

CAHIER DU LAMSADE

Laboratoire d'Analyse et Modélisation de Systèmes pour l'Aide à la Décision
(Université de Paris-Dauphine)
Unité de Recherche Associée au CNRS n° 825

APPROXIMABILITY PRESERVING REDUCTIONS FOR NP-COMPLETE PROBLEMS

CAHIER N° 112
janvier 1993

V. Th. PASCHOS
L. RENOTTE

CONTENTS

	<u>Pages</u>
Résumé	i
Abstract	ii
1. Introduction, notations and recalls	1
2. Set covering hierarchy	2
3. Hitting set hierarchy	8
4. Vertex-covering-by-cliques hierarchy	9
5. Vertex covering hierarchy	13
6. Expansions of reductions between problems in different hierarchies	16
6.1 Set covering and minimal join problem	16
6.2 Set covering and vertex covering by cliques	17
6.3 Set covering and vertex covering	19
7. Conclusions and discussion	20
References	22

Réductions Préservant l'Approximabilité des Problèmes NP-Complets

Résumé

Nous construisons des réductions qui préservent l'approximation (continues) pour un certain nombre de problèmes combinatoires. La conservation du rapport d'approximation à travers ces réductions établit une certaine continuité entre les problèmes considérés entraînant un transfert possible des résultats positifs, négatifs ou conditionnels d'un problème à un autre par le biais de ces réductions. Nous nous sommes plus particulièrement intéressés à la continuité des réductions à l'intérieur des hiérarchies de problèmes combinatoires. Ces réductions étant composables et transitives, nous avons également étudié la continuité des réductions entre des membres de hiérarchies différentes. Pour un problème combinatoire, on obtient une hiérarchie en imposant des restrictions sur des instances du problème. Nous construisons tout d'abord une hiérarchie pour le problème de la *couverture d'ensembles* et explorons la complétude de ces membres à travers des réductions préservant leur rapport d'approximation (réductions continues). Nous construisons ensuite une hiérarchie pour le problème du *transversal d'un hypergraphe* et donnons des résultats de continuité. Ensuite nous relierons un autre problème d'optimisation en bases de données (*minimisation des jointûres*) à la hiérarchie concernant le problème de couverture d'ensembles en proposant une réduction et en prouvant sa continuité. Nous établissons une hiérarchie pour le problème de *couverture des sommets par des cliques* ainsi que pour le problème de *couverture de sommets*. Pour les membres de chaque hiérarchie nous cherchons des réductions continues les reliant. Nous proposons également des réductions reliant les membres de hiérarchies différentes. Nous formons ainsi des sous-classes de la classe NP-complet à l'intérieur desquelles le rapport d'approximation est préservé.

Mots-clés: Problème NP-complet, réduction polynomiale, approximation polynomiale

Approximability Preserving Reductions for NP-Complete Problems

Abstract

We conceive some approximability preserving (continuous) reductions among a number of combinatorial problems. The preservation of the approximation ratio along these reductions establishes a kind of continuity between the involved problems, continuity resulting to a possible transfer (up to multiplication by the so called expansion of the reduction) of positive, negative or conditional results along chains of such reductions. We are, more particularly interested in continuity of reductions in the interior of hierarchies of combinatorial problems as well as, given that approximability preserving reductions are composable and transitive, we explore the continuity of reductions between members of different hierarchies. For a combinatorial problem, a hierarchy is obtained when we impose additional restrictions on its instances. We construct first a hierarchy for *set covering* problem and we explore the completeness of its members under reductions that preserve their approximation ratio (*continuous reductions*). We construct also a hierarchy for *hitting set* and we give continuity results for it. Moreover, we relate another *NP*-complete database query optimization problem, the *minimal join problem* to set covering hierarchy by proposing a reduction and by proving its continuity. After, we establish hierarchies for the *vertex covering by cliques* problem as well as for the *vertex covering* problem. For the members of each hierarchy we investigate the continuity of reductions relating them. Also, we propose such reductions connecting problems from distinct hierarchies. So, subclasses of *NP*-complete problems related by approximability preserving reductions are systematically constructed.

keywords: NP-complete problem, polynomial reduction, polynomial time approximation

1 Introduction, Notations and Recalls

The notion of the reduction is a *key-point* for proving and understanding the behaviour of numerous combinatorial problems. This major tool seems to be, however, a somewhat weak when one tries to investigate approximate solutions of *NP*-complete problems. This means that, although the inclusion in the class *NP*-complete establishes a kind of “equivalence” between included problems, this equivalence considers in general the optimal solutions and not the sub-optimal (approximate) ones, in the sense that nearly optimal solutions for one problem reveal very poor and unsatisfactory when translated to another one. We are interested in reductions that have the property to preserve not only the optimality of the solutions between two problems Π and Π' , but also the approximation ratio ([6]) from Π to Π' (up to a multiplicative factor).

In what follows we use the term *continuous reduction* introduced by Simon in [22] to define exactly an approximability-preserving reduction. The notion of continuity resides implicitly in many reductions of [6,7,10,11] even if, up to now, this notion is not defined appropriately and uniformly. Very interesting works on this concept are (among other) the ones of Papadimitriou and Yannakakis ([18,17], Panconesi and Ranjan ([15]), Crescenzi and Panconesi ([5]), Kolaitis and Thakur ([12,13]) and of course the one of Simon ([22]). We define firstly the adopted notion of continuity as it is expressed in [22].

Given two combinatorial problems Π, Π' a reduction $\omega : \Pi \propto \Pi'$ is called continuous with an expansion of e if each ρ' -approximation algorithm $\mathcal{A}'$ for Π' can be converted into a ρ -approximation algorithm $\mathcal{A}$ for Π such that $\rho \leq e\rho'$. The corresponding notation is $\Pi \xRightarrow{e} \Pi'$. The notion of the approximation ratio used throughout of the paper is the one defined in [6,16].

Π and Π' are called *equivalent* (*asymptotic equivalent*) if there exist mutual continuous reductions with $e = 1$ ($e = 1 + \epsilon, \forall \epsilon > 0$) in both directions.

Lemma 1. ([22]) *Continuous reductions are transitive. More formally:*

$$\text{If } \Pi \xRightarrow{e} \Pi' \text{ and } \Pi' \xRightarrow{e'} \Pi'' \text{ then } \Pi \xRightarrow{e \cdot e'} \Pi'' .$$

In what follows we use the notation $\Pi \xrightarrow{e} \Pi'$ to denote a reduction with expansion e which depends on the instances of Π or Π' .

In fact, the notion of continuity is more interesting than one may thinks. Let us use an example to explain this. There are two well known *NP*-complete problems of graph theory, the *vertex covering* (*VC*) and the *independent set* problem (*IS*).

Given a graph $G = (V, E)$ a *vertex cover* is a subset $V' \subseteq V$ such that for each edge $uv \in E$ at least one of u and v belongs to V' and the *minimum vertex cover problem* (*VC*) is to find a minimum size vertex cover.

Also, given a graph $G = (V, E)$, an *independent set* is a subset $V' \subseteq V$ such that, for every pair (v_i, v_j) of vertices in V' , the edge $v_i v_j$ does not belong to E , and the *maximum independent set problem* (*IS*) is to find a maximum size independent set.

There is an easy, and immediate, relation between vertex cover and independent set in a graph G ([6,16]) as a *vertex cover* is the complement of an *independent set* with respect to the vertex set of G .

But this very intuitive relationship, that constitutes also a reduction between the two problems, is not reflected on the level of the approximation results for these problems. Really, the existence of an approximation algorithm for VC ([6,16]) with a ratio 2 does not lead to a corresponding approximation algorithm with a bounded error for IS .

In this paper, we establish first hierarchies for combinatorial optimization problems as *set covering*, *hitting set*, *vertex covering by cliques* and VC .

For a combinatorial problem a hierarchy is obtained when we impose additional restrictions on its instances.

After, for each of the established hierarchies we propose reductions and we explore their expansions. We study also expansions for reductions connecting problems of different hierarchies.

Moreover, we define a database query optimization problem, the *minimal join problem*, we investigate its complexity and we prove that it is linked to set covering by reduction continuous and of expansion 1.

Finally, we give results constituting examples of how the notion of continuity becomes significant and helpful in making easier the estimation of the approximation performance for many algorithms.

