Recognizing Recursive Circulant Graphs $G(cd^m, d)$

Guillaume Fertin, André Raspaud

LaBRI U.M.R. 5800, Université Bordeaux I
351 Cours de la Libération, F33405 Talence Cedex

{fertin,raspaud}@labri.u-bordeaux.fr

Abstract

Recursive circulant graphs $G(N, d)$ have been introduced in 1994 by Park and Chwa [PC94] as a new topology for interconnection networks. Recursive circulant graphs $G(N, d)$ are circulant graphs with N nodes and with jumps of powers of d. A subfamily of recursive circulant graphs (more precisely, $G(2^k, 4)$) is of same order and degree than the hypercube of dimension k, with sometimes better parameters, such as diameter [PC94, GMR98]. Embeddings among recursive circulant graphs, hypercubes and Knödel graphs of order 2^k have also been studied in [PC, FR98b]. Here, following a question raised in [CFG99], we give, thanks to a sharp structural analysis of such graphs, an $O(cd^{m+2} \cdot (2m)^d)$ algorithm to determine if a given graph is a recursive circulant graph of the form $G(cd^m, d)$, for any $d \geq 3$, except in the case where c is even while d is odd. Applying this algorithm to recursive circulant graphs $G(2^k, 4)$ gives us an $O(2^k \cdot k^4)$ recognition algorithm for such graphs.

1 Introduction

Recursive circulant graphs $G(N, d)$ have been introduced in 1994 by Park and Chwa [PC94]. Since that time, recursive circulant graphs have been widely studied [Mic96, PC, GMR98, FR98a, Fer99], and some more properties have been shown (Hamiltonian decomposition, edge-forwarding index, embeddings, etc.). More precisely, the subfamily of recursive circulant graphs of the form $G(2^k, 4)$ was presented by Park and Chwa as a new topology for multicomputer networks, because to its nice properties concerning their diameter, routing schemes, mean internode distance, etc. For instance, it has the same order and degree as the hypercube of dimension k, H_k, but has a smaller diameter : $\lceil \frac{3k-1}{4} \rceil$ compared to k for the hypercube.

Below, we give a general definition of a Circulant Graph $C_N(a_1, a_2, \ldots a_m)$ and of a Recursive Circulant Graph $G(N, d)$.

Definition 1 (Circulant Graph) *A* circulant graph *on N vertices $C_N(a_1, a_2, \ldots a_m)$, with $a_i \in \mathbb{N}^*$ and $a_1 < a_2 < \ldots < a_m < N$, has vertex set $V = \{0, 1, \ldots, N-1\}$ and edge set $E = \{(u, v) \mid \exists\, a_i,\ 1 \leq i \leq m$ such that $u + a_i \equiv v \pmod N\}$.*

Definition 2 (Recursive Circulant Graph $G(N, d)$ [PC94]) *The* recursive circulant graph *of order N is denoted $G(N, d)$. It has vertex set $V = \{0, 1, 2, \ldots, N-1\}$ and edge set $E = \{(v, w) \mid \exists\, i,\ 0 \leq i \leq \lceil \log_d N \rceil - 1$, such that $v + d^i \equiv w \bmod N\}$.*

We note that $G(N, d)$ is isomorphic to the circulant graph $C_N(1, d, d^2, d^3, \ldots, d^{\lceil \log_d N \rceil - 1})$. Note also that, in $G(N, d)$, any edge that connects a vertex v to a vertex $v \pm d^i \bmod N$, with $0 \leq i \leq \lceil \log_d N \rceil - 1$, is said to be *in dimension i*. Figure 1 shows two examples of recursive circulant graphs.

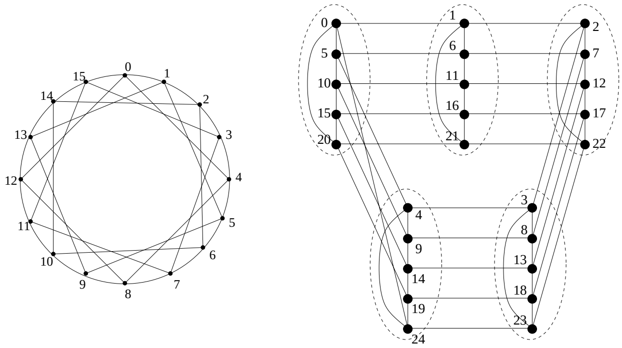

Figure 1: $G(16, 4)$ (left) and $G(25, 5)$ (right)

Recursive circulant graphs $G(N, d)$ are Cayley graphs defined on the abelian group $(\mathbb{Z}/N\mathbb{Z}, +)$, and therefore are vertex-transitive [PC94]. They also are Hamiltonian decomposable [GMR98].

In this paper, we will mainly focus on the particular case where $N = cd^m$, $1 \leq c < d$ is a multiple of a power of d. The recursive circulant graphs $G(cd^m, d)$ have been widely studied, notably in [Mic96, PC94]. For any $d \geq 3$, recursive circulant graphs $G(cd^m, d)$ are regular of degree $2m$ if $c = 1$, $2m + 1$ if $c = 2$ and $2m + 2$ otherwise. Moreover, $G(2^k, 4)$ have been shown to be Minimum Broadcast Graphs, Minimum Gossip Graphs and Minimum Linear Gossip Graphs [FR98a]. From this point of view, it is interesting to get an algorithm which recognizes any recursive circulant graph of the form $G(2^k, 4)$. More generally, the question of the complexity of an algorithm which, for a given graph G, determines if G is isomorphic to the recursive circulant graph $G(N, d)$ is of interest ; indeed, this would help to answer the even more general question of recognizing circulant graphs. Note that [MT98] have partially answered the question by giving an algorithm allowing to recognize any circulant graph of order N, when N is prime, in time polynomial in N.

Moreover, in [CFG99], the authors have raised a question about the existence of algorithms and techniques to recognize any bipartite incident graph, that is a given family of bipartite graphs constructed from circulant graphs. Hence, the understanding of circulant graphs could also help in some way to answer the question.

