Decomposable trees : a polynomial algorithm for Tripodes*

D. Barth†, O. Baudon‡, J. Puech§

† PRiSM, Université de Versailles-S^t Quentin en Yv., 45 Bld des Etats Unis, F-78000 VERSAILLES
‡ LaBRI - UMR 5800, Université Bordeaux I, 350 Cours de la Libération, F-33405 TALENCE
§ Département de Mathématique, Bât. 425, Centre d'Orsay, Université Paris-Sud, F-91405 ORSAY CEDEX

email : barth@prism.uvsq.fr, puech@lri.fr, baudon@labri.u-bordeaux.fr

Abstract

In this article, we deal with graphs modelling interconnection networks of parallel systems (parallel computers, networks of workstations,..). We want to share the nodes of such a network between many users, each one needing a given number of nodes. Thus, a graph G with N vertices is said to be *decomposable* if for each set $\{n_1, \ldots, n_k\}$ which sum is equal to N, there exists a partition $V_1, \ldots, V_k$ of $V(G)$ such that for each i, $1 \le i \le k$, $|V_i| = n_i$ and the subgraph induced by V_i is connected.
We show that determining if a given tripode (three disjoint chains connected by one extremity to a same new vertex) is decomposable can be done by a polynomial algorithm.

Keywords : parallel systems, network sharing, graph decomposition, tripodes, integer decomposition, polynomial algorithm.

1 Introduction

In this paper, we deal with graphs modeling parallel systems (parallel computers, networks of workstations,..), considered as networks connecting different computing resources [7, 10, 13]. Such a parallel system could be simultaneously used by different distributed applications [1, 8]. In this case, a subpart of the network (in term of nodes) has to be efficiently assigned to each application, that is without as far as possible overlaping communications of two different applications. We here focus on the capacity of efficiently share the resources of a parallel system between many users (see [2, 3, 6] for different technics and applications about networks sharing). Let G be a graph of order N (that is with N vertices) modelling a parallel system. Consider k different users such that the i^{th} one needs n_i resources in G with $\sum_{1\le i\le k} n_i \le N$. The aim is then to attribute the good number of resources to each user such that

*This work was mainly realised when the two first authors work in the LRI, Université Paris-Sud.

(i) the subgraph of G induced by the set of resources attributed to each user is connected,
(ii) a resource is attributed to at most one user.

Conditions (i) and (ii) guaranty that the communications generated by the application of one user will not disturb the other applications, and that each user works on its own local network. This resources attribution can always be realised in a graph G containing an hamiltonian path (that is a traceable graph). Note that, by just knowing that G is a traceable graph, it is still a difficult problem to design an algorithm performing such an attribution.

More generally, the problem of knowing whether a given resources attribution is always possible, has been recently shown to be NP-complete [12] (see [9] for definitions about NP-completeness). We does not know if this problem is NP-complete in case of trees. The interest of studying trees is also motivated by the fact that if a spanning tree of a graph is decomposable, then so is this graph. In our article, as a first study on this problem, we consider the particular case of *Tripodes*, that is trees made of three disjoint chains connected by one extremity to a same new vertex called *root*. The problem of knowing whether a tripode is decomposable is equivalent to a number theory problem (see Problem 2) which seems to be difficult at first sight. From the graph theory point of view, this problem can also be seen as a generalisation of the problem of knowing if a graph contains a perfect matching if $|V(G)|$ is even (that is if $\{2, 2, \ldots, 2\}$ configures G) or a quasi-perfect matching if $|V(G)|$ is odd (that is if $\{2, \ldots, 2, 1\}$ configures G) [11]. Thus, all necessary conditions for a graph to contain a perfect matching (or a quasi-perfect matching) is a necessary condition for a graph to be decomposable.

The article is organised as follows. In the rest of this section, we give definitions and we present the problems we deal with. Section 2 presents the main result, that is a non trivial necessary and sufficient condition for a tripode to be decomposable, and we show that this condition can be checked in polynomial time.

1.1 Definitions

We consider classical graphs notations and definitions (see [5]).

Definition 1 *Let G be a graph with n vertices. Consider $d = \{n_1, n_2, \ldots, n_k\}$ a set of integers such that $n_1 \geq n_2 \ldots \geq n_k$ and $\sum_{i=1}^{k} n_i = n$; such a set d is called a decomposition of n. We denote by max_d the greatest integer in d and by $\mathcal{D}(n)$ the set of all the decompositions of n.*

Definition 2 *We say that a decomposition $d = \{n_1, n_2, \ldots, n_k\} \in \mathcal{D}(n)$* configures *$G$ if there exists a set partition $V_1, V_2, \ldots, V_k$ of $V(G)$, called (G, d)-partition, such that for each i, $1 \leq i \leq k$,*
- *$|V_i| = n_i$,*
- *the subgraph of G induced by V_i is connected.*

We say that V_i is the n_i-set of the (G, d)-partition, and that l-sets are all the sets V_i of same cardinality $l \in d$.

