} is the one with the best *extrinsic* score relatively to $\mathcal{C}_{Dep}$ and (ii) SGC_{54} is the one with the best *intrinsic* score ; i.e. $\forall \kappa \in \mathbb{N},\ 0 < \kappa \leq |V_{em}|$:

$$\textbf{(i)}\ F_{0.5}(R^{Dep}_{\kappa}, P^{Dep}_{\kappa}) < F_{0.5}(R^{Dep}_{24}, P^{Dep}_{24}) \tag{6}$$

with $R^{Dep}_{\kappa} = R(\widehat{SGC_{\kappa}},\ \widehat{\mathcal{C}_{Dep}})$ and $P^{Dep}_{\kappa} = P(\widehat{SGC_{\kappa}},\ \widehat{\mathcal{C}_{Dep}})$;

$$\textbf{(ii)}\ F_{0.5}(R^{em}_{\kappa}, P^{em}_{\kappa}) < F_{0.5}(R^{em}_{54}, P^{em}_{54}) \tag{7}$$

with $R^{em}_{\kappa} = R(\widehat{SGC_{\kappa}}, G_{em})$ and $P^{em}_{\kappa} = P(\widehat{SGC_{\kappa}}, G_{em})$

[2]Available at `https://snap.stanford.edu/data/email-Eu-core.html`

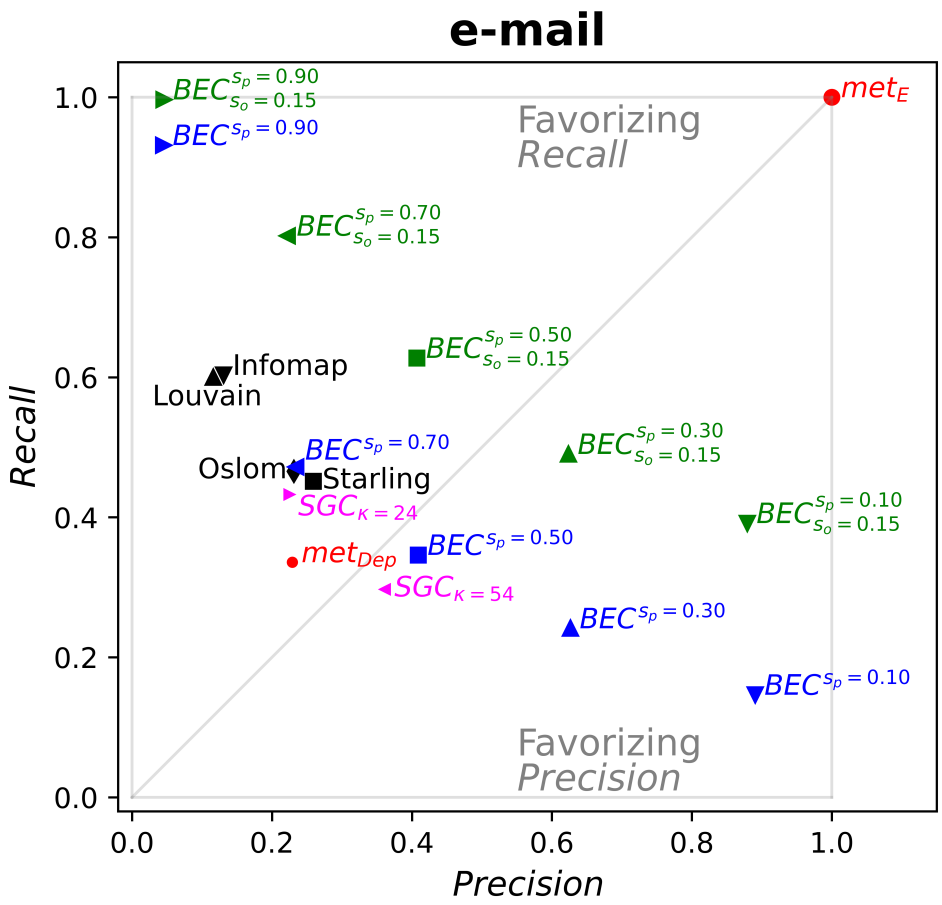


Figure 4: **Performances in the 2-dimensional space** $precision \times recall$ **of clustering methods when applied to the e-mail graph** G_{em}**.** The Oracle method met_{Dep} and the Omniscient method met_E are highligthed in red.

Fig. 4 displays methods applied to G_{em} on the $precision \times recall$ plane. It highlights the trade-off made by each clustering methods in terms of $precision$ and $recall$. Several lessons can be drawn from this visualization:

• **First:** Non parameterized methods like $Louvain$, $Infomap$, $Starling$ or $Oslom$ differ in the trade-offs they make.

• **Second:** Intrinsicaly, the family methods $\{SGC_\kappa\}_{0<\kappa\leq|V_{em}|}$ perform less well on both dimensions than the family $\{BEC^{s_p=\sigma}\}_{\sigma\in[0\ 1]}$: both $precision$ and $recall$ of $BEC^{0.50}$ are greater than these of $SGC_{\kappa=54}$ (and see Formula 7).

• **Last:** The Oracle method met_{Dep}, returning the 'ground-truth' $\mathcal{C}_{Dep}$, has poor $precision$ and $recall$ scores, which calls into question the relevance of $\mathcal{C}_{Dep}$ as a 'ground-truth' reference.