In this paper, we give an answer concerning these questions for most of the recursive circulant graphs of the form $G(cd^m, d)$. More precisely, we answer the question for any $d \geq 3$, except in the case where c is even while d is odd (cf. Table 1 for a summary of the results). In order to achieve this, we give in Section 2 definitions and notations that will help us solving the problem ; we also give some preliminary technical results which will be the core of the recognition algorithm, and which will show its correctness. In Section 3, we develop these above mentioned algorithms and compute their complexity. This will lead to the main Theorem of the paper (Theorem 4). We finally apply this Theorem to recognize recursive circulant graphs of the form $G(2^k, 4)$, for which we get an $O(2^k \cdot k^4)$ algorithm.

2 Description of the Technique and Preliminary Results

In this section, we present some definitions, notations and results that will be useful to determine our recognition algorithm. First, we briefly explain the technique : it consists in distinguishing in the given graph G, those edges that are possibly in dimension 0 in the recursive circulant graph $G(cd^m, d)$. Once this is done, it is easy to check if G, of order cd^m, is isomorphic to $G(cd^m, d)$. Indeed, the edges in dimension 0 in $G(cd^m, d)$ form a Hamiltonian cycle by definition ; hence, once the edges in dimension 0 are distinguished, it suffices to relabel the vertices accordingly, from 0 to $N - 1 = cd^m - 1$. Once this is done, it suffices to check that any vertex $0 \leq v \leq N - 1$ is connected to every vertex of the form $v \pm d^i \bmod N$, for any $1 \leq i \leq \lceil \log_d N \rceil - 1$, as in the definition of a recursive circulant graph (cf. Definition 2).

Non surprisingly, the most difficult and time consuming part of the algorithm consists in isolating those edges in dimension 0. This is done thanks to a two-steps technique, which is as follows :

(a) **Step 1** : G being given, distinguish the edges that are possibly in the highest dimension in $G(cd^m, d)$ (dimension $m-1$ when $c=1$, dimension m otherwise). This will be done by different methods, depending on the parity of d and/or c (cf. Section 3.1).

(b) **Step 2** : we recursively distinguish the edges in dimension $m-i$ ($2 \leq i \leq m$ when $c=1$, $1 \leq i \leq m$ otherwise) by calculating, in $G(cd^m, d)$, the number of pairwise distinct cycles of length $d+1$ that go through an edge in dimension $m-i+1$. This is done using the same routine m or $m-1$ times (depending on the value of c), cf. Theorem 1, starting from the highest dimension (whose edges have been distinguished in **Step 1**), in order to reach dimension 0.

As mentioned above, most of the proofs here rely on the computation of the number of pairwise *distinct* cycles of fixed length that go through a particular edge in the graph, where we define two cycles to be *distinct* iff they differ on at least one edge.

Moreover, in order to simplify proofs, we will use in the rest of this paper the following notation for cycles : a cycle C_L of length L in $G(cd^m, d)$ will be denoted thanks to the dimensions used by the edges of C_L. When the cycle uses an edge $(v, v+d^k)$, this will be denoted by $\oplus dim\ k$; when it uses an edge $(v, v-d^k)$, this will be denoted by $\ominus dim\ k$.

For instance, the following notation : $\oplus dim\ \alpha\ [\ominus dim\ \beta]^\gamma$ means that we first use an edge $(v, v+d^\alpha)$, then γ times an edge of the type $(w, w-d^\beta)$. Clearly, this notation does not define a unique cycle in $G(cd^m, d)$, since it is vertex transitive ; it only defines a pattern that exists in the graph, such that this pattern corresponds to cycles in the graph.

Finally, we recall that, in the following, N will denote the number of vertices of the considered graph (very often in this paper, N will be equal to cd^m), and that all the computations will be made modulo N.

Starting from this point, we can now give some preliminary results, which will later be used to prove the correctness of our recognition algorithm.

Proposition 1 *Let $p \geq 1$ and $d \geq 2$ be two integers. The following equation : $d^p + \sum_{i=1}^{q} \varepsilon_i d^{r_i} = 0$, with $1 \leq q \leq d$, $\varepsilon_i \in \{-1, +1\}$ and $0 \leq r_i \leq p-1$ ($r_i \in \mathbb{N}$) $\forall\ 1 \leq i \leq q$, has only one solution. It is the following : $q = d$, $\varepsilon_i = -1$ and $r_i = p-1\ \forall\ 1 \leq i \leq d$.*

Proof : By induction on p. Let $p = 1$; in that case, by definition $r_i = 0\ \forall\ 1 \leq i \leq q$. Hence, we need to solve $d + \sum_{i=1}^{q} \varepsilon_i = 0$. However, it is easy to see that $\sum_{i=1}^{q} \varepsilon_i = -d$ has only one solution : $q = d$ and $\varepsilon_i = -1\ \forall\ 1 \leq i \leq d$.

Suppose now that the above Proposition is true for a fixed $p \geq 1$, and let us show it then also holds for $p+1$. Let $S = d^{p+1} + \sum_{i=1}^{q} \varepsilon_i d^{r_i}$, $0 \leq r_i \leq p$, and let us solve $S = 0$. Let $k = \sum_{i \text{ s.t.} r_i = 0} \varepsilon_i$. In that case, let $S_1 = S - d^{p+1} - k = \sum_{r_i \neq 0} \varepsilon_i d^{r_i}$. Clearly, we have $k \in [-q; q]$. However, $S_1 \equiv 0$ mod d and $d^{p+1} \equiv 0$ mod d (since $p \geq 1$) ; hence, since we have $S_1 + d^{p+1} + k = 0$, necessarily we must have $k \equiv 0$ mod d. There are 3 cases here :

(a) $k = -d$. This means that $q = d$, $\varepsilon_i = -1$ and $r_i = 0\ \forall\ 1 \leq i \leq d$. Thus, $S_1 = 0$ and we have to solve $d^{p+1} - d = 0$. This is possible iff $p = 0$, since $d \geq 2$; however, since we supposed $p \geq 1$, this equation has no solution here.