An example is given on a tree $\mathcal{T}$ in Figure 1. We now define what we call a decomposable graph.

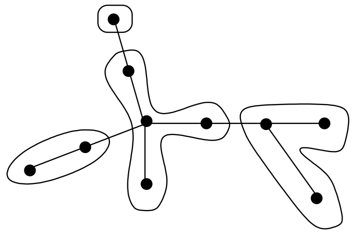

Figure 1: A tree $\mathcal{T}$ with a set partition from a decomposition $\{4, 3, 2, 1\}$.

Definition 3 *A graph G is said* decomposable *if each decomposition of $\mathcal{D}(n)$ configures G.*

In this article, we especialy focus on *k-podes* $T(t_1, t_2, ..., t_k)$, with $k > 2$ and $t_1, t_2, \ldots, t_k$ being k strictly positive integers such that $t_1 \geq t_2 \geq \ldots \geq t_k$. The k-podes $T(t_1, t_2, ..., t_k)$ is constructed by connecting one extremity of each of k chains, respectively of length $t_1 - 1, \ldots, t_k - 1$ (the chain of length t_i in the k-podes is denoted by T_i), to a new vertex called *root* (see Figure 4(a)). The order of this graph is then $n = t_1 + \ldots + t_k + 1$.

1.2 Two problems

Problem 1 : decomposable graphs.
Given : a graph G.
Question : Is G a decomposable graph ?

Recently, it has been shown that, given a graph G and a decomposition $d \in \mathcal{D}(|V(G)|)$, the problem of knowing if d configures G is NP-complete (see [12]). It is still an open question to know if Problem 1 for any graph G can be solved in a polynomial time, function of $|V(G)|$.

If G is decomposable, then for any $d \in \mathcal{D}(|V(G)|)$, there is a covering subtree T of G such that d configures T. Then, any graph with a decomposable spanning tree is decomposable. Thus, as a first study on Problem 1, we focus here on trees. Let T be a tree with n vertices, x be a vertex of degree $k > 2$ in T and $C_x = \{t_1, t_2, \ldots, t_k\}$ be the set of the number of vertices of all the connected subtrees obtained by deleting x in T. Consider $t_1 \geq t_2 \ldots \geq t_k$. It is easy to see that, if T is decomposable, then for any vertex x of degree $k > 2$ from T, with $C_x = \{t_1, \ldots, t_k\}$, the k-podes $T(t_1, \ldots, t_k)$ is decomposable. Unfortunately, this is not a necessary and sufficient condition, as it is shown in Figure 2. From this relation between

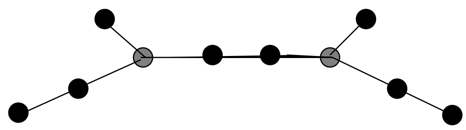

Figure 2: A no-decomposable tree with 11 vertices, knowing that the 3-podes $T(6, 3, 1)$ is decomposable.

trees and k-podes, we have obtained in [4] different necessary conditions for a tree to be decomposable. For any vertex x in T with $C_x = \{t_1, \ldots t_k\}$, we have shown for example that

(a.) $t_1 \geq \frac{n-1}{2}$ and for any i, $1 \leq i \leq k-1$, we have $t_i \geq \sum_{j=i+1}^{k} t_j$,

(b.) let $c_o = |\{t \in C_x : \ t \text{ is odd}\}|$; if n is even then $c_o = 1$ else $c_o \in \{0, 2\}$,

(c.) $\Delta(T) \leq log_2(n-1) \ +1$ and for any $n \geq 5$ and $n \neq 6$, there exists a decomposable tree with n vertices different from a chain.

Every chain is clearly decomposable. The first question is to know if the proplem is also easy to solve in trees having exactly one vertex with degree more than 2. It is why in this paper, we mainly deal with 3-podes, also called *tripodes*. Note that by using Lemma 1 and Definition 4 given later, Problem 1 for Tripode $T(t_1, t_2, t_3)$ is equivalent to this new problem :

Problem 2 : 3-partition of n.
Given : three integers t_1, t_2 and t_3 with $t_1 \geq t_2 \geq t_3 > 0$.
Question : For all decompositions d of $n = t_1 + t_2 + t_3 + 1$, does there exist a partition d_1, d_2, d_3 of $d - \{n_1\}$ such that $\sum_{v \in d_1} v = t_1 - a$, $\sum_{v \in d_2} v = t_2 - b$ and $\sum_{v \in d_3} v = t_3 - c$ where a, b and c are positive integers with $a + b + c = n_1 - 1$?

In this article, we show that Problem 2 (i.e., Problem 1 for tripodes) can be solved in polynomial time, function of $n = t_1 + t_2 + t_3 + 1$.

