Solutions to the Communication Minimization Problem for Affine Recurrence Equations

Vincent Loechner and Catherine Mongenet

Université Louis Pasteur de Strasbourg, Laboratoire ICPS,
Bd. S. Brant, F-67400 Illkirch, France

Abstract. This paper deals with communication optimization which is a crucial issue in automatic parallelization. From a system of parameterized affine recurrence equations, we propose a heuristic which determines an efficient space-time transformation. It reduces first the distant communications and then the local communications.
Keywords. automatic parallelization techniques, communication optimization, system of recurrence equations.

1 Introduction

Among the many works dealing with automatic parallelization techniques, one of the important issues is concerned with communication minimization. Our paper focuses on this question. It discusses strategies of communication optimization for systems of parameterized affine recurrence equations (PARE's) [10] which formalize single assignment affine loop nests. The parallelization of a PARE is classically based on an affine space-time transformation [5]. It assigns each computation to a virtual processor defined by an allocation function and to an execution time defined by a schedule function. The communications between virtual processors result from the projection of the dependences of the PARE.

We distinguish between two types of communications: local communications and distant communications. While local communications can be efficiently implemented on most target architectures since they are regular, distant communications are inefficient and should be avoided. Therefore our strategy of communication optimization mainly focuses on dependences which generate distant communications in order to localize them as much as possible. The localization requires certain conditions to be satisfied by the allocation function and in some cases by the schedule function.

This technique relies on the dependence information given by parameterized utilization sets and utilization vectors [9, 6]. This dependence modeling allows the classification of the dependences according to the potential number of distant communications they may induce.

This paper is organized as follows. Section 2 recalls the basis notions of PARE parallelization. Communication minimization is discussed in section 3, where two conditions to eliminate distant communications are presented. A heuristic to optimize communications using these two techniques is described in section 4 and illustrated with an example in section 5. In section 6 we compare our approach with related works and propose different ways to improve the heuristic.

2 Parallelization of Systems of Parameterized Affine Recurrence Equations

A *parameterized affine recurrence equation* (PARE) is defined by:

$$X[z] = f(..., Y[g(z,p)], ...) \qquad z \in \mathcal{D}(p) \qquad (E)$$

where X and Y are variable names, p is a vector of $\mathbb{Z}^q$ defining the problem parameters, and $\mathcal{D}(p) \subset \mathbb{Z}^n$ is a parameterized convex bounded polyhedron called the *iteration space*. It is defined by a set of k linear constraints of the form $\mathcal{D}(p) = \{z \in \mathbb{Z}^n \mid Cz + C'p + c \geq 0\}$, where C is a $k \times n$ matrix, C' a $k \times q$ matrix, and c a k-vector. The *index function* $g(z,p)$ is defined by $g(z,p) = Rz + Qp + r$, where R and Q are integer matrices, and r is an integer vector.

As detailed in [9], the dependence information extracted from such a PARE is characterized by its *utilization* and *emission* sets. The utilization set corresponding to occurrence $Y[z_0]$ in equation (E) is:

$$Util_{E,Y}(z_0, p) = \{z \in \mathcal{D}(p) \mid g(z,p) = z_0\}.$$

The utilization set is a convex polyhedron parameterized by z_0 and p. Its vertices are denoted by v_i. They can be determined as functions of z_0 and p using the algorithm developed by Loechner and Wilde in [6]: $v_i = A_i z_0 + B_i p + c_i$. This algorithm deals with rational polyhedra, and therefore finds rational vertices of such polyhedra[1]. Used in the context of utilization set derivation, it computes the rational vertices of the rational convex hull of $Util_{E,Y}$. Notice however, that in most practical problems, utilization sets are characterized by integer vertices. In the following, we develop methods to minimize communications relatively to the vertices of utilization sets. They only apply to integer vertices.

The lineality space of any utilization set associated with index function $g(z,p) = Rz + Qp + r$ is $Ker(R)$. Such a set is therefore characterized by a basis of vectors called the *utilization vectors* and denoted by $u_{E,Y,j}$ with $1 \leq j \leq dim_{Util_{E,Y}}$ and $dim_{Util_{E,Y}} = dim(\mathcal{D}(p)) - Rank(R)$. They can be computed using the Hermite normal form decomposition of matrix R. They verify $Ru_{E,Y,j} = 0$. In the particular case of a full rank matrix, $Ker(R) = 0$ and the utilization set is reduced to one single point. We call this a *degenerate* utilization set.

