THE WEIZMANN INSTITUTE OF SCIENCE

THE CARL F. GAUSS MINERVA CENTER FOR SCIENTIFIC COMPUTATION

POB 26, 76100 REHOVOT ISRAEL

Multilevel Evaluation of Integral Transforms on adaptive grids

Achi Brandt & Kees Venner

Report WI/GC–5

August 1996

Multilevel evaluation of integral transforms on adaptive grids

A. Brandt [a] and C.H. Venner[a,b]

[a] Weizmann Institute of Science, Rehovot, Israel.
[b] University of Twente, Enschede, The Netherlands.

Abstract

Fast numerical evaluation of integral transforms on an adaptive grid, i.e. using local grid refinement, requires an algorithm that relies on smoothness properties only of the continuum kernel, independent of its discrete form. The basic outline of such an algorithm was given in [6], where it was shown that already on a uniform grid this algorithm was more efficient than earlier fast evaluation algorithms [4, 5]. In this paper we outline its detailed formulation for the actual case of local grid refinements. Numerical results are presented for a model problem with a singularity. First it is shown that on a *uniform* grid this singularity dictates a much deteriorated relation between work and accuracy in comparison with the regular case (where accuracy is measured in terms of approximating the *continuum* transform, of course). Next we demonstrate that with the presented fast evaluation algorithm on a *non-uniform* grid one can restore the regular work to accuracy relation, i.e., obtain the same efficiency as for the case without a singularity.

1 Introduction

In [6] we presented a new algorithm for the fast evaluation of integral transforms (multi-integrals) of the type:

$$Gu(x) = \int_\Omega G(x,y)u(y)dy, \qquad x \in \Omega \subset I\!R^d \tag{1}$$

with u a given function, and $G(x,y)$ a given kernel. This kernel $G(x,y)$ is assumed to be "asymptotically smooth", i.e smooth at large $|y-x|$. (For more precise definitions, see [5]).

Multilevel algorithms developed earlier for this task [4, 5] relied on the smoothness properties of the discretized kernel (matrix). Consequently they depended on grid uniformity for their efficiency. However, in practical applications, e.g. in contact mechanics [7, 8, 9], and in lubrication [10, 11], a substantial increase of efficiency may be obtained by using non-uniform grids, since the solution is smooth in large parts of the domain, and large gradients will occur only locally. It is only in these latter regions that a fine grid is really needed. In such cases local grid *refinement* (grid adaptivity) can significantly reduce computing times without loss of accuracy. Moreover, if u has some singularity, as indeed is the case in [7] and for some of the problems discussed in [8], local grid refinements may even be essential to maintain an efficient work-accuracy relationship.

The multilevel methodology in principle allows such local grid refinements in a very natural way, e.g. see [1, 2, 3], but to implement these techniques for integral transforms, a *new* algorithm for the fast evaluation had to be developed. This new algorithm only relies on the smoothness of the *continuum* kernel for its efficiency. It thereby facilitates the introduction of local refinements, wherever needed. Moreover, it is generally faster than fast evaluation algorithms developed earlier [4, 5], as was demonstrated in [6] for a model problem using uniform grids covering the entire domain. In the present report we outline the formulation of the algorithm and its application to a situation with local grid refinements.

As in [6], for simplicity we will restrict ourselves in the description to a one dimensional problem with $\Omega = [a,b]$. In the examples we will use a problem from contact mechanics with the kernel $G(x,y) = \ln|x-y|$.

2 Composite grids

The discretization, e.g. of the function u, will use a sequence of uniform grids with meshsizes $h_0, h_1 = h_0/2, \ldots, h_K = h_0/2^K$. The coarsest meshsize h_0 is such that $(b-a)/h_0$ is a small integer; hence, we may assume that a discretization of the integral transform on it would yield a small matrix by which it is easy to multiply.

The grid with meshsize h_k will have an origin a_k, i.e., it will include points y such that $(y-a_k)/h_k$ is an integer. Often $a_0 = a_1 = \cdots = a_K = a$, and so it is in all our tests, but

other cases, such as cell centered grids are often of interest.

We say that a certain grid h_k (i.e., the grid with meshsize h_k) *covers* a given interval $[a', b']$ if it includes *all* the points $y \in [a', b']$ such that $(y - a_k)/h_k$ is an integer.