2 Decomposable tripodes

Theorem 1 *For any tripode $T(t_1, t_2, t_3)$, Problem 1 can be solved in polynomial time function of $n = t_1 + t_2 + t_3 + 1$.*

This theorem is a direct consequence of the two next propositions. Proposition 1 gives a necessary and sufficient condition for a tripode to be decomposable. We show that all the decompostions of n configure $\mathcal{T} = T(t_1, t_2, t_3)$ iff this is true for two subsets of decompositions called q-decompositions and $\overline{q}$-decompositions. In fact, we show that if there exists a decomposition a that does not configure $\mathcal{T}$, then there exists a q-decomposition or a $\overline{q}$-decomposition c that does not configure $\mathcal{T}$ (see Figure 3). Proposition 2 shows that knowing

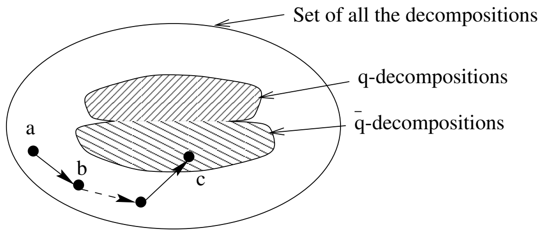

Figure 3: The decompositions set with a transformation from decomposition a to decomposition c.

if all the q-decompositions and the $\overline{q}$-decompositions configure $\mathcal{T}$ can be verified by using a polynomial algorithm.

2.1 A necessary and sufficient condition

We first define two kinds of decompositions of an integer n and then give our first main result on tripodes.

Definition 4 *Let n and q be two integers, with $n > q > 0$. We call a q-*decomposition *of n a decomposition made of u occurences of q and v occurences of $q+1$, where u and v are two integers such that $n = uq + v(q+1)$ and $v \neq 0$. We denote by $(u,v)_q$ this q-decomposition.*

Definition 5 *Let n and q be two integers, with $n > q > 0$. We call a $\overline{q}$-*decomposition of n *a decomposition made of u occurences of q, v occurences of $q+1$ and an integer m, where*
- *$n = uq + v(q+1) + m$,*
- *$v \neq 0$ and $1 \leq m \leq q-1$.*

We denote by $\overline{(u,v)}_q$ this $\overline{q}$-decomposition.

For instance, consider $n = 18$. Then $(2,2)_4 = \{4,4,5,5\}$ and $\overline{(1,1)}_7 = \{7,8,3\}$.

Proposition 1 *The tripode $\mathcal{T} = T(t_1,t_2,t_3)$ of order n is decomposable, if and only if, all q-decompositions of n, with $q \leq t_2 + t_3$, and all $\overline{q'}$-decompositions of n, with $q' \leq t_3 - 2$, are configuring $\mathcal{T}$.*

We now establish a very useful result that concerns the position of the max_d-set of any decomposition d of n on any decomposable tripode.

Definition 6 *Let $d \in \mathcal{D}(n)$ be a decomposition configuring a tripode $\mathcal{T}$. A $(\mathcal{T},d)$-partition p is said to be* good *if the root of $\mathcal{T}$ is in a max_d-set V of p. In this case we denote by α_j the cardinality of $V \cap T_j$, with $j = 1,2,3$.*

Lemma 1 *If $d \in \mathcal{D}(n)$ is a decomposition configuring a tripode $\mathcal{T}$ of order n, then there exists a good $(\mathcal{T},d)$-partition (note that this result is also true for k-podes).*

Proof. Consider a $(\mathcal{T},d)$-partition p. Without loss of generality, we can suppose that a max_d-set V_i lies on the path T_1 of length $t_1 - 1$ in $\mathcal{T}$. Let V_j be the n_j-set containing the root of $\mathcal{T}$. Since it is always possible to move the sets of p that lie on the same path, we can consider that V_i is just near V_j (that is a vertex of V_i is adjacent to a vertex of V_j) on T_1. We give a sketch of a construction of a new $(\mathcal{T},d)$-partition p' from p (see Figure 4); it consists in
- adding vertices from T_1 to V_j such that its cardinality becomes n_1 and the subgraph induced by V_j is still connected,
- extracting vertices from V_i such that its cardinality becomes n_j,

It is easy to see that p' is then a good $(\mathcal{T},d)$-partition. □

We want to characterize the fact that a decomposition d of n configures a tripode, and in order to do that, we need the following definition.

Definition 7 *Let $T(t_1,t_2,t_3)$ be a tripode of order n and d a decomposition of n. Let Σ be the sum of a subset of $d - \{n_1\}$. We say that Σ satisfies the Property $(\mathcal{H})$ on d if $1 + t_2 + t_3 - n_1 \leq \Sigma \leq t_2$.*

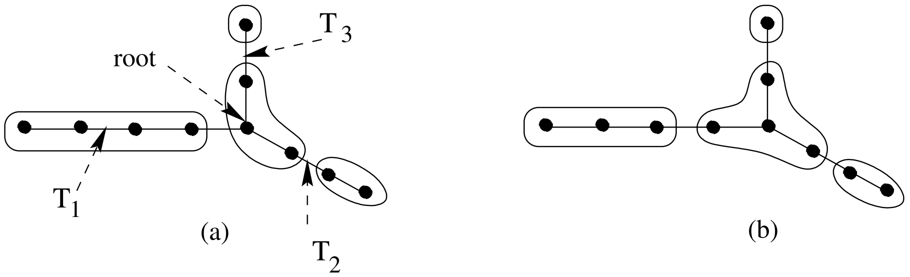

Figure 4: A $(T(4,3,2),\{4,3,2,1\})$-partition (see (a)) and a good $(T(4,3,2),\{4,3,2,1\})$-partition (see (b)) .