The dependence from Y to X in equation (E) is expressed by a set of dependence vectors of the form $d = z - z_0$, where $z \in Util_{E,Y}(z_0, p)$. Since the utilization set is a convex polyhedron, this dependence is characterized by a set of extremal dependence vectors. For any integer vertex v_i of $Util_{E,Y}(z_0, p)$ the corresponding extremal vector is denoted by $d_i = v_i - z_0$.

For a given dependence, the set of all the points z_0 which originate the dependence vectors is called the emission set:

$$\begin{aligned} Emit_{E,Y}(p) &= \{z_0 \in \mathbb{Z}^n \mid \exists z \in \mathcal{D}(p) \text{ such that } z_0 = g(z,p)\} \\ &= \{Rz + Qp + r \mid z \in \mathcal{D}(p)\} \end{aligned}$$

[1] Notice that the number of vertices may depend on symbolic parameters p. This is taken into account in [6] by splitting the problem in subdomains relatively to p.

Its dimension is $dim_{Emit_{E,Y}} = Rank(R)$. It is contained in an affine space of dimension $Rank(R)$, whose basis vectors are $\varepsilon_{E,Y,j}$ with $1 \leq j \leq dim_{Emit_{E,Y}}$.

A parallel solution to a set of PARE is characterized by an affine space-time transformation [5]. It is expressed, for each variable X, by a full rank $n \times n$ transformation matrix $T_X = \begin{pmatrix} \lambda_X \\ \sigma_X \end{pmatrix}$ where σ_X is a $d_\sigma \times n$ integer matrix representing the *allocation* on a virtual processor space of dimension d_σ, and λ_X is a $d_\lambda \times n$ integer matrix defining the *multi-dimensional schedule*: $t_X(z) = \lambda_X z + \alpha_X$. The row vectors of σ_X define the basis vectors of the virtual processor space. The allocation function is defined by $alloc_X(z) = \sigma_X z + \gamma_X$. The row vectors of λ_X are called the *schedule vectors*. The schedule must satisfy the causal constraints which are classically expressed for any dependence from Y to X by $t_Y(z_0) \prec t_X(z)$, $\forall z \in Util_{E,Y}(z_0, p)$ where $\prec$ is the lexicographical *lower than* operator.

3 Communication Minimization

Let us now focus on the core of this paper: communication minimization. We classically distinguish between two types of communications: local communications between neighboring processors and distant communications. Local communications can be efficiently implemented on most target architectures whereas distant ones have to be avoided as much as possible.

3.1 How to Eliminate Distant Communications

A communication results from the projection or allocation of a dependence vector on two different virtual processors. For a dependence from Y to X, the resulting communication vector is: $alloc_X(z) - alloc_Y(z_0)$. Using equality $z_0 = Rz + Qp + r$ with $z \in Util_{E,Y}(z_0, p)$, we have:

$$\begin{aligned} alloc_X(z) - alloc_Y(z_0) &= alloc_X(z) - alloc_Y(Rz + Qp + r) \\ &= (\sigma_X - \sigma_Y \cdot R)z - \sigma_Y(Qp + r) - \gamma_Y + \gamma_X \\ &= (\sigma_X - \sigma_Y \cdot R)z + \rho(p) \end{aligned} \tag{1}$$

where $\rho(p) = -\sigma_Y(Qp + r) - \gamma_Y + \gamma_X$ does not depend on z and can therefore be seen as a constant term. From this equation we can deduce a sufficient condition to eliminate distant communications [2, 4, 8]:

Theorem 1 [Strong condition]. *The dependence from Y to X does not induce any distant communication if $\sigma_Y \cdot R = \sigma_X$.*

When this theorem is verified all the utilization points are allocated to the same virtual processor. Since the allocation matrices must be full row rank, this condition can only be satisfied if $Rank(R) \geq d_\sigma$.

One of the results of this paper is to show that the condition given in Theorem 1 is often too strong. As proved in [8], this condition is necessary only in the case of degenerate utilization sets. However it is sufficient but not necessary in the case of non-degenerate utilization sets, i.e. one or multi-dimensional sets. Our objective is to propose for such utilization sets a weaker condition to eliminate distant communications.