2 Set Covering Hierarchy

Given a collection $\mathcal{S}$ ($|\mathcal{S}| = n$) of subsets of a finite set C ($|C| = m$), a *cover* is a subcollection $\mathcal{S}' \subseteq \mathcal{S}$ such that $\bigcup_{S_i \in \mathcal{S}'} S_i = C$ and the *minimum set cover problem* (SC) is to find a cover of minimum size.

We consider then the following hierarchy for SC .

- WSC** Weighted SC . Each subset has an integer non-negative weight and the objective is to find a cover of minimum total weight.
- PWSC(k)** Polynomially weighted SC . This is like WSC with the restriction that the weights are all bounded above by n^k , where k is any fixed integer and $n = |\mathcal{S}|$. We note that $PWSC(0) \equiv SC$ (recall that the weights are integers).
- RSSC** Regular on $\mathcal{S}$ SC . It is a set covering problem with the additional restriction that the cardinalities of the members of $\mathcal{S}$ are all equal.
- RCSC** Regular on C SC . It is a set covering problem where all the elements of the ground set C belong to the same number of subsets.
- BCSC(Δ)** Bounded on C SC . This is a set covering problem where the number of subsets every element belongs, is bounded above by Δ (which can be a function of n). We denote by $BCSC(O(1))$ the instances of $BCSC(\Delta)$ with Δ a universal constant.
- PWBCSC(k, Δ)** Polynomially weighted and bounded on C SC . It is a set covering problem polynomially weighted by n^k and bounded on C by Δ . We use the notation $PWBCSC(k,0,\Delta)$ to denote the instances of $PWBCSC(k,\Delta)$ where the weights are bounded above by an

integer (absolute) constant.

CSC Conformal *SC*. This is like *SC* with the additional constraint that for every three sets S_i, S_j, S_k of $\mathcal{S}$, the set $(S_i \cap S_j) \cup (S_j \cap S_k) \cup (S_i \cap S_k)$ is contained in a set of $\mathcal{S}$.

There is a graph formalism for *SC* which will be used in what follows.

Definition 1. For every instance I of *SC*, the characteristic graph of I is the bipartite graph $B = (S, C, E)$ where the vertex set S denotes the family $\mathcal{S}$ and the vertex set C the elements of the set $\mathcal{C}$. The graph B contains the edge $s_i c_j$ if $c_j \in S_i$. ■

For instance, *RSSC* (*RCSC*) is characterised by bipartite graphs (definition 1) in which the S -vertices (C -vertices) have all the same degree and for *BCSC*(Δ) the corresponding characteristic graph has the degrees of its C -vertices bounded above by Δ .

We shall now show how the members of the *SC* hierarchy defined above are related by continuous reductions.

Theorem 1. If σ is the number of the sets participating to the optimal solution of the instance of *WSC* and k any fixed constant, then

$$\begin{aligned} SC &\stackrel{1+\epsilon}{\Longleftrightarrow} RSSC \\ SC &\stackrel{1}{\Longleftrightarrow} RCSC \\ WSC &\stackrel{1+\epsilon}{\Longleftrightarrow} PWSC(k), k > 1 \\ WSC &\stackrel{2}{\Longleftrightarrow} PWSC(1) \\ WSC &\stackrel{\sigma}{\longrightarrow} SC \\ PWBCSC(k, 0, \log n) &\stackrel{1}{\Longleftrightarrow} BCSC(\log n) \\ PWBCSC(k, O(1)) &\stackrel{1}{\Longleftrightarrow} BCSC(O(1)). \end{aligned}$$

Also, if the weights need not to be integer, then

$$WSC \stackrel{1}{\Longleftrightarrow} PWSC(k).$$

Proof: For all cases the direction $\stackrel{1}{\Longleftarrow}$ is trivially true as for every line, problems on righthand side are subproblems of the ones on lefthand side.

$$SC \stackrel{1+\epsilon}{\Longleftrightarrow} RSSC$$

Let us suppose that an instance I of *SC* is given and moreover that we have a ρ -approximation algorithm $\mathcal{A}'$ for *RSSC*.

Let B the characteristic graph of I (definition (1)) and let $\gamma = \max_{s_i \in S} |\Gamma(s_i)|$, $\bar{\gamma} = \min_{s_i \in S} |\Gamma(s_i)|$ denoting the maximum and minimum degrees, respectively, of the S -vertices of B . We insert $\gamma - \bar{\gamma}$ C -vertices in B and for each point s_j of the graph B , such that $|\Gamma(s_j)| \leq \bar{\gamma}$, we link arbitrarily $\gamma - |\Gamma(s_j)|$ of these vertices to s_j .

We obtain so an instance I' of $RSSC$ on which we apply $\mathcal{A}'$.

Let λ the cardinality of the obtained solution. This solution constitutes of course a solution of the same cardinality for the instance I . For the optimal solutions β, β' of I, I' , respectively, we have $\beta' \leq \beta + 1$. Indeed, if the size of the optimal solution of I is β and as the ground set C is covered by them, there remain the newly added C -vertices in I' to be covered. All these elements can be covered by just one set among the minimum cardinality sets in I and therefore at most $\beta + 1$ sets of I' are sufficient to cover the ground set of I' .

We have thus

$$\rho \geq \frac{\lambda}{\beta'} \geq \frac{\lambda}{\beta + 1}$$

which leads to

$$\frac{\lambda}{\beta} \leq \rho \left(1 + \frac{1}{\beta}\right) \leq \rho(1 + \epsilon)$$

for instances where $\beta \geq \frac{1}{\epsilon}$ (or equivalently β is not bounded).

*

$$SC \xrightarrow{1} RCSC$$

Let us suppose that an instance I of SC is given and moreover that we have a ρ -approximation algorithm $\mathcal{A}'$ for $RCSC$. Let us also denote this time by γ the maximum degree of the C -vertices of the incidence graph B expressing I . We construct then easily an instance I' of $RCSC$ by adding for each c_i such that $|\Gamma(c_i)| \leq \gamma$, $\gamma - |\Gamma(c_i)|$ S -vertices linked to c_i . The so obtained bipartite graph B' expresses an instance of $RCSC$ in which every solution corresponds to a solution of I with the same size and vice-versa. So this transformation is continuous with expansion 1.

*

$$WSC \xrightarrow{1+\epsilon} PWSC(k), \quad k > 1$$

$$WSC \xrightarrow{2} PWSC(1)$$

In a general way, from an instance of WSC we can construct an instance of $PWSC(k)$ where we transform the weights of the former by taking, for $i = 1, 2, \dots, n$, $w'_i = \lceil n^k \frac{w_i}{w} \rceil$. Obviously all the new weights are integer and bounded by n^k . Let us call φ the transformation just described.

Let us consider an instance I of WSC , and let us suppose the existence of a polynomial time approximation algorithm $\mathcal{A}$, with a (universally) constant approximation ratio ρ , supposed to solve $PWSC(k)$.

We apply φ in order to construct an instance I' of $PWSC(k)$, after we apply $\mathcal{A}$ in I' and we store the obtained solution. We take off the maximum weight sets of I (without removing the elements that these sets contain), obtaining so a new instance I_1 of WSC in which we apply φ ; let I'_1 the obtained instance of $PWSC(k)$, in which we apply $\mathcal{A}$ and

we store the obtained solution if it is feasible for I' (and consequently for I). We iterate the process until no feasible solutions for I are produced. Finally we consider the smallest solution between the so obtained ones.

Let w'_i the set weights produced by φ , Λ' the family of sets participating in this solution, λ' its value and β' , the value of the optimal solution of I' .

It is easy to see that there may be a number of consecutive iterations for which the solution found when applying the process just described does not change. If this is the case for the solution Λ' , we will reason with respect to the last iteration in which solution Λ' has been obtained. We call f this iteration. Let $\bar{I}$ the instance of WSC in the beginning of iteration f , and w the maximum set weight in $\bar{I}$. Obviously, Λ' contains at least one set of weight w . For λ' we have the following.

$$\lambda' = \sum_{S_i \in \Lambda'} w'_i = \sum_{S_i \in \Lambda'} \lceil n^k \frac{w_i}{w} \rceil \geq \sum_{S_i \in \Lambda'} n^k \frac{w_i}{w} = \frac{n^k}{w} \sum_{S_i \in \Lambda'} w_i. \quad (1)$$

We select the same family as solution Λ of I ($\Lambda = \Lambda'$) and let λ its value. Also, let B ($|B| = \sigma$) be the sets of the optimal solution of I and let β the value of this solution.