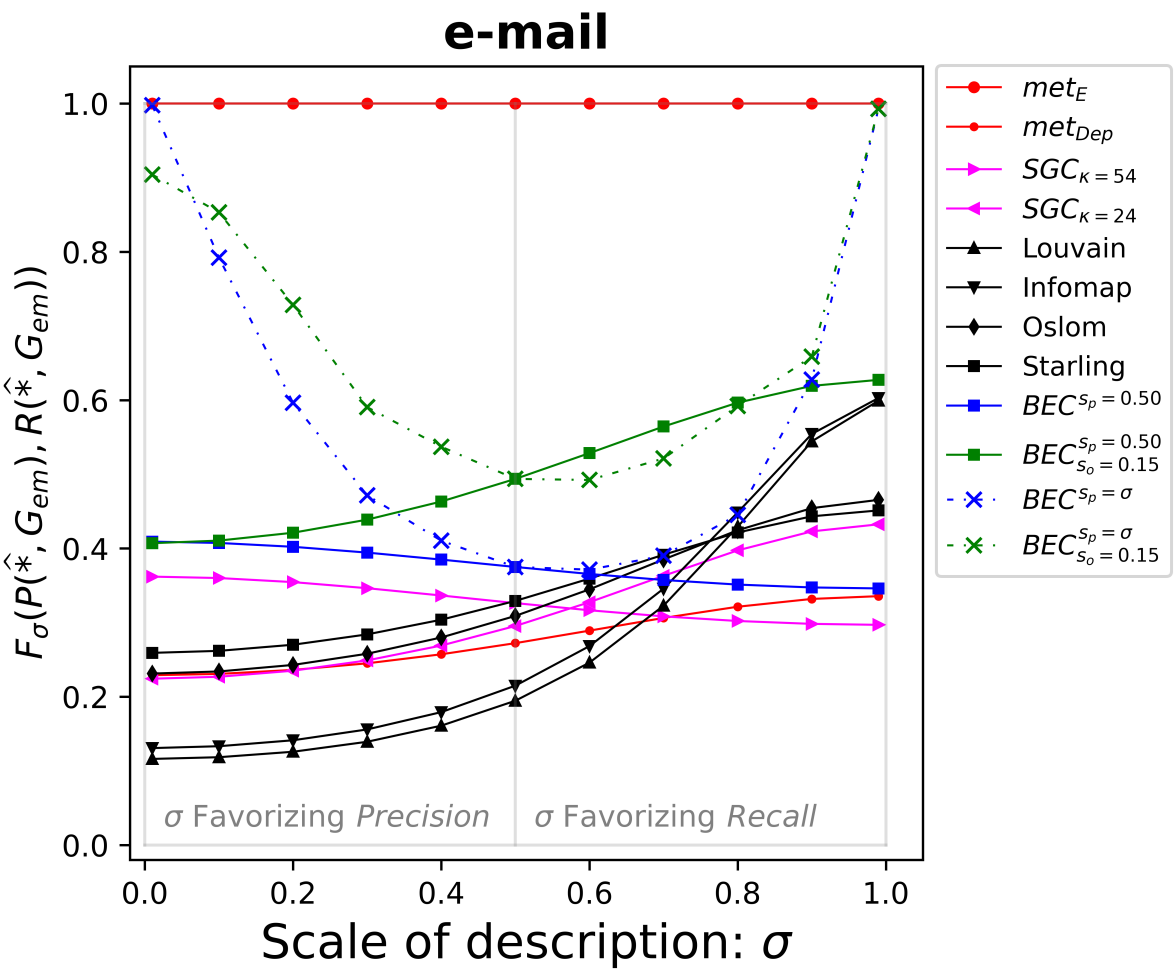


Figure 5: **Performances $F_\sigma(P(\hat{*}, G_{em}), R(\hat{*}, G_{em}))$ of clustering methods as derived graphs $\hat{*} = \widehat{met(G_{em})}$ under the description scale σ.**

Fig. 5 displays the performances of each clustering method under the scale of description $\sigma \in [0, 1]$. A key result is that the family of partitional clusterings $\{BEC^{s_p=\sigma}\}_{\sigma\in[0\ 1]}$ outperforms all other partitional methods in terms of the quality function F_σ. This could be understood by the fact that $BEC^{s_p=\sigma}$ is optimizing F_σ (s_p the optimized scale by $BEC^{s_p=\sigma}$ coincide with the description scale σ used for evaluation). Removing these family of clustering methods from the comparison, none of the partitional methods tested outperforms the others for all scales of description σ (neither does $BEC^{s_p=0.5}$).

	Intrinsicaly against $\mathbf{G_{em}}$			Extrinsicaly against $\widehat{\mathcal{C}_{Dep}}$		
Methods	**P**	**R**	$\mathbf{F_{0.5}}$	**P**	**R**	$\mathbf{F_{0.5}}$
$\mathbf{met_E}$	100	100	100	34	23	27
$\mathbf{met_{Dep}}$	23	34	27	100	100	100
$\mathbf{SGC_{54}}$	36	30	33	56	31	40
$\mathbf{SGC_{24}}$	22	43	30	46	60	52
Louvain	12	**(60)**	20	22	**(77)**	35
Infomap	13	60	22	22	70	34
Starling	26	45	33	51	61	**(56)**
Oslom	23	48	31	46	66	54
$\mathbf{BEC^{0.50}}$	**(41)**	35	**(38)**	**(59)**	34	43
$\mathbf{BEC^{0.50}_{0.15}}$	**41**	**63**	**49**	40	42	41

Table 3: **Intrinsic and extrinsic scores of clustering methods.** Each row gives the $100 \times precision$, $100 \times recall$ and $100 \times F_{\sigma=0.5}$ score for the clusterings: intrinsicaly against the original graph G_{em} and extrinsicaly against the derived graph $\widehat{\mathcal{C}_{Dep}}$ from the 'ground-truth' departments clustering $\mathcal{C}_{Dep}$. On each column, best scores are highlighted in red, and best scores as partitional clustering are highlighted in red parentheses.

Table 3 compares, at scale of description $\sigma = 0.5$, methods intrinsically against the original graph G_{em}, and extrinsically against the derived graph $\widehat{\mathcal{C}_{Dep}}$ from the 'ground-truth' $\mathcal{C}_{Dep}$.

• **Intrinsically:** The best result –both best $precision$ and $recall$– is obtained with $BEC^{0.5}_{0.15}$, and the best $F_{\sigma=0.5}$ score for partitional clustering is obtained with $BEC^{0.5}$. This could be understood by the fact that $BEC^{0.5}$ is optimizing $F_{0.5}$ (s the optimized scale by BEC^s is equal to the description scale σ used for evaluation). Surprisingly three partitional methods ($Starling$, $Oslom$ and $BEC^{0.5}$) obtain better $precision$ and at same time better $recall$ than met_{Dep}. This means that $\widehat{Starling(G_{em})}$, $\widehat{Oslom(G_{em})}$ and $\widehat{BEC^{0.5}(G_{em})}$ detect the edges of G_{em} better than $\widehat{C_{Dep}}$.

• **Extrinsically:** The omniscient method strikingly presents the worst $F_{\sigma=0.5}$ score, which again calls into question the relevance of $\mathcal{C}_{Dep}$ as a 'ground-truth' reference: research departments of a research institution are apparently not the best structures to explain patterns in e-mail exchanges among its employees. As was noted in (*21*): *We can think that two individuals from the same department can communicate in real life more often than two individuals from different departments which means that individuals from the same department do not necessarily need to communicate more by email than two individuals from different departments*.

As demonstrated in (*43*), defining a proper 'ground-truth' reference is a difficult task. Even experts often disagree with each other even when their judgements are based on the same pro-

tocol (*36*). Defining the desired scale of description and measuring quality with the F_σ function can help to define 'ground-truth' in a more consensual way in the future.

Prior to this work, it was impossible to compute Table 3 due to the lack of a framework for comparing partitional and overlapping clusterings to both the original graph and a 'ground-truth' clustering. This is a key result of the existence of a such graph clustering framework.

4.2 Clusterings comparison on artificial graphs

In this section we intrinsically evaluate the clusterings returned by different methods on artificial graphs accompanied by their expected modules.

Because we need to know κ the number of groups of nodes in advance in the Inputs of the Spectral Graph Clustering method $SGC(G, \kappa)$, to be able to compare Spectral Graph Clustering with BEC we define the family of clustering method for $s \in [0, 1]$:

$$SGC^s(G) = SGC(G, \kappa = |BEC^s(G)|)$$

To simplify the notations for readability, let's note hereafter $\mathcal{P}(\mathcal{C}, G) = P(\widehat{\mathcal{C}}, G)$; $\mathcal{R}(\mathcal{C}, G) = R(\widehat{\mathcal{C}}, G)$ and $\mathcal{F}_s(\mathcal{C}, G) = F_s(P(\widehat{\mathcal{C}}, G), R(\widehat{\mathcal{C}}, G))$.