In the case of a non-degenerate utilization set, the data computed at a given emission point is used by a convex polyhedron: the utilization set. An efficient way to transmit this data to the whole utilization set is to send it along one of its extremal dependence vectors, and then to send it from neighbor to neighbor using the utilization vectors. Since the utilization vectors are constant, these latter communications are local. Therefore the only distant communications that may occur are the ones induced by the chosen extremal dependence vector d_i. Notice that this *propagation technique* imposes new constraints on the schedule as discussed in section 3.3.

3.2 A Weak Condition to Eliminate Distant Communications

Let us focus on non-degenerate utilization sets and apply the propagation technique. In this case, the only distant communication that may occur is the one related to the chosen integer vertex v_i. This communication is characterized by vector $alloc_X(v_i) - alloc_Y(z_0)$. The necessary condition which eliminates distant communications according to this dependence must guarantee that the projection of this vector on the processor space does not depend on z_0:

$$\begin{aligned} alloc_X(v_i) - alloc_Y(z_0) &= alloc_X(A_i z_0 + B_i p + c_i) - alloc_Y(z_0) \\ &= (\sigma_X \cdot A_i - \sigma_Y) z_0 + \sigma_X \cdot B_i p + \sigma_X c_i + \gamma_X - \gamma_Y \\ &= (\sigma_X \cdot A_i - \sigma_Y) z_0 + \rho'(p) \end{aligned} \tag{2}$$

where $\rho'(p) = \sigma_X \cdot B_i p + \sigma_X c_i + \gamma_X - \gamma_Y$ is a constant function according to z_0.

Theorem 2 [Weak condition]. *The dependence from Y to X does not induce any distant communication if and only if: $(\sigma_X \cdot A_i - \sigma_Y)\, \varepsilon_{E,Y,j} = 0$ for all $j \in [1, dim_{Emit_{E,Y}}]$, where $\varepsilon_{E,Y,j}$ are the basis vectors of the affine space supporting $Emit_{E,Y}(p)$.*

Proof. Let $z_0, z_1 \in Emit_{E,Y}(p)$. We have $z_0 = z_1 + \sum_{j=1}^{d} (\alpha_j\, \varepsilon_{E,Y,j})$ where $\alpha_j \in \mathbb{Z}$ and d denotes $dim_{Emit_{E,Y}}$. From equation (2) we deduce that there is no distant communication if and only if $(\sigma_X \cdot A_i - \sigma_Y) z_0$ is constant, $\forall z_0 \in Emit_{E,Y}(p)$. Therefore we have: $(\sigma_X \cdot A_i - \sigma_Y)(z_0 - z_1) = 0$

$$\Leftrightarrow\ (\sigma_X \cdot A_i - \sigma_Y) \sum_{j=1}^{d} (\alpha_j\, \varepsilon_{E,Y,j}) = 0 \quad \forall \alpha_j \in \mathbb{Z}$$
$$\Leftrightarrow\ (\sigma_X \cdot A_i - \sigma_Y)\, \varepsilon_{E,Y,j} = 0 \qquad \forall 1 \le j \le d$$

□

3.3 Constraints on the Schedule

The whole process of space-time transformation requires the computation of both a schedule function and an allocation function. Since we deal with communication minimization, our first concern is the allocation as discussed above.

However, in order to obtain a valid space-time transformation the determination of both schedule and allocation functions must be conducted simultaneously. When applying the strong condition (Theorem 1) each utilization set is projected onto a single virtual processor and therefore its points have to be scheduled at successive time steps. However this condition is already expressed by the fact that transformation matrices are full rank and there are no new constraints on the schedule in this case. Conversely in the case of the propagation technique new constraints on the schedule have to be stated as discussed hereunder. They are added to the initial causal constraints.

The propagation technique consists in sending the data from the virtual processor associated with z_0 to the processor associated with a chosen vertex v_i and then to all the processors related to the other utilization points. Therefore the execution time step related to vertex v_i must be less than or equal to any of the other utilization points. This results into the following theorem.

