Decomposition of Energy Spaces and Applications

Roland GLOWINSKI, Jacques-Louis LIONS, and Olivier PIRONNEAU

R. G. : Department of Mathematics, University of Houston, Houston, Texas 77204, USA;

J.-L. L. : Collège de France, 3 Rue D'Ulm, 75231 Paris cedex 05, France;

O. P. : Laboratoire D'Analyse Numérique, Université P. et M. Curie, Tour 55-65, 5^e étage, 4 Place Jussieu, 75252 Paris cedex 05, France.

Abstract. In this note we discuss the solution of linear variational problems in Hilbert spaces V, such that $V = \sum_{i=1}^m V_i$, with the possibility that $V_i \cap V_j \neq \{0\}$ if $i \neq j$. We take advantage of the above decomposition of V to solve the linear variational problems by conjugate gradient algorithms operating in $\prod_{i=1}^m V_i$. For some situations, the method discussed in this note leads to novel domain decomposition methods with good convergence and parallelization properties. We conclude this note with the results of numerical experiments concerning the solution of elliptic problems on an L-shaped domain considered as the union of two overlapping rectangles and on a two-dimensional domain which is the union of a rectangle and a disk.

Sur la Décomposition d'Espaces d'Energie. Applications

Résumé. Dans cette note on étudie la résolution de problèmes variationnels linéaires dans des espaces d'Hilbert V, tels que $V = \sum_{i=1}^m V_i$, avec la possibilité que $V_i \cap V_j \neq \{0\}$ si $i \neq j$. Sur la base de la décomposition ci-dessus ou peut résoudre les problèmes variationnels linéaires par des algorithmes de gradient conjugué opérant dans $\prod_{i=1}^m V_i$. Dans certaines situations la méthode décrite dans cette note conduit à des méthodes de décomposition de domaines, d'un type nouveau (à notre connaissance tout au moins). On conclut cette note par la présentation de résultats numériques relatifs à la résolution de problèmes elliptiques sur un domaine en forme de L, considéré comme la réunion, avec recouvrement, de deux rectangles et sur un domaine bi-dimensionnel qui est la réunion d'un rectangle et d'un disque.

1 Space decomposition for linear variational problems

Let V be a real Hilbert space (for the scalar product $(\cdot,\cdot)$ and the associated norm $\|\cdot\|$), the so-called *energy space*. In V we consider the following linear variational problem

(1) $$u \in V;\ a(u,v) = L(v), \forall v \in V,$$

where $a : V \times V \to \mathbb{R}$ is bilinear, continuous, and *coercive* (i.e., there exists $\alpha > 0$ so that

Note présentée par Roland Glowinski.

$a(v,v) \geq \alpha \|v\|^2$, $\forall v \in V$) and where $L : V \to \mathbb{R}$ is linear and continuous. We do not suppose that $a(\cdot,\cdot)$ is symmetric. It is well known that if the above conditions on V, a, and L hold, then problem (1) has a unique solution.

Remark 1.1. As shown in [1], the linearity of (1) is *by no means* necessary to what follows. □

Let us assume now that the space V can be *decomposed* as

$$V = V_1 + V_2 + \cdots + V_m, \tag{2}$$

where for $i = 1, \ldots, m$, V_i is a closed subspace of V. Decomposition (2) means that, $\forall v \in V$, there exists $v_i \in V_i$ such that

$$v = v_1 + v_2 + \cdots + v_m. \tag{3}$$

Decomposition (3) *is not unique* if it happens that

$$V_i \cap V_j \neq \{0\},\ for\ some\ pairs\ \{i,j\},\ i \neq j. \tag{4}$$

We want now to find solution methods for problem (1) taking advantage of decomposition (2).