4.2.1 Clusterings comparison on $\mathbf{Benchmark_{ER}}$

$Benchmark_{ER}$ is the class of Random graphs studied by Erdös and Rényi (*18, 19*) with parameters N the number of nodes and p the connection probability between two nodes. Random graphs do not have meaningful group structures, and they can be used to test if the algorithms are able to recognize the absence of group structures. An algorithm is able to recognize the absence of group structures in a random graph G_p^N if its returned clustering $\mathcal{C}$ is such that $|\mathcal{C}| = 1$ or $|\mathcal{C}| = N$. Therefore, we set $N = 128$ and we will study the accuracy of the methods with $Benchmark_{ER}$ according to p.

Let $G_{ER} = (V_{ER}, E_{ER})$ a random graph built by $Benchmark_{ER}$, $\Gamma_{ER} = \{V\}$ with only one cluster, and $Oracle_{ER}(G_{ER}) = \Gamma_{ER} = \{V\}$ the Oracle's method who knows $\Gamma_{ER} = \{V\}$ but ignores E_{ER} the concretly constructed edges.

We show in Figure 1 (*cf.* SI B.1) the accuracy of the methods according to p. We can see that:

- $\mathbf{Oracle_{ER}}$: It knows $\Gamma_{ER} = \{V\}$, but does not know the concretely constructed edges E_{ER}. Its number of clusters is always $= 1$. Its $precision$ increases when p increases, because $density$ increases. Its $recall$ is always $= 1$. Its $\mathcal{F}_{\sigma=0.5}$ increase;

- **Louvain** : $|Louvain(G_p)| = 1$ only when $p = 1$, it therefore does not recognize the absence of group structures in Random Graphs;
- **Infomap** : As soon as $p >= 0.14$, $|Infomap(G_p)| = 1$, it therefore recognizes the absence of group structures in Random Graphs. That is because $Infomap$ is able to identify the absence of strong structures (see (*21*));
- **Oslom** : As soon as $p >= 0.12$, $|Oslom(G_p)| = N$, it therefore recognizes the absence of group structures in Random Graphs;
- **Starling** : $|Starling(G_p)| = 1$ only when $p = 1$, it therefore does not recognize the absence of group structures in Random Graphs. That is because $Starling$ identifies the presence of weak structures (see (*21*));
- $\mathbf{BEC^{s_p}}$ and $\mathbf{BEC^{s_p}_{S_o=0.15}}$: When p is large enough, they returns one alone cluster $\{V\}$, thus recognizing the absence of group structures in Random Graphs. The larger s_p, the less p needs to be large to they recognize the absence of group structure;
- $\mathbf{SGC^s} : \forall p \in]0,1], \forall s \in [0,1], \mathcal{F}_{\sigma=0.5}(SGC(G_p, \kappa = |BEC^s(G_p)|), G_p) \leqslant \mathcal{F}_{\sigma=0.5}(BEC^s(G_p), G_p)$. That is, at middle point of view with $\sigma = 0.5$, for an equivalent number of modules, $\widehat{BEC(G_p)}$ predicts the edges of G_p better than $\widehat{SGC(G_p)}$.

4.2.2 Clusterings comparison on $\mathbf{Benchmark_{LFR}}$

In order to allow to systematically study the behavior of clustering methods relative to the complex nature of the community structure in real networks, Lancichinetti, Fortunato and Radicchi proposed $Benchmark_{LFR}$ (*27*). The graphs $G_\mu^{N,k,a,b,on,om}$ in $Benchmark_{LFR}$ are parameterized [3] with:

- N their number of nodes;
- k their average degree;
- a the power law exponent of their degree distribution;
- b the power law exponent of their community sizes distribution;
- $\mu \in [0,1]$ their mixing parameter: Each node shares a fraction $1-\mu$ of its links with the other nodes of its community and a fraction μ with the other nodes of the graph.
- on the number of overlapping nodes;

[3] Code to generate $Benchmark_{LFR}$ graphs can be downloaded from Andrea Lancichinetti's homepage `https://sites.google.com/site/andrealancichinetti/home`.

- om the number of memberships of the overlapping nodes.

With $Benchmark_{LFR}$, when the mixing parameter μ is weak, the overconnected regions are well separated from each other, and when μ increases, the overconnected regions are less clear.

Performance on $\mathbf{Benchmark^{NO}_{LFR}}$ without overlaps: We set $on = 0$, $om = 0$, $N = 1000$, and $k = 15$ or $k = 25$, $(a = 2, b = 1)$ or $(a = 2, b = 2)$ or $(a = 3, b = 1)$ and for each of these six configurations, we will study the accuracy of the methods according to μ.

Let $G^{NO}_{LFR} = (V^{NO}_{LFR}, E^{NO}_{LFR})$ a graph built by $Benchmark^{NO}_{LFR}$, Γ^{NO}_{LFR} its expected modules as expected overconnected regions, and $Oracle^{NO}_{LFR}(G^{NO}_{LFR}) = \Gamma^{NO}_{LFR}$ the Oracle's method which knows Γ^{NO}_{LFR} but ignores E^{NO}_{LFR} the concretly constructed edges. We show in Figure 2 and 3 (*cf.* SI B.2) the accuracy of the methods according to μ. We can see that:

- $\mathbf{Oracle^{NO}_{LFR}}$: It knows Γ^{NO}_{LFR}, but does not know the concretely constructed edges E^{NO}_{LFR}. Its number of clusters is always $|\Gamma^{NO}_{LFR}|$. Its $precision$ decreases when μ increase, because there are more and more non-edges in the expected modules, but $Oracle^{NO}_{LFR}$ does not know it. Its $recall$ decreases when μ increase, because there are more and more edges outside the expected modules, but $Oracle^{NO}_{LFR}$ does not know it. Its $\mathcal{F}_{\sigma=0.5}$ decreases when μ increase;
- $\mathbf{Louvain}$: Always $\mathcal{F}_{\sigma=0.5}(Louvain(G_\mu), G_\mu) \leqslant \mathcal{F}_{\sigma=0.5}(Oracle^{NO}_{LFR}, G_\mu)$ and often its $\mathcal{F}_{\sigma=0.5}$ is the lowest of our four non-parameterized methods;
- $\mathbf{Infomap}$: [When overconnected regions are **clear**, $\mathcal{F}_{\sigma=0.5}$ are good (i.e. equivalent to that of $Oracle^{NO}_{LFR}$)]; [When overconnected regions are **less clear**, $\mathcal{F}_{\sigma=0.5}$ collapses with a very clear promotion of $recall$ and a too much degradation of $precision$, because returning one single module. When $Infomap$ returns a single module, it means that there is no way with the $Map\ Equation$ to compress the path description of a random walker on the graph (see (*47*))];
- $\mathbf{Oslom}$: [When overconnected regions are **clear**, $\mathcal{F}_{\sigma=0.5}$ are good]; [When overconnected regions are **very unclear**, $\mathcal{F}_{\sigma=0.5}$ collapses with a promotion of $recall$ and a too much degradation of $precision$];
- $\mathbf{Starling}$: $\mathcal{F}_{\sigma=0.5}$ are good, and with $k = 15$ even better than that of $Oracle^{NO}_{LFR}$ when overconnected regions are **less clear**, with a degradation of $recall$ but compensated by a very good $precision$ (i.e. better to that of $Oracle^{NO}_{LFR}$);
- $\mathbf{BEC^{S_P}}$: $\mathcal{F}_{\sigma=0.5}$ are always good, and with $k = 15$ even better than that of $Oracle^{NO}_{LFR}$ when overconnected regions are **less clear**;