Theorem 3. *The propagation related to vertex v_i of $Util_{E,Y}(z_0, p)$ can be applied only if the schedule function satisfies: $\lambda_k \cdot \overline{u_j} \geq 0$, $\forall k \in [1, d_\lambda]$, where $\overline{u_j}$ denote the set of generating vectors of the smallest cone whose origin is v_i and which contains the utilization set.*

Notice that vectors $\overline{u_j}$ are the vectors supported by the edges of $Util_{E,Y}$ at vertex v_i. They do not depend on parameters z_0 and p, as proved in [6].

Theorem 3 guarantees the existence of a valid schedule. It does not necessarily avoid broadcast communications. As classically stated in the literature, there is a broadcast communication along $\overline{u_j}$ if and only if $\lambda_k \cdot \overline{u_j} = 0$, $\forall k \in [1, d_\lambda]$.

3.4 Special cases

Full rank index matrix. When index matrix R is full rank the dependence is degenerate and there is only one utilization point z related to a given emission point $z_0 = Rz + Qp + r$. Therefore, point z is vertex $v_1 = A_1 z_0 + B_1 p + c_1$. Since R is full rank we have $z = R^{-1} z_0 - R^{-1} \cdot Qp - R^{-1} r = A_1 z_0 + B_1 p + c_1$. This implies $A_1 = R^{-1}$. In this case, Theorem 2 can be rewritten as:

$$(\sigma_X \cdot R^{-1} - \sigma_Y)\, \varepsilon_{E,Y,j} = 0 \quad \forall j \in [1, dim_{Emit_{E,Y}}].$$

If $dim_{Emit_{E,Y}} = n$, the basis vectors $\varepsilon_{E,Y,j}$ generate the whole n-dimensional space, and the above condition is equivalent to $\sigma_X - \sigma_Y \cdot R = 0$ and it proves that in this particular case, the strong condition *is necessary* as proved in [8].

Self-dependence. In the case of an self-dependence from a given variable X to itself, the communication vector is:

$$alloc_X(z) - alloc_X(z_0) = (\sigma_X - \sigma_X \cdot R)z + \rho(p).$$

The strong condition to avoid distant communications is $\sigma_X \cdot (R - Id) = 0$. This condition can only be verified if $Rank(R - Id) \leq n - d_\sigma$.

Recall that this condition allocates all the utilization points to the same virtual processor. The alternative to avoid distant communications is to use the propagation technique. It works in the same way as a cross-dependence from Y to

X and is based on the choice of an integer vertex v_i. Conditions on the schedule are also given by Theorem 3. The weak condition on the allocation matrix (given in Theorem 2) simplifies into: $\sigma_X \cdot (A_i - Id)\,\varepsilon_{E,Y,j} = 0$, $\forall j \in [1, dim_{Emit_{E,Y}}]$.

Notice that in the case of a uniform self-dependence, $R = Id$ and there is no distant communication, since Theorem 2 simplifies into $\sigma_X = \sigma_X$.

4 A Heuristic for Communication Optimization

A system of affine recurrence equations is of course characterized by several dependences which most of the time are incompatible in terms of communication elimination. In many practical cases, communication-free solutions result in only one virtual processor. Therefore, our objective is to propose a heuristic which:

– first eliminates as many distant communications as possible, using either the strong or the weak condition.

– then reduces as much as possible the remaining local communications by zeroing the constant terms $\rho(p)$ or $\rho'(p)$.

This is done by successively analyzing the dependences. They are ordered so as to deal first with the dependences which could result in the largest number of distant communications. Recall that uniform self-dependences always result in local communications and are therefore taken into account after all the affine and cross dependences.

As it has been proved in this paper, the number of distant communications accountable to one dependence is in the worst case equal to $Card(Emit_{E,Y}(p))$. Therefore, the dependences are ordered according to $dim_{Emit_{E,Y}}$.

The algorithm we propose consists in building a tree of solutions. Each level of the tree corresponds to one dependence; each node on a level defines one set of constraints on the schedule and the allocation related to the dependence for a given strategy of communication minimization. The root of the tree defines the set of causal constraints on the schedule and the set of *compatibility constraints* between allocation and schedule, in order to ensure that the transformation matrices are full rank. The constraints are propagated from ancestors to descendants. A given node contains all the constraints of its direct ancestor plus the ones corresponding to the strategy used at this node. The intersection of all the constraints is determined, and if it has no solution the node is removed. All the paths in the tree have the same length. The leaves represent the various parallel solutions to the PARE.