Remark 1.2. Examples of decompositions of the energy space V as in (2) abund. In finite dimension (the final situation for applications) the V_i's may correspond, for example, to different triangulations of the domain Ω where (1) is considered. They can also correspond to a *domain decomposition* (see Section 3) or to situations where one *mixes* finite element and spectral methods. □

Next, we introduce a family of bilinear functionals $s_i(\cdot,\cdot)$ with the following properties

$$s_i(\cdot,\cdot)\ is\ continuous,\ symmetric\ and\ coercive. \tag{5}$$

Remark 1.3. ¿From a theoretical point of view, $s_i(\cdot,\cdot)$ can be *any* bilinear functional satisfying (5). In practice the $s_i(\cdot,\cdot)$ will be chosen so that the iterative methods (of the preconditioned conjugate gradient type) discussed in Section 2 will have good convergent properties. These notions will appear more clearly with the computational experiments discussed in Section 3. □

Let denote by W the product space $\prod_{i=1}^m V_i$. To each $\boldsymbol{\mu} = \{\mu_i\}_{i=1}^m \in W$ we associate $\mathbf{u}(\boldsymbol{\mu}) = \{u_i(\boldsymbol{\mu})\}_{i=1}^m \in W$ obtained from $\boldsymbol{\mu}$ via the solution of the *virtual state system*

$$\begin{cases} u_i(\boldsymbol{\mu}) \in V_i, \\ s_i(u_i(\boldsymbol{\mu}) - \mu_i, v_i) + a(\sum_{j=1}^m \mu_j, v_i) = L(v_i),\ \forall v_i \in V_i, \\ i = 1, \ldots, m. \end{cases} \tag{6}$$

The m equations in (6) can be solved in parallel and the *simplest* $s_i(\cdot,\cdot)$ *is the simplest they are to solve*. Given the *virtual control* $\boldsymbol{\mu} = \{\mu_i\}_{i=1}^m \in W$, system (6) defines a *unique state* vector $\{u(\boldsymbol{\mu})\}_{i=1}^m \in W$. *The method of virtual control* ([2], [3]) *consists then in choosing (if possible) the* μ_i*'s in such a manner that one recovers the solution* u of (1) from the $\{u_i(\boldsymbol{\mu}_i)\}_{i=1}^m$. This approach has been introduced in [2], [3] for various types of decompositions (including domain decomposition). Here, the choice of $\{\mu_i\}_{i=1}^m$ satisfying the above requirements is simple: it suffices to find a *fixed point* of the mapping

$$\boldsymbol{\mu} \to \{u(\boldsymbol{\mu})\}_{i=1}^m\ : W \to W. \tag{7}$$

Indeed, if $\boldsymbol{\lambda} = \{\lambda_i\}_{i=1}^m$ is such a fixed point, we have $u_i(\boldsymbol{\lambda}) = \lambda_i$, $\forall i = 1, \cdots, m$, and (6) reduces to

$$a(\sum_{j=1}^m \lambda_j, v_i) = L(v_i),\ \forall v_i \in V_i,\ \forall i = 1, \ldots, m; \tag{8}$$

since $V = \sum_{i=1}^{m} V_i$, it follows from (8) that

(9) $$a(\sum_{j=1}^{m} \lambda_j, v) = L(v),\ \forall v \in V,$$

which implies that $\sum_{j=1}^{m} \lambda_j$ is the solution of problem (1).

Remark 1.4. If (4) takes place, mapping (7) *does not have a unique fixed point* and indeed any $\boldsymbol{\lambda} = \{\lambda_i\}_{i=1}^{m} \in W$ such that $\sum_{j=1}^{m} \lambda_j = u$, with u solution of problem (1), is a fixed point of mapping (7). □

It is worth observing that the above conditions can also be thought of in the framework of *least square methods* as in, e.g., [4], [5]. Indeed, let us define $\mathbf{y} = \{y_i\}_{i=1}^{m}$ by

(10) $$y_i = \mu_i - u_i(\boldsymbol{\mu}),\ \forall i = 1, \dots, m,$$

then the y_i's are solution of

(11) $$\begin{cases} \forall i = 1, \dots, m,\ y_i \in V_i \ and \\ s_i(y_i, v_i) = a(\sum_{j=1}^{m} \mu_j, v_i) - L(v_i),\ \forall v_i \in V_i. \end{cases}$$

Finding a fixed point of application (7) amounts to finding $y_i = y_i(\boldsymbol{\lambda})$, such that $y_i(\boldsymbol{\lambda}) = 0$, $\forall i = 1, \dots, m$, which can be defined "*constructively*" by *minimizing the least-squares functional* (virtual cost function!)