- $\mathbf{BEC^{S_P}_{S_o=0.15}}$: For $S_p = 0.30$ and $S_p = 0.50$, with overlaps their $\mathcal{F}_{\sigma=0.5}$ are always equal or greater than those of $Oracle^{NO}_{LFR}$ without overlaps;

- $\mathbf{SGC^s}$: Always $\mathcal{F}_{\sigma=0.5}(SGC(G_\mu, \kappa = |BEC^s(G_\mu)|), G_\mu) \leqslant \mathcal{F}_{\sigma=0.5}(BEC^s(G_\mu), G_\mu)$. That is, at middle point of view with $\sigma = 0.5$, for an equivalent number of modules, $\widehat{BEC(G_\mu)}$ predicts the edges of G_μ better than $\widehat{SGC(G_\mu)}$.

Performance on $\mathbf{Benchmark^{OV}_{LFR}}$ with overlaps: We set $on = 100$ and $om = 4$, $N = 200$, and $k = 15$ or $k = 25$, $(a = 2, b = 1)$ or $(a = 2, b = 2)$ or $(a = 3, b = 1)$ and for each of these six configurations, we will study the accuracy of the methods according to μ.

Let $G^{OV}_{LFR} = (V^{OV}_{LFR}, E^{OV}_{LFR})$ a graph built by $Benchmark^{OV}_{LFR}$, Γ^{OV}_{LFR} its expected modules as expected overconnected regions, and $Oracle^{OV}_{LFR}(G^{OV}_{LFR}) = \Gamma^{OV}_{LFR}$ the Oracle's method which knows Γ^{OV}_{LFR} but ignores E^{OV}_{LFR} the concretly constructed edges. We show in Figure 4 and 5 (*cf.* SI B.2) the accuracy of the methods according to μ. We can see that:

- $\mathbf{Oracle^{OV}_{LFR}}$: It knows Γ^{OV}_{LFR}, but does not know the concretely constructed edges E^{OV}_{LFR}. Its number of clusters is always $|\Gamma^{OV}_{LFR}|$. Its $precision$ decreases when μ increase, because there are more and more non-edges in the expected modules, but $Oracle^{OV}_{LFR}$ does not know it. Its $recall$ decreases when μ increase, because there are more and more edges outside the expected modules, but $Oracle^{OV}_{LFR}$ does not know it. Its $\mathcal{F}_{\sigma=0.5}$ decreases when μ increase;

- $\mathbf{Louvain}$: [When overconnected regions are **clear**, $\mathcal{F}_{\sigma=0.5}$ are low]; [When overconnected regions are **less clear**, $\mathcal{F}_{\sigma=0.5}$ are better than that of $Oracle^{OV}_{LFR}$];

- $\mathbf{Infomap}$: [When overconnected regions are **clear**, $\mathcal{F}_{\sigma=0.5}$ are very low]; [When overconnected regions are **very unclear**, $\mathcal{F}_{\sigma=0.5}$ are better than that of $Oracle^{OV}_{LFR}$];

- $\mathbf{Oslom}$: $\mathcal{F}_{\sigma=0.5}$ are always low, especially when overconnected regions are **less clear**;

- $\mathbf{Starling}$: [When overconnected regions are **clear**, $\mathcal{F}_{\sigma=0.5}$ are low]; [When overconnected regions are **less clear**, $\mathcal{F}_{\sigma=0.5}$ are better than that of $Oracle^{OV}_{LFR}$];

- $\mathbf{BEC^{S_P}}$: [When overconnected regions are **clear**, $\mathcal{F}_{\sigma=0.5}$ are low]; [When overconnected regions are **less clear**, $\mathcal{F}_{\sigma=0.5}$ are better than that of $Oracle^{OV}_{LFR}$];

- $\mathbf{BEC^{S_P}_{S_o=0.15}}$: [When overconnected regions are **clear**, $\mathcal{F}_{\sigma=0.5}$ are low]; [When overconnected regions are **less clear**, $\mathcal{F}_{\sigma=0.5}$ are better than that of $Oracle^{OV}_{LFR}$];

- $\mathbf{SGC^s}$: For $S_p = 0.30$ and $S_p = 0.50$, $\mathcal{F}_{\sigma=0.5}(SGC(G_\mu, \kappa = |BEC^s(G_\mu)|), G_\mu) \leqslant \mathcal{F}_{\sigma=0.5}(BEC^s(G_\mu), G_\mu)$.

5 Discussion

Graphs are one of the main conceptual structures for modeling complex systems, where nodes represent the basic lowest-level entities and edges represent their interactions. Clustering algorithms are then used to identify densely connected sets of nodes of these graphs as representations of entities at higher scales of observation in these systems, from the study of which we can deduce particular characteristics or functions of these systems. This operation defines *de facto* a *scale of description* and the different scales of description offer distinct insights on the system under study. For example, the study of living systems built from different types of basic entities such as cells can focus on the higher-order structures such as biological tissues, organs, individuals, etc.

This poses two legitimate questions: **(1)** *How to decompose a complex system?* **(2)** *How to define the appropriate scales for these decompositions?* The proposed $nPnB$ framework and the associated BEC clustering families provide an original answer to the first question: identifying the sub-parts of a complex system is basically linked to the question of the chosen scale of observation ; the choice of a clustering algorithm only makes sense once a scale of observation has been chosen. This means that the second question is in fact primary and requires to be able to define observation scales intrinsic to a given complex system that could serve as natural entry points for the clustering.

This is a central question to the study of complex systems (*30, 32, 40*) and a detailed answer is beyond the scope of this paper. Nevertheless, we can sketch out how our approach allows intrinsic scales to be defined, in line with previous work (*46*), but with simpler metrics.