In the case of *uniform* discretizations, all the grids cover the *entire* problem domain $[a, b]$. The purpose of having several grids that cover the same interval is to use the coarser of them in multigrid algorithms to obtain a fast *multi-integration*, i.e. fast evaluation of the many discrete integrals on the finest of the grids. As explained above, however, in actual applications the desired (finest) grid may not actually be uniform. We shall produce such non-uniformity by having some of the finer uniform grids covering not the entire domain $[a, b]$ but only those parts of the domain where such a refinement is desired. Those parts will also be covered by coarser grids, for the purpose of multigrid processing. Thus, generally, each uniform grid h_k will cover all those subintervals of Ω where the desired meshsize is h_k *or smaller*. Denoting the closure of the union of these subintervals by P_k, it is clear that

$$\emptyset = P_{K+1} \subset P_K \subset \cdots \subset P_{k+1} \subset P_k \subset P_{k-1} \subset \cdots \subset P_0 = [a, b].$$

We denote by P^k the set of all grid-h_k points i.e., the set of points $y \in P_k$ such that $(y - a_k)/h_k$ is an integer. A *patch* in P^k is any *maximal* (i.e., not strictly contained in another) subsequence in P^k with distance h_k between its consecutive points, i.e., without any gaps. Thus, each patch corresponds to one subinterval where the desired meshsize is h_k or smaller, and P^k will generally be a union of such patches.

The *composite grid* $\Omega^{(K)}$ is defined as the set of uniform grids

$$\Omega^{(K)} = \{P^0, P^1, \ldots, P^K\}.$$

A *composite function* $u^{(K)}$ is a function defined on $\Omega^{(K)}$:

$$u^{(K)} = \{u^0, u^1, \ldots, u^K\} \tag{2}$$

i.e., it is a *set* of discrete functions where u^k is defined on P^k, $(k = 0, 1, \ldots, K)$.

In the special case of a *uniform* discretization, $P_k = \Omega = [a, b]$ for all $k = 0, 1, \ldots, K$. Often in that case $a_k = a$ and $h_k = (b - a)/(2^k N_0)$.

2.1 Data grid, integration grid, and evaluation grid

For the discretization of the integral transform we can distinguish three different grids; a *data grid* of points $\{z_j\}$ on which u is given, an *integration grid* of points $\{y_j\}$ which subdivides the domain into subintervals for piecewise integration, and an *evaluation grid* of points $\{x_i\}$ at which the discrete transform is to be evaluated. In general these three grids can be different. For example, data points may be midway between integration grid points.

The above description of a composite grid can be considered as the definition of the *integration* composite grid $\Omega^{(K)}$. For this grid however we need $a_k = a$. Furthermore, the end points of each patch of P^k must coincide with points of P^{k-1}. A gridpoint of P^k will be denoted by $y_j^k = a_k + j\ h_k$ with $0 \leq j \leq n_k$, or when sufficiently clear simply by its index j.

A similar description will apply to the *data* composite grid with points z_j^k, and to the *evaluation* composite grid. This latter grid will be denoted by $\bar{\Omega}^{(K)} = \{\bar{P}^0, \bar{P}^1, \ldots, \bar{P}^K\}$. Its uniform subgrids $\bar{P}^k$ will each consist of patches, each patch being a sequence of gridpoints h_k apart. A point of $\bar{P}^k$ will be denoted x_i^k.

For simplicity one can think of the case where the evaluation and integration grids are identical $y_j^k = x_j^k$ and $\bar{P}^k = P^k$, $(k = 0, 1, \ldots, K)$. However, even in that case it is still useful to keep the separate notation, not just for generality but mainly for clarity, since the grids have different roles in the algorithm.

3 Discretization

A composite (integration) grid as explained in the previous section effectively subdivides the domain Ω into a sequence of say n closed integration intervals $[y_j, y_{j+1}]$, where $a = y_0 < y_1 < ... < y_n = b$, with their length matching the meshsize of the locally finest grid $y_{j+1}^k - y_j^k = h_k$ for $(y_j^k, y_{j+1}^k) \subseteq P_k - P_{k+1}$. On the composite grid we define $\tilde{u}(y)$ to be an $s-$order polynomial interpolation from the values of u given on the data grid of points $\{z_l\}$. That is, for any k and $y \in [y_j^k, y_{j+1}^k]$ such that $(y_j^k, y_{j+1}^k) \subseteq P_k - P_{k+1}$ we define

$$\tilde{u}_j^k(y) = \sum_{t=0}^{s-1} c_{j,t}^k (y - y_j^k)^t \tag{3}$$

where the s coefficients $c_{j,t}^k$ are such that $\tilde{u}_j^k(z_l)$ coincides with $u(z_l)$ on s of the data grid points. Note that these s points need not themselves all belong to the interval, but they must of course be in its neighborhood.

For any x, the contribution of an interval $[y_j^k, y_{j+1}^k] \subseteq P_k - P_{k+1}$ to the discrete integral transform $Gu^{(K)}(x)$ is now defined by:

$$G_{k,j}u^{(K)}(x) = \int_{y_j^k}^{y_{j+1}^k} G(x,y)\tilde{u}_j^k(y)dy. \tag{4}$$

The local discretization error per unit length of integration, caused by the average interpolation error is bounded by:

$$|\tau| = (\gamma_1 h)^s |u^{(s)}||G| \tag{5}$$

where h is a local bound on the datagrid meshsize, $|u^{(s)}|$ the maximum of the s derivative of u in the interval used by the interpolation, and $|G|$ the average of $|G(x,y)|$ over the

integration interval. $\gamma_1 \approx 0.5$ if the length of the integration interval is also bounded by h, and the s interpolation points are suitably chosen on both its sides.