For λ, β we have:

$$\begin{aligned} \lambda &= \sum_{S_i \in \Lambda} w_i \leq \frac{w}{n^k} \lambda' \\ \beta' &\leq \frac{n^k}{w} \beta + \sigma. \end{aligned} \quad (2)$$

By the hypothesis on the ratio of $\mathcal{A}$ and by (1) and (2), we have:

$$\frac{\frac{n^k}{w} \sum_{S_i \in \Lambda} w_i}{\frac{n^k}{w} \beta + \sigma} \leq \frac{\lambda'}{\beta'} \leq \rho$$

or equivalently

$$\frac{\lambda}{\beta} \leq \rho \left(1 + \frac{w\sigma}{n^k \beta} \right). \quad (3)$$

Thus the expansion given by (3) is equal to $\left(1 + \frac{w\sigma}{n^k \beta} \right)$.

Now, consider a solution of value λ_j obtained on an instance I_j for which the maximum weight $w_j = w_\beta$, where w_β the maximum weight in the optimal solution of WSC . Of course, this assures that such a solution (of value λ_j) exists and is feasible for I (the initial instance of WSC) and moreover that the optimum for I_j is the same as the optimum for I (it is easily seen that for this optimum holds $\beta \geq w_\beta + \sigma - 1$).

From (3) we have then

$$e = 1 + \frac{w_\beta \sigma}{n^k \beta} \leq 1 + \frac{w_\beta \sigma}{n^k (w_\beta + \sigma - 1)} \stackrel{\sigma \leq n}{\leq} 1 + \frac{1}{n^{k-1}}.$$

As $\lambda \leq \lambda_j$, the approximation ratio provided by (3) holds also for the solution of value λ . Hence for $k \geq 2$, and for $n \geq \frac{1}{\epsilon}$ (we can suppose that n is not bounded by an absolute constant) we have $e \leq 1 + \epsilon$, while for $k = 1$ we have $e \leq 2$.

*

$$WSC \xrightarrow{\sigma} SC$$

The proof of the last part of the theorem is analogous to the proof given just above, but a little different.

Let us suppose that a ρ -approximation algorithm $\mathcal{A}'$ is given for SC as well as an instance I of WSC . We shall show how $\mathcal{A}'$ can be used to solve I .

We take off the weights on the sets of I and we execute $\mathcal{A}'$ on the resulting instance. After, we take the weights into account to evaluate the value of the so obtained solution and we store it. We take off the maximum weight sets and we repeat the same procedure on the resulting instance. We restore the obtained solution (always if it is feasible) and we reiterate the process until no feasible solution is produced. Finally we take as solution the one of the smallest value between the so obtained solutions.

Let $\bar{\lambda}$ the value of this solution. Also, let I_i the instance (in iteration i) where its maximum weight equals w_β (the maximum weight of the optimal solution of I), let $\lambda_i, \beta_i, \beta$, the values of the solution provided by $\mathcal{A}'$ on I_i , the optimal solution of I_i and the optimal solution of I respectively and let λ'_i the solution provided by $\mathcal{A}'$ on the unweighted instance I_i .

Obviously

$$\beta_i = \beta \geq w_\beta + \sigma - 1$$

where the first equality holds because despite of the deletion of some sets all the sets constituting the optimal solution of I are present in I_i , while the second inequality holds because the optimal solution of I contains at least a set of size w_β .

Also, by the hypothesis on the approximation ratio of $\mathcal{A}'$, $\lambda'_i \leq \rho\sigma$, where σ is as above. This is because if σ_{opt} , the cardinality of the optimal solution of the unweighted instance I_i , then $\sigma_{\text{opt}_i} \leq \sigma$ and therefore $\lambda'_i \leq \rho\sigma_{\text{opt}_i} \leq \rho\sigma$ (we recall that none of the sets constituting the optimal solution of I has been removed yet in iteration i).

For the approximation ratio of $\mathcal{A}'$ when running on I_i we have:

$$\frac{\lambda_i}{\beta_i} = \frac{\lambda_i}{\beta} \leq \frac{w_\beta \lambda'_i}{w_\beta + \sigma - 1} \leq \lambda'_i \leq \rho\sigma.$$

Therefore, as $\bar{\lambda} \leq \lambda_i$, the approximation ratio of $\mathcal{A}'$ remains the same for the finally selected solution of value $\bar{\lambda}$ and consequently, an expansion equal to σ is found.

*

However, for some restricted cases of WSC we can obtain straightforward continuous reductions with expansion equal to 1, as we prove in what follows.

$$\begin{aligned} PWBCSC(k, 0, \log n) &\xrightarrow{1} BCSC(\log n) \\ PWBCSC(k, O(1)) &\xrightarrow{1} BCSC(O(1)) \end{aligned}$$

Given an instance I of WSC we can obtain an instance of SC in the following way.

Let us first denote by $w_i, 1 \leq i \leq n$ the weight of the set S_i in I .

By referring to B we replace each point s_i by w_i points and each point c_j by $\prod_{k=1}^{|\Gamma(c_j)|} w_{j_k}$ points, where $\Gamma(c_j) = \{s_{j_1}, s_{j_2}, \dots, s_{j_k}\}$. We create so S -groups $\overline{S}_i, 1 \leq i \leq n$, one for every point s_i of B (or equivalently for every set S_i of I) and also C -groups $\overline{C}_j$ one for every point c_j of B (or equivalently for every element c_j of I). After, for every group $\overline{S}_{j_l}$ corresponding to a vertex of B adjacent to the vertex c_j we link the points of $\overline{S}_{j_l}$ to the points of $\overline{C}_j$ in such a way that every point of $\overline{S}_{j_l}$ is adjacent to exactly $\frac{\prod_{k=1}^{|\Gamma(c_j)|} w_{j_k}}{w_{j_l}}$ points of $\overline{C}_j$ and not two points of $\overline{S}_{j_l}$ are adjacent to the same point of $\overline{C}_j$. In other words for each c_j , we can order, arbitrarily, its neighbours in B , say $s_{j_1}, s_{j_2}, \dots, s_{j_{|\Gamma(c_j)|}}$, for each corresponding group, say $\overline{S}_k$, and number its points by $1, 2, \dots, w_k$ arbitrarily. Finally, we number the points of $\overline{C}_j$ by $1, 2, \dots, \prod_{k=1}^{|\Gamma(c_j)|} w_{j_k}$. Then we express each one of those numbers in $|\Gamma(c_j)|$ integer coordinates, the l th coordinate taking all the integer values from 1 to w_{j_l} . Now it is easy to see that the k th point of a group, for example of the group $\overline{S}_{j_l}$, is linked in the incidence graph B' of the new instance I' , to exactly those numbers having their j_l th coordinate equal to k .

The so constructed bipartite graph B' is the incidence graph of an instance I' of SC .

We claim that any minimal solution for SC in I' has to include only whole $\overline{S}$ -groups and not parts of them.

Indeed, let us suppose that there is a solution Λ' , ($|\Lambda'| = \lambda$), for which there is a $\overline{C}$ -group, say $\overline{C}_j$, for which none of its adjacent $\overline{S}$ -groups is entirely contained in Λ' . The way B' is conceived implies that if from the $\Gamma(c_j)$ groups incident to $\overline{C}_j$, Λ' misses in the k_1 th, k_2 th, $\dots, k_{|\Gamma(c_j)|}$ th element respectively, then the point expressed in integer coordinates by $k_1 k_2 \dots k_{|\Gamma(c_j)|}$ will be not covered by Λ' . Thus for each $\overline{C}_j$ at least one of its incident $\overline{S}$ -groups is entirely contained in every solution of I' .

Of course, as we can assume that Λ' is minimal, by the previous discussion we can conclude there that no strict subset of a $\overline{S}$ -group is included in this solution.

It is obvious now, that given an instance I of WSC , we can transform it, by the way just discussed, into an instance I' of SC in which by applying Λ' we obtain a solution Λ' of cardinality λ' . This solution is straightforwardly associated to a solution Λ of WSP of the same value, by picking from every $\overline{S}$ -group of I' included in Λ' , the corresponding set in I and by including it in Λ .

Obviously, for the corresponding optimal solutions β, β' we have $\beta' \leq \beta$.

Henceforth, once $\mathcal{A}'$ is supposed to have approximation ratio ρ , we have:

$$\frac{\lambda}{\beta} \leq \frac{\lambda'}{\beta'} \leq \rho.$$

Thus the reduction just described is continuous with expansion 1.

