The Computational Complexity of Graph Problems with Succinct Multigraph Representation

Marek Karpinski *
Institut für Informatik, Universität Bonn, Wegelerstrasse 6, D-5300 Bonn

Klaus W. Wagner+
Institut für Mathematik, Universität Augsburg, Memminger Strasse 6, D-8900 Augsburg

1. Introduction

Special succinct (polylogarithmic in the number of vertices) representations of graphs used in VLSI design and other contexts cause a blow-up of the complexity of some graph problems which become hard for higher time and space complexity classes. This interesting phenomenon was studied recently in a number of papers (see [L 82], [GW 83], [W 84], [L 86], [PY 86], [LW 87a]).

In the present paper we study the computational complexity of graph problems when the graphs are described by *vertex multiplicity graphs*. The vertex multiplicity graph representation (for short: *VMG-representation*) enables us to describe an independent set of vertices which are connected with the remaining vertices of the graph in the same way by giving only one vertex and the size of the independent set. We prove that both UNARY NETWORK FLOW and PERFECT MATCHING problems are P-complete in VMG-representation. It is believed that none of theses problems is P-complete in standard representation (as both of them are in RNC, [KUW 85], [MVV 87]), and it is known that some special cases are in NC [GK 87]. (Let us note also the

* Supported in part by Leibniz Center for Research in Computer Science and DFG Grant KA 673/2 - 1.

+ Supported in part by DFG Grant WA 594/1 - 1.

recent result in [DK 87] that PERFECT MATCHING restricted to 3-regular graphs is still AC^0-hard for the general PERFECT MATCHING problem.) An interesting consequence of our result is that the VMG-versions of UNARY NETWORK FLOW and PERFECT MATCHING are not efficiently parallelizable unless **P** = NC (cf. [C 85]).
Surprisingly we are also able to prove that the following three NP-complete problems do not have a complexity blow-up in VMG-representation: CLIQUE, MAXIMUM INDEPENDENT SET and CHROMATIC NUMBER. The are all proven to remain NP-complete problems.

Finally we prove the existence of a graph problem whose VMG-version does have a complexity blow-up. This problem is some modification of the circuit value problem which is P-complete in standard representation and PSPACE-complete in VMG-representation.

2. Notation and Basic Definition

A *(un)directed vertex multiplicity graph* is a triple G = (V,E,b) where (V,E) is an ordinary (un)directed graph and b: $V \mapsto \mathbb{N}^+$ gives the multiplicity b(v) for every vertex v. Such a vertex multiplicity graph G = (V,E,b) is considered to be the description or representation of the graph $L_{VM}(G)$ with vertex set $\{(v,i): v \in V \text{ and } 1 \le i \le b(v)\}$ and edge set $\{((v,i),(u,j)): (u,v) \in E, 1 \le i$ $b(n) \text{ and } 1 \le j \le b(n)\}$.

The subscript VM to a graph problems means that the graphs are given in VMG-representation. Since the graphs can possibly described more succinctly using the VMG-representation, the complexity of a VMG-version of a graph problem cannot be less than the complexity of the graph problem in standard representation.

The complexity of the VMG-version of a graph problem heavily depends on how the vertex multiplicities b(v) are described in an instance to thaisproblem. If they are described in unary then the VMG-representation of a graph cannot be very succinct and the complexity of the VMG-version of a graph problem will not differ very much from the complexity of that graph problem in standard representation. In what follows we consider only the case that the vertex multiplicities are described in binary.

3. Network Flow and Matching

In this section we mainly consider the two problems UNARY NETWORK FLOW and PERFECT MATCHING which are not known to be **P**-complete (and believed not to be because they are in RNC, see [C 85], [KUW 85], [MVV 87]). We show that the VMG-versions of these problems become **P**-complete.

UNARY NETWORK FLOW

Instance: A directed acyclic graph $G = (V,E)$ with the *source* $s \in V$ and the *sink* $t \in V$, a *capacity function* $c\colon E \mapsto \mathbb{N}^+$ whose values are given in unary, a natural number f.