Lemma 2 *Let $d = \{n_1, n_2, \ldots, n_k\}$ be a decomposition of $\mathcal{D}(n)$ such that $n_1 \geq n_2 \ldots \geq n_k$ and let $\mathcal{T} = T(t_1, t_2, t_3)$ be a tripode such that $n_k \geq t_3 + 1$. Then, the decomposition d configures $\mathcal{T}$, if and only if, there exists a subset S of $d - \{n_1\}$ such that the sum of S satisfies the Property $(\mathcal{H})$ on d.*

Proof. Since we have $n_i \geq t_3 + 1$ for every $i \in \{1, 2, \ldots, k\}$, observe that every set of a $(\mathcal{T}, d)$-partition either contains the path T_3 and the root of $\mathcal{T}$ or intersects none of them.
• If part: Suppose that there exists a subset S of $d - \{n_1\}$ such that the sum Σ of S satisfies $1 + t_2 + t_3 - n_1 \leq \Sigma \leq t_2$ (note that S may be an empty set). Hence $0 \leq t_2 - \Sigma \leq n_1 - t_3 - 1$. We can now construct a good $(\mathcal{T}, d)$-partition in the following way. Every a-set where $a \in S$ lies on T_2, the max_d-set contains the rest of T_2, the root and the path T_3 (we can do that since $0 \leq t_2 - \Sigma \leq n_1 - t_3 - 1$). Then, it is easy to decide where the remaining a-sets lie on the remaining subpath of T_1. Thus the decomposition d configures $\mathcal{T}$.
• Only if part: Suppose that the decomposition d configures $\mathcal{T}$ with a good $(\mathcal{T}, d)$-partition. In particular the max_d-set V contains the root of $\mathcal{T}$ and therefore it also contains the path T_3. Since the decomposition d configures $\mathcal{T}$, there exists a subset S of $d - \{n_1\}$ such that the sum Σ of S satisfies $t_2 = \Sigma + \alpha_2$ where $\alpha_2 = |V \cap T_2|$. Since $0 \leq \alpha_2 \leq n_1 - t_3 - 1$, we have $1 + t_2 + t_3 - n_1 \leq \Sigma \leq t_2$ and therefore Σ satisfies the Property $(\mathcal{H})$ on d. □

To prove Proposition 1, we also need the next lemma and the next definition.

Lemma 3 *Let $d = \{n_1, n_2, \ldots, n_k\}$ be a decomposition of n such that d is non-configuring a tripode $\mathcal{T} = T(t_1, t_2, t_3)$ of order n. Every partition $\{A_1, A_2, \ldots, A_l\}$ of d induces the decomposition $d' = \{m_1, m_2, \ldots, m_l\}$ of n where $M_i = \sum_{a \in A_i} a$. Then the decomposition d' does not configure $\mathcal{T}$ if one of the two following conditions holds :*

a. *$n_1 = max_{d'}$.*

b. *$M_i \geq t_3 + 1$ and there exists $a_i \in A_i$ such that $max_i - a_i \leq t_3$ for every $i \in \{1, 2, \ldots, l\}$.*

Proof. Suppose that $d' = \{m_1, m_2, \ldots, m_l\}$ is a decomposition configuring $\mathcal{T}$. By Lemma 1, there exists a good $(\mathcal{T}, d')$-partition. Let i_0 be the integer such that the m_{i_0}-set contains the root of $\mathcal{T}$ (in particular $max_{d'} = m_{i_0}$). Observe that if case a holds, then $m_{i_0} = max_{d'} = n_1$, and without loss of generality we can suppose that $A_{i_0} = \{n_1\}$ (indeed possibly $|A_{i_0}| \geq 2$, but in this case there exists an integer i_1 such that $A_{i_1} = \{n_1\}$ for otherwise $max_{d'} > n_1$). If $|A_{i_0}| = 1$ then $A_{i_0} = \{n_1\}$ (in particular this is the case when case a holds). If $|A_{i_0}| \geq 2$, since case b holds, the m_{i_0}-set V_{i_0} of d' contains the path T_3 and the root of $\mathcal{T}$. Moreover, we can split the set V_{i_0} such that the a_{i_0}-set of d lies on the part of V_{i_0} that contains the root of $\mathcal{T}$ and the two sets $V_{i_0} \cap T_1$ and $V_{i_0} \cap T_2$. In either case, we obtain a $(\mathcal{T}, d)$-partition by splitting every m_i-set of d' in a-sets of d with $a \in A_i$. $\square$