5.1 Intrinsic scales

Let's focus on a simple case. Let G be a graph composed of two levels of clusters (cf. Fig. 6): the vertices are grouped into three large clusters, which are themselves decomposed into three smaller clusters. Formally, we define a graph $G = (V, E)$ such that V is the union of nine sets $\Delta_1 \ldots \Delta_9$ of 20 vertices each. These vertices are grouped into three sets $\Gamma_1 = \Delta_1 \cup \Delta_2 \cup \Delta_3$, $\Gamma_2 = \Delta_4 \cup \Delta_5 \cup \Delta_6$, $\Gamma_3 = \Delta_7 \cup \Delta_8 \cup \Delta_9$. An edge e between two vertices u and v is created with the following probabilities:

- $p_1 = 0.5$ if the two vertices belong to a same set Δ ($\exists i$ such $u, v \in \Delta_i$);
- $p_2 = 0.05$ if they belong to distinct sets Δ but to a same set Γ ($\nexists i$ such $u, v \in \Delta_i$ but $\exists j$ such $u, v \in \Gamma_j$);
- $p_3 = 0.005$ if they belong to distinct sets Γ ($\nexists i$ such $u, v \in \Gamma_i$)

By construction, G has two "natural" sub-structures: one composed of nine small modules $\mathcal{C}_A = \{\Delta_1, \ldots, \Delta_9\}$, the other composed of three bigger modules $\mathcal{C}_B = \{\Gamma_1, \Gamma_2, \Gamma_3\}$.

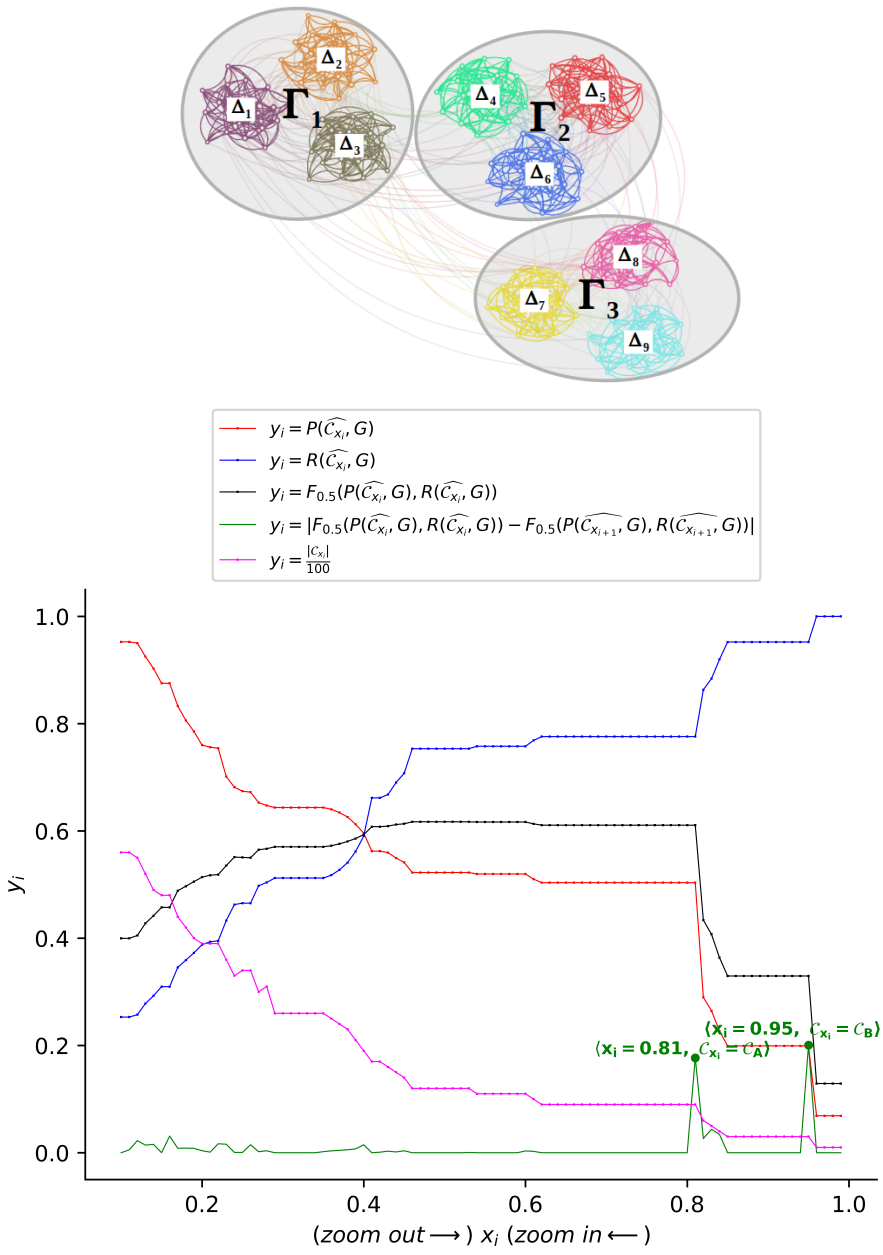


Figure 6: **Informations of interest to define intrinsic scales as well as relevant 'ground-truths' for multi-scale graphs.**

Let's now consider the clustering family $\mathcal{C}_{x_i} = BEC^{s_p=x_i}(G)$ that describes G at different *scales of description* x_i. Fig. 6 shows how the $precision$: $\mathcal{P}(\mathcal{C}_{x_i}, G)$, $recall$: $\mathcal{R}(\mathcal{C}_{x_i}, G)$ and $\mathcal{F}_{0.5}score$: $\mathcal{F}_{0.5}(\mathcal{C}_{x_i}, G))$ vary with x_i. We can observe that the variations of the $\mathcal{F}_{0.5}score$

(i.e. the curve $y_i = |\mathcal{F}_{0.5}(\mathcal{C}_{x_i}, G)) - \mathcal{F}_{0.5}(\mathcal{C}_{i+1}, G))|$) displays two local maxima at the values: $x_a = 0.81$ and $x_b = 0.95$. That is to say that when an observer zooms out ($x_i : 0 \rightsquigarrow 1$, as in Fig 6), then x_{a+1} and x_{b+1} are the *scales of description* for which the $\mathcal{F}_{0.5}score$ suddenly drops (in relation to the zoom out action). Moreover $\mathcal{C}_{x_a} = \mathcal{C}_A$ and $\mathcal{C}_{x_b} = \mathcal{C}_B$.