Question: Does there exist a flow of at least f units from s to t in G, i.e. does there exist a ϕ: E $\mapsto \mathbb{N}^+$ such that $\phi \le c$, $\Sigma_{(u,v)\in E}\phi(u,v) = \Sigma_{(v,w)\in E}\phi(v,w)$ for all $v \in V \setminus \{s,t\}$ and $\Sigma_{(s,w)\in E}\phi(s,w) = \Sigma_{(u,t)\in E}\phi(u,t) \ge f$?

If the values of the capacity function are given in binary we obtain the problem BINARY NETWORK FLOW which is shown to be $\le^{\log}{}_m$-complete for **P** in [LW 87b]. As mentioned above, UNARY NETWORK FLOW is not believed to be $\le^{\log}{}_m$-complete for **P**. We are able to prove

Theorem 1. UNARY NETWORK FLOW$_{VM}$ is $\le^{\log}{}_m$-complete for **P**.

Proof. By reduction from BINARY NETWORK FLOW. An edge (v,w) with capacity c is replaced by the two edges (v,u) and (u,w) with capacity 1 where u is a new vertex with multiplicity c. The vertices v and w obtain multiplicity 1.

That UNARY NETWORK FLOW$_{VM}$ is in **P** is seen by the following reduction to BINARY NETWORK FLOW:

$(V,E,c,b) \in$ UNARY NETWORK FLOW$_{VM}$ $\Leftrightarrow$ $(V,E,c') \in$ BINARY NETWORK FLOW

where $c'(v,w) = c(v,w)\cdot b(v)\cdot b(w)$, for all (v,w). □

Since the latter reduction also works for BINARY NETWORK FLOW$_{VM}$ we have

Corollary 2. BINARY NETWORK FLOW_{VM} is $\leq^{\log}{}_m$-complete for **P**. □

Now consider the perfect matching problem.

PERFECT MATCHING

Instance: An undirected graph $G = (V,E)$.

Question: Is there an $E' \subseteq E$ such that every $v \in V$ belongs to exactly one $e \in E$?

PERFECT BIPARTITE MATCHING is the restriction of PERFECT MATCHING to bipartite graphs. PERFECT MATCHING can be considered as a subproblem of the following problem.

PERFECT B-MATCHING WITH CAPACITIES.

Instance: An undirected graph $G = (V,E)$, functions $b: V \mapsto \mathbb{N}^+$ and $c: E \mapsto \mathbb{N}^+$.

Question: Is there a function $c': E \mapsto \mathbb{N}^+$ such that $c' \leq c$ and $\Sigma_{e \in E, v \text{ in } e} c'(e) = b(v)$ for all $v \in V$?

Obviously,

$(V,E,b) \in$ PERFECT MATCHING_{VM} $\Leftrightarrow$ $(V,E,b,c) \in$ PERFECT B-MATCHING WITH CAPACITIES

where $c(v,u) = b(v)+b(u)$.

Consequently, since PERFECT B-MATCHING WITH CAPACITIES is in **P** (cf. [GLS 87]) we obtain that PERFECT MATCHING_{VM} is in **P**. As mentioned above it is not believed that PERFECT MATCHING is **P**-complete. However, we can prove that even PERFECT BIPARTITE MATCHING_{VM} is **P**-complete.

Theorem 3. PERFECT BIPARTITE MATCHING_{VM} is $\leq^{\log}{}_m$-complete for **P**.

Proof. We reduce the **P**-complete problem BINARY NETWORK FLOW to PERFECT BIPARTITE MATCHING_{VM}. We start with the classical reduction of UNARY NETWORK FLOW to PERFECT BIPARTITE MATCHING which works as follows (see e.g. [KUW 87]):

Given a flow network $G = (V,E)$ with capacity function $c: E \mapsto \mathbb{N}^+$ and a natural number f. Let s (t) be the source (sink) node of G which has indegree (outdegree) 0. It is not hard to see that G has a maximum flow greater or equal to f if and only if the subsequently defined graph $G' = (V_1 \cup V_2, E')$

has a perfect matching.