Definition 8 *We denote by $S_{i,j}$ the bijection on $\mathcal{D}(n)$ defined as follows : given a decomposition $d = \{n_1, n_2, \ldots, n_k\}$ of $\mathcal{D}(n)$, $S_{i,j}(d)$ is the decomposition of $\mathcal{D}(n)$ defined by $S_{i,j}(d) = \{n_l | l \neq i, j\}_{1 \leq l \leq k} \cup \{n_i - 1, n_j + 1\}$.*

Proof of Proposition 1. Suppose that $\mathcal{T} = T(t_1, t_2, t_3)$ is not decomposable, and let $d = \{n_1, n_2, \ldots, n_k\}$ be a decomposition of $\mathcal{D}(n)$ which does not configure $\mathcal{T}$. Recall that $n_1 \geq n_2 \ldots \geq n_k$. We will show that there exists a q-decomposition of n or a $\overline{q'}$-decomposition of n with $q' \leq t_3 - 2$, which does not configure $\mathcal{T}$. If $n_1 = t_3$ then d clearly configures $\mathcal{T}$, a contradiction. Therefore we will consider the following two cases.

a. Suppose that $n_1 \leq t_3 - 1$.

Claim 1: Without loss of generality, $i \neq j$ implies $n_i + n_j > n_1$.

• Otherwise, since $n_i + n_j \leq n_1$, we can apply Lemma 3.a to the decomposition $\{n_i + n_j\} \cup \{n_l \mid l \neq i, j\}$ of n. We repeat this operation untill Claim 1 holds.•

Claim 2: If $n_{k-1} \leq n_1 - 2$, then the decomposition $S_{1,k-1}(d)$ does not configure $\mathcal{T}$.

• Suppose that $n_{k-1} \leq n_1 - 2$ and that $S_{1,k-1}(d)$ configures $\mathcal{T}$. By Lemma 1, we can consider a good $(\mathcal{T}, S_{1,k-1}(d))$-partition p. The same T_j does not contain both (n_1-1)-sets and $(n_{k-1}+1)$-sets, for otherwise p gives us a good $(\mathcal{T}, d)$-partition by transfering one vertex from a $(n_{k-1}+1)$-set to a (n_1-1)-set. Moreover, if $n_1 > n_2$, then $(n_1 - 1) = max_{S_{1,k-1}(d)}$ and by Lemma 1, it is easy to obtain from p a good $(\mathcal{T}, d)$-partition of d, a contradiction. So, we have $n_1 = n_2$.
Thus, $max_{S_{1,k-1}(d)} = n_1$ and therefore we can assume that there exists $(i_1, i_2, i_3) \in \mathcal{S}_3$ (the set of the bijective functions on $\{1, 2, 3\}$) such that T_{i_1} contains a $(n_{k-1}+1)$-set and T_{i_2} contains a (n_1-1)-set. We have $\alpha_{i_2} = 0$ (see Definition 6), for otherwise we can transfer one vertex from the $max_{S_{1,k-1}(d)}$-set to a (n_1-1)-set of T_{i_2}. We now prove that $\alpha_{i_1} \leq n_1 - n_{k-1} - 2$, that is $\alpha_{i_3} \geq n_{k-1} + 1$ (indeed, observe that $max_{S_{1,k-1}(d)} = n_1 = \alpha_{i_3} + \alpha_{i_1} + 1$). As a matter of fact, suppose to the contrary that $\alpha_{i_1} > n_1 - n_{k-1} - 2 = (n_1 - 1) - (n_{k-1} + 1)$. Then, since by hypothesis $n_{k-1} + 1 \leq n_1 - 1$, we can exchange a $(n_{k-1}+1)$-set on T_{i_1} with a $(n_1 - 1)$-set on T_{i_2}. Since after this exchange $\alpha_{i_1} \geq 1$ we can transfer one vertex from the $max_{S_{1,k-1}(d)}$-set to the exchanged (n_1-1)-set, which contradicts that the decomposition d does not configure $\mathcal{T}$. We finally prove that only max_d-sets lie on T_{i_3}. Indeed, suppose that there is a n_j-set on T_{i_3} such that $n_j \leq n_1 - 1$. Since $n_j + \alpha_{i_3} \geq n_k + n_{k-1} + 1 > n_1$ (see Claim 1) we can exchange a n_j-set on T_{i_3} with a (n_1-1)-set on T_{i_2} that can be extended to a max_d-set of a $(\mathcal{T}, d)$-partition, a contradiction.
Thus the n_k-set lies on T_{i_1} or on T_{i_2}. Since $\alpha_{i_3} \geq n_{k-1} + 1 > n_k$, we can suppose that the n_k-set is lying on $V(T_{i_3})$, which gives a contradiction with the fact that only max_d-sets lie on T_{i_3}.•

Apply $S_{1,k-1}$ to the decomposition d until we get either a q-decomposition of n or a $\overline{q}$-decomposition of n with $q \leq t_3 - 2$ and which does not configure $\mathcal{T}$.

b. Suppose that $n_1 \geq t_3 + 1$.