Let $G^k(x,y)$ denote as in [6] the family of kernels defined by:

$$G^0(x,y) = G(x,y)$$

;

$$G^l(x,y) = \int_x^y G^{l-1}(x,\eta)d\eta. \tag{6}$$

For example, for $G(x,y) = \ln|y-x|$:

$$G^l(x,y) = \frac{1}{l!}(y-x)^l(\ln|y-x| - \sum_{j=1}^{l}\frac{1}{j}). \tag{7}$$

Integrating (4) by parts s times using (6) one obtains:

$$G_{k,j}u^{(K)}(x) = \sum_{l=1}^{s}(-1)^l\tilde{u}_j^{k,(l-1)}(y_j^k)G^l(x,y_j^k) - \sum_{l=1}^{s}(-1)^l\tilde{u}_j^{k,(l-1)}(y_{j+1}^k)G^l(x,y_{j+1}^k), \tag{8}$$

where $\tilde{u}_j^{k,(l-1)}$ denotes the $l-1$ derivative of $\tilde{u}_j^k$.

Subsequently $Gu^{(K)}(x)$ approximating $Gu(x)$ is obtained by summing up over all integration intervals. Let P^q denote the finest grid for which $y_0^q = a \in P_q$, and let $P^{\bar{q}}$ be the finest grid for which $y_{n_{\bar{q}}}^{\bar{q}} = b \in P_{\bar{q}}$. By (3) and (8) we then obtain:

$$Gu^{(K)}(x) = B^{(K)}(x) + \sum_{l=1}^{s}(-1)^l S^{(K),l}(x), \tag{9}$$

where

$$B^{(K)}(x) = \sum_{l=1}^{s}(-1)^l\left[\tilde{u}_0^{q,(l-1)}(a)G^l(x,a) - \tilde{u}_{n_{\bar{q}}-1}^{\bar{q},(l-1)}(b)G^l(x,b)\right], \tag{10}$$

and

$$S^{(K),l}(x) = \sum_{k=0}^{K}\sum_{j\in P^k} G^l(x,y_j^k)U_j^{k,l}, \tag{11}$$

with

$$U_j^{k,l} = \begin{cases} 0 & \text{for } y_j^k = a \text{ or } y_j^k = b \\ A_j^{k,l} - B_j^{k,l} & \text{otherwise,} \end{cases} \tag{12}$$

where

$$A_j^{k,l} = \begin{cases} \tilde{u}_j^{k,(l-1)}(y_j^k) & \text{if } (y_j^k, y_j^k + h_k) \subseteq P_k - P_{k+1} \\ 0 & \text{otherwise,} \end{cases} \tag{13}$$

$$B_j^{k,l} = \begin{cases} \tilde{u}_{j-1}^{k,(l-1)}(y_j^k) & \text{if } (y_j^k - h_k, y_j^k) \subseteq P_k - P_{k+1} \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

In the case of a uniform discretization $U_j^{l,k} = 0$ for all $k < K$, hence in that case the double summation in (11) reduces to a summation over P^K as in [6], and the coarser grids will only play a role in the fast evaluation; see Sec. 2.

Depending on the choice of the integration grid intervals relative to the datagrid intervals there will be cancellations, i.e. $U_j^{k,l} = 0$ for some l, for nearly all points $y_j^k \in P^k - P^{k+1}$. This for example holds for $l = 1$ if the integration grid intervals coincide with the intervals of the datagrid: cancellations occur (as explained in [6] for uniform grids) at all points except for the internal boundaries (endpoints of the patches). Such cancellations also occur for any s even and l odd if the integration grid and datagrid coincide (as is natural for s even), and for s odd and l even if the data gridpoints are midpoints of the integration grid (as is natural for s odd).

At interior points of $P^k - P^{k+1}$ where (12) does not vanish we have:

$$|U_j^{k,l}| = 2(\gamma_2 h_k)^{s-l+1} |u^{(s)}(y_j^k)| + O(h_k^{s-l+2}), \tag{15}$$

with $\gamma_2 \approx 0.5$ due to the grid uniformity in such regions. This is no longer true for internal boundary points. However, the total factor multiplying $G(x, y)$ at such a point (with nearly cancelling contributions from the two neighbouring patches) is still smaller than the right hand side of (15), with $\gamma_2 = O(1)$ but possibly $\gamma_2 > 1$.

As noted in Sec. 2 we have to evaluate (9) for all points $\{x_i\}$ of a *non-uniform* evaluation grid, i.e. for all $x_i^k \in \bar{P}^k - \bar{P}^{k+1}$ of the composite evaluation grid. From a computational point of view the main task then is the evaluation of the discrete subtransforms, i.e. (11), for which the aforementioned cancellations do not occur. Restricting ourselves to a single transform, the algorithm for the fast evaluation of such a transform is outlined in the following sections.