$V_1 = \{(e,1,i) : e \in E \text{ and } 1 \leq i \leq c(e)\} \cup \{a_1,\ldots,a_f\}$,

$V_2 = \{(e,2,i) : e \in E \text{ and } 1 \leq i \leq c(e)\} \cup \{b_1,\ldots,b_f\}$, and

$E' = \{((e,1,i),(d,2,j)) : e,d \in E,\ e = (v_1,v_2) \text{ and } d = (v_2,v_3) \text{ for some } v_1,v_2,v_3 \in V\} \cup$

$\{((e,2,i),(e,1,i)) : e \in E\} \cup \{(a_j,(e,2,i)) : e \text{ is incident with } s\} \cup \{((e,1,i),b_j) : e \text{ is incident with } t\}$.

Obviously, G' can be described by the vertex multiplicity graph G" having vertex sets $V'_1 = \{(e,1) : e \in E\} \cup \{a\}$ and $V'_2 = \{(e,2) : e \in E\} \cup \{b\}$ and multiplicities $b'(e,1) = b'(e,2) = c(e)$ and $b'(a) = b'(b) = f$.

So far the unary case. However, the above reduction $(G,c) \to G"$ does also work for flow networks with capacities given in binary (i.e. with capacities which can be exponentially in the size of the input). □

Corollary 4. PERFECT B-MATCHING WITH CAPACITIES is $\leq^{\log}{}_m$-complete for P. □

4. Problems Whose VMG-Versions Have No Complexity Blow-Up

One example of such a problem is already given in Corollary 2. In this section we shall see that some of the most popular NP-complete problems belong to this category. To prove this we have only to check whether the VMG-versions of the problems in question are still in NP. For the following three problems this is not very hard.

CLIQUE

Instance: An undirected graph $G = (V,E)$, a natural number k.

Question: Does there exist a clique of size k in G, i.e. does there exist a $V' \subseteq V$ such that $\#V' = k$ and $V' \times V' \subseteq E$?

MAXIMUM INDEPENDENT SET

Instance: An undirected graph $G = (V,E)$, a natural number k.

Question: Does there exist a independent set of size k in G, i.e. does there exist a $V' \subseteq V$ such that $\#V' = k$ and $V'xV' \subseteq VxV \smallsetminus E$?

CHROMATIC NUMBER

Instance: An undirected graph $G = (V,E)$, a natural number k.

Question: Does there exist a coloring of G with k colors, i.e. does there exist a $c: V \mapsto \{1,...,k\}$ such that $(v,w) \in E$ implies $c(v) \neq c(w)$?

It is well-known that these problems are NP-complete (see [K 72]). We prove that their VMG-versions remain NP-complete.

Theorem 5. CLIQUE_{VM}, MAXIMUM INDEPENDENT SET_{VM} and CHROMATIC NUMBER_{VM} are $\leq^{\log}{}_m$-complete for NP.

Proof. Let $G = (V,E,b)$ be an undirected vertex multiplicity graph. The following obvious statements show how to reduce the VMG-versions of the above problems to their standard versions.

Since different vertices (v,i) and (v,j) of $L_{VM}(G)$ cannot belong to one clique we obtain

$$L_{VM}(G) \text{ has a clique of size } k \Leftrightarrow G \text{ has a clique of size } k.$$

Since with (v,i) all other (v,j) can be included in an independent set we obtain

$$L_{VM}(G) \text{ has an independent set of size } k \Leftrightarrow G \text{ has an independent set } V' \text{ such that } \textstyle\sum_{v \in V'} b(v) \geq k.$$

Since for a fixed $v \in V$ all (v,i) can have the same color we obtain

$$L_{VM}(G) \text{ has a coloring with } k \text{ colors} \Leftrightarrow G \text{ has a coloring with } k \text{ colors.} \qquad \square$$

The proof of the membership in NP is more involved for the VMG-version of the following problem.