Claim 3: Without loss of generality, $n_k \geq t_3 + 1$.

• Otherwise consider the following construction of sets of integers $A_1, \ldots, A_\alpha$. Let $\epsilon_1 = \{n_k, n_{k-1}, \ldots, n_1\}$ be an ordered set of integers, where $n_k \leq n_{k-1} \leq \ldots \leq n_1$, and i be an integer.

Begin the construction with $i = 1$
While the smallest element e_1 *of the ordered set* $\epsilon_i = \{e_1, e_2, \ldots, e_m\}$, *where* $e_1 \leq e_2 \leq \ldots \leq e_m$, *is such that* $e_1 \leq t_3$ *do*

◇ *Let* l *be the integer such that* $e_1+e_2+\ldots+e_l \leq t_3$ *and* $e_1+e_2+\ldots+e_l+e_{l+1} \geq t_3+1$ *(since* $n_1 \geq t_3+1$, *the integer* l *exists).*
◇ $A_i = \{e_1, \ldots, e_{l+1}\}$
◇ ϵ_{i+1} *is the ordered set obtain from* $\epsilon_i - A_i$
◇ $l_i = l$
◇ $i = i + 1$

After this construction, we obtain disjoint subsets $A_1, \ldots, A_\alpha$ of $\{n_k, n_{k-1}, \ldots, n_1\}$ and an ordered set $\epsilon_{\alpha+1} = \{n_r, \ldots, n_1\}$. Then, for any j, $\alpha + 1 \leq j \leq \alpha + r$, we define $A_j = \{n_{r+\alpha+1-j}\}$. If $n_2 > t_3$ then $m_1 = |A_1| = n_1$. Else, by construction, $m_i \geq t_3 + 1$ for any $i \in \{1, 2, ..., \alpha + r\}$ and $m_i - n_{l_i} \leq t_3$. Thus, by Lemma 3, the associated decomposition d' does not configure $\mathcal{T}$. •

Claim 4: If $n_2 > n_k + 1$, then the decomposition $S_{2,k}(d)$ does not configure $\mathcal{T}$.

• Since $n_1 \geq n_k + 1$, we have $max_d = max_{S_{2,k}(d)} = n_1$. Let Σ' be the sum of a subset of $S_{2,k}(d) \setminus \{n_1\}$. We consider three cases.
First suppose that Σ' contains both or none of $n_2 - 1$ and $n_k + 1$. Then Σ' can be seen as the sum of a subset of $d \setminus \{n_1\}$, Therefore the sum Σ' does not satisfy Property $(\mathcal{H})$ on $S_{2,k}(d)$ (see Definition 7), since d does not configure $\mathcal{T}$ and by Lemma 2.
Now suppose that Σ' contains $n_k + 1$ but not $n_2 - 1$. Then $\Sigma = \Sigma' - 1$ can be seen as the sum of a subset of $d \setminus \{n_1\}$ and therefore does not satisfy the Property $(\mathcal{H})$ on d. Hence Σ' satisfies Property $(\mathcal{H})$ on d, if and only if, $\Sigma' = 1 + t_2 + t_3 - n_1$. Then $\Sigma = \Sigma' + n_2 - (n_k + 1) = t_2 + t_3 + n_2 - n_1 - n_k$ can be seen as the sum of a subset of $d \setminus \{n_1\}$. Since $n_2 > n_k + 1$, we have $\Sigma > \Sigma' = 1 + t_2 + t_3 - n_1$. Hence $\Sigma > t_2$, and so $n_2 - n_1 > n_k - t_3$. Since $n_1 \geq n_2$, we obtain that $n_k < t_3$, a contradiction.
Finally suppose that Σ' contains $n_2 - 1$ but not $n_k + 1$. Then $\Sigma = \Sigma' + 1$ can be seen as the sum of a subset of $d \setminus \{n_1\}$ and therefore does not satisfy the Property $(\mathcal{H})$ on d. Hence Σ' satisfies the Property $(\mathcal{H})$ on d, if and only if, $\Sigma' = t_2$. Then $\Sigma = \Sigma' + n_k - (n_2 - 1) = t_2 + n_k - n_2 + 1$ can be seen as the sum of a subset of $d \setminus \{n_1\}$. Since $n_2 - 1 > n_k$, we have $\Sigma < \Sigma' = t_2$. Hence $\Sigma < 1 + t_2 + t_3 - n_1$. Therefore $n_2 - n_1 > n_k - t_3$ and we get the same contradiction as above. In either case, we conclude that Σ' does not satisfy Property $(\mathcal{H})$ on $S_{2,k}(d)$, and therefore $S_{2,k}(d)$ does not configure $\mathcal{T}$.•

Apply $S_{2,k}$ to d as many times the hypothesis of Claim 4 ($n_2 > n_k + 1$) holds. At the end, we obtain a decomposition $d' = \{m_1, m_2, \ldots, m_k\}$ of $\mathcal{D}(n)$, where $m_1 \geq m_2 \ldots \geq m_k$, which does not configure $\mathcal{T}$ and which is made of u occurences of an integer q, v occurences of

$q+1$, with $n = uq + v(q+1) + m_1$.