In order to estimate the efficiency of the various solutions, the algorithm determines at each step the *volume* of distant and local communications in terms of dimensions. A similar approach based on communication volume is used in [4]. The volume is recorded in two arrays where any entry d corresponds to the number of d-dimensional communications ($1 \le d \le n$). The strategies applied at each node to build its direct descendants related to a given dependence are:

1a. **strong condition and** $\rho(p)$**=0** (Theorem 1 and Eq. (1)). Creation of a node with $\sigma_Y \cdot R = \sigma_X$ and $\rho(p) = 0$ as new constraints if they can be satisfied.

The resulting allocation function projects onto the same virtual processor all the utilization points and their corresponding emission points. There are therefore no new communications. Notice that the initial constraints on the schedules (causal and compatibility constraints) ensure that they are valid relatively to these allocations.

1b. **strong condition** (Theorem 1) alone. If $\sigma_Y \cdot R = \sigma_X$ can be satisfied, a new node is created, with one new set of local communications of dimension dim_{Emit} due to $\rho(p)$.

2a. **weak condition** successively applied to each integer vertex v_i of the utilization set, and **elimination of the local communication due to** $\rho'(p)$ (Eq. (2)). The new constraints on the allocation are given by Theorem 2, and by condition $\rho'(p) = 0$. The new constraints on the schedule are given by Theorem 3. If they can all be satisfied, a new node is created with all these new constraints and with a new set of local communication of dimension $(dim_{Emit} + dim_{Util})$ due to the propagation along the utilization vectors.

2b. **weak condition** successively applied to each integer vertex v_i of the utilization set. The new constraints on the allocation are given by Theorem 2 and the new constraints on the schedule by Theorem 3. If they can be satisfied, a new node is created with two new sets of local communication: the first one due to $\rho'(p)$ of dimension dim_{Emit} and the second one of dimension $(dim_{Emit} + dim_{Util})$ resulting from the propagation along the utilization vectors.

3. **no constraints**. Creation of a new node without adding any new constraints, with a new set of distant communication of dimension dim_{Emit}, and a new set of local communication of dimension $(dim_{Emit} + dim_{Util})$. This node ensures that at least one space-time transformation exists.

5 An Example

Let the following PARE depending on parameter N be mapped onto a two dimensional array of virtual processors using a unique schedule function.

$$X[i,j,k] = Y[j,i-1,0] + 2 \quad D_{(3)} = \{i,j,k \mid 0 \le j,k \le N, 1 \le i \le N\} \quad (3)$$
$$Y[i,j,k] = X[0,0,k-1]/5 \quad D_{(4)} = \{i,j,k \mid 0 \le i,j \le N, 1 \le k \le N\} \quad (4)$$

Since the resulting processor array is two dimensional, the schedule is one dimensional. Let $\sigma_X = (x_1, x_2, x_3)$ and $\sigma_Y = (y_1, y_2, y_3)$ where x_i and y_i are 2-dimensional column vectors. Let $\lambda_X = \lambda_Y = \lambda = (\lambda_1, \lambda_2, \lambda_3)$ and $\alpha_X = \alpha_Y$. When restricting the solutions to constant (not depending on N) schedule vectors, the causal constraints are: $\lambda_1 = \lambda_2 > 0, \lambda_3 > 0$.

Utilization set $Util_{(3),Y}$ has dimension 1 and has therefore two vertices, while $Util_{(4),X}$ has dimension 2 and four vertices. These vertices depend on parameters $z_0 = (i_0, j_0, k_0)$ and N. In order to apply the propagation technique, new conditions on the schedule (related to each chosen vertex) must be satisfied as stated by Theorem 3. These new conditions are not always compatible with the causal constraints. For utilization set $Util_{(3),Y}$, propagation is possible only for vertex $v_1 = (j_0 + 1, i_0, 0)$ and the new constraint on the schedule is $\lambda_3 \geq 0$. For utilization set $Util_{(4),X}$, propagation is possible only for vertex $v_2 = (0, 0, k_0 + 1)$ and the new constraints on the schedule are $\lambda_1, \lambda_2 \geq 0$.