Of course this reduction is not performed in polynomial time for every function Δ .

In fact if an element of C is contained in $O(n)$ sets of $\mathcal{S}$ (we recall here that $n = |\mathcal{S}|$) having weights $O(n)$, then the number of points of the corresponding $\overline{C}$ -group is of $O(n^{O(n)})$, hence exponential on the size of I .

It is easy to see that the polynomiality of the reduction is preserved whenever $w_i = O(1)$ and $\Gamma_B(c_i) = O(\log n), \forall i \leq |C|$ or whenever $w_i = O(n^k)$ and $\Gamma_B(c_i) = O(1), \forall i \leq |C|$ (these are the extremal cases with respect to w_i and $\Gamma_B(c_i)$).

Let us remark before completing the proof of the theorem, that if the weights are not integer we can easily prove that $WSC \xleftrightarrow{1} PWSC(k)$.

Let w the maximum weight of an instance I of WSC and let us suppose, without loss of generality, that $w > n^k$. Then, the instance I' of $PWSC(k)$ remains the same as I but every weight w_i of I is scaled down to $w'_i = n^k \frac{w_i}{w}$ in I' . Obviously, all the weights of this new instance are smaller than n^k . Now, given a solution of I with cost c we can obtain a solution for I' with cost $c' = n^k \frac{c}{w}$ by choosing exactly the same subsets to belong to the solution of I' , while a solution of cost c' in I' is immediately associated to a solution of cost $c = c' \frac{w}{n^k}$ in I .

Thus, the reduction is continuous with expansion 1.

With this last reduction we complete the proof of the theorem. ■

3 Hitting Set Hierarchy

Let $\mathcal{C} = \{C_1, \dots, C_m\}$ a family of m subsets drawn from an n -element set S . A subset $H \subseteq S$ is a *hitting set* for the family $\mathcal{C}$ if H has a non-empty intersection with each element of this family and the *minimum hitting set problem* is that of finding a hitting set of minimum cardinality.

HS is the dual, via the interchange of the two vertex sets of the incidence graph defined in definition 1, SC , this duality establishing a continuity between the two problems expressed in terms of the following proposition.

Proposition 1. $SC \xleftrightarrow{1} HS$.

Consequently, we can, in a similar way as in the case of SC , define the corresponding HS -hierarchy, where the roles of the family of sets and of the ground set are interchanged. Of course for the hitting set the weights are considered on the elements of the ground set S .

Let WHS , $PWHS(k)$, $RCHS$, $RSHS$, $B_cHS(\Delta)$, $PWB_cHS(k, \Delta)$, the corresponding problems of the HS hierarchy.

Theorem 2. If σ is the number of the elements of $\mathcal{C}$ participating to the optimal solution of the instance of WHS and k any fixed constant, then

$$\begin{aligned}
 HS &\xleftrightarrow{1+\epsilon} RCHS \\
 HS &\xleftrightarrow{1} RSHS \\
 WHS &\xleftrightarrow{1+\epsilon} PWHS(k), k > 1 \\
 WHS &\xleftrightarrow{2} PWHS(1) \\
 WHS &\xrightarrow{\sigma} HS \\
 BCHS(\log n) &\xleftrightarrow{1} PWBCHS(k, 0, \log n) \\
 BCHS(O(1)) &\xleftrightarrow{1} PWBCHS(k, O(1)).
 \end{aligned}$$

Also, if the weights need not to be integer, then

$$WHS \stackrel{1}{\Longleftrightarrow} PWHS(k).$$

The discussion of sections 2 and 3 is summarized by the following theorem.

Theorem 3.

- *SC, RSSC, RCSC, HS, RCHS, RSHS are approximate equivalent.*
- *BCSC($\log n$), PWBCSC($k, 0, \log n$), BC $HS(\log n)$, PWBC $HS(k, 0, \log n)$ are equivalent.*
- *WSC, WHS, PWSC(k), PWHS(k), $k > 1$ are approximate equivalent.*
- *PWSC(1), PWHS(1) are equivalent.*
- *BCSC($O(1)$), PWBCSC($k, O(1)$), BC $HS(O(1))$, PWBC $HS(k, O(1))$ are equivalent.*

Recently some researchers ([1]), have proved the following.

Theorem 4. ([1]). *VC even for bounded degree graphs and IS for bounded degree graphs do not admit a polynomial time approximation schema unless $P = NP$.*

As VC is a subproblem of SC, the following corollary concludes this section.

Corollary 1. *If $P \neq NP$ then there exists an $\epsilon > 0$ such that there is no approximation algorithm for SC, RSSC, RCSC, HS, RCHS, RSHS, WSC, WHS, PWSC(k), PWHS(k), guaranteeing an approximation ratio smaller than $1 + \epsilon$.*

4 Vertex-Covering-by-Cliques Hierarchy

Given a graph G , a *partition of the vertices of G into cliques* (covering of the vertices of G by cliques) is a system of cliques of G such that each vertex is contained in only one (at least one) clique and the *minimum partition into cliques* (PC) (*minimum vertex covering by cliques* (VCC)) problem is to find such a partition (covering) of minimum cardinality. Also, given a graph $G = (V, E)$, a *coloring* is a mapping f associating a color to each vertex in such a way that adjacent vertices have different colors and the *minimum graph coloring problem* (GC) is to find the minimum number of colors that constitute a coloring for G .

Finally, given a graph $G = (V, E)$, an *edge covering by cliques* is a system of cliques which covers all the edges of G , and the minimum edge covering by cliques problem (ECC) is

to find a minimum cardinality such system.

Obviously, each color induces an independent set in G and a clique in $\overline{G} = (V, \{v_i v_j : v_i v_j \notin E\})$, the complement of G , therefore PC and GC are equivalent. On the other hand, PC and VCC are equivalent as every partition constitutes also a covering and from every covering we can immediately obtain a partition of the same cardinality.

In fact the following holds ([14]).

Proposition 2.

$$\begin{array}{ccc} PC & \xleftrightarrow{1} & VCC \\ VCC & \xleftrightarrow{1+\epsilon} & ECC \end{array}$$

We construct now the following hierarchy for VCC .

- WVCC** Weighted VCC . Given a graph with non-negative integer weights, the weight of a clique can be defined as the maximum of the weights on its vertices, and the objective is to minimize the sum of the weights of the cliques participating in the partition.
- PWVCC(k)** Polynomially weighted VCC . This is like $WVCC$ with the additional constraint that all the weights are bounded by n^k , where n the order of the graph.
- RVCC** Regular VCC . This is a VCC with the additional constraint that the instance of the problem is a regular graph (all vertices have the same degree).
- VCC(L)** This is a VCC problem whose instances have cliques not exceeding L , for a fixed constant L .

It is easy to see that the weighted version of VCC defined above is NP -hard. Plainly, if an exact polynomial time algorithm is given for $WVCC$, then we can execute this algorithm on a graph whose all of vertices have weights equal to 1. In this case, the maximum weight of every clique is also equal to 1 and consequently, the optimal solution for the weighted case constitutes a minimum covering by cliques.

Theorem 5. *If σ is the number of the cliques participating in the optimal solution of the instance of $WVCC$ and k any fixed constant, then*

$$\begin{array}{ccc} VCC & \xleftrightarrow{2} & RVCC \\ WVCC & \xleftrightarrow{1+\epsilon} & PWVCC(k), k > 1 \\ WVCC & \xleftrightarrow{2} & PWVCC(1) \\ WVCC & \xrightarrow{\sigma} & VCC. \end{array}$$

Also, if the weights need not to be integer, then

$$WVCC \xleftrightarrow{1} PWVCC(k).$$

Proof: Here also the $\xleftarrow{1}$ part is easy for all cases as problems on righthand side are subproblems of the ones on lefthand side.

$$VCC \xrightarrow{2} RVCC$$

Let an instance $G = (V, E)$ ($|V| = n$) of VCC and let us suppose the existence of a polynomial ρ -approximation algorithm $\mathcal{A}$ for $RVCC$ (ρ an absolute constant as usual). From G we can construct an instance G' of $RVCC$ by taking two disjoint copies G_1, G_2 of G and the following (cross) edges between them: a vertex v_{i_1} of copy G_1 is linked to a vertex v_{j_2} of copy G_2 if and only if $v_i v_j \notin E$. The so constructed G' is regular as its vertices have all degree n .