Claim 5: If $m_1 \geq m_k + 2$, then the decomposition $S_{1,k}(d')$ does not configure $\mathcal{T}$.

• Observe that $max_{S_{1,k}(d')} = m_1 - 1$. Let Σ' be the sum of a subset of $S_{1,k}(d') \setminus \{m_1 - 1\}$. If Σ' does not contain $m_k + 1$, then Σ' can be seen as the sum of a subset of $d \setminus \{m_1\}$, and therefore Σ' does not satisfy the Property $(\mathcal{H})$ on $S_{1,k}(d')$. Otherwise Σ' contains $max_k + 1$. Then $\Sigma = \Sigma' - 1$ can be seen as the sum of a subset of $d \setminus \{max_1\}$ and therefore does not satisfy the Property $(\mathcal{H})$ on d. That is either $\Sigma > t_2$ or $\Sigma < 1 + t_2 + t_3 - m_1$. Since $\Sigma = \Sigma' - 1$, we obtain that either $\Sigma' > t_2 + 1 > t_2$ or $\Sigma' < 1 + t_2 + t_3 - (m_1 - 1)$. Therefore Σ' still does not satisfy the Property $(\mathcal{H})$ on $S_{1,k}(d')$, and by Lemma 2, we know that $S_{1,k}(d')$ does not configure $\mathcal{T}$. •

Apply $S_{1,k}$ to d' until we get a q-decomposition of n that does not configure $\mathcal{T}$. □

2.2 A complexity result

Proposition 1 shows that to know if a tripode $\mathcal{T}$ is decomposable, it is necessary and sufficient to check if each q-decomposition and each $\overline{q}$-decomposition configures $\mathcal{T}$. Therefore, to prove Theorem 1, we just have to prove the following.

Proposition 2 *For any tripode $\mathcal{T} = T(t_1, t_2, t_3)$, with $n = t_1 + t_2 + t_3 + 1$ vertices, knowing if all q-decompositions of n, with $q \leq t_2 + t_3$,and all $\overline{q'}$-decompositions of n, with $q' \leq t_3 - 2$, are configuring $\mathcal{T}$ can be realised in a polynomial time function of n.*

We first see how to check that a q-decomposition configures a tripode $T(t_1, t_2, t_3)$.

Definition 9 *A (q, u, v)-sum is an integer S such that there exist two integers a and b such that $S = aq + b(q+1)$ with $a \leq u$ and $b \leq v$.*
Let $\phi(S)$ be the integer a and $\psi(S)$ be the integer b.

Lemma 4 *Let $d = (u, v)_q$ be a q-decomposition of n.*
Then, d configures $T(t_1, t_2, t_3)$ if and only if there exists a $(q, u, v-1)$-sum R such that $t_3 - q \leq R \leq t_3$ and a $(q, u - \phi(R), v - \psi(R) - 1)$-sum S such that $t_2 + t_3 - q - R \leq S \leq t_2$.

Proof. The decomposition d configures $T(t_1, t_2, t_3)$ with a good $(\mathcal{T}, d)$-partition, if and

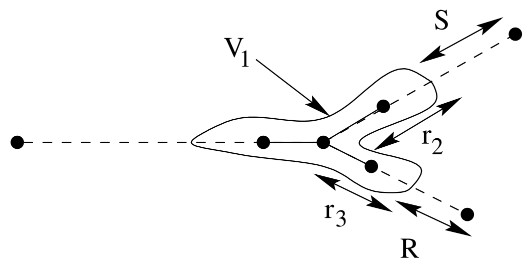

Figure 5: A d-partition of $T(t_1, t_2, t_3)$.

only if, considering T_3, there exists a sum R among the set of integers $d - \{q+1\}$ such that

$t_3 = R{+}\alpha_3$ where $\alpha_3 \leq q$, and then considering T_2 there exists a sum S among the set of integers $d - \{q{+}1$,the integers of $R\}$ such that $t_2 = S{+}\alpha_2$ where $\alpha_2 \leq q{+}R{-}t_3$ (see Figure 5). $\square$

Proof of Proposition 2. Consider a tripode $\mathcal{T} = T(t_1, t_2, t_3)$ of order $n = t_1 + t_2 + t_3 + 1$ and observe that a $\overline{q}$-decomposition $\overline{(u, v)_q}$ of n configures $T(t_1, t_2, t_3)$, if and only if, the q-decomposition $(u, v)_q$ of $n{-}m$ configures $T(t_1{-}m, t_2, t_3)$ or $T(t_1, t_2{-}m, t_3)$ or $T(t_1, t_2, t_3{-}m)$, where $m = n - (uq + v(q + 1))$. Therefore, it is sufficient to prove the following claim.