4 Kernel softening

For the purpose of fast evaluation we will rewrite a given discrete transform into a *softened transform*, i.e. the same discrete transform but with a softened kernel, and a *correction*. This softened kernel is designed such that the softened transform can accurately be obtained using coarser grids, and that the correction is a quantity that can be obtained by local operations only. This procedure of kernel softening is described in [6], but for the sake of completeness we briefly repeat it here.

We assume that the kernel $G(x,y)$ is asymptotically smooth. By this we mean that $G(x,y)$ is increasingly smooth for larger $|y-x|$, in such a way that for any given "allowed error" $\epsilon > 0$ there exist nonnegative integers m and p for which a "softened kernel" $G_H(x,y)$ can be defined at any "softening scale" $H > 0$ with the properties:

(*i*) *Locality of softening*: $G_H(x,y) = G(x,y)$ for $|y-x| \geq mH$.

(*ii*) $G_H(x,y)$ is *suitably smooth* on the scale H. By this it is meant that, both as a function of x for any fixed y and as a function of y for any fixed x, $G_H(x,y)$ can be approximated up to an error smaller than ϵ by a p-order interpolation from its values on any uniform grid with meshsize H (or smaller). This translates into the requirement that

$$(\gamma_3 H)^p |G_H^{(p)}(x,y)| \leq O(\epsilon)$$

for any (x,y), where $G_H^{(p)}(x,y)$ stands for any p-order derivative of G_H with respect to either x or y, and γ_3 is a constant depending on the interpolation geometry, $\gamma_3 = 1/2$ for the usual central interpolations.

With the exception of oscillatory kernels treated in [2], most kernels arising in physics are smooth in a way that a "softening" satisfying (i) and (ii) can easily be provided with m and p rising only slowly for decreasing ϵ; e.g. for the family of kernels $G^l(x,y)$ defined by (7) softened kernels with $m = O(\ln \frac{1}{\epsilon})$ and $p = O(\ln \frac{1}{\epsilon})$ are described in [6].

Let $G_L^l(x,y)$ denote the softened kernel of $G^l(x,y)$ on the scale h_L. By (ii), for any $y \in P_L$, the value of $G_L^l(x,y)$ can be interpolated from the grid values $G_L^l(x,y_J^L)$. In particular, for any y_j^{L+1} of grid P^{L+1} there are interpolation weights w_{jJ}^L such that

$$G_L^l(x,y_j^{L+1}) = \sum_{J \in P^L} w_{jJ}^L G_L^l(x,y_J^L) + O(\epsilon) \tag{16}$$

for all x. The summation actually only extends over just p terms (e.g. the terms for which $|y_j^{L+1} - y_J^L| < ph_L/2$ if even p and central interpolation are used). In the same way for any point $x_i^{L+1} \in \bar{P}^{L+1}$ there are interpolation weights $\bar{w}_{iI}^L$ such that

$$G_L(x_i^{L+1},y) = \sum_{I \in \bar{P}^L} \bar{w}_{iI}^L G_L(x_I^L,y) + O(\epsilon). \tag{17}$$

For smallest errors these p–order interpolations should preferably be central. Near the boundaries (of the domain) non-central interpolations can be used such that all interpolation points are within the domain. However, usually $G(x,y)$ is well defined beyond the boundaries, and central interpolations can be used throughout. Note however that to facilitate central interpolation, some patches should sometimes be slightly extended to ensure that every point of P^{L+1} ($\bar{P}^{L+1}$) is interior to a patch of P^L ($\bar{P}^L$) and is at a distance at least $ph_L/2$ from the patch endpoints.

5 Softened transform

We define the softened transform $S_L^{(K),l}$ by:

$$S_L^{(K),l}(x) = \sum_{k=0}^{K} \sum_{j \in P^k} G_L^l(x, y_j^k) U_j^{k,l} = \sum_{k=0}^{K-1} \sum_{j \in P^k} G_L^l(x, y_j^k) U_j^{k,l} + \sum_{j \in P^K} G_L^l(x, y_j^K) U_j^{K,l}. \quad (18)$$

By definition $G_L^l(x, y)$ is smooth with respect to y on scale h_L. So for $L < K$, up to $O(\epsilon)$:

$$S_L^{(K),l}(x) = \sum_{k=0}^{K-1} \sum_{j \in P^k} G_L^l(x, y_j^k) U_j^{k,l} + \sum_{j \in P^K} \sum_{J \in P^{K-1}} w_{jJ}^{K-1} G_L^l(x, y_J^{K-1}) U_j^{K,l}.$$