(b) $k = d$. This means that $q = d$, $\varepsilon_i = +1$ and $r_i = 0\ \forall\ 1 \leq i \leq d$. Thus, $S_1 = 0$, and we have to solve $d^{p+1} + d = 0$. Since we supposed $d \geq 2$, this is impossible.

(c) $k = 0$. If $S_1 = 0$, then we need to solve $d^{p+1} = 0$, which is impossible since we supposed $d \geq 2$. If $S_1 \neq 0$, this means that S_1 is a sum of terms, each of which is a non-zero power of

d. Let q' be the number of terms contained in S_1 ; clearly, $1 \leq q' \leq q$. In that case, we need to solve : $d^{p+1} + \sum_{j=1}^{q'} \varepsilon_j d^{r_j}$, where we have $r_j \geq 1 \; \forall \; 1 \leq j \leq q'$. Hence, since $d \neq 0$, we can factorize this expression by d. Thus we need now to solve the following equation **(E')** : $d^p + \sum_{j=1}^{q'} \varepsilon_j d^{r_j - 1} = 0$, where $1 \leq q' \leq q \leq d$, $0 \leq r_j - 1 \leq p$ and $\varepsilon_j \in \{-1, +1\} \; \forall \; 1 \leq j \leq q'$. By induction hypothesis, this equation has only one solution : $q' = d$, $r_j - 1 = p - 1$ and $\varepsilon_j = -1 \; \forall \; 1 \leq j \leq q'$. Thus, we get $q = q' = d$, and $\varepsilon_j = -1$ and $r_j = p \; \forall \; 1 \leq j \leq d$, which shows that the Proposition also holds for $p + 1$ provided that it holds for p.

Thus the Proposition is proved by induction on p. □

Theorem 1 *In $G(cd^m, d)$, with $d \geq 3$, $m \geq 1$ and $1 \leq c \neq 2 < d$, there is only one cycle of length $d + 1$ that goes through an edge in dimension $1 \leq p$ and no edge in dimension $q \geq p$. It is of the form : $\oplus dim \; p \; [\ominus dim \; (p - 1)]^d$.*

In the case $c = 2$, the above cycle exists in $G(2 \cdot d^m, d)$, but is not unique when $p = m$. Indeed, in that case there exists a second cycle that satisfies those constraints. It is of the form : $\oplus dim \; m \; [\oplus dim \; (m - 1)]^d$.

Proof : Due to the definition of $G(cd^m, d)$, finding such a cycle is equivalent to solve the following equation : $d^p + \sum_{i=1}^{d} \varepsilon_i d^{r_i} \equiv 0 \mod cd^m$, where $\varepsilon_i \in \{-1, +1\}$ and $0 \leq r_i \leq p - 1 \; \forall \; 1 \leq i \leq d$. Let $S = d^p + \sum_{i=1}^{d} \varepsilon_i d^{r_i}$. It is easy to see that $S \in [0; 2 \cdot d^p]$. We distinguish 3 cases here :

(a) $c = 1$. In that case, we will suppose here that $m \geq 2$, since when $m = 1$, $G(d^m, d)$ is isomorphic to the cycle of length d, C_d, and thus no cycle of length $d + 1$ can exist in this graph. Thus, we will suppose here $m \geq 2$. We have $p \leq m - 1$ by definition of $G(d^m, d)$. However, $2 \cdot d^p < d^m$, since $p \leq m - 1$ and $d \geq 3$ by hypothesis. Hence the only possibility is $S = 0$. By Proposition 1, we see that this is only possible when $\varepsilon_i = -1$ and $r_i = p - 1 \; \forall \; 1 \leq i \leq d$. Thus, this gives in $G(d^m, d)$ a cycle of the form : $\oplus dim \; p \; [\ominus dim \; (p - 1)]^d$.

(b) $c = 2$. In that case, we have $p \leq m$ by definition of $G(2 \cdot d^m, d)$. Since $S \in [0; 2 \cdot d^p]$, there are two possibilities here :

- $S = 2 \cdot d^m$. This is only possible when $p = m$, $\varepsilon_i = +1$ and $r_i = m - 1 \; \forall \; 1 \leq i \leq d$. Thus, this gives in $G(2 \cdot d^m, d)$ a cycle of the form : $\oplus dim \; m \; [\oplus dim \; (m - 1)]^d$.
- $S = 0$. In that case, thanks to Proposition 1, we know that this is only possible when $\varepsilon_i = -1$ and $r_i = p - 1 \; \forall \; 1 \leq i \leq d$. Thus, this gives in $G(2 \cdot d^m, d)$ a cycle of the form : $\oplus dim \; p \; [\ominus dim \; (p - 1)]^d$.

(c) $c > 2$. Here, we also have $p \leq m$ by definition of $G(cd^m, d)$. We know that $S \in [0; 2 \cdot d^p]$; however, $2 \cdot d^p < cd^m \; \forall \; 1 \leq p \leq m$, since $c > 2$. Hence the only possibility for S to satisfy $S \equiv 0 \mod cd^m$ is when $S = 0$. By Proposition 1, we know that the only solution for this is when $\varepsilon_i = -1$ and $r_i = p - 1 \; \forall \; 1 \leq i \leq d$. Thus, this gives in $G(cd^m, d)$ a cycle of the form : $\oplus dim \; p \; [\ominus dim \; (p - 1)]^d$.

Altogether, the above analysis proves the Theorem. □

Proposition 2 *For all integers $d \geq 2$ and odd $1 \leq c \leq d$ the equation : $\sum_{i=1}^{c} \varepsilon_i d^{r_i} = 0$ with $\varepsilon_i \in \{-1, +1\}$ and $r_i \in \mathbb{N} \; \forall \; 1 \leq i \leq c$, has no solution.*

Proof : By induction on c. First, let us suppose $c = 1$. Then, the equation we need to solve is $\varepsilon_1 d^{r_1} = 0$. Since $d \neq 0$ and $\varepsilon_1 \in \{-1, +1\}$, clearly this equation has no solution.