Claim : Knowing if all q-decompositions of n are configuring $\mathcal{T}$ can be realised in a polynomial time function of n.

• A q-decomposition is characterized by q, the number u of occurences of q and the number v of occurences of $q + 1$. Therefore the number of q-decomposition is at most $O(n^3)$. Moreover, by Lemma 4, knowing if a given q-decomposition of n is configuring $\mathcal{T}$ can be decided in at most $O(n^4)$ steps. Thus, the complexity of the problem given in the claim is at most in $O(n^7)$.•

This completes the proof of the proposition. $\square$

Note that in [4], we give a detailled polynomial algorithm to solve Problem 1 for tripodes, with some computational results (see Figure 6).

3 Conclusion

In this article, we deal with trees having only one vertex with degree more than 2. In [4], we give some preliminar results about k-podes with $k > 3$. A detailled polynomial algorithm to decide if such a given tree is decomposable is also given in [4]. In figure 5, we give all the decomposable tripodes of order n, with $n \leq 20$.
Knowing that Problem 1 is or not NP-complete is still an open problem. This can be considered in two ways :

• Does there exist a constant $c > 3$ such that Problem 1 can be poynimialy solved for any k-podes, with $k \leq c$?

• Does there exist a constant c such that this problem can be polynomialy solved for any tree (eventualy with a bounded degree $d \geq 3$) with less than c vertices and a degree greater than 2?

	t_1	t_2	t_3
$n=5$	2	1	1
$n=7$	3	2	1
	4	1	1
$n=8$	4	2	1
$n=9$	4	3	1
	6	1	1
$n=10$	6	2	1
$n=11$	5	3	2
	5	4	1
	6	3	1
	7	2	1
	8	1	1
$n=12$	6	4	1
$n=13$	6	4	2
	6	5	1
	7	4	1
	8	3	1
	9	2	1
	10	1	1
$n=14$	7	4	2
	8	4	1
	10	2	1
$n=15$	7	6	1
	10	3	1
	12	1	1
$n=16$	8	6	1
	10	4	1
	12	2	1

	t_1	t_2	t_3
$n=17$	8	5	3
	8	6	2
	8	7	1
	9	5	2
	9	6	1
	10	5	1
	11	4	1
	12	3	1
	13	2	1
	14	1	1
$n=18$	9	6	2
	10	6	1
	12	4	1
$n=19$	9	6	3
	9	7	2
	9	8	1
	10	7	1
	11	6	1
	12	4	2
	12	5	1
	13	4	1
	14	3	1
	15	2	1
	16	1	1
$n=20$	10	8	1
	12	6	1
	16	2	1

Figure 6 : Decomposable tripode $T(t_1, t_2, t_3)$ of order n, $1 \leq n \leq 20$.

References

[1] R.H. Arpaci, D.A. Patterson. *The interaction of parallel and sequential workloads on a network of workstations.* UC Berkeley Technical Report CS-94-830, 1994.

[2] B. Awerbuch, A. Goldberg, M. Luby, S. Plotkin. *Network decomposition and locality in distributed computing,* In Proc. 30th IEEE Symp. on Foundation On Computer Science, May 1989.

[3] B. Awerbuch, D. Peleg. *Sparse Partitions.* Proc. 31st IEEE Symp. on Found. of Comp. Science, pp 503-513, oct. 1990.

[4] D. Barth, O. Baudon, J. Puech, *Networks sharing : a polynomial algorithm for Tripodes,* internal report n. 1164, L.R.I, Université de Paris Sud, 1998.

[5] C. Berge. *Graphes et Hypergraphes,* (Gauthier-Villard, 1971).

[6] S.L. Bezrukov *On k-partitioning the n-cube,* Lect. Notes in Comp. Sci., vol. 1197, Springer Verlag, pp. 44-55, 1997.

[7] F. Cappello, D. Etiemble. *Communication in parallel architectures and networks of workstation : from standards to new standards.* Proc. of Parallel Computer Technologies, Lect. Notes in Comp. Science 1277, pp 298-310, 1997.

[8] P. Deliste, J. Wang, X. Zheng. *Utopia: A load sharing facility for large, heterogeneous distributed computing system*, Technical Report CSR1-257, University of Toronto, 1992.

[9] M.R. Garey, D.S. Johnson. *Computers and intractability, a guide to the theory of NP-completeness.* (Freeman and Co. Ed.) , 1979.

[10] K. Hwang. *Advanced computer architecture*, (Mc Graw Hill Int. Ed.), 1993.

[11] L. Lovašz, M. Plummer. *Matchings theory*, (NorthHolland Ed.), 1987.

[12] M. Robson. *Private communication* (preprint), Université Bordeaux I, 1998.

[13] J. de Rumeur. *Communications dans les réseaux d'interconnexion* (Ed. Masson, 1994).