Furthermore, looking for sub-modules, if the observer zooms in on the system ($x_i : 1 \rightsquigarrow 0$), then for these two same values, stable sub-modules $\mathcal{C}_{x_a} = \mathcal{C}_A$ and $\mathcal{C}_{x_b} = \mathcal{C}_B$ emerge (in relation to the zoom in action: for $x_b = 0.95$: $\ldots \langle x_{b-1}, \mathcal{C}_{b-1} \neq \mathcal{C}_B\rangle \overset{zoom\ in}{\longrightarrow} \langle x_b, \mathcal{C}_b = \mathcal{C}_B\rangle \overset{zoom\ in}{\longrightarrow} \langle x_{b+1}, \mathcal{C}_{b+1} = \mathcal{C}_B\rangle \overset{zoom\ in}{\longrightarrow} \langle x_{b+2}, \mathcal{C}_{b+2} = \mathcal{C}_B\rangle \overset{zoom\ in}{\longrightarrow} \ldots$ and then for $x_a = 0.81$: $\ldots \langle x_{a-1}, \mathcal{C}_{a-1} \neq \mathcal{C}_A\rangle \overset{zoom\ in}{\longrightarrow} \langle x_a, \mathcal{C}_a = \mathcal{C}_A\rangle \overset{zoom\ in}{\longrightarrow} \langle x_{a+1}, \mathcal{C}_{a+1} = \mathcal{C}_A\rangle \overset{zoom\ in}{\longrightarrow} \langle x_{a+2}, \mathcal{C}_{a+2} = \mathcal{C}_A\rangle \overset{zoom\ in}{\longrightarrow} \ldots$

The two values x_a and x_b are thus salient because they correspond to abrupt structural changes in the description of the system when one dives into the details of its organization. In this sense, they are thus good candidates to define *intrinsic scales* as well as relevant 'ground-truth' clusterings: $\langle x_a = 0.81, \mathcal{C}_a = \mathcal{C}_A\rangle$ and $\langle x_b = 0.95, \mathcal{C}_b = \mathcal{C}_B\rangle$.

5.2 Finding Graph Clusterings

In the $nPnB$ framework, any metric ✠ based on the four metrics TP, TN, FP, FN, is such that ✠ $\in M_{P,In} \cap M_{P,Ex} \cap M_{O,In} \cap M_{O,Ex}$ (for exemple $precision$, $recall$, $\mathcal{F}_s$, $Jaccard$, $RandIndex$ and MCC). In particular, ✠ $\in M_{P,In} \cap M_{O,In}$ and any ✠ may be used as intrinsic metric to be optimized to find graph clusterings with or without overlaps. However $\mathcal{F}_s$ has the advantage of allowing a trade-off between $precision$ and $recall$ which faithfully formalizes the two antagonistic properties P_{DC} and P_{WC}.

5.3 Comparing Graph Clusterings in $nPnB$ framework

Let $\mathcal{C}$ and $\mathcal{C}^{ref}$ be two clusterings of a graph $G = (V, E)$. The $nPnB$ framework makes it possible to measure the capacity of the graph $\widehat{\mathcal{C}}$ to detect the edges of the graph $\widehat{\mathcal{C}^{ref}}$. If $\mathcal{C}^{ref} = E$ then $\widehat{\mathcal{C}^{ref}} = G$, it is an intrinsic measurement else it is extrinsic. We can measure this capacity:

■ **In the 2-dimensional space precision $\times$ recall:** with $\mathcal{P}(\mathcal{C}, \widehat{\mathcal{C}^{ref}})$ and $\mathcal{R}(\mathcal{C}, \widehat{\mathcal{C}^{ref}})$, where $\mathcal{P}(\cdot,\cdot)$ the metric $precision$ faithfully formalizes the P_{DC} property and $\mathcal{R}(\cdot,\cdot)$ the metric $recall$ faithfully formalizes the P_{WC} property.

Then we can objectively conclude that $\mathcal{C}_1$ performs better or equal than $\mathcal{C}_2$ in regard of the gold standard $\mathcal{C}^{ref}$ **iff**

$$\Big(\mathcal{P}(\mathcal{C}_1, \widehat{\mathcal{C}^{ref}}), \mathcal{R}(\mathcal{C}_1, \widehat{\mathcal{C}^{ref}})\Big) \in \Big[\mathcal{P}(\mathcal{C}_2, \widehat{\mathcal{C}^{ref}}), 1\Big] \times \Big[\mathcal{R}(\mathcal{C}_2, \widehat{\mathcal{C}^{ref}}), 1\Big]$$

■ **In 1-dimensional space:** Since $precision$ and $recall$ are antagonistic, optimizing both of them is a multi-objective optimization that most of the time does not have a unique optimal solution. If we cannot conclude one way or the other in the 2-dimensional space $precision \times recall$, the choice of a clustering can be reduced to 1-dimensional space, provided that we combine:

- **A subjective decision:** defining a trade-off between $precision$ and $recall$ through the choice of a *scale of description* $\sigma \in [0, 1]$;
- **An objective methodology:** using then $\mathcal{F}_\sigma$ as objective criteria for the evaluation of graph clustering methods in regard of the subjective *scales of description* σ.

Then we can objectively conclude that $\mathcal{C}_1$ performs better or equal than $\mathcal{C}_2$ in regard of the gold standard $\mathcal{C}^{ref}$ under the subjective *scale of description* σ **iff**

$$\mathcal{F}_\sigma(\mathcal{C}_1, \widehat{\mathcal{C}^{ref}}) \geqq \mathcal{F}_\sigma(\mathcal{C}_2, \widehat{\mathcal{C}^{ref}})$$

In a 1-dimensional decision space, it is necessary to choose a subjective σ description scale *before* being able to objectively compare two clusterings with $\mathcal{F}_\sigma$. Note that the subjective middle point of view $\sigma = 0.5$ has some salience since the associated objective criterion $\mathcal{F}_{0.5}$ gives the same weight to the two antagonistic properties P_{DC} and P_{WC}. Then $\mathcal{F}_{0.5}$ is a symetrical criterion and has both $homogeneity$ and $completeness$ properties. The entities represented by the clusters for this scale of description could be considered as the *'median entities of the system'*.

6 Conclusion

We have introduced $nPnB$ a framework to evaluate graph clusterings as cohesive sets of nodes that present several advantages.

Unification: It allows to evaluate intrinsically or extrinsically any clustering with or without overlap (for exemple, it was so far not possible to construct a table similar to table 3). This is the key result of this paper.

Simplicity: evaluation in this framework is based on the two metrics, $precision$ and $recall$ widely used in science and therefore easily understandable by most users of real-world graphs. It naturally includes the notion of *description scale* found in other forms in many algorithms (for example such κ the desired number of observed modules for Spectral Graph Clustering).

Ground-truths: the proposed framework makes it possible to intrinsically assess the relevance of 'ground-truth' clusterings.

Effectiveness: it is effective in the sense that on the one hand it provides inspiration for new clustering algorithms (the BEC method is the first clustering algorithm based on $\mathcal{F}_s$ optimiza-

tion); and on the other hand it allows us to produce information of interest on the intrinsic scales of description of a complex system.

Limitation: nPnB only takes into account the edges of the graphs

We have shown that the two main properties expected from a graph clustering, P_{DC} and P_{WC} are faithfully formalized by the two metrics $precision$ and $recall$ which are antagonistic in $nPnB^P$ framework. However, these two properties only take into account the structures of the graphs, i.e. only their edges without taking into account all other possible characteristics of the nodes. This is the main limitation of the nPnB framework, indeed evaluating the quality of a graph clustering by taking into account other characteristics such as node attributes is entirely dependent on the type of these attributes, what they represent and especially what is expected of the modules with respect to these attributes.

Perspectives

Many complex networks have directed weighted edges and have a temporal dimension. Those attributes are not been addressed in the proposed framework and should be part of further developements.