Now suppose the above Proposition holds for any odd integer less than or equal to c. where c is also odd, and let us show it then holds for $c + 2$ (provided that $c + 2 \leq d$). Let $S = \sum_{i=1}^{c+2} \varepsilon_i d^{r_i}$, and let us solve $S = 0$. For this, let $r_{min} = \min\{r_i, \; 1 \leq i \leq c + 2\}$, and let us factorize S by $d^{r_{min}}$.

Thus we need to solve $d^{r_{min}} \cdot S' = 0$, where $S' = \sum_{i=1}^{c+2} \varepsilon_i d^{r'_i}$, with $r'_i = r_i - r_{min}\ \forall\ 1 \leq i \leq c+2$ (thus, $r'_i \geq 0\ \forall\ i$). Since $d \neq 0$, it suffices to solve $S' = 0$. Now let z be the number of terms in S' such that $r'_i = 0$, and let $k = \sum_{i \text{ s.t.} r'_i=0} \varepsilon_i$. By definition of r'_i, necessarily $z \geq 1$; moreover, since $c+2 \leq d$, $k \in [-d; d]$. Thus we now need to solve $k + S'_1 = 0$, where $S'_1 = \sum_{r'_i \neq 0} \varepsilon_i d^{r'_i}$. However, $S'_1 \equiv 0 \bmod d$, thus we need to have $k \equiv 0 \bmod d$. Three cases arise here :

(a) $k = -d$. This means $S'_1 = 0$, and thus we need to solve $-d = 0$, which is impossible.

(b) $k = d$. Similarly to the previous case, this is impossible since $d \neq 0$.

(c) $k = 0$. This necessarily implies that z is even, thus $z \geq 2$. Since S'_1 contains $c+2-z$ terms, the equation we need to solve becomes : $\sum_{j=1}^{c+2-z} \varepsilon_i d^{r'_i}$. Let $c' = c+2-z$. Since $z \geq 2$ is even and c is odd, c' is odd and $c' \leq c$. By induction, we know that there is no solution to this equation, and the above Proposition holds for $c+2$.

Thus, Proposition 2 above is proved by induction on c. □

Theorem 2

(a) In $G(d^m, d)$ with $m \geq 2$ and odd $d \geq 3$, there is only one cycle of length d. It is of the form : $[\oplus dim\ (m-1)]^d$;

(b) In $G(cd^m, d)$ with $m \geq 1$ and odd $3 \leq c < d$, there is only one cycle of length c. It is of the form : $[\oplus dim\ m]^c$.

Proof : Note that in Theorem 2(a), we deal with $G(d^m, d)$ with odd d, which could be considered as a particular case of $G(cd^{m-1}, d)$ with odd $c = d$. Hence, case Theorem 2(a) can be considered as a particular case of case (b). However, for clarity reasons, we keep them separate here.

Let us first prove (a). Finding a cycle of length d in $G(d^m, d)$ is equivalent to solve the following equation : $\sum_{i=1}^{d} \varepsilon_i d^{r_i} \equiv 0 \bmod d^m$, where $\varepsilon_i \in \{-1, +1\}$ and $0 \leq r_i \leq m-1\ \forall\ 1 \leq i \leq d$. However, we see that $S = \sum_{i=1}^{d} \varepsilon_i d^{r_i} \in [-d^m; d^m]$, hence there are 3 possibilities :

- $S = d^m$. In that case, we necessarily have $\varepsilon_i = -1$ and $r_i = m-1\ \forall\ 1 \leq i \leq d$. Hence the corresponding cycle C_1 in $G(d^m, d)$ is of the form : $[\oplus dim\ (m-1)]^d$.
- $S = -d^m$. In that case, we necessarily have $\varepsilon_i = -1$ and $r_i = m-1\ \forall\ 1 \leq i \leq d$. Hence the corresponding cycle C_2 in $G(d^m, d)$ is of the form : $[\ominus dim\ (m-1)]^d$. However, we can easily see that C_2 uses exactly the same edges as cycle C_1 above, and thus is not distinct from C_1.
- $S = 0$. However, we know by Proposition 2 that this equation has no solution. Thus, this case is impossible.

Hence, the only cycle of length d in $G(d^m, d)$ is of the form : $[\oplus dim\ (m-1)]^d$.

To prove (b), we use exactly the same arguments as above : we solve the equation $\sum_{i=1}^{c} \varepsilon_i d^{r_i} \equiv 0 \bmod cd^m$, and note that $S = \sum_{i=1}^{c} \varepsilon_i d^{r_i} \in [-cd^m; cd^m]$. Since $S = 0$ is impossible by Proposition 2, we see that we either have $S = -cd^m$ or $S = cd^m$. Both cases lead to the same cycle in $G(cd^m, d)$, which is of the form : $[\oplus dim\ m]^c$. □

Theorem 3

(a) In $G(d^m, d)$ with $m \geq 2$ and even $d \geq 4$, there is only one cycle of length $d+1$ going through a given edge in dimension $m-1$. It is of the form : $\oplus dim\ (m-1)\ [\ominus dim\ (m-2)]^d$;

(b) In $G(2d^m, d)$ with $m \geq 1$ and even $d \geq 4$, there are exactly 2 distinct cycles of length $d+1$ going through a given edge in dimension m. They are of the form : $\oplus dim\ m\ [\ominus dim\ (m-1)]^d$ and $\oplus dim\ m\ [\oplus dim\ (m-1)]^d$;

(c) *In $G(cd^m, d)$ with $m \geq 1$, even $d \geq 4$ and $3 \leq c < d$, there is only one cycle of length $d+1$ going through a given edge in dimension m. It is of the form : $\oplus dim\ m\ [\ominus dim\ (m-1)]^d$.*

Proof : We note that the above Theorem is close in its form to Theorem 1 in the case $p = m-1$ when $c = 1$ (resp. $p = m$ when $c \geq 2$). Yet, the distinction here is that the restriction consisting in not using any other edge in dimension $m-1$ (resp. in dimension m when $c \geq 2$) is not necessary here. However, we will see that the proof relies on the same type of arguments.