By changing the order of $\Sigma_{j \in P^K}$ and $\Sigma_{J \in P^{K-1}}$ in the last double summation we obtain

$$S_L^{(K),l}(x) = \sum_{k=0}^{K-1} \sum_{j \in P^k} G_L^l(x, y_j^k) U_j^{k,l} + \sum_{J \in P^{K-1}} G_L^l(x, y_J^{K-1}) \sum_{j \in P^K} w_{jJ}^{K-1} U_j^{K,l},$$

which can be rewritten as

$$S_L^{(K),l}(x) = \sum_{k=0}^{K-2} \sum_{j \in P^k} G_L^l(x, y_j^k) U_j^{k,l} + \sum_{J \in P^{K-1}} G_L^l(x, y_J^{K-1}) \tilde{U}_J^{K-1,l}, \quad (19)$$

where for *any* $k < K$ and $J \in P^k$ we recursively define

$$\tilde{U}_J^{k,l} = U_J^{k,l} + \sum_{j \in P^{k+1}} w_{jJ}^k \tilde{U}_j^{k+1,l}, \quad (20)$$

and $\tilde{U}_j^{K,l} = U_j^{K,l}$.

Notice that (19) is of the same form as (18). Therefore, if $h_{K-1} < h_L$ the procedure of splitting off the top level and replacing its summation by a coarse grid summation can be repeated. In fact it can be repeated as long as the meshsize of the finest grid in the summation is smaller than h_L. As a result, up to $O(\epsilon)$ error:

$$S_L^{(K),l}(x) = \sum_{k=0}^{L-1} \sum_{j \in P^k} G_L^l(x, y_j^k) U_j^{k,l} + \sum_{j \in P^L} G_L^l(x, y_j^L) \tilde{U}_j^{L,l}, \quad (21)$$

for any x. This will serve as the composite grid approximation to (18). To evaluate $S_L^{(K),l}(x)$ fast for all $x_i^L \in \bar{P}^L$ notice that:

$$S_L^{(K),l}(x) = \sum_{k=0}^{L-1} \sum_{j \in P^k} G_{L-1}^l(x, y_j^k) U_j^{k,l} \quad + \sum_{j \in P^L} G_{L-1}^l(x, y_j^L) \tilde{U}_j^{L,l} + M_{L-1}^{L,l}(x), \quad (22)$$

where

$$M_{L-1}^{L,l}(x) = \sum_{k=0}^{L-1} \sum_{j \in P^k} (G_L^l(x, y_j^k) - G_{L-1}^l(x, y_j^k)) U_j^{k,l} \qquad (23)$$
$$+ \sum_{j \in P^L} (G_L^l(x, y_j^L) - G_{L-1}^l(x, y_j^L)) \tilde{U}_j^{L,l}.$$

By (16) with $L-1$ instead of L, i.e. using the smoothness of $G_{L-1}^l(x,y)$ with respect to y (on the scale h_{L-1}), up to $O(\epsilon)$ error

$$S_L^{(K),l}(x) = \sum_{k=0}^{L-2} \sum_{j \in P^k} G_{L-1}^l(x, y_j^k) U_j^{k,l} + \sum_{j \in P^{L-1}} G_{L-1}^l(x, y_j^{L-1}) \tilde{U}_j^{L-1,l} + M_{L-1}^{L,l}(x). \qquad (24)$$

Using (17) with $L-1$ instead of L, i.e., using the smoothness with respect to x (on the scale h_{L-1}) of $G_{L-1}^l(x,y)$ in (24), we obtain for any $x_i^L \in \bar{P}^L$,

$$S_L^{(K),l}(x_i^L) = \sum_{I \in \bar{P}^{L-1}} \bar{w}_{iI}^{L-1} S_{L-1}^{(K),l}(x_I^{L-1}) + M_{L-1}^{L,l}(x_i^L) + O(\epsilon), \qquad (25)$$

where $S_{L-1}^{(K),l}$ is the expression given in (21) ($L-1$ replacing L).

Thus, for all $x_i^L \in \bar{P}^L$ we can obtain $S_L^{(K)}(x_i^L)$ by interpolating $S_{L-1}^{(K)}(x_i^L)$ from the coarser grid $\bar{P}^{L-1}$ followed by the addition of a local correction, $M_{L-1}^{L,l}(x_i^L)$. Notice that both operations are local. For the interpolation this is obvious, i.e. the summation over $I \in P^{L-1}$ extends over p terms only. However, also $M_{L-1}^{L,l}(x_i^L)$ can be obtained by a small amount of work, since in (23) all terms j for which $|y_j^k - x_i^L| > m h_{L-1}$ are zero by the definition of the softened kernel.

This procedure can of course be repeated recursively until a grid is reached where straightforward evaluation using (21) can be done cheaply.

At this point it should be noted that the equations presented in this section are a natural extension of those given in [6]. Indeed, for a uniform discretization $U_j^{k,l} = 0$ for all $k < K$, and as a result the first terms in the right hand sides of equations (18), (19) through (24) vanish.