Every solution for G' is constituted either from cliques, each one comprising vertices of only one copy, or from cliques some of them comprising vertices of both copies. To obtain a solution for G , in the first case we have only to choose the part of the solution constituting the smaller system, covering all vertices of one copie, and in the second case to take for each clique formed by vertices of both copies, the vertices including in one of them (a subset of vertices of a clique forms also a clique). Thus, in every case, if λ, λ_R (β, β_R) are respectively the solutions given by $\mathcal{A}$ on G and G' (optimal solutions for G and G') we have from the above discussion

$$\begin{aligned} \lambda &\leq \lambda_R \\ \beta &\geq \frac{\beta_R}{2} \end{aligned}$$

and consequently

$$\frac{\lambda}{\beta} \leq 2\rho.$$

So, the expansion of the described reduction is equal to 2.

*

$$\begin{aligned} WVCC &\xrightarrow{1+\epsilon} PWVCC(k), k > 1 \\ WVCC &\xrightarrow{2} PWVCC(1) \end{aligned}$$

Given an instance I of WSC we construct an instance I' of $PWSC(k)$ by scaling the weights $w_i, i = 1, 2, \dots, n$ of the former down to $w'_i = \lceil n^k \frac{w_i}{w_{\max}} \rceil$ for the latter, where $w_{\max}$ the maximum weight of a set in I . Obviously all the new weights are integer and bounded by n^k .

Let us now suppose the existence of a polynomial time approximation algorithm $\mathcal{A}$, with a (universally) constant approximation ratio ρ , supposed to solve $PWSC(k)$ and denote by Λ' the system of cliques participating in the solution provided by $\mathcal{A}$, λ' its value and β' , the value of the optimal solution of I' .

The arguments deriving expression (1) (theorem 1) hold also for this case by considering weights of cliques in place of subset weights and thus

$$\lambda' \geq \frac{n^k}{w} \sum_{C_i \in \Lambda'} w_i$$

where $\{C_i\}$, $i = 1, 2, \dots, |\Lambda'|$ the system of cliques covering the vertices of I' .

We select the same system as solution Λ of I ($\Lambda = \Lambda'$) and let λ its value. Also, let B ($|B| = \sigma$) the cliques of the optimal solution of I and let β the value of this solution.

Via the changing of sets to cliques the arguments giving expressions (2) and (3) (theorem 1) hold always and thus we conclude that

$$\frac{\lambda}{\beta} \leq \rho \left(1 + \frac{w\sigma}{n^k\beta} \right)$$

and consequently to an expansion $e = \left(1 + \frac{w\sigma}{n^k\beta} \right)$. Let us remark first that $\sigma \leq n$. Also we have $w \leq \beta$. Indeed, as the weight of a clique is the maximum weight of one of its vertices, and as we have a system covering all the vertices of the graph, at least one of the cliques constituting the optimal solution has to cover the maximum weight vertex and thus to have weight equal to w .

Thus finally,

$$e \leq \frac{1}{n^{k-1}}.$$

Hence for $k \geq 2$ and for $n \geq \frac{1}{\epsilon}$ we have $e \leq 1 + \epsilon$, while for $k = 1$ we have $e \leq 2$.

*

$$WVCC \xrightarrow{\sigma} VCC$$

Given an instance of $WVCC$ and an algorithm for VCC we can remove the weights of the vertices resolve the unweighted graph and after by re-inserting the weights evaluate the value of a solution for $WVCC$. If λ , β (λ' , β') are respectively the values of the approximate and optimal solution of $WVCC$ (VCC) and w the maximum weight of the instance of $WVCC$, we have

$$\begin{aligned} \lambda &\leq w\lambda' \\ \beta &\geq w \end{aligned}$$

and consequently

$$\frac{\lambda}{\beta} \leq \lambda' \leq \rho\beta'.$$

As $\beta' \leq \sigma$ we derive an expansion equal to σ for the described reduction.

*

For the last part of the theorem, where we relax the constraint of the integer weights the proof is similar to the corresponding case of theorem 1. ■

5 Vertex Covering Hierarchy

VC can be seen as a restriction of SC by considering, for every graph $G = (V, E)$ instance of VC , the set $V(G)$ as the set family and the set $E(G)$ as the ground set of an instance $(V(G), E(G))$ of SC , each vertex containing the edges incident to it. In fact such an instance is characterized by a graph B (definition 1) that has the degree of its C -vertices all equal to 2 and thus for all integers j

$$\mathcal{I}_j(VC) \subseteq \mathcal{I}_j(BCSC(2))$$

where given a problem Π , $\mathcal{I}_j(\Pi)$ is the set of instances of Π of size j .

The VC hierarchy considered in this paper is the following.

- WVC** Weighted VC . Each vertex has an integer non-negative weight and the objective is to find a vertex covering of minimum total weight.
- PWVC(k)** Polynomially weighted VC . This is like WVC with the restriction that the weights are all bounded above by n^k , $\forall k$, where n is the order of the graph.
- RVC** Regular VC . It is a vertex covering problem with the additional restriction that all the vertices of G have the same degrees.

Theorem 6. *If σ is the number of the sets participating to the optimal solution of the instance of WVC and k any fixed constant, then*

$$\begin{aligned} VC &\stackrel{\frac{5}{2}}{\Longleftrightarrow} RVC \\ WVC &\stackrel{1+\epsilon}{\Longleftrightarrow} PWVC(k), k > 1 \\ PWVC(k) &\stackrel{1}{\Longleftrightarrow} VC, \forall k \end{aligned}$$

Also, if the weights need not to be integer, then

$$WVC \stackrel{1}{\Longleftrightarrow} PWVC(k).$$

Proof: $\stackrel{1}{\Longleftarrow}$ part is, once more, easy for all cases, as problems on righthand side are subproblems of the ones on lefthandside.

$$VC \stackrel{\frac{5}{2}}{\Longrightarrow} RVC$$

Let us suppose that there exists a polynomial ρ -approximation algorithm for RVC . Given an instance $G = (V, E)$ of VC we can obtain an instance $G' = (V', E')$ of RVC by applying the construction described in theorem 5 for the corresponding case of VC , RVC .

A solution for VC in G' (distributed in both G_1, G_2) has to cover all edges of both G_1 and G_2 as well as all the cross edges. Moreover, for every solution T' of G' , if we denote by T'_1, T'_2 the part of T' in G_1, G_2 respectively, then T'_1 (T'_2) constitutes a vertex covering (perhaps not minimal) for G_1 (G_2). Thus, if hypothetical algorithm $\mathcal{A}$ gives a solution T'_R for VC in G' , one can easily obtain a solution T' for G by taking $T' = \min\{T'_1, T'_2\}$,

and by minimizing it. On the other hand, for any solution in G' , once the edges of the two copies have been covered, the only cross edges to be covered are the edges linking vertices of an independent set with respect to T'_1 in G_1 to vertices of another independent set with respect to T'_2 in G_2 . Consequently if τ and α respectively are the cardinalities of a minimum vertex covering and of a maximum independent set in G , τ_R the optimal solution of VC in G' , $|T'_R| = \tau'_R$ and $|T'| = \tau'$, we have obviously

$$\tau_R \leq 2\tau + \alpha.$$

From the previous discussion we have

$$\frac{2\tau'}{2\tau + \alpha} \leq \frac{\tau'_R}{\tau_R} \leq \rho$$

and consequently

$$\frac{\tau'}{\tau} \leq \rho \left(1 + \frac{\alpha}{2\tau}\right). \quad (4)$$

If $\alpha \leq \tau$, then expression (4) gives an expansion of $\frac{3}{2}$ for the reduction just described.

Let us now study the case where $\alpha \geq \tau$. We obtain a maximum matching M in G (let us denote by m the cardinality of M) and we take into account graph $G_m = (V_m, E_m)$ induced by the $2m$ vertices incident to edges of M . Let n_m the order, τ_m and α_m the cardinalities of a minimum vertex covering and of a maximum independent set respectively of G_m .

We have $\alpha_m \leq \tau_m$ because if $\alpha_m > \tau_m$, then $\alpha_m > \tau_m \geq m$ ([2]) and consequently $n_m > 2m$ which is impossible as by construction $n_m = 2m$ (recall that M is also a maximum matching of G_m and moreover it is perfect for this graph).