HAMILTONIAN CIRCUIT

Instance: An undirected graph $G = (V,E)$.

Question: Does G have a hamiltonian circuit, i.e. does there exist a repitition-free enumeration $v_1, v_2, \ldots, v_m$ of V such that $(v_1,v_2), (v_2,v_3), \ldots, (v_{m-1},v_m), (v_m,v_1) \in E$?

It is well-known that this problem is NP-complete (see [K 72]). We prove

Theorem 6. HAMILTONIAN CIRCUIT$_{VM}$ is $\leq^{\log}{}_m$-complete for NP.

Proof. Let G = (V,E,b) be an undirected vertex multiplicity graph. Assume that $L_{VM}(G)$ has a Hamiltonian circuit C. For every (v,u)∈ E we guess the number d(v,u) of occurences of (v,u) in C (more exactly: the number of different ((v,i),(u,j)) in C). The function d is called the *trace* of C in $L_{VM}(G)$.

Consider some (v,u) ∈ E and assume that d(v,u) ≥ 3, i.e. there exist $i_1, i_2, i_3, j_1, j_2, j_3$ such that $((v,i_1),(u,j_1))$, $((v,i_2),(u,j_2))$ and $((v,i_3),(u,j_3))$ are in C. There are essentially two cases how these edges are included in C. They are shown in Figure 1.

(Figure 1)

In either case we can take away two exemplars of (v,u), one exemplar of v and one exemplar of u, and we obtain a new graph in which the remaining parts of the Hamiltonian circuit C form a new Hamiltonian circuit C'. This is shown in Figure 2.

(Figure 2)

And conversely, starting with such a graph and a Hamiltonian circuit C' as shown in Figure 2 we can add one exemplar of v, one exemplar of u and two exemplars of (v,u) to obtain the corresponding graph shown in Figure 1 with Hamiltonian circuit C.

Repeating this construction we obtain from b,d the functions b',d' such that d(v,u) ∈ {1,2}, $d'(v,u) \equiv d(v,u) \pmod 2$, $b'(v) = b(v)-1/2(d(v,u)-d'(v,u))$, $b'(u) = b(u)-1/2(d(v,u)-d'(v,u))$ and

$$L_{MV}(V,E,b) \text{ has a HC with trace } d \Leftrightarrow L_{MV}(V,E,b') \text{ has a HC with trace } d'.$$

Doing so for every edge (v,u)∈ E we eventually obtain the functions $\underline{b}$, $\underline{d}$ fulfilling

(1) $\underline{d}(v,u) \leq 2$ for all (v,u)∈ E,

(2) $\underline{b}(v) \leq \#V$ for all v∈ V, and

(3) $L_{MV}(V,E,b)$ has a HC with trace d ⇔ $L_{MV}(V,E,\underline{b})$ has a HC with trace $\underline{d}$.

Because of (2) the property " $L_{MV}(V,E,\underline{b})$ has a HC with trace $\underline{d}$" can be tested nondeterministically in polynomial time. □

5. An Example of a Problem Whose VMG-Version Has a Complexity Blow-Up

To obtain such examples we have to make essential use of graphs with high vertex multiplicities. In the preceding section we have seen that in many cases adding "equivalent" vertices (i.e. vertices which are connected with all other vertices of the graph in the same way as one particular vertex of the graph) does not really complicate the problem in question. Thus we are looking for a canonical way to make equivalent vertices non-equivalent.

Let $G = (V,E)$ be an arbitrary digraph. The vertices v and u of G are said to be equivalent iff $(\{v\}xV)\cap E = (\{u\}xV)\cap E$ and $(Vx\{v\})\cap E = (Vx\{u\})\cap E$. Define $\#v = \#\{u\text{: u equivalent to v}\}$. Let $p\colon V \mapsto \mathbb{N}$ be any function such that $\{p(u)\text{: u equivalent to v}\} = \{1,2, \dots ,\#v\}$. Define $G(p) = (V,\{(v,u) : (v,u)\in E \text{ and } p(u) = p(v)+1\})$. Obviously, for two such function p,q the graphs G(p) and G(q) are isomorphic. Hence the definition $G^* = G(p)$ is independent of the special choice of p.