The following table describes the new set of constraints associated with the strong and weak conditions respectively given by Theorems 1 and 2. The constraints to be verified for eliminating all communications due to $\rho(p)$ and $\rho'(p)$ result respectively from equation (1) and (2). The new constraints on the schedule imposed by the propagation technique are included in the causal constraints and are therefore not reported.

dep.	method	new constraints (case b.)	new constraints to eliminate the communications due to $\rho(p)$ or $\rho'(p)$ (case a.)
$(3), Y$	strong cond. (case 1.)	$x_1 = y_2$ $x_2 = y_1$ $x_3 = 0$	$\gamma_Y = \gamma_X + y_2$
	weak cond. (v_1) (case 2.)	$x_1 = y_2$ $x_2 = y_1$	$\gamma_Y = \gamma_X + x_1$
$(4), X$	strong cond. (case 1.)	$y_1 = 0$ $y_2 = 0$ $y_3 = x_3$	$\Rightarrow \nexists\, \sigma_Y$ full row $\Rightarrow$ the strong condition can not be satisfied
	weak cond. (v_1) (case 2.)	$x_3 = y_3$	$\gamma_X = \gamma_Y + y_3$

Computation of the tree of solutions. Notice that there is only one incompatibility between the constraints in the table above: $\gamma_Y = \gamma_X + y_2$ corresponding to the strong condition in dependence $[(3), Y]$ is incompatible with $\gamma_X = \gamma_Y + y_3$ from the weak condition in dependence $[(4), X]$. Since $x_3 = y_3$ and $x_3 = 0$ we have $y_3 = -y_2 = x_3 = 0$ and there is no full row rank matrix σ_Y satisfying the equalities $y_3 = y_2 = 0$.

The tree of solution is represented figure 1. The volume of communication is computed at each node of the tree by two vectors representing respectively the dimensions (1, 2 or 3) of the sets of distant and local communications as described section 4. The *no constraint* leaf is only informative. It shows that the worst case would result in one set of 1-dimensional distant communications, one set of 2-dimensional distant communications and two sets of 3-dimensional local communications.

The leftmost leaf in figure 1 is the best solution. It results in only two sets of local communications, one of dimension 3 and the other of dimension 1. Its constraints are: the causal constraints $\lambda_1 = \lambda_2 > 0, \lambda_3 > 0$ and the constraints on the allocation $\sigma_X = (x_1, x_2, 0), \sigma_Y = (x_2, x_1, 0), \gamma_X = \gamma_Y + x_1$ with x_1 and x_2 two independent 2-dimensional vectors. A possible solution is:
$\sigma_X = \begin{pmatrix} 1\,0\,0 \\ 0\,1\,0 \end{pmatrix}$, $\gamma_X = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $\sigma_Y = \begin{pmatrix} 0\,1\,0 \\ 1\,0\,0 \end{pmatrix}$, $\gamma_Y = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, $\lambda = (1, 1, 1)$ and $\alpha = 0$.

This example clearly shows that the weak condition is useful. Hence the strong condition for the second dependence can not be satisfied in the case of a

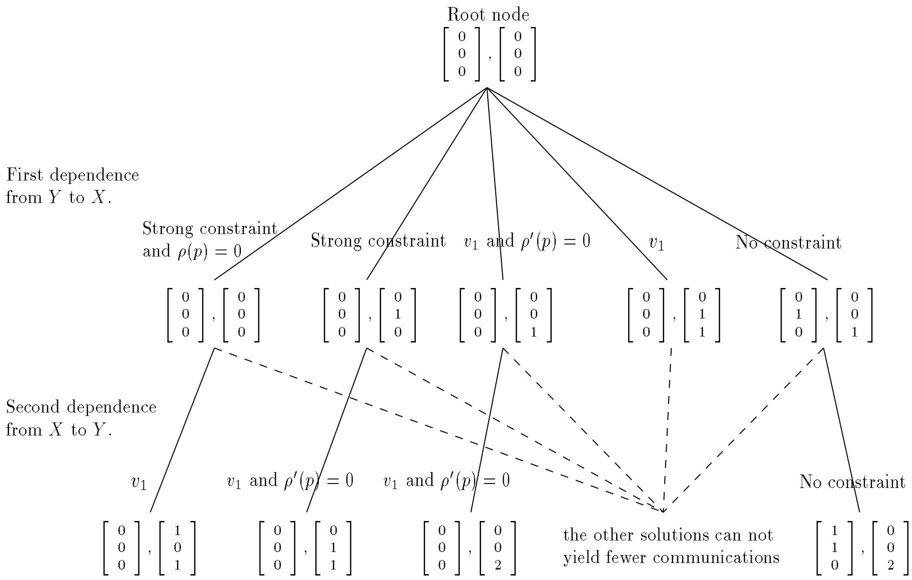

Fig. 1. The tree of solutions

2-dimensional processor array. If the strong condition was the only method to remove distant communications, there would be no way to localize the second dependence.