Hence, if we apply in G_m the reduction described previously, we obtain

$$\frac{\tau'_m}{\tau_m} \leq \frac{3}{2}\rho$$

where τ'_m the cardinality of the solution of RVC on graph G'_m constructed as described above.

As the so obtained vertex covering for G_m can constitute part of a vertex covering for the whole of G , we have now to examine how we can cover edges in $E \setminus E_m$.

Obviously, V_m is a vertex covering for G as it is the set of vertices incident to a maximum matching of G ([2,6,16]) and $m \leq \tau_m \leq \tau'_m \leq n_m$. Thus, there are less than m vertices of V_m not taken into account in the solution of G_m and if now we take also these vertices into account we form a solution for the whole of G of cardinality less than $\tau'_m + m$ for which holds

$$\frac{\tau'_m + m}{\tau} \leq \frac{\tau'_m}{\tau} + \frac{m}{\tau} \stackrel{\tau_m \leq \tau}{\leq} \frac{\tau'_m}{\tau_m} + \frac{m}{\tau} \leq \frac{3}{2}\rho + 1 = \rho \left(\frac{3}{2} + \frac{1}{\rho}\right) \stackrel{\rho \geq 1}{\leq} \frac{5}{2}\rho.$$

Thus, given a graph G and a polynomial ρ -approximation algorithm for RVC the minimum between the set of vertices incident to the edges of a maximum matching in G and the solution provided by the reduction described assures an expansion equal to $\frac{5}{2}$.

In fact, from a practical point of view, an expansion $\frac{5}{2}$ for VC is not significant at all. By saying “practical point of view”, we mean that if the concept of continuity could serve to

conceive efficient approximation algorithms for problems, by using efficient approximation algorithms for some easier problems, then the found expansion would not lead to a new approximation algorithm for VC more efficient than the known one (with approximation ratio 2, ([6,16])) even on the hypothesis that a polynomial time approximation schema would be conceived for RVC . But we have written down this reduction in order to show that RVC is almost as difficult as VC and moreover that both problems could belong in the same subclass of problems, where a single algorithm for the more restrictive between them, can lead to algorithms with a constant error for all other problems in the class.

On the other hand, if *random graph theory* ([3]) provides us with some intuition on some properties possessed by “almost all” graphs, then we can conclude that the class of graphs where $\alpha \leq \tau$ is not only not empty, but much bigger than the one where $\alpha \geq \tau$ (think that $\alpha \geq \tau$ contradicts the existence of a matching of cardinality $\lfloor \frac{n}{2} \rfloor$ in a graph of order n , and that the conjecture on the existence of such a matching is true in almost all graphs even if the edge probabilities tend slowly to 0).

Thus, if there exists an approximation algorithm for RVC with approximation ratio $\leq \frac{4}{3}$, then the solutions of VC in all graphs admitting a perfect matching can be approximated in a ratio strictly smaller than 2. In fact, if G admits a perfect matching, the reduction described above has expansion $\frac{3}{2}$ and consequently the result becomes already more significant.

*

$WVC \xrightarrow{1+\epsilon} PWVC(k), k > 2$ is obtained immediately from the corresponding case of theorem 1 for SC , as VC and its variants constitute subproblems of the corresponding variants of SC .

*

$$PWVC(k) \xleftrightarrow{1} VC, \forall k$$

Given an instance I of $PWVC(k)$ we can construct an instance I' of VC by replacing every vertex v_i of weight w_i by an independent set of size w_i and if two vertices v_i, v_j are linked by an edge in I , we draw a complete bipartite graph between the two independent sets of sizes w_i, w_j . Then, the proof is completed with arguments similar to the ones of theorem 1 for the cases $BCSC(\log n) \xleftrightarrow{1} PWBCSC(k, 0, \log n)$ and $BCSC(O(1)) \xleftrightarrow{1} PWBCSC(n, O(1))$.

*

Finally, the equivalencies $WVC \xrightarrow{1} PWVC(k)$ (if the weights need not to be integer) can be proved in a similar way as in the case of SC . ■

Corollary 2. VC, WVC and $PWVC(k), \forall k$ are approximate equivalent.

6 Expansions of Reductions between Problems in Different Hierarchies

We study in this section expansion for reductions linking problems of the different hierarchies established above.

But first let us discuss a *query optimization problem* from the *theory of relational databases* ([23]) and its relation with *SC* with respect to its approximation.

6.1 Set Covering and Minimal Join Problem

Let us introduce *minimal join problem (MJP)* ([19]) by means of the following example.

Example 1. Let us suppose a relational schema $\mathcal{R}$ and the following query Q defined on a subset $\{R_1, R_2, R_3, R_4\}$ of $\mathcal{R}$. Let us suppose also that Q is expressed in terms of *QBE* ([24]), where the underlined symbols are called *example elements*. If there are attributes in the relations participating in Q containing the same example element (here for example $R_1.A_1$ and $R_2.A_2$), then only tuples containing the same value for these attributes have to be taken into account in the evaluation of Q .

R_1	A_1	B_1	C_1	R_2	A_2	B_2	R_3	A_3	B_3	R_4	A_4	B_4
	<u>x</u>	<u>y</u>	<u>z</u>		<u>x</u>	P		<u>x</u>	<u>z</u>		<u>y</u>	c_1

Condition Box
$3\underline{x} + 5\underline{y} \geq 8\underline{z} + 2$

In order to evaluate the above query, we have first to evaluate the expression in the condition box, and for doing that, we have first to decide from which relation every example element takes its values. We have thus to associate relational names to example elements in condition box. Once this association has been performed, we have to perform the joins implied by it. If for instance we associate R_1 to $\underline{x}$, R_4 to $\underline{y}$, R_3 to $\underline{z}$, two joins have to be performed in order the condition to be evaluated. If on the other hand we associate relation R_1 to all three example elements, we have not to perform any join at all. To conclude the example, we mention that once the necessary joins being performed, we evaluate the condition itself and we complete the actions on condition box by performing the selections on the relations, that is by taking into account only those tuples containing A_1- , B_1- , C_1- , A_2- , A_3- , B_3- , A_4- values that satisfy the condition in condition box.

■

It is easy to see that *if n relations are associated to the set of the example elements of the condition, then $n - 1$ joins have to be performed for the evaluation of the condition.*

Thus, given a relational schema $\mathcal{R}$ and a query Q defined over $\mathcal{R}$ and containing an algebraic expression E having example elements as variables, *MJP* is the problem of finding the least number of relations associating to the set of example elements of E .

As we can see, the solution of an instance of *MJP* is not dependent neither on the kind of the algebraic expression E to be evaluated, nor on the nature of the query Q . In fact, in the step of the processing of Q concerning the evaluation of E , the only interesting information obtained from Q is exactly the association between the example elements of E and the relations using these elements in Q .

Theorem 7. *MJP is NP-complete.*

Proof: We reduce *MJP* to *SC*.

Let us suppose that we are given an instance (S, C) of *SC*, where S denotes the family of sets and C the ground set. We construct an instance of *MJP* by associating C to the example elements appearing in the expression E and S to the relations containing these elements in the body of Q , relation S_i containing an example element if and only if the corresponding set element is contained in the set S_i .

Let us suppose now that a solution Λ is given for *MJP*. We can easily construct a solution for *SC* by simply picking the elements of S corresponding to the elements of Λ . Conversely, if a solution Λ' is given for *SC* by selecting the relations corresponding to the elements of Λ' , we form a solution for *MJP*, because covering all the elements of C is equivalent to finding a subset of relations associated to all example elements of E . ■

The previous reduction is continuous as, to every instance of *MJP* (*SC*) corresponds an instance of *SC* (*MJP*) and moreover the solutions of both instances have the same cardinality. Therefore, an approximation algorithm solving one of these two problems solves also the other within the same approximation ratio.

The following corollary associates *MJP* to *SC*- and *HS*-hierarchy.

Corollary 3. $HS \xleftrightarrow{1} SC \xleftrightarrow{1} MJP$

6.2 Set Covering and Vertex Covering by Cliques

In this section, we conceive some simple continuous reductions between restricted instances of *SC* and *VCC*.