Let us first prove (b) and (c), that is we suppose $c \geq 2$. For this, we need to solve the following equation : $d^m + \sum_{i=1}^{d} \varepsilon_i d^{r_i} \equiv 0 \bmod cd^m$, with $\varepsilon_i \in \{-1, +1\}$ and $0 \leq r_i \leq m\ \forall\ 1 \leq i \leq d$. We see that $S = d^m + \sum_{i=1}^{d} \varepsilon_i d^{r_i} \in [d^m - d^{m+1}; d^m + d^{m+1}]$ in all the cases. Let us then split the possible cases in two categories :

- $S \equiv 0 \bmod cd^m$ and $S \neq 0$. Let us prove that this happens iff $c = 2$, $r_i = m-1$ and $\varepsilon_i = +1\ \forall\ 1 \leq i \leq d$. For this, let us suppose $S = \alpha c d^m$, with $\alpha \in \mathbb{Z}^*$. Hence the above equation becomes : $\sum_{i=1}^{d} \varepsilon_i d^{r_i} = (\alpha c - 1) d^m$. However, $\alpha c - 1 \neq 0$ since we supposed $c \geq 2$. Let $k = \sum_{\text{s.t.} r_i = 0} \varepsilon_i$ and $S_1 = S - k$. We then have to solve $S_1 + k = (\alpha c - 1) d^m$, with $k \in [-d; d]$. Since $S_1 \equiv 0 \bmod d$ and $(\alpha c - 1) d^m \equiv 0 \bmod d$, we must have $k \equiv 0 \bmod d$. Hence 3 cases arise :

  (1) $k = +d$, thus $\varepsilon_i = +1$ and $r_i = 0\ \forall\ 1 \leq i \leq d$, and we need to solve $d = (\alpha c - 1) d^m$. This happens iff $\alpha = 1$, $c = 2$ and $m = 1 = r_i + 1$.

  (2) $k = -d$, thus $S_1 = 0$ and we need to solve $-d = (\alpha c - 1) d^m$. Clearly, this is impossible.

  (3) $k = 0$. In that case, since $d \neq 0$, the equation can be factorized by d, and leads to an equation of the same form, for which we can apply the same arguments as above.

  Thus, this equation holds iff $c = 2$, $\varepsilon_i = +1$ and $r_i = m-1\ \forall\ 1 \leq i \leq d$, and this gives in $G(cd^m, d)$ a cycle of the form : $\oplus dim\ m\ [\oplus dim\ (m-1)]^d$.

- $S = 0$. By similar arguments to the ones developed above, we can show that the only case for which we have $S = 0$ happens when $\varepsilon_i = -1$ and $r_i = m-1\ \forall\ 1 \leq i \leq d$ (due to lack of space, the proof is omitted here) ; thus, we get a cycle of the form : $\oplus dim\ m\ [\ominus dim\ (m-1)]^d$.

Similarly, when $c = 1$ and $d \geq 4$ is even, we show by the same type of arguments that the only cycle of length $d+1$ that goes through a given edge in dimension $m-1$ is of the form : $\oplus dim\ (m-1)\ [\ominus dim\ (m-2)]^d$. (the proof is omitted here). □

3 Recognizing $G(cd^m, d)$

In this section, we give a recognition algorithm for recursive circulant graphs $G(cd^m, d)$, for all $d \geq 3$, except in the case where c is even while d is odd. In all the cases, we recall that the recognition algorithm always works in 2 steps (cf. Section 2) : first, distinguish those edges possibly in highest dimension (i.e., dimension m or $m-1$ depending on the value of c). This will be done thanks to the results of Section 3.1. Second, distinguish those edges possibly in dimension $m-i$ from those in dimension $m-i+1$, till we reach dimension 0. From that point, it is easy to check that every edge corresponds to a "jump" of a power of d. This will be undertaken in Section 3.2.

3.1 Distinguishing Edges in Highest Dimension (Step 1)

Step 1 of the algorithm is undertaken in different ways depending on the value of c in $G(cd^m, d)$. Hence, we need to split the proof in three parts.

3.1.1 $c = 1$

Before developing the algorithm realizing **Step 1**, let us recall the results we already have : first, when d is odd, we can use Theorem 2(a) to get those edges possibly in dimension $m-1$. Indeed, Theorem 2(a) yields that the only cycle of length d in $G(d^m, d)$ only uses edges in dimension $m-1$. However, when d is even, Theorem 3 yields that there exists only one cycle of length $d+1$ going through an edge in dimension $m-1$; but this cycle also uses edges in dimension $m-2$. Hence, in order to distinguish those two dimensions ($m-1$ and $m-2$), we give the following Proposition. We note that this Proposition is also valid for $c \neq 1$, which will be useful in Section 3.1.3.

Proposition 3 *For any even $d \geq 4$:*

(a) in $G(d^m, d)$ with $m \geq 2$, there are at least d pairwise distinct cycles of length $d+1$ that go through a given edge e_i in dimension i, for any $0 \leq i \leq m-2$.

(b) in $G(cd^m, d)$ with $m \geq 1$ and $2 \leq c < d$, there are at least d pairwise distinct cycles of length $d+1$ that go through a given edge e_i in dimension i, for any $0 \leq i \leq m-1$.

Proof : Let us first prove (a) : let $d \geq 4$ be even, and let us give, in $G(d^m, d)$, d pairwise distinct cycles of length $d+1$ that go through a given edge e_i in dimension $0 \leq i \leq m-2$. For this, we use the following pattern : $\oplus dim\ (i+1)\ [\ominus dim\ i]^d$.