Now we are able to modify every graph problem in such a way that, given a graph G, the graph G* has to fulfill the property associated with the problem. We shall demonstrate this for the circuit value problem.

CIRCUIT VALUE*

Instance: A directed graph G whose vertices are labelled by boolean functions, a special vertex of G which is called the output node, a natural number r and Boolean values $a_1, \dots ,a_r$.

Question: Is G* a correct Boolean circuit and does it provide output 1 for input $a_1, \dots ,a_r$? This means: 1. If v is a vertex of G* with indegree $m > 0$ then it is labelled with an m-ary Boolean function. 2. If $v_1, \dots v_s$ are the vertices of G* with indegree 0 (given in a fixed order) then $s = r$ and G* provides the value 1 at the output node if $a_1, \dots ,a_r$ are given as inputs to $v_1, \dots ,v_r$.

We are able to prove that CIRCUIT VALUE* has the same complexity as CIRCUIT VALUE (i.e. it is P-complete) and that CIRCUIT VALUE*_{VM} is PSPACE-complete.

Theorem 7. CIRCUIT VALUE* is $\leq^{\log}_{m}$-complete for P.

Proof. Since constructing G^* from G can be performed in polynomial time we obtain CIRCUIT VALUE*$\in$ **P**.

For the **P**-hardness we reduce CIRCUIT VALUE to CIRCUIT VALUE*. Let C be a Boolean circuit with the nodes $v_1, \dots ,v_r$. From C we construct the Boolean circuit D by adding new nodes $u_1, \dots ,u_r$ and an edge from v_i to u_i for every $i = 1, \dots ,r$. Every u_i is labelled with the identity function. Since the new nodes do not have any influence to the output computed by D the Boolean circuits C and D have the same input-output behaviour. However; all nodes of D are pairwise unequivalent. Hence $D^* = D$ and, consequently, C and D^* have the same input-output behaviour.□

Theorem 8. CIRCUIT VALUE$^*_{VM}$ is $\leq^{log}_{m}$-complete for **PSPACE**.

Scetch of the proof. Every Boolean circuit presented by a vertex multiplicity graph G has a level structure such that every level has at most $|G|$ nodes and all input nodes to level i come from level i-1. Thus such a Boolean circuit can be evaluated in polynomial space.

The **PSPACE**-hardness can be proved by a master reduction from an arbitrary problem $A \in$ **PSPACE** to CIRCUIT VALUE*. Let M be a one-tape Turing machine accepting A using space $p(|x|)$ for an input x where p is a suitable polynomial. It is very easy to construct to a given input x to M a vertex multiplicity graph which describes a Boolean circuit C_x with the following properties:

- C_x has $c^{p(|x|)}$ levels each consisting of $d \cdot p(|x|)$ nodes (for some constants $c,d>0$),
- if the inputs to the first level correspond to 0-1-encoding of the initial configuration of M on input x then the values of the nodes of the (i+1)th level correspond to the encoding of the i-th configuraion of the work of M on input x,
- the value of the first node of the $c^{p(|x|)}$-th level is 1 if and only if M accepts x.

The details of this construction are left to the reader. □

It would be interesting to study also the complexity of A^*_{VM} for other problems A, for example those dealt with in Section 4.

Acknowledgement

We are grateful to Martin Grötschel, Augsburg, for pointing out the similarity of the VMG-version of the matching problem and the problem of b-matching with capacities.

6. References

[C 85] Cook., S.A., A taxonomy of problems with fast parallel algorithms, Information and Control 64(1985), 2-22.

[DK 87] Dahlhaus, E., Karpinski, M., Perfect matching for regular graphs is AC^o-hard for the general matching problem, Research Report #RJ 5902, IBM Research, San Jose (1987), 1-7.