Theorem 8.

$$\begin{aligned} VCC(L) &\xrightarrow{1} SC \\ CSC &\xrightarrow{1} VCC \end{aligned}$$

Proof: Let us suppose that an approximation algorithm $\mathcal{A}$ is given for *SC*. Then, given an instance I of *VCC*(L), we construct an instance I' of *SC* by considering every clique of I as a set containing its vertices (recall that if L is an absolute constant, then we can find all cliques of I in time polynomial on the size of I). It is easy to see that a set covering of I' corresponds to a covering by cliques of the vertices of I with the same cardinality,

repeat

select a subset S_i of maximum cardinality; put it in the solution for SC ;
delete the elements of S_i ; update the cardinalities of the remained sets

until all elements of C are deleted;

Algorithm 1.

repeat

select a maximum clique in G ; put it in the solution of $VCC(L)$;
remove the vertices constituting it

until all vertices are removed

Algorithm 2.

thus, the hypothetical algorithm $\mathcal{A}$ has the same approximation ratio when treating both I and I' .

For the second result of the theorem, let us suppose that an instance I of CSC is given. We construct then the graph $G_C = (C, E, l)$ having vertex set C representing the ground set C , $E = \{c_i c_j : \{c_i c_j\} \subseteq S_k \in \mathcal{S}\}$ and l an application $E(G_C) \rightarrow \mathcal{S}$ such that $l(c_i c_j) = S_k$ if $S_k \supseteq \{c_i c_j\}$.

This graph which, of course, can be constructed for every instance of SC , in the case of CSC has the property that *every maximal clique of G_C represents a set of $\mathcal{S}$* ([2]).

Therefore, if we suppose that an approximation algorithm $\mathcal{A}$ is given for VCC , the clique covering $\mathcal{A}$ produces for G_C is immediately transformed to a set covering for I , thus the reduction is continuous with expansion 1. ■

In this point, we give an example on how the notion of continuity can be useful in conception and analysis of approximation algorithms.

Let us consider the natural greedy algorithm 1 for SC . Johnson ([9]) and Chvátal ([4]) have evaluated its approximation ratio and have given an upper bound equal to $1 + \ln \Delta$ for this ratio, where Δ the maximum cardinality of a set in $\mathcal{S}$.

The way the first reduction of theorem 8 is conceived, preserves the equality between the order of a clique in the instance of $VCC(L)$ and the cardinality of the corresponding subset of the instance of SC . Thus, algorithm 1 can be transformed to the greedy algorithm 2 for $VCC(L)$.

Moreover, the observation made on the relation between degrees of G and set cardinalities of the sets in $\mathcal{S}$, prevents us to make a quite complicated analysis in order to evaluate the approximation ratio of algorithm 2, as the following corollary 4 is immediately deduced.

Corollary 4. *Algorithm 2 resolves polynomially $VCC(L)$ within an approximation ratio bounded above by $1 + \ln L$.*

6.3 Set Covering and Vertex Covering

We explore now the expansion of a reduction between SC and VC .

Theorem 9. *If Δ the maximum number of sets, an element of C is contained in the instance of SC , then*

$$SC \xrightarrow{\frac{\Delta}{2}} VC.$$

Proof: Let us remark first that $VC \xrightarrow{1} SC$ as VC is a subproblem of $BCSC(2)$.

Let us consider the graph $G_S = (S, E, l)$ of an instance I of SC having vertex set S representing the family $\mathcal{S}$, $E = \{s_i s_j : S_i \cap S_j \neq \emptyset\}$ and l an application $E(G_S) \rightarrow C$ such that $l(s_i s_j) = c_k$ if $S_i \cap S_j = c_k$.

It is easy to see that, given an instance $I = (\mathcal{S}, C)$ of SC in the graph G_S , every element $c_i \in C$ corresponds to a clique whose vertices are the ones representing the sets containing c_i , and moreover, the edges of the clique are labelled by c_i .

Let us call such a clique *maximal* and let us denote it by C_i . Let us also denote by $|C_i|$ the order of C_i .

A solution of SC on the instance I can be seen in terms of G_S as a set S' of vertices that “covers” all the edge labels of G_S (there is no label that does not appear in an edge incident to a member of S').

Let us now reduce G_S as follows:

- For each maximal clique $C_i = (s_{i_1}, s_{i_2}, \dots, s_{i_k})$ with $|C_i| \geq 3$, we delete all the edges $s_{i_l} s_{i_m}, l > m + 1$. So, for every C_i , we are left with a cycle $C'_i = (s_{i_1}, s_{i_2}, \dots, s_{i_k}, s_{i_1})$.
- If $|C_i|$ is even, then we delete from C'_i the edges $s_{i_l} s_{i_m}, l = 2j, m = l + 1, j = 1, 2, \dots, \frac{|C_i|}{2}$.
If $|C_i|$ is odd, then we delete from C'_i the edges $s_{i_l} s_{i_m}, l = 2j, m = l + 1, j = 1, 2, \dots, \frac{|C_i|-1}{2}$.

Let G'_S denote the so obtained graph.

Let us suppose that a ρ -approximation algorithm $\mathcal{A}$ for VC is given. We execute $\mathcal{A}$ on G'_S .

The solution obtained from this execution constitutes a solution for I . In fact, as the obtained set is a vertex covering for G'_S by covering all its edges, it “covers” also all the labels on its edges, thus it satisfies the property of a set covering.

Let λ the cardinality of this solution.

On the other hand, let β, β' the cardinalities of the optimal solutions of SC on I and for VC on G'_S respectively.

It is easy to see that the solution for vertex covering on G'_S covers at most $\frac{\Delta}{2}$ times the

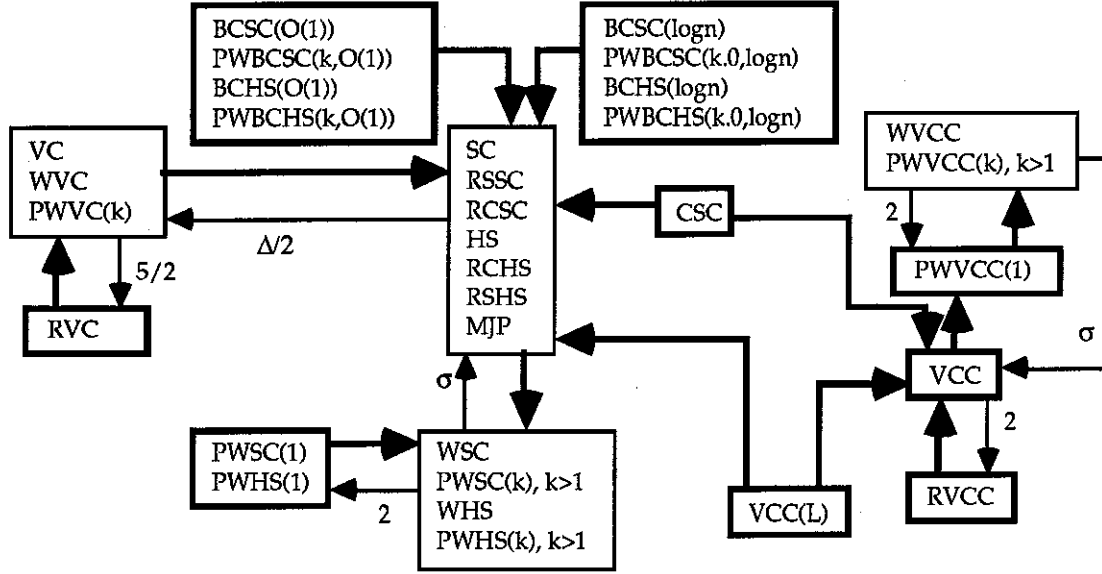


Figure 1: Reductions between combinatorial problems. Boxes with thick (thin) border-lines include equivalent (approximate equivalent) problems. Thick arcs represent reductions of expansion 1.

same label, whereas in the optimal solution for SC (in terms of G'_S), every label is covered at least one time.

Thus

$$\frac{\beta'}{\beta} \leq \frac{\Delta}{2}$$

and consequently

$$\frac{\lambda}{\beta} \leq \frac{\Delta}{2} \frac{\lambda}{\beta'} \leq \frac{\Delta}{2} \rho.$$

So an expansion of $\frac{\Delta}{2}$ is found. ■

The fact that a 2-approximation algorithm exists for VC together with theorem 9 results to a Δ -approximation algorithm for SC (see also [8,20]).

7 Conclusions and Discussion

We have studied the continuity of reductions in the interior of hierarchies of SC , HS , VC , VCC as well as of reductions associating members of different hierarchies. Figure 1 surveys all the reductions proposed above.

However, there are reductions in this paper not only non-continuous but also having not constant expansion. A good open problem is the conception of other ones that could assure at least constant expansions.