6 Conclusion

We have proposed a heuristic to optimize communications resulting in a set of constraints on the allocation and schedule functions. Besides the strong condition which is classical in other approaches, we have introduced a weak condition. It is a less restrictive condition using the intrinsic information of the utilization sets. It offers more flexibility to the communication optimization problem and results in a wider set of solutions. As seen section 5 on the example it may be successful when the strong condition fails. This condition is a generalization of the well-known uniformization problem [10]. The uniformization process consists in transforming at the equation level the affine dependences into uniform ones. The propagation technique can be seen as a less restrictive uniformization approach. It works at the solution level and moreover it does not enforce a full uniformization: only the most communication-expensive dependences are partially uniformized.

Other works focus on this question often referred as *placement* or *alignment* problem. Since this problem has been proved to be NP-complete [2] various heuristics have been proposed. Some works do not consider the owner compute rule and therefore allocate independently instructions and data. In [3], Darte and

Robert introduced the notion of communication graph and proposed a method to optimize communications in uniforms loop nests. This approach has been extended to affine loop nests by Dion and Robert in [2], using a more precise graph called the access graph. Both approaches use a condition similar to the strong condition presented in this paper. Other heuristics are given with respect to the owner compute rule. In [4] Feautrier proposes a greedy algorithm that cuts the edges of the data flow graph according to their volume of communication.

As mentioned in this paper, allocation and schedule are strongly related since the transformation matrices must be full rank. This important concept is not taken into account by these various approaches which deal exclusively with the allocation function. Conversely, our heuristic integrates both the derivation of the schedule and allocation functions. As it has been proved in section 3.3 this is particularly important in order to apply the propagation technique where new conditions on the schedule have to be added to the causal constraints.

The heuristic we propose can be refined in several ways. In the current version of the algorithm the volume of communication is estimated using the dimensions of the utilization and emission sets. It could be computed exactly as function of the parameters using the Ehrhart algorithm [1]. The current version only applies the propagation technique to integer vertices of the utilization sets. This should be extended to deal with rational vertices. The propagation technique only focus on the chosen integer vertex. In order to reduce the number of local communications related to the utilization points, we could apply well known techniques to further reduce this number of local communications [7]. They define a set of constraints between the allocation matrices and the utilization vectors.

References

1. Clauss, Ph., Loechner, V.: Parametric Analysis of Polyhedral Iteration Spaces. *ASAP'96*, Chicago, Illinois, August 1996.
2. Dion, M., Robert, Y.: Mapping affine loop nests: New results. *HPCN'95*, Milan, LNCS 919, 184–189, 1995.
3. Darte, A., Robert, Y.: A graph-theoretic approach to the alignment problem. *Parallel Processing Letters.* **4(3)**, 1994.
4. Feautrier, P.: Towards automatic distribution. *Parallel Processing Letters.* **4(3)**: 233–244, 1994.
5. Lengauer, Ch.: Loop parallelization in the polytope model. *CONCUR'93*, LNCS 715, 398–416, 1993.
6. Loechner, V., Wilde, D. K.: Parameterized polyhedra and their vertices. Research Report ICPS 96-09. `http://icps.u-strasbg.fr/pub-96/`
7. Mongenet, C.: Mappings for communication minimization using distribution and alignment. *PACT'95*, Limassol, Cyprus, ACM Press, 185-193, 1995.
8. Mongenet, C.: Affine dependence classification for communications minimization. Research Report ICPS 96-08. `http://icps.u-strasbg.fr/pub-96/`
9. Mongenet, C., Clauss, Ph., Perrin, G. R.: A geometrical coding to compile affine recurrence equations on regular arrays. *IPPS'91*, Anaheim, California, 1991.
10. Quinton, P., Van Dongen, V.: The Mapping of Linear Recurrence Equations on Regular Arrays. *Journal of VLSI Signal Processing,* **1**:95–113, 1989.