We can see that in this pattern, we have d edges in dimension i, and it is not difficult to see that for a fixed edge e_i in dimension $0 \leq i \leq m-2$, there are at least d pairwise distinct cycles going through e_i. Indeed, we get each time a different cycle when $e_i = (\alpha, \alpha - d^i)$ lies in j-th position in the above pattern ($1 \leq j \leq d$).

To prove (b), we use exactly the same pattern, and we note that, since $c \geq 2$, we can add the case $i = m-1$. Therefore, we also show that through any edge e_i in dimension $0 \leq i \leq m-1$ lie at least d distinct cycles of length $d+1$ in $G(cd^m, d)$. □

Now we are ready to give an algorithm that realizes **Step 1**.

Property 1 *For any $d \geq 3$ and $m \geq 2$, there exists an $O(d^{m+2} \cdot (2m)^d)$ algorithm to realize* **Step 1** *of a recognition algorithm of $G(d^m, d)$.*

Proof : We first note that if $m = 1$, then $G(d^m, d)$ is isomorphic to the cycle C_d of length d, which can be recognized in $O(d)$. Hence, we will consider here $m \geq 2$.

Consider the following algorithm, that allows to isolate in a given graph G those edges that are possibly in dimension $m-1$ in the recursive circulant graph $G(d^m, d)$.

Algorithm *Step 1 ($c = 1$)*
Input : a regular graph $G = (V, E)$ of $N = d^m$ vertices ($d \geq 3$, $m \geq 2$) and of degree $2m$.
Output : S_{m-1}, a set of edges in G that are possibly in dimension $m-1$ in $G(d^m, d)$.

Phase 1. For every edge $e \in E(G)$, compute the number n_e of cycles in G that go through e, and that are of length d if d is odd, and $d+1$ if d is even.
Phase 2. Identify the set $S_{m-1} \subset E$ of edges that are possibly edges in dimension $m-1$ in $G(d^m, d)$. If $|S_{m-1}| \neq d^m$, then G is not a recursive circulant graph.

Let us first suppose d is odd. Then the correctness of the above algorithm is given by Theorem 2(a) : the only cycle of length d in $G(d^m, d)$ only uses edges in dimension $m-1$.

If d is even, then by Theorem 3(a) we know that there is only one cycle of length $d+1$ going through an edge in dimension $m-1$. However, this cycle uses edges in dimension $m-2$. Yet, Proposition 3(a) yields that at least $d \geq 3$ cycles of length $d+1$ go through a given edge in dimension $m-2$ in $G(d^m, d)$, thus there is no ambiguity, and the correctness of the above algorithm also holds for even d.

The complexity of this algorithm is as follows : first, assuming node-adjacency testing in constant time, realizing Phase 1 takes $O(d^2 \cdot (2m)^{d-1})$ for each edge $e \in E(G)$. Indeed, let $e = (u, v)$, and let us build the tree representing the possible paths of length $d-2$ when d is odd (resp. $d-1$ when d is even) having u as an endpoint. This tree holds $O((2m)^{d-2})$ leaves when d is odd (resp. $O((2m)^{d-1})$ when d is even), because $G(d^m, d)$ is $2m$-regular. We then test the possible adjacency between any leaf l and v. If l and v are adjacent, then we look at the path from u to l and eliminate those which contain twice the same vertex, or which contain v. This costs $O(d^2)$ for each leaf. Since there are $O((2m)^{d-1})$ leaves at worse (i.e. when d is even), we end up with a complexity of $O(d^2 \cdot (2m)^{d-1})$ for each edge $e \in G$. Hence, Phase 1 runs in $O(m \cdot d^{m+2} \cdot (2m)^{d-1})$, i.e. in $O(d^{m+2} \cdot (2m)^d)$, since there are $m \cdot d^m$ edges in G. Phase 2 runs in $O(m \cdot d^m)$, which is the number of edges in $G(d^m, d)$. Clearly, Phase 1 is the most time consuming and thus we get an algorithm running in $O(d^{m+2} \cdot (2m)^d)$. □

3.1.2 Odd $c \geq 3$

Property 2 *For any $m \geq 1$ and odd $3 \leq c < d$, there exists an $O(cd^{m+2} \cdot (2m)^{c-1})$ algorithm to realize* **Step 1** *of a recognition algorithm of $G(cd^m, d)$.*

Proof : Consider the following algorithm :

Algorithm *Step 1 (odd $c \geq 3$)*
Input : a regular graph $G = (V, E)$ of $N = cd^m$ vertices (with odd $3 \leq c < d$) and of degree $2m+2$.
Output : S_m, a set of edges in G that are possibly in dimension $m-1$ in $G(cd^m, d)$.

Phase 1. For every edge $e \in E(G)$, compute the number n_e of cycles of length c that go through e.
Phase 2. Identify the set $S_m \subset E$ of edges that are possibly edges in dimension m in $G(cd^m, d)$. If $|S_m| \neq cd^m$, then G is not a recursive circulant graph.

The correctness of the algorithm is given by Theorem 2(b), which yields that the only cycle of length c in $G(cd^m, d)$ with odd $3 \leq c < d$ only uses edges in dimension m. The complexity of the above algorithm is, as in Property 1, fixed by Phase 1, since Phase 2 runs in $O((m+1) \cdot cd^m)$, corresponding to the number of edges. Phase 1 runs in $O((m+1) \cdot cd^{m+2} \cdot (2m)^{c-2})$, i.e. in $O(cd^{m+2} \cdot (2m)^{c-1})$, since each cycle of length c going through a given edge e in G costs $O(d^2 \cdot (2m)^{c-2})$ (G being $(2m+2)$-regular), and since there are $(m+1) \cdot cd^m$ edges to consider. □

3.1.3 Even c – Even $d \geq 4$

Property 3 *For any $m \geq 1$, even $d \geq 4$ and even $2 \leq c < d$, there exists an $O(cd^{m+2} \cdot (2m)^d)$ algorithm to realize* **Step 1** *of a recognition algorithm of $G(cd^m, d)$.*

Proof : Consider the following algorithm :

Algorithm *Step 1 (even c, even $d \geq 4$)*
Input : a regular graph $G = (V, E)$ of $N = cd^m$ vertices (even $d \geq 4$, even c) and of degree $2m+1$ (if $c = 2$) or $2m+2$ (if $c \geq 4$).
Output : S_m, a set of edges in G that are possibly in dimension m in $G(cd^m, d)$.