(12) $$J(\boldsymbol{\mu}) = \ 1/2 \sum_{i=1}^{m} s_i(y_i)$$

(where $s_i(y_i) = s_i(y_i, y_i)$), and where $y_i = y_i(\boldsymbol{\mu})$ is the solution of (11). Apparently, there is still a "catch" if situation (4) holds; no matter what we do, the fixed point of application (7) is not *unique*. Therefore the minimization problem

(13) $$\boldsymbol{\lambda} \in W,\ J(\boldsymbol{\lambda}) \leq J(\boldsymbol{\mu}),\ \forall \boldsymbol{\mu} \in W$$

has infinity many solutions. We want to find an algorithm which converges, as quickly as possible, to one of these solutions. The conjugate gradient algorithm discussed in Section 2, hereafter, will have these convergence properties.

Remark 1.5. Given v in V, there is a *unique* decomposition

$$v = \sum_{i=1}^{m} v_i,\ v_i \in V_i,\ \forall i = 1, \dots, m$$

which minimizes $\Phi : W \to \mathbb{R}$ continuous, strictly convex and coercive. Functional J in (12) corresponds to

(14) $$\Phi(v_1, \dots, v_m) = \frac{1}{2} \sum_{i=1}^{m} s_i(v_i).$$

Other choices are possible, such as

(15) $$\Phi(v_1, \dots, v_m) = \frac{1}{2}[s_1(v_1) + s_2(v_2 - v_1) + \dots + s_m(v_m - (v_1 + v_2 + \dots + v_{m-1}))]$$

which may be of practical interest to derive multigrid-like algorithms.

2 A space decomposition based conjugate gradient algorithm for the solution of problem (1)

In order to solve problem (1) via the *least-squares formulation*

(16) $$\boldsymbol{\lambda} \in W,\ J(\boldsymbol{\lambda}) \le J(\boldsymbol{\mu}),\ \forall \boldsymbol{\mu} \in W,$$

where J defined by (11), (12), we need to know the differential $J'(\boldsymbol{\mu})$ of J at $\boldsymbol{\mu}$, $\forall \boldsymbol{\mu} \in W$. Using a simple *perturbation analysis* shows that

(17) $$\delta J(\boldsymbol{\mu}) =< J'(\boldsymbol{\mu}), \delta\boldsymbol{\mu} >= \textstyle\sum_{i=1}^m s_i(y_i, \delta y_i),$$

(18) $$s_i(v_i, \delta y_i) = a(\textstyle\sum_{j=1}^m \delta\mu_j, v_i), \forall v_i \in V_i,\ \forall i = 1, \ldots, m,$$

where $< \cdot, \cdot >$ denotes the duality pairing between the dual space W' of W and W itself. Taking $v_i = y_i$ in (18) and comparing to (17) shows that , $\forall \boldsymbol{\mu}, \mathbf{v} \in W$, we have

(19) $$< J'(\boldsymbol{\mu}), \mathbf{v} >= a(\sum_{i=1}^m v_i, \sum_{i=1}^m y_i) = a^*(\sum_{i=1}^m y_i, \sum_{i=1}^m v_i).$$

A conjugate gradient algorithm for the solution of (16) is as follows:

(20) $$\boldsymbol{\lambda}^0 = \{\lambda_i^0\}_{i=1}^m\ is\ given\ in\ W\ (\boldsymbol{\lambda}^0 = 0,\ for\ example);$$

$\forall i = 1, \ldots, m$, *we solve*

(21) $$y_i^0 \in V_i;\ s_i(y_i^0, v_i) = a(\textstyle\sum_{j=1}^m \lambda_j^0, v_i) - L(v_i),\ \forall v_i \in V_i,$$

(22) $$g_i^0 \in V_i;\ s_i(g_i^0, v_i) = a(v_i, \textstyle\sum_{j=1}^m y_j^0), \forall v_i \in V_i,$$

and we set

(23) $$\mathbf{g}^0 = \{g_i^0\}_{i=1}^m, \mathbf{w}^0 = \mathbf{g}^0.$$

For $n \ge 0$, $\boldsymbol{\lambda}^n$, $\mathbf{g}^n$, $\mathbf{w}^n$ *being known we compute* $\boldsymbol{\lambda}^{n+1}$, $\mathbf{g}^{n+1}$ *and, if necessary,* $\mathbf{w}^{n+1}$ *as follows:*
We solve, for $i = 1, \ldots, m$,