[GW 83] Galperin, H., Wigderson, A., Succinct representations of graphs, Information and Control 56(1983), 183-198.

[GK 87] Grigoriev, D. Yu., Karpinski, M., The matching problem for bipartite graphs with polynomially bounded permanents is in NC, Proc. 28th IEEE FOCS (1987), 166-172.

[GLS 87] Grötschel, M., Lovasz, L., Schrijver, A., Geometric Algorithms and Combinatorial Optimization, to appear in Springer Verlag 1988.

[K 72] Karp, R.M., Reducibilities among combinatorial problems, Proc. IBM Symposium on Complexity of Computer Computation, Plenum Press, New York 1972, 85-103.

[KUW 85] Karp, R.M., Upfal, E., Wigderson, A., Constructing a perfect matching is in random NC, Proc. 17th STOC (1985), 22-32.

[L 82] Lengauer, T., The complexity of compacting hierarchically specified layouts of integrated circuits, Proc. 23rd IEEE FOCS (1982), 358-368.

[L 86] Lengauer, T., Exploiting hierarchy in VLSI design, Proc. Aegean Workshop on Computing, Loutraki, LNCS 226 (1986), 180-193.

[LW 87a] Lengauer, T., Wagner, K.W., The correlation between the complexities of the non-hierarchical and hierarchical versions of graph problems, Proc. STACS 1987, LNCS 247 (1987), 100-113.

[LW 87b] Lengauer, T., Wagner, K.W., The binary network flow problem is logspace complete for P, TR 138, Institute of Mathematics, University of Augsburg, 1987.

[MVV 87] Mulmuley, K., Vazirani, U.V., Vazirani, V.V., Matching is as easy as matrix inversion, Proc. 19th ACM STOC (1987), 345-354.

[PY 86] Papadimitriou, Ch.H., Yannakakis, M., A note on succinct representations of graphs, Information and Control 71 (1986), 181-185.

[W 84] Wagner, K.W., The complexity of problems concerning graphs with regularities, Proc. of MFCS Conference 1984, LNCS 176 (1984), 544-552.