6 Fast evaluation

For all points of the evaluation grid we will calculate the transforms in (9) by writing

$$S^{(K),l}(x_i^L) = S_L^{(K),l}(x_i^L) + M_L^{(K),l}(x_i^L), \qquad \forall x_i^L \in \bar{P}^L - \bar{P}^{L+1} \quad (0 \le L \le K), \qquad (26)$$

with $S_L^{(K),l}$ for any $L \le K$ defined by (25), and

$$M_L^{(K),l}(x) = \sum_{k=0}^{K} \sum_{j \in P^k} (G^l(x, y_j^k) - G_L^l(x, y_j^k)) U_j^{k,l}. \tag{27}$$

Thus the evaluation of the original multi-summation is replaced by the evaluation of a softened transform followed by a correction. This correction, $M_L^{(K),l}(x_i^L)$, can be obtained by local operations since, by the definition of the softened kernel, it involves only those terms j for which $|x_i^L - y_j^k| < mh_L$ and since it needs to be evaluated only for $x_i^L \in \bar{P}^L - \bar{P}^{L+1}$. Therefore, by (25) and (23) the evaluation of (26) can be done using local operations only.

We now introduce the following algorithm for the fast evaluation of (26) on a composite grid (K):

(I) $\forall L \leq K$ and $x \in \bar{P}^L$ evaluate $S_L^{(K),l}(x)$

This task is performed by the following recursive algorithm:

(i) *Anterpolation from* P^K *to* P^{K-1}: calculate $\hat{U}_j^{K-1,l}$ for all $y_j^{K-1} \in P^{K-1}$ by (20).

(ii) *Coarse grid evaluation*:
$\forall L \leq K-1$ and $x \in \bar{P}^L$ evaluate $S_L^{(K),l}(x)$.

(iii) *Interpolation from grid* $\bar{P}^{K-1}$ *to grid* $\bar{P}^K$ to obtain $S_{K-1}^{(K),l}(x)$ for all $x \in \bar{P}^K$.

(iv) *Grid* $\bar{P}^K$ *correction*:
$\forall x \in \bar{P}^K$ add $M_{K-1}^{K,l}(x)$ as defined by (23) to obtain $S_K^{(K),l}(x)$.

Step (iii) and (iv) together are described by (25) above (K replacing L) Notice the recursion involved in step (ii): It can be done using the same algorithm (I) but with $L \leq K-1$ instead of $L \leq K$. This recursion can be repeated until a (coarse) grid L is reached where (ii) can be performed more cheaply by straightforward evaluation(s).

(II) $\forall L \leq K$ and $x \in \bar{P}^L - \bar{P}^{L+1}$ evaluate $S^{(K),l}(x)$ by computing $M_L^{(K),l}(x)$ as defined by (27) and adding it to $S_L^{(K),l}(x_i^L)$.

For a uniform discretization this algorithm reduces to the one described in [6]. Indeed, in that case $\bar{P}^{L+1} - \bar{P}^L = \emptyset$ for $L < K$, hence the corrections in (II) need only be computed for $L = K$, and can then be combined with the corrections of step (iv) for $L = K$ into a single correction $M_{K-1}^{(K),l}(x)$.

7 Parameter optimization

The values of the parameters p and m on a given grid h_L are determined in the same way as in [6] for a uniform grid, i.e. by minimizing the computational work under the constraint that the incremental evaluation error is smaller than an estimate for the original fine-grid discretization error.

For any x the discretization error per unit length of integration, see Sec. 3, is bounded by:

$$|\tau^h| \leq (\gamma_1 h)^s |u^{(s)}| \ |G|. \tag{28}$$

Due to (15) and (16), the incremental evaluation error resulting from the transfer of (part of) an evaluation from grid h_{L+1} to grid h_L $(0 \leq L \leq K-1)$, e.g. the error that results from replacing (22) by (24), per unit length, is given by:

$$|e^L| = \gamma_2 (\gamma_2 h)^{s-l} |G_L^l - I\!I_{h_L}^{h_{L+1}} G_L^l| \ |u^{(s)}|, \tag{29}$$

where $G_L^l(x,y) - I\!I_{h_L}^{h_{L+1}} G_L^l(x,y)$ stands for the order p interpolation error made with the interpolation of G_L^l from grid h_L to a point x or a point y of grid h_{L+1}, and $|G_L^l - I\!I_{h_L}^{h_{L+1}} G_L^l|$ is the average of its absolute value over the integration points y. Generally this error is bounded by

$$|G_L^l - I\!I_{h_L}^{h_{L+1}} G_L^l| < (\gamma_3 h_L)^p |G_L^{l(p)}|, \tag{30}$$

where $|G_L^{l(p)}|$ stands for the average of the absolute value of the p^{th} derivative of G_L^l, and γ_3 depends on the geometry of the interpolation, i.e., $\gamma_3 = 1/2$ for central interpolation. Generally, by (28), (29) $|e^L| \leq |\tau^h|$ requires:

$$(\gamma_2)^{s-l+1} h^{-l} |G_L^l - I\!I_{h_L}^{h_{L+1}} G_L^l| \leq (\gamma_1)^s \ |G|. \tag{31}$$

Under this constraint we want to minimize the incremental work per grid h_{L+1} point $W = O(p+4m)$ related to transferring (part of) an evaluation from grid h_{L+1} to grid h_L. This work estimate was obtained as follows. Taking an operation to mean a combination of one multiplication and one addition, the number of operations in the computation of $\hat{U}^{L,l}$ on grid h_L is $p/2$ per grid h_{L+1} point (since $d = 1$, for half of the values the transfer is trivial). Similarly $p/2$ operations per grid h_{L+1} point are used to interpolate $S_L^{(K),l}$ from grid h_L to grid h_{L+1}. Finally the corrections are added which involve summations over roughly $4m$ grid h_{L+1} points.

