Rappelons que A la matrice est stockée sous forme de 3 valeurs a, b et c uniquement.

Algorithme 1 : Produit matriciel $Au = v$

$nproc \leftarrow$ nombre de proc.
$me \leftarrow$ numéro du proc courant.
$iBeg_{me} \leftarrow$ indice de début dont s'occupe le proc me.
$iEnd_{me} \leftarrow$ indice de fin dont s'occupe le proc me.
$S_{me} = iEnd_{me} - iBeg_{me} + 1$.
$v_{me} \leftarrow$ vecteur de taille S_{me} destiné à accueillir le vecteur $Au = v$ sur me.

pour chaque $k = 0 : nproc - 1$ **faire**
  $iBeg_k \leftarrow$ indice de début dont s'occupe le proc k
  $iEnd_k \leftarrow$ indice de fin dont s'occupe le proc k

  $S_k = iEnd_k - iBeg_k + 1$
  $v_k \leftarrow$ vecteur temporaire de taille S_k initialisé à zéro.

  pour $p = iBeg_k : iEnd_k$ **faire**
    $i = reste(p, N_x)$ et $j = quotient(p, N_x)$

    pour $n = iBeg_{me} : iEnd_{me}$ **faire**
      si $j > 0$ *(ie* $P(i,j) \notin \Gamma_{bas}$*)* **alors**
        si $n == p - N_x$ **alors**
          $v_k[p - iBeg_k] \mathrel{+}= c\, u[n - iBeg_{me}]$ $\quad (p \mapsto n = \mathrm{P}(i, j-1))$
        fin
      fin
      si $i > 0$ *(ie* $P(i,j) \notin \Gamma_{gauche}$*)* **alors**
        si $n == p - 1$ **alors**
          $v_k[p - iBeg_k] \mathrel{+}= b\, u[n - iBeg_{me}]$ $\quad (p \mapsto n = \mathrm{P}(i-1, j))$
        fin
      fin

      si $n == p$ **alors**
        $v_k[p - iBeg_k] \mathrel{+}= a\, u[n - iBeg_{me}]$ $\quad (p \mapsto n = p)$
      fin

      si $i < N_x$ *(ie* $P(i,j) \notin \Gamma_{droit}$*)* **alors**
        si $n == p + 1$ **alors**
          $v_k[p - iBeg_k] \mathrel{+}= b\, u[n - iBeg_{me}]$ $\quad (p \mapsto n = \mathrm{P}(i+1, j))$
        fin
      fin

      si $j < N_y$ *(ie* $P(i,j) \notin \Gamma_{haut}$*)* **alors**
        si $n == p + N_x$ **alors**
          $v_k[p - iBeg_k] \mathrel{+}= c\, u[n - iBeg_{me}]$ $\quad (p \mapsto n = \mathrm{P}(i, j+1))$
        fin
      fin
    fin

    MPI_Reduce (&v_k[0], &v_{me}[0], S_k, MPI_DOUBLE, MPI_SUM, k,
     MPI_COM_WORLD)
  fin
  retourner v_{me}
fin