Phase 1. For every edge $e \in E(G)$, compute the number n_e of cycles of length $d+1$, and that go through e.
Phase 2. Identify the set $S_m \subset E$ of edges that are possibly edges in dimension m in $G(cd^m, d)$. If $|S_m| \neq d^m$ when $c = 2$ (resp. if $|S_m| \neq cd^m$ when $c \geq 4$), then G is not a recursive circulant graph.

The correctness of the algorithm is given by Theorem 3(b)-(c) (no more than two cycles of length $d+1$ going through a given edge in dimension m) and Proposition 3(b) (at least $d \geq 4$ cycles of length $d+1$ going through a given edge in dimension $i < m$).

The complexity of the above algorithm is fixed by Phase 1, since Phase 2 runs in $O((m+1) \cdot cd^m)$, corresponding to the number of edges. Phase 1 runs in $O(m \cdot cd^{m+2} \cdot (2m)^{d-1})$, i.e. in $O(cd^{m+2} \cdot (2m)^d)$ since each cycle of length $d+1$ going through a given edge e in G costs $O(d^2 \cdot (2m)^{d-1})$, and since there are $O(m \cdot cd^m)$ edges to consider. □

3.2 Getting to Dimension 0 (Step 2)

In this Section, we develop **Step 2** of the algorithm. The technique we use here is, from any set $S_p \in E(G)$ of edges possibly in dimension p, to determine the set S_{p-1} of edges possibly in dimension $p-1$. We start with $p = m-1$ when $c = 1$, and $p = m$ when $c \geq 2$. For this, we take an edge e_p in S_p, search for cycles of length $d+1$ that go through e_p and no other edge in S_q, $q \geq p$. By Theorem 1, if such a cycle C exists, then all the edges distinct from e_p in C are in dimension $p-1$, thus they belong to S_{p-1}. We repeat this till we reach dimension 0, from which we can construct the graph and check its isomorphism to $G(cd^m, d)$.

Property 4 *For any $m \geq 1$, $d \geq 3$ and $1 \leq c < d$, there exists an $O(cd^{m+2} \cdot (2m)^d)$ algorithm realizing* **Step 2** *of a recognition algorithm of $G(cd^m, d)$.*

Proof : For clarity reasons, we will distinguish the cases $c = 1$ and $c \neq 1$, though both cases are similar. Thus, consider the two following algorithms :

Algorithm *Step 2 ($c = 1$)*
Input : a regular graph $G = (V, E)$ of $N = d^m$ vertices, $d \geq 3$ and $m \geq 2$, and a set S_{m-1} of edges possibly in dimension $m-1$ in $G(d^m, d)$.
Output : tell whether G is isomorphic to the recursive circulant graph $G(d^m, d)$.

Phase 3. Let $i = m-1$.
do
For every edge e_i in S_i, compute the number of cycles of length $d+1$ that go through e_i, and that go through no other edge in S_j for $j \geq i$.
Identify the set $S_{i-1} \subset E$ of edges that are possibly edges in dimension $i-1$ in $G(d^m, d)$. If $|S_{i-1}| \neq d^m$, then G is not a recursive circulant graph.
$i = i - 1$
while $i \geq 1$
Phase 4. Label the vertices accordingly to a Hamiltonian cycle using edges in S_0. If this construction fails, then G is not a recursive circulant graph. Otherwise, G is a recursive circulant graph $G(d^m, d)$ iff for every vertex $v \in V$, v is connected to every vertex w of the form $v + d^i$ (mod d^m), for every $0 \leq i \leq m-1$, and there is no extra edge in G.

Algorithm *Step 2 ($c \geq 2$)*
Input : a regular graph $G = (V, E)$ of $N = cd^m$ vertices, $d \geq 3$ and $m \geq 2$, and a set S_m of edges possibly in dimension m in $G(cd^m, d)$.
Output : tell whether G is isomorphic to the recursive circulant graph $G(cd^m, d)$.

Phase 3. Let $i = m$.
do
For every edge e_i in S_i, compute the number of cycles of length $d+1$ that go through e_i, and that go through no other edge in S_j for $j \geq i$.
Identify the set $S_{i-1} \subset E$ of edges that are possibly edges in dimension $i-1$ in $G(cd^m, d)$.

If $|S_{i-1}| \neq cd^m$, then G is not a recursive circulant graph.
$i = i - 1$
while $i \geq 1$
Phase 4. Label the vertices accordingly to a Hamiltonian cycle using edges in S_0. If this construction fails, then G is not a recursive circulant graph. Otherwise, G is a recursive circulant graph $G(cd^m, d)$ iff for every vertex $v \in V$, v is connected to every vertex w of the form $v + d^i \pmod{cd^m}$, for every $0 \leq i \leq m$, and there is no extra edge in G.

The correctness of both these algorithms is given by Theorem 1. Let us now compute the maximum complexity of those two algorithms : for each edge of the set S_i, using Theorem 1 we look at cycles of length $d+1$. Thus it takes $O(d^2 \cdot (2m)^{d-1})$ for each of these edges ($G(cd^m, d)$ is regular of degree at most $2m$, $2m+1$ or $2m+2$ depending on the value of c). Since $|S_i| \leq cd^m$ for any i, and since we repeat this process at most m times (to get from dimension m to dimension 0), this leads to a complexity in $O(cd^{m+2} \cdot (2m)^d)$. □

Altogether, Properties 1, 2, 3 and 4 give the main theorem of this paper.