The notion of continuous reductions is usefull not only for better understanding of the structure of the class of NP -complete problems, but also because it provides some new instruments for obtaining approximation results. Let us take for example VCC and SC .

For VCC (we recall that VCC is equivalent to GC), there is a very strong negative result, namely there is no algorithm approximating the optimal solution of VCC within a relative error less than or equal to 100%. The generalization of the first result of theorem 8 ($VCC(L) \xrightarrow{1} SC$) on all instances of VCC (i.e. $VCC \xrightarrow{1} SC$) would produce immediately a lower bound equal to 2 for the value of the approximation ratio for SC and this would be an interesting negative result for a well known open problem as the approximability of SC , for which we know too little things.

As we can see, we have reductions associating SC or HS to VC , even if these reductions are not continuous. In any case, even when non-continuity appears, the conceived reductions can be useful as they describe in a certain way relations between classes of problems, each class containing problems equivalent with regard to their approximability. In what concerns the approximability relation between VCC on the one hand and one of SC , HS , VC on the other hand, we have no reductions (even non-continuous) relating them, and here we have to remark that the continuous reductions between restricted instances of SC and VCC in theorem 8 give no significant information regarding the approximability relation between VCC -hierarchy and one of SC -, HS -, VC -hierarchy. We propose henceforth the conception of such (even non-continuous) reductions as another open question.

Finally, we have to remark once more that all the conceived reductions are composable and transitive.

We are going now to discuss the relation between SC and *set packing problem* with respect to their approximations.

Given a collection $\mathcal{S}$ ($|\mathcal{S}| = n$) of finite sets, a *packing* is a subcollection $\mathcal{S}' \subseteq \mathcal{S}$ all of whose members are mutually disjoint and the *maximum set packing problem* (SP) is to find a packing of maximum size.

SC is greedily solved by algorithm 3, which in each round, solves a secondary problem, SP for the present case, and uses the solution of this last as partial solution for SC .

This technique, is the well known *master-slave technique* studied by Johnson in [9] and also by Simon in [21,22]. The master problem is a minimization problem (in our case SC) which calls a maximization problem, the slave problem (SP), and the algorithm determines the solution of the former by solving the latter. Johnson has pointed out that optimal algorithms for the slave problem lead to approximation algorithms for the corresponding master problem whose performance ratios are bounded above by $\ln n$, where n the size of the instance for the master problem and Simon in [21] has extended this by proving that if a ρ_s -polynomial time approximation algorithm exists for the slave problem then the master-slave technique leads to a ρ_m -polynomial time approximation algorithm for the master problem with $\rho_m \leq \ln n \rho_s$.

We have then the following corollary.

Corollary 5.

$$SC \xrightarrow{\ln n} SP$$

Let us also remark here that, for $VCC(L)$, algorithm 2 has approximation ratio $\ln n$ (the ratio given by corollary 4 is obviously better), as the slave problem called by the algorithm

repeat

 obtain a maximal set packing $\mathcal{SP}$ on $\mathcal{S}$; put $\mathcal{SP}$ in the solution for $\mathcal{SC}$;
 remove the elements of $\mathcal{C}$ covered by the sets of $\mathcal{SP}$; remove $\mathcal{SP}$ from $\mathcal{S}$

until all elements of $\mathcal{C}$ are removed;

Algorithm 3.

is polynomial.

The continuity adopted throughout this paper and also by Simon in [22] is much less restrictive than the one implied by the L -reduction of Papadimitriou and Yannakakis in [18]. There, the authors define a broad class of maximization problems called *MAXSNP*, whose main property was the completeness of its members under polynomial time approximation schemas. In [1] is proved that *there does not exist polynomial time approximation schema for any MAXSNP-hard problem unless $P=NP$.*

But the concept of the continuity (either the one adopted here and in [22], or the one adopted in [18]) inside the class of NP -complete problems can never be able to capture some intuitive equivalencies as for example the one between VC and IS , or more generally between a problem and its variants resulting from affine transformations of its objective function (under the same set of constraints), or the equivalence (under approximation) between a minimization and a maximization problem. The study of such a stronger and more representative notion of continuity, requiring novel definitions of some notions as optimization problem, polynomial reduction, approximation performance, etc, would contribute to the definition of a new more consistent polynomial approximation theory, better adapted to the real nature of the hard optimization problems.

References

- [1] S. Arora, C. Lund, R. Motwani, M. Sudan and M. Szegedy, *Proof Verification and Hardness of Approximation Problems*, Proc. of 33rd Annual IEEE Conference on Foundations of Computer Science, FOCS'92, 1992.
- [2] C. Berge, *Graphs and Hypergraphs*, North Holland, Amsterdam, 1973.
- [3] B. Bollobás, *Random Graphs*, Academic Press, London, 1985.
- [4] V. Chvátal, *A Greedy - Heuristic for the Set Covering Problem*, Math. Oper. Res., Vol. 4, pp. 233-235, 1979.
- [5] P. Crescenzi and A. Panconesi, *Completeness in Approximation Classes*, Proc. of Fundamentals of Computation Theory, FCT'89, LNCS, Vol. 380, pp. 116-126, Springer Verlag, 1989.

- [6] M. R. Garey and D. S. Johnson, *Computers and Intractability. A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, San Francisco, 1979.
- [7] M. Gondran et M. Minoux, *Graphes et Algorithmes*, eds. Eyrolles, Paris, 1985.
- [8] D. S. Hochbaum, *Approximation Algorithms for the Set Covering and Vertex Cover Problems*, SIAM J. Comput. vol. 11 (3), pp. 555-556, 1982.
- [9] D. S. Johnson, *Approximation Algorithms for Combinatorial Problems*, J. Comput. System Sci. Vol. 9, pp. 256-278, 1974.
- [10] R. M. Karp, *Reducibility Among Combinatorial Problems*, in R. E. Miller and J. W. Thatcher (eds.), *Complexity of Computer Computations*, Plenum Press, New York, pp. 85-103, 1972.
- [11] R. M. Karp, *On the Complexity of Combinatorial Problems*, Networks, Vol. 5, pp. 45-68, 1975.
- [12] Ph. G. Kolaitis and M. N. Thakur, *Logical Definability of NP Optimization Problems*, Technical Report UCSC-CRL-90-48, University of California at Santa Cruz, September 1990.
- [13] Ph. G. Kolaitis and M. N. Thakur, *Approximation Properties of NP Minimization Classes*, Technical Report UCSC-CRL-91-10, University of California at Santa Cruz, April 1991.
- [14] L. T. Kou, L. J. Stockmeyer and C. K. Wong, *Covering Edges by Cliques with regard to Keyword Conflicts and Intersection Graphs*, Comm. ACM, Vol. 21, pp. 135-139, 1978.
- [15] A. Panconesi and D. Ranjan, *Quantifiers and Approximation*, Proc. of 22nd Annual ACM Symp. on Theory of Computing STOC, pp. 446-456, 1990.
- [16] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice Hall, New Jersey, 1981.
- [17] C. H. Papadimitriou and M. Yannakakis, *The Travelling Salesman Problem With Distances One and Two*, Math. of Operations Research, 1990.
- [18] C. H. Papadimitriou and M. Yannakakis, *Optimization, Approximation and Complexity Classes*, J. Comput. System Sci., Vol. 43, pp. 425-440, 1991.
- [19] V. Th. Paschos, *Some Complexity Results for the Join Indicator Problem in Relational Databases*, Rapport de Recherche # 576, LRI, Université de Paris-Sud, Centre d'Orsay, 1990.
- [20] V. Th. Paschos, *A "Tour d'Horizon" about how Optimal Solutions to Set Cover and Set Packing can be Approximated*, manuscript, 1991.

- [21] H. U. Simon, *The Analysis of Dynamic and Hybrid Channel Assignment*, Research Report SFB 124-B1 10, Fachbereich Informatik, Universität des Saarlandes, 1988, also Proc. of Fundamentals of Computation Theory, FCT'89, 1989.
- [22] H. U. Simon, *On Approximate Solutions for Combinatorial Optimization Problems*, SIAM J. of Disc. Math., Vol. 3(2), pp. 294-310, 1990.
- [23] J. D. Ullman, *Principles of Database Systems*, Computer Science Press, 1982.
- [24] M. M. Zloof, *Query-By-Example: A Database Language*, IBM Syst. J., Vol. 4, pp. 324-343, 1974.