Due to the use of integrated kernels it will usually be possible to employ $m = 0$ and a certain minimal p (derived from (31) and depending only on l) for several of the finest coarsening h_L (provided the basic meshsize h is sufficiently small). At large h_L (i.e., after several coarsening stages), m and p will start to rise, reaching finally the typical size (e.g. $O(\ln \frac{1}{\epsilon})$).

For the case $G = \ln|y - x|$ this optimization, see [6] yielded that p should be taken the maximum of the first non-negative integer satisfying

$$p \geq -0.83 \ln(g) + l + 1, \tag{32}$$

and $l + 1$, and m the first integer satisfying

$$m \geq 1.23(p - l - 1), \tag{33}$$

where

$$g = \frac{\gamma_1^s}{\gamma_3^{l+1}\gamma_2^{s-l+1}} \frac{h^l}{h_L^{l+1}}|G|. \tag{34}$$

From (34) it can be seen that:

$$\ln(g) = c_l + l \ \ln(h) - (l+1)\ln(h_L), \tag{35}$$

with c_l some constant depending on the geometry of the interpolation, order of discretization, and l. From (32), (33), and (35) it follows that, for a given finest grid mesh size h, m and p will increase logarithmically with increasing coarse grid mesh size. On the other hand, with decreasing h_L, both m and p decrease, and for $h_L \approx h$ they reach the limits $m = 0$ and $p \geq l + 1$.

For the case of a uniform finest grid (32) and (33) can simply be used with h being the finest grid mesh size. However, in the case of a composite grid h is a local quantity. For simplity of implementation p and m should be constant on a given coarse grid. We therefore propose the use of Eq. (35) with $h = h_K$, the mesh size of the finest mesh, i.e. we determine p and m by the accuracy needed in the interpolation to the finest grid patches present. For additional considerations see Sec. 11.1.

8 Mesh optimization

In [1, 2, 3] a framework of multigrid techniques for the solution of singular partial differential problems by adaptable discretization is presented. The same analysis and techniques can be applied to the evaluation of an integral transform in the case of a singularity in u. The relevant theory is extensively explained in [1].

In general the choice of discretization parameters should be guided by the objective to minimize the work invested to obtain a given accuracy, or, in other words, minimizing the error obtained in a given amount of (computational) work. In principle one can optimize both the order of approximation and the meshsize. However, here we'll restrict ourselves to meshsize adaptation.

The work invested to obtain the integral transform is roughly given by

$$W = \int_\Omega \frac{w}{h^d} dy \tag{36}$$

where d is the dimension of the problem's domain, e.g. $d = 1$ for the problem considered here, $h = h(y)$ is the meshsize of the integration grid, and w is the work per gridpoint. More precisely, w is the sum of the anterpolation (step (i) in Sec. 6) work per integration gridpoint and the interpolation (step (iii)) and correction (step (iv) and (II)) per evaluation grid point. (We assume here the same meshsize distribution $h = h(x) = h(y)$ for both the integration and evaluation grids.) Since the work of anterpolation and of interpolation per gridpoint is proportional to the interpolation order p, and the correction work per gridpoint is proportional to the softening parameter m, we have $w = w_1 m + w_2 p$ with constant w_1 and w_2.

For any x the discretization error in the discrete integral transform can be given by:

$$E = \int_\Omega \tau(y) dy \tag{37}$$

with τ from (5). The value of $|G|$ in (5) depends on x, the evaluation point of interest. However, assuming we are equally interested in accuracy at all evaluation points, $|G|$ in (5) may be considered to be just a constant, which can be absorbed into the error unit, hence, omitted. The Euler differential equation for optimizing h by minimizing E under fixed W (or minimizing W under fixed E) is:

$$\frac{\partial E}{\partial h(y)} + \lambda \frac{\partial W}{\partial h(y)} = 0, \tag{38}$$

or, by (36) and (37)

$$\frac{\partial \tau}{\partial h} = \lambda w h^{-d-1}, \tag{39}$$

where λ is a constant independent of y expressing the marginal rate of exchanging optimal accuracy for work, i.e. $\lambda = -dE_{min}/dW$. Referred to as *the exchange rate*, it will serve as the chief control parameter specifying the cost at which additional accuracy can be afforded.