Theorem 4 (Recognition of $G(cd^m, d)$) *For any $d \geq 3$ and $m \geq 1$, there exists an $O(cd^{m+2} \cdot (2m)^d)$ algorithm to recognize any recursive circulant graph of the form $G(cd^m, d)$, except in the case when c is even while d is odd.*

Below, we give a summary of the results given in this paper about the existence of a recognition algorithm of $G(cd^m, d)$. The "XXX" entries are cases where $c \geq d$ (thus they do not arise by definition), and the entries "Not Solved" indicate the cases for which this paper does not give an answer. In all the other entries, reference is given to the Properties used to realize **Step 1** and **Step 2** of the recognition algorithm.

	$d = 2$	Even $d \geq 4$	Odd d
$c = 1$	Not Solved	Properties 1 and 4	
Odd $c \geq 3$	XXX	Properties 2 and 4	
Even c	XXX	Properties 3 and 4	Not Solved

Table 1: Summary of the results

As a Corollary of Theorem 4, we give an $O(2^k \cdot k^4)$ algorithm to recognize recursive circulant graphs $G(2^k, 4)$. As indicated in the Introduction, it is interesting to get a recognition algorithm for those graphs, since $G(2^k, 4)$ competes well with the hypercube of dimension k and the Knödel graph of order 2^k, and since they have good broadcasting and gossiping properties.

Corollary 1 (Recognition of $G(2^k, 4)$) *There exists an $O(2^k \cdot k^4)$ algorithm to recognize any recursive circulant graph of the form $G(2^k, 4)$.*

Proof : Let $N = 2^k$, and let us split the proof depending on the parity of k. If we suppose $k = 2m$ is even, then $G(2^k, 4)$ is formally defined as $G(4^m, 4)$, that is $d = 4$ and $c = 1$. When $k = 2m + 1$ is odd, the recursive circulant graph is $G(2 \cdot 4^m, 4)$, i.e. $c = 2$ and $d = 4$. By Theorem 4, we get in both cases a recognition algorithm running in $O(N \cdot m^4)$, that is in $O(2^k \cdot k^4)$. □

4 Conclusion and Open Problems

In this paper, we have given an $O(cd^{m+2} \cdot (2m)^d)$ algorithm to recognize recursive circulant graphs of the form $G(cd^m, d)$, for all $d \geq 3$, with the exception of the case where c is even while d is odd. We note that when $d = O(1)$, this then gives an $O(N \cdot (\log(N))^{O(1)})$ algorithm, with

$N = cd^m$.

Moreover, a corollary of this result is the existence of an $O(2^k \cdot k^4)$ algorithm to recognize the family of recursive circulant graphs $G(2^k, 4)$. This last result is to be compared with the rather simple $O(2^k)$ algorithm to recognize the hypercube of dimension k, and the result of [CFG99], which gives an algorithm to recognize the Knödel graph $W_{k,2^k}$ in $O(2^k \cdot k^3)$.

As a conclusion, we give three open problems related to this paper, and growing in difficulty :

- Can we extend the result to any recursive circulant graph of the form $G(cd^m, d)$, with even c and odd d ? For any $d = 2$? We conjecture that the answer to this question is positive.
- Can we extend the result to any circulant graph of the form $G(N, d)$? That is, what happens if we do not restrict ourselves to the case $N = cd^m$? We note that this has been partially answered in [Bar99] for chordal rings of degree 3.
- Can we apply the same kind of techniques for the more general case of any circulant graph $C_N(p_1, p_2, \ldots p_k)$, that is when both N and the "jumps" are not constrained ? We note that there already exists an algorithm in the case where N is a prime integer [MT98], which is polynomial in the number N of vertices.

References

[Bar99] L. Barrière. Triangulations and chordal rings. In *Proc. 6th International Colloquium on Structural Information and Communication Complexity (SIROCCO '99)*, volume 5 of *Proceedings in Informatics*, pages 17–31. Carleton University Press, 1999.

[CFG99] J. Cohen, P. Fraigniaud, and C. Gavoille. Recognizing bipartite incident-graphs of circulant digraphs. In 25^{th} *International Workshop on Graph-Theoretic Concepts in Computer Science (WG'99)*, volume 1665 of Lecture Notes in Computer Science. Springer-Verlag, 1999. 215-227.

[Fer99] G. Fertin. *Étude des Communications dans les Réseaux d'Interconnexion.* PhD thesis, Université Bordeaux I, 1999.

[FR98a] G. Fertin and A. Raspaud. Families of graphs having broadcasting and gossiping properties. *In Proc. of the 24th International Workshop on Graph-Theoretic Concepts in Computer Science (WG'98), Smolenice. LNCS*, 1517:63–77, 1998.

[FR98b] G. Fertin and A. Raspaud. A survey on Knödel graphs. Technical report, Laboratoire Bordelais de Recherche en Informatique, 1998. Submitted for publication.

[GMR98] G Gauyacq, C. Micheneau, and A. Raspaud. Routing in recursive circulant graphs : Edge forwarding index and hamiltonian decomposition. *In Proc. of the 24th International Workshop on Graph-Theoretic Concepts in Computer Science (WG'98), Smolenice. LNCS*, 1517:227–241, 1998.

[Mic96] C. Micheneau. *Graphes Récursifs Circulants : Structure et Communications. Communications Vagabondes et Simulation.* PhD thesis, Université Bordeaux I, 1996.

[MT98] M.E. Muzychuk and G. Tinhofer. Recognizing circulant graphs of prime order in polynomial time. *The Electronic Journal of Combinatorics*, 5(1):R25, 1998.

[PC] J-H. Park and K-Y. Chwa. Recursive circulants and their embeddings among hypercubes. Manuscript. Submitted for Publication.

[PC94] J.-H. Park and K.-Y. Chwa. Recursive circulant : a new topology for multicomputers networks (extended abstract). In *Proc. Int. Symp. Parallel Architectures, Algorithms and Networks ISPAN'94, Kanazawa, Japan*, pages 73–80, 1994.