Weighted and directed networks: Our approach could easily be adapted to the case of weighted networks $G = (V, E, w)$ where the weight of each edge $\{u, v\} \in E$ is given by the function $w : E \longrightarrow \mathbf{R}^{*+}$. Only three lines of code need to be modified (*cf.* SI C.1) [4].

For directed networks, when no specific meaning is given to directionnality of edges in clusters, our approach can be extended by simply counting all directional edges in the $precision$ and $recall$ metrics. However, when dealing with directed modules (*42*), new strategies should be developped in the agglomeration phase since agglomeration has to satisfy both the non-decrease of $\mathcal{F}_s$ and the constraint of defining directed modules.

Temporal graphs: As for the temporal dimension, the BEC clustering algorithm could be used as the base algorithm for phylomemy reconstruction (*14*) in order to generate temporal clusterings which clusters satisfy both a $precision/recall$ constraint on links relatively other clusters in the same temporal slice and a $precision/recall$ constraint on nodes distribution among temporal clusters (phylomemetic branches).

[4] The C^{++} implementation used for this paper already allow weighted networks. By default, we have set $w : E \longrightarrow \mathbf{R}^{*+}, w(\{u, v\}) = 1$.

Attributed graph clustering: When nodes of the graph also have some attributes, it is possible to define clusterings that take these attributes into account (see for exemple (*4–6, 12, 15, 23, 51, 56*)). Attributed graph clustering can be perfectly hybridised with our approach, but it will still be necessary to assess beforehand the relative weight given to the information contained in the links and that contained in the attributes of the nodes (see SI. C.2.)

Intrinsic scales: When graphs are less trivial than that of figure 6, the analysis of the function $f(x,y) = 1 - R(\widehat{BEC^{x=s_p}(G)}, \widehat{BEC^{s_p=x}_{s_o=y}(G)})$ can also produce informations of interest (for exemple, $\forall x, y \in [0,1], f(x,y) = 0$ **iff** G is a set of unconnected cliques, and local maxima produce information of interest on the intrinsic scales of observation of multi-scale graphs modeling real complex systems). Such function is well defined only if we can compare non overlapping clustering with overlapping clustering, which is possible with the $nPnB$ framework. This paves the way to evaluating the improvement obtained by allowing clusters to overlap, compared with strict compliance with the partition constraint.

Competing interests

The authors declare no competing interests.

Data availability

All data analysed during this study are included in this published article [and its supplementary information files]. The datasets used during the current study available from the corresponding author on reasonable request.

Contributions

Bruno Gaume initiated the research, wrote the first draft and carried out the digital implementation. Bruno Gaume, Ixandra Achitouv and David Chavalarias further developed the writing of the article and the analyses. Bruno Gaume and David Chavalarias wrote the final version of the manuscript.

Aknowledgments

This work was supported by the Complex Systems Institute of Paris Île-de-France (ISC-PIF) and the EU NODES project (LC-01967516).

References

1. GarganText, collaborative and decentralized LibreWare, (2023).

2. E. AMIGÓ, J. GONZALO, J. ARTILES, AND F. VERDEJO, A comparison of extrinsic clustering evaluation metrics based on formal constraints, Information Retrieval Journal, 12 (2009), pp. 461–486.

3. K. ASMI, D. LOTFI, AND M. EL MARRAKI, Overlapping community detection based on the union of all maximum spanning trees, Library Hi Tech, ahead-of-print (2020).

4. K. BERAHMAND, S. BAHADORI, M. N. ABADEH, Y. LI, AND Y. XU, Sdac-da: Semi-supervised deep attributed clustering using dual autoencoder, IEEE Transactions on Knowledge and Data Engineering, (2024).

5. K. BERAHMAND, Y. LI, AND Y. XU, Dac-hpp: deep attributed clustering with high-order proximity preserve, Neural Computing and Applications, 35 (2023), pp. 24493–24511.

6. K. BERAHMAND, M. MOHAMMADI, R. SHEIKHPOUR, Y. LI, AND Y. XU, Wsnmf: Weighted symmetric nonnegative matrix factorization for attributed graph clustering, Neurocomputing, 566 (2024), p. 127041.

7. V. D. BLONDEL, J.-L. GUILLAUME, R. LAMBIOTTE, AND E. LEFEBVRE, Fast unfolding of communities in large networks, Journal of Statistical Mechanics: Theory and Experiment, 2008 (2008), p. P10008.

8. S. BOCCALETTI, V. LATORA, Y. MORENO, M. CHAVEZ, AND D. U. HWANG, Complex networks: Structure and dynamics, Physics Reports, 424 (2006), pp. 175–308.

9. M. BUCKLAND AND F. GEY, The relationship between Recall and Precision, Journal of the American Society for Information Science, 45 (1994), pp. 12–19. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/%28SICI%291097-4571%28199401%2945%3A1%3C12%3A%3AAID-ASI2%3E3.0.CO%3B2-L.

10. M. BW., Comparison of the predicted and observed secondary structure of t4 phage lysozyme, Biochim Biophys Acta, (1975).

11. H. CAI, V. W. ZHENG, AND K. C.-C. CHANG, A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications, IEEE Transactions on Knowledge and Data Engineering, 30 (2018), pp. 1616–1637. Conference Name: IEEE Transactions on Knowledge and Data Engineering.

12. J. CAO, J. FANG, Z. MENG, AND S. LIANG, Knowledge Graph Embedding: A Survey from the Perspective of Representation Spaces, ACM Comput. Surv., 56 (2024), pp. 159:1–159:42.

13. T. CHAKRABORTY, A. DALMIA, A. MUKHERJEE, AND N. GANGULY, Metrics for community analysis: A survey, ACM Comput. Surv., 50 (2017).

14. D. CHAVALARIAS, Q. LOBBÉ, AND A. DELANOË, Draw me Science: Multi-level and multi-scale reconstruction of knowledge dynamics with phylomemies, Scientometrics, (2021).

15. P. CHUNAEV, Community detection in node-attributed social networks: a survey, Computer Science Review, 37 (2020), p. 100286.

16. H. CRAMÉR, Mathematical Methods of Statistics, Princeton University Press, Princeton, 1946.

17. J. DJOLONGA, M. LUCIC, M. CUTURI, O. BACHEM, O. BOUSQUET, AND S. GELLY, Precision-Recall Curves Using Information Divergence Frontiers, in Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, PMLR, June 2020, pp. 2550–2559. ISSN: 2640-3498.

18. P. ERDÖS AND A. RÉNYI, On random graphs i, Publicationes Mathematicae Debrecen, 6 (1959), pp. 290–297.

19. ——, On the evolution of random graphs, Publ. Math. Inst. Hungary. Acad. Sci., 5 (1960), pp. 17–61.

20. S. FORTUNATO AND M. E. J. NEWMAN, 20 years of network community detection, Nature Physics, 18 (2022), pp. 848–850. Number: 8 Publisher: Nature Publishing Group.