(24) $$\bar{y}_i^n \in V_i;\ s_i(\bar{y}_i^n, v_i) = a(\sum_{j=1}^m w_j^n, v_i),\ \forall v_i \in V_i,$$

(25) $$\bar{g}_i^n \in V_i;\ s_i(\bar{g}_i^n, v_i) = a(v_i, \sum_{j=1}^m \bar{y}_j^n),\ \forall v_i \in V_i,$$

and set

(26) $$\rho_n = \sum_{i=1}^{m} s_i(g_i^n) / \sum_{i=1}^{m} s_i(\bar{g}_i^n, w_i^n),$$

(27) $$\lambda_i^{n+1} = \lambda_i^n - \rho_n w_i^n,\ \forall i = 1, \ldots, m,$$

(28) $$g_i^{n+1} = g_i^n - \rho_n \bar{g}_i^n,\ \forall i = 1, \ldots, m.$$

If $\sum_{i=1}^{m} s_i(g_i^{n+1}) / \sum_{i=1}^{m} s_i(g_i^0) \le \epsilon$*, take* $u = \sum_{i=1}^{m} \lambda_i^{n+1}$*; else compute*

(29) $$\gamma_n = \sum_{i=1}^{m} s_i(g_i^{n+1}) / \sum_{i=1}^{m} s_i(g_i^n),$$

(30) $$w_i^{n+1} = g_i^{n+1} + \gamma_n w_i^n,\ \forall i = 1, \ldots, m.$$

Do $n = n + 1$ *and return to (24).*

In order to discuss the convergence of algorithm (20)-(30) we introduce $W_0 \subset W$ defined by

(31) $$W_0 = \{\mathbf{v} \mid \mathbf{v} = \{v_i\}_{i=1}^m \in W,\ \sum_{i=1}^{m} v_i = 0\},$$

and denote by W_1 the orthogonal of W_0 in W for the scalar product

(32) $$\{\mathbf{v}, \mathbf{w}\} \to \sum_{i=1}^{m} s_i(v_i, w_i).$$

We have then $W = W_0 \oplus W_1$ and we can prove that *the functional* J *defined by (11), (12) is strictly convex and coercive over* W_0. From these properties and from general results on the convergence of conjugate gradient algorithms (see, e.g., [6]) we can show that

(33) $$\lim_{n \to +\infty} \boldsymbol{\lambda}^n = \mathbf{u} + \boldsymbol{\lambda}_0^0,$$

where, in (33), $\mathbf{u}$ is the unique element of W_1 so that $\sum_{i=1}^{m} u_i = u$ and where $\boldsymbol{\lambda}_0^0$ is the component of $\boldsymbol{\lambda}^0$ belonging to W_0 in the decomposition $\boldsymbol{\lambda}^0 = \boldsymbol{\lambda}_0^0 + \boldsymbol{\lambda}_1^0$ ($\boldsymbol{\lambda}_k^0 \in W_k$, $k = 0, 1$). Starting from $\boldsymbol{\lambda}^0 = \mathbf{0}$ implies that $\lim_{n \to +\infty} \boldsymbol{\lambda}^n = \mathbf{u}$.

Remark 2.1. Algorithm (20)-(30) is well suited to parallel computing.

Remark 2.2. The methods discussed in this note are general and apply to systems of partial differential equations, nonlinear problems; such generalization are under investigation and will be presented in [1].

3 Numerical experiments

3.1 A first test problem

Let $\Omega \subset \mathbb{R}^2$ be the union of the rectangles $\Omega_1 = (0, 2) \times (0, 1)$ and $\Omega_2 = (0, 1) \times (0, 2)$. In order to solve in $H_0^1(\Omega)$, the *Poisson-Dirichlet* problem