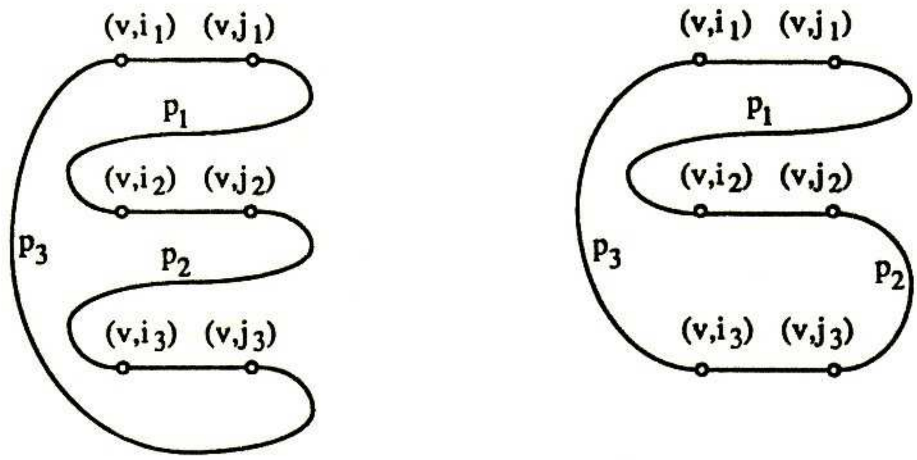

Figure 1. Case 1 Case 2

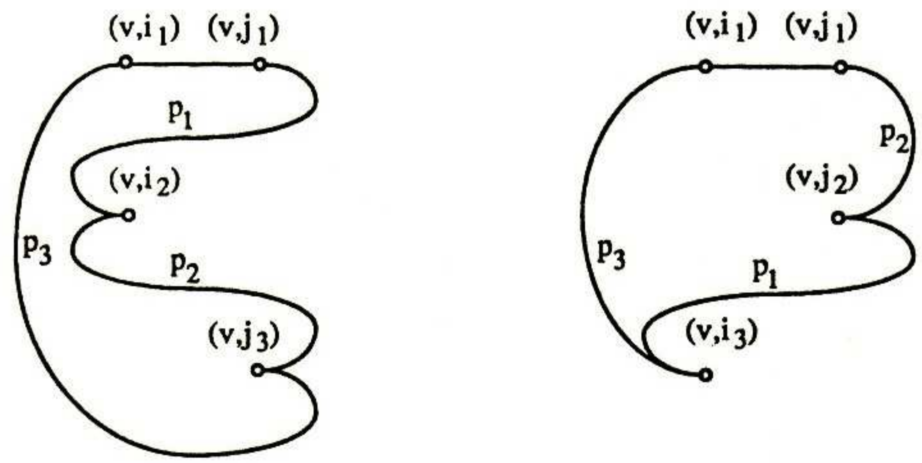

Figure 2. Case 1 Case 2

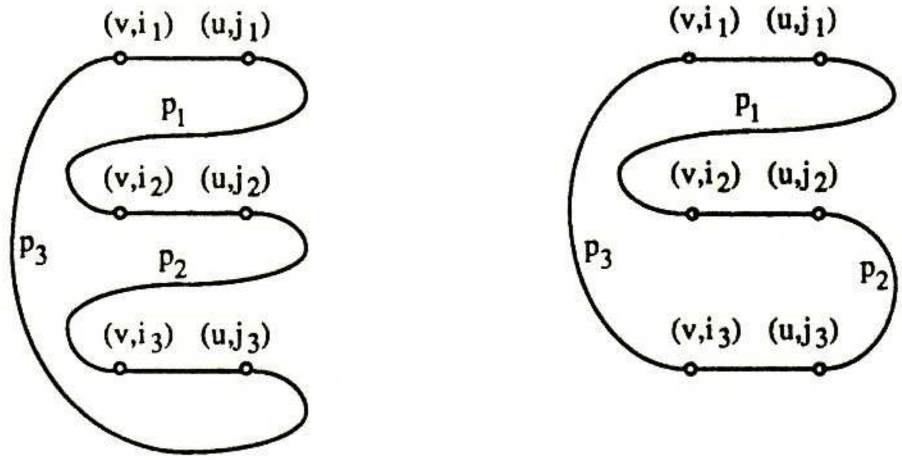

Figure 1. Case 1 Case 2

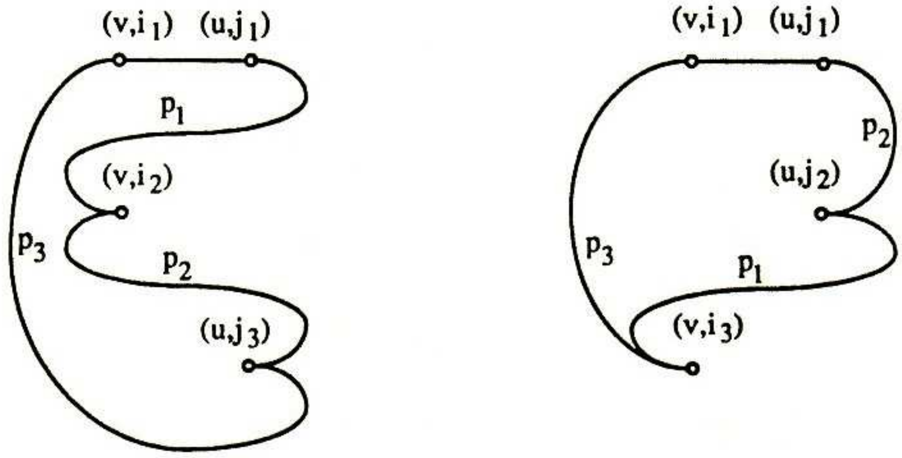

Figure 2. Case 1 Case 2