21. B. GAUME, Starling: Introducing a mesoscopic scale with confluence for graph clustering, PLOS ONE, 18 (2023), pp. 1–30.

22. B. GAUME, F. MATHIEU, AND E. NAVARRO, Building real-world complex networks by wandering on random graphs, Revue I3, 10 (2010), pp. 73–91.

23. C. HE, X. FEI, Q. CHENG, H. LI, Z. HU, AND Y. TANG, A survey of community detection in complex networks using nonnegative matrix factorization, IEEE Transactions on Computational Social Systems, 9 (2021), pp. 440–457.

24. P. JACCARD, Distribution de la flore alpine dans le bassin des dranses et dans quelques régions voisines., Bulletin de la Societe Vaudoise des Sciences Naturelles, 37 (1901), pp. 241–72.

25. J. KLEINBERG, An impossibility theorem for clustering, Advances in neural information processing systems, 15 (2002).

26. D. E. KNUTH, The Art of Computer Programming: Fundamental algorithms, The Art of Computer Programming, Addison -Wesley, 1968.

27. A. LANCICHINETTI, S. FORTUNATO, AND F. RADICCHI, Benchmark graphs for testing community detection algorithms, Physical Review E, 78 (2008), pp. 046110+.

28. A. LANCICHINETTI, F. RADICCHI, AND J. J. RAMASCO, Statistical significance of communities in networks, Physical Review E, 81 (2010).

29. A. LANCICHINETTI, F. RADICCHI, J. J. RAMASCO, AND S. FORTUNATO, Finding statistically significant communities in networks, PLOS ONE, 6 (2011), pp. 1–18.

30. G. LI, W. GE, J. ZHANG, AND M. KWAUK, Multi-Scale Compromise and Multi-Level Correlation in Complex Systems, Chemical Engineering Research and Design, 83 (2005), pp. 574–582. Publisher: Elsevier.

31. H. LI, R. ZHANG, Z. ZHAO, AND X. LIU, Lpa-mni: An improved label propagation algorithm based on modularity and node importance for community detection, Entropy, 23 (2021), p. 497.

32. J. LI, J. ZHANG, W. GE, AND X. LIU, Multi-scale methodology for complex systems, Chemical Engineering Science, 59 (2004), pp. 1687–1700.

33. U. LUXBURG, A tutorial on spectral clustering, Statistics and Computing, 17 (2007), p. 395–416.

34. B. W. MATTHEWS, Comparison of the predicted and observed secondary structure of t4 phage lysozyme., Biochimica et biophysica acta, 405 2 (1975), pp. 442–51.

35. P. J. MUCHA, T. RICHARDSON, K. MACON, M. A. PORTER, AND J.-P. ONNELA, Community structure in time-dependent, multiscale, and multiplex networks, science, 328 (2010), pp. 876–878.

36. G. C. MURRAY AND R. GREEN, Lexical Knowledge and Human Disagreement on a WSD Task, Computer Speech & Language, 18 (2004), pp. 209–222.

37. E. NAVARRO, Métrologie des graphes de terrain, application à la construction de ressources lexicales et à la recherche d'information. November 2013.

38. M. E. J. NEWMAN, The Structure and Function of Complex Networks, SIAM Review, 45 (2003), pp. 167–256.

39. M. E. J. NEWMAN AND M. GIRVAN, Finding and evaluating community structure in networks, Physical Review E, 69 (2004).

40. Z. NUSSINOV, P. RONHOVDE, D. HU, S. CHAKRABARTY, M. SAHU, B. SUN, N. A. MAURO, AND K. K. SAHU, Inference of hidden structures in complex physical systems by multi-scale clustering, ArXiv, abs/1503.01626 (2015).

41. J. OPITZ, A closer look at classification evaluation metrics and a critical reflection of common evaluation practice, Transactions of the Association for Computational Linguistics, 12 (2024), pp. 820–836.

42. G. PALLA, I. J. FARKAS, P. POLLNER, I. DERÉNYI, AND T. VICSEK, Directed network modules, New Journal of Physics, 9 (2007), p. 186.

43. L. PEEL, D. B. LARREMORE, AND A. CLAUSET, The ground truth about metadata and community detection in networks, Science Advances, 3 (2017), p. e1602548.

44. D. M. W. POWERS, Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation, (2020).

45. W. RAND, Objective criteria for the evaluation of clustering methods, Journal of the American Statistical association, 66 (1971), pp. 846–850.

46. P. RONHOVDE AND Z. NUSSINOV, Multiresolution community detection for megascale networks by information-based replica correlations, Physical Review E, 80 (2009).

47. M. ROSVALL AND C. T. BERGSTROM, Maps of random walks on complex networks reveal community structure, Proceedings of the National Academy of Sciences, 105 (2008), pp. 1118–1123.

48. B. SCHÖLKOPF, A. J. SMOLA, F. BACH, ET AL., Learning with kernels: support vector machines, regularization, Optimization, and Beyond. MIT press, 1 (2002).

49. H. SCHÜTZE, C. D. MANNING, AND P. RAGHAVAN, Introduction to information retrieval, 39 (2008).

50. S. H. STROGATZ, Exploring complex networks, Nature, 410 (2001), pp. 268–276. Number: 6825 Publisher: Nature Publishing Group.

51. X. SU, S. XUE, F. LIU, J. WU, J. YANG, C. ZHOU, W. HU, C. PARIS, S. NEPAL, D. JIN, ET AL., A comprehensive survey on community detection with deep learning, IEEE Transactions on Neural Networks and Learning Systems, (2022).

52. A. TANDON, A. ALBESHRI, V. THAYANANTHAN, W. ALHALABI, AND S. FORTUNATO, Fast consensus clustering in complex networks, Phys. Rev. E, 99 (2019), p. 042301.

53. A. THARWAT, Classification assessment methods: a detailed tutorial, (2018).

54. U. VON LUXBURG, A tutorial on spectral clustering, Statistics and Computing, 17 (2007), pp. 395–416.

55. U. VON LUXBURG, R. C. WILLIAMSON, AND I. GUYON, Clustering: Science or art?, (2012), pp. 65–79.

56. C. WANG, S. PAN, P. Y. CELINA, R. HU, G. LONG, AND C. ZHANG, Deep neighbor-aware embedding for node clustering in attributed graphs, Pattern Recognition, 122 (2022), p. 108230.

57. D. J. WATTS AND S. H. STROGATZ, Collective Dynamics of Small-World Networks, Nature, 393 (1998), pp. 440–442.

58. J. YANG AND J. LESKOVEC, Defining and evaluating network communities based on ground-truth, 2012 IEEE 12th International Conference on Data Mining, (2012), pp. 745–754.

59. H. YIN, A. R. BENSON, J. LESKOVEC, AND D. F. GLEICH, Local higher-order graph clustering, Proceedings of Conference on Knowledge Discovery and Data Mining, (2017), p. 555–564.

60. G. U. YULE, On the methods of measuring association between two attributes, Journal of the Royal Statistical Society, 75 (1912), pp. 579–652.