(34) $$-\triangle u = f \ in\ \Omega,\ \ u = 0\ on\ \partial\Omega$$

(whose variational formulation is given by

$$u \in H_0^1(\Omega); \int_\Omega \nabla u \cdot \nabla v \, d\mathbf{x} = \int_\Omega f v \, d\mathbf{x}, \; \forall v \in H_0^1(\Omega)), \tag{35}$$

by the methodology discussed in Sections 1 and 2, we observe that, for example, $H_0^1(\Omega) = H_0^1(\Omega_1) \cup H_0^1(\Omega_2)$. Since $H_0^1(\Omega_1) \cap H_0^1(\Omega_2) = H_0^1(\Omega_1 \cap \Omega_2)$, we clearly have $H_0^1(\Omega_1) \cap H_0^1(\Omega_2) \neq \{0\}$. In order to apply the techniques of Sections 1 and 2 we take

$$V = H_0^1(\Omega), \; and \, \forall i = 1, 2, \; V_i = H_0^1(\Omega_i), \; s_i(v_i, w_i) = \int_{\Omega_i} \nabla v_i \cdot \nabla w_i \, d\mathbf{x}, \; \forall v_i, w_i \in V_i. \tag{36}$$

Next we approximate (35) using a classical finite element method based on piecewise linear approximations and on the triangulation $\mathcal{T}_h$ shown in Figure 1; the subproblems associated to Ω_i are approximated using the restriction of $\mathcal{T}_h$ to each $\overline{\Omega_i}$. The decomposition approach discussed in Sections 1 and 2 can also be applied to the finite element approximation of (35). Let us observe that if the trapezoidal rule is employed to compute the contribution of f in (34), (35), the discrete analogue of problem (34), (35) corresponds to the finite difference discretization of (34) by the classical five point formula. The same comment applies to the subproblems, taking place in the Ω_i's. From this remark the four discrete elliptic problems to solve at each iteration of algorithm (20)-(30) can be solved by *fast Poisson solvers* based on *cyclic reduction* (see [7] for a discussion of cyclic reduction methods). In the following we denote by h the length of the edges adjacent to the right angle of the triangles of $\mathcal{T}_h$. Concerning the convergence of algorithm (20)-(30) we have taken $\epsilon = 10^{-14}$ for stopping criterion. If one takes $f = 1$ in (34), (35), algorithm (20)-(30) converges in 7 iterations, for $h = 1/32$, $1/64$, $1/128$. The graph of the solution has been visualized on Figure 2 and the contours on Figure 3.

3.2 A second test problem

Let $\Omega \subset \mathbb{R}^2$ be the union of $\Omega_1 = \{\mathbf{x} \mid \mathbf{x} \in \mathbb{R}^2, \; \|\mathbf{x}\| < 1\}$ and $\Omega_2 = (0,2) \times (0,1)$. On Ω we consider the following *non-symmetric* (except if $\alpha = 0$) *elliptic problem*:

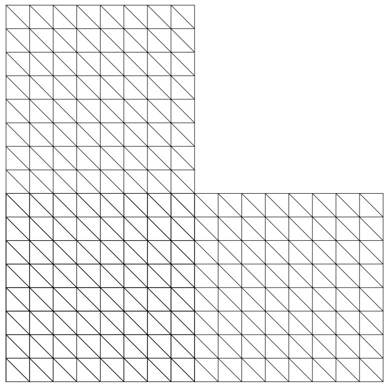
$$-\triangle u + \alpha \frac{\partial u}{\partial x_1} = 1 \; in \; \Omega, \;\; u = 0 \; on \; \partial\Omega, \tag{37}$$

for $\alpha = 0, 1, 2, 3$. The two parts of Ω are triangulated independently implying that the meshs differ on $\Omega_1 \cap \Omega_2$ (see Figure 4, top left and right grids). Algorithm (20)-(30) with $s_i(\cdot, \cdot)$ defined as in (36) has been used to solve problem (37). Figure 5 shows the convergence of the method for the four values of α. The convergence is fast and marginally affected by the non-symmetry of the operator in (37). The computed solutions are very close to those obtained without decomposition, using the bottom grid on Figure 4. The local solutions obtained via the decompositions of Ω and $H_0^1(\Omega)$ have been visualized on Figure 6 and their sum as well.

References

[1] Glowinski, R., Lions, J.L., Pan, T.-W., Pironneau, O., Decomposition of energy spaces, virtual control and applications (to appear).

[2] Lions, J.L., Pironneau, O., 1998. Algorithmes parallèles pour la solution de problèmes aux limites, *C.R. Acad. Sci. Paris*, t. 327, Série 1, pp. 947-952.



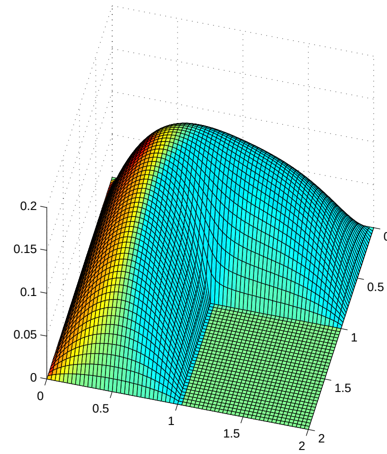


Figure 1: Figure 1: Triangulation of Ω ($h = 1/8$) for the 1^{st} test problem. Figure 2: Graph of the solution of the 1^{st} test problem ($h = 1/32$).

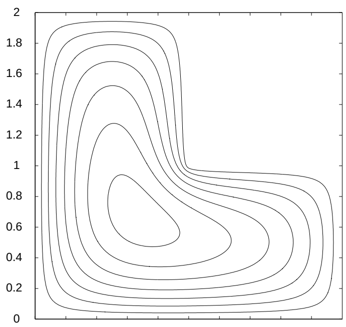
[3] Lions, J.L., Pironneau, O. 1999. Domain decomposition methods for CAD, *C.R. Acad. Sci. Paris*, t. 328, Série 1, pp. 73-80.

[4] Bristeau, M.O., Glowinski, R., Periaux, J., Pironneau, O., Poirier, G., 1985. On the numerical solution of nonlinear problems in fluid dynamics by least squares and finite element methods (II). Application to transonic flow simulations, *Comput. Methods Appl. Mech. Engrg.*, 51, pp. 363-394.

[5] Glowinski, R., 1984. *Numerical Methods for Nonlinear Variational Problems*, Springer-Verlag, New York.

[6] Daniel, J., 1970. *The Approximate Minimization of Functionals*, Prentice Hall, Englewood Cliffs, NJ.

[7] Golub, G.H., Van Loan, C., 1996. *Matrix Computation*, 3rd edition, Johns Hopkins University Press, Baltimore, MD.



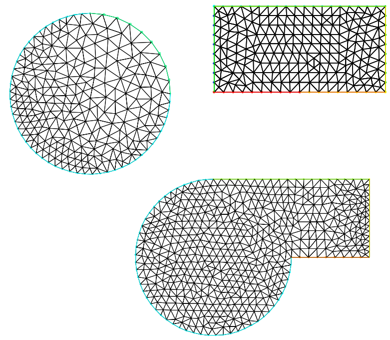
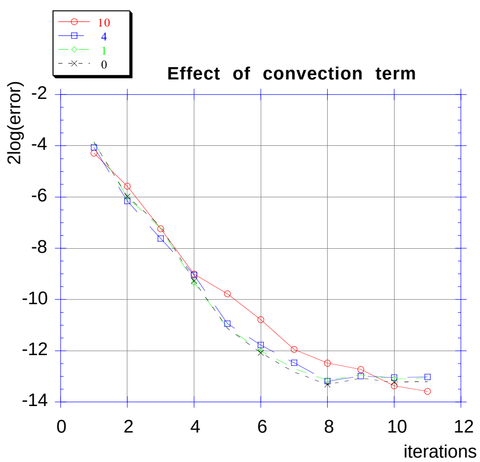


Figure 2: Figure 3: Contours of the solution of the 1^{st} test problem ($h = 1/128$). Figure 4: Grids for Ω_1, Ω_2 and Ω for the 2^{nd} test problem.



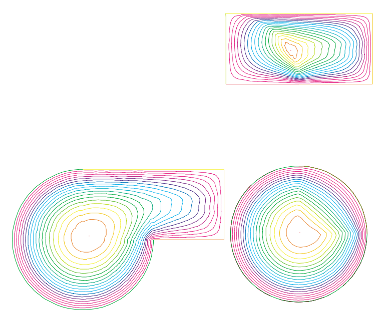


Figure 3: Figure 5: Convergence history for the different values of α (2^{nd} test problem). Figure 6: Contours of the local solutions (right figures) and of the global solution (left figure) for the 2^{nd} test problem.