9 Model problem: Singularity in $u(y)$

As a model problem we take the evaluation of

$$Gu(x) = \int_{-1}^{1} u(y) \ln|x - y| dy \tag{40}$$

with

$$u(y) = \begin{cases} \sqrt{1 - \left(\frac{y}{r_0}\right)^2} & \text{for } \left|\frac{y}{r_0}\right| \le 1 \\ 0 & \text{otherwise,} \end{cases} \tag{41}$$

For a given $r_0 \leq 1$.

The problem is symmetric, i.e. there are two singularities, at $y = \pm r_0$. Because they have the same (polluting) effect on the solution they need not be discussed separately. In the following we will therefore just consider the singularity at $y = -r_0$. Let r denote the distance from this singularity for points $-r_0 \leq y \leq 0$; i.e. $r = y + r_0$. From (41) it follows that

$$|u^{(s)}| = O(r^{\frac{1}{2}-s}). \tag{42}$$

Consequently, if the integral transform is computed using a uniform integration grid (without local refinement), it follows from (5) that:

$$E \approx \int_{O(h)}^{R_0} \tau(r)dr = O(h^s + h^{3/2}) \tag{43}$$

with $R_0 = O(1)$. From (36) with $d = 1$ it follows that $W = O(1/h)$, hence, $E = O(W^{-s} + W^{-3/2})$. For $s \geq 2$ this behaviour is much worse than the regular behaviour (if $|u^s(y)|$ is bounded) where $E = O(h^s) = O(W^{-s})$. The objective now is to introduce local refinements (around the singularities at $y = \pm r_0$) in such a way that the regular relation between W and E, where a higher order discretization yields a more accurate solution for the same work, is restored.

For the local discretization error given by (5), Eq. (39) gives:

$$h = O(\lambda^{\frac{1}{s+1}} r^{\frac{2s-1}{2(s+1)}}). \tag{44}$$

As we are only interested in a relatively crude approximation for now we neglect dependance of m and p on x (see Sec. 11.1). With the non-uniform meshsize (44) we then obtain from (36)

$$W \approx \int_0^{R_0} \frac{1}{h(r)} dr = O(\lambda^{\frac{-1}{s+1}}), \tag{45}$$

and from (37)

$$E \approx \int_0^{R_0} \tau dr = O(\lambda^{\frac{s}{s+1}}), \tag{46}$$

hence, by choosing a refinement pattern according to (44) we anticipate to restore $W = O(E^{-s})$, *as in the regular case*.

Eq. (44) will be used to determine the gross refinement pattern around the singularities at $y = \pm r_0$. However, instead of creating a non-uniform grid by using (44) to determine the local meshsize, (44) will be used inversely, i.e. to determine the distance r from the singularity (in the direction of the origin $y = 0$) to which a patch of grid P^k with meshsize $h = h_k$ should extend, namely $r(h) = O(h^{\frac{2(s+1)}{2s-1}} \lambda^{-\frac{2}{2s-1}})$. The size of the patch on

the other side of the singularity is trivial: only the $s/2$ points needed for the definition of the interpolation polynomial $\tilde{u}(y)$.

For any fixed λ, Eq. (44) then fully defines a composite grid. Note that the maximum number of grids (K) thus created is always finite (natural stopping): $r(h) = 0(h^\beta)$ with $\beta > 1$, hence, for sufficiently small h_k the radius to which this grid should extend will be smaller than h_k, which means that the grid is empty.

10 Numerical tests

Below we present results of numerical tests for the model problem. To demonstrate the need for local grid refinement we first present results obtained using a uniform discretization. Subsequently in Sec. 10.2 the mesh refinement required to restore the optimal work-accuracy efficiency is tested.

10.1 Uniform grids

The integration grid, datagrid, and evaluation grid were chosen to be the same uniform grid with meshsize h. For $s = 2$, i.e. discretization using a piecewise linear polynomial, Eq. (9) reduces to:

$$G^h u^h(x_i) = B^h(x_i) + S^{2,h}(x_i), \tag{47}$$

where

$$S^{2,h}(x_i) = \sum_{j=1}^{n-1} G^2(x_i, y_j) U_j^{2,h} \tag{48}$$

with

$$U_j^{2,h} = \frac{1}{h}(u_{j-1}^h - 2u_j^h + u_{j+1}^h), \tag{49}$$

and

$$\begin{aligned} B^h(x_i) \;=\; & u_n^h G^1(x_i, y_n) - u_0^h G^1(x_i, y_0) + \\ & \frac{1}{h}(u_1^h - u_0^h) G^2(x_i, y_0) - \\ & \frac{1}{h}(u_n^h - u_{n-1}^h) G^2(x_i, y_n). \end{aligned} \tag{50}$$

To get a clear picture of the discretization error, the discrete transform (48) has been computed by simple summation. We will present results for a sequence of grids with meshsizes H, $H/2$, etc. These grids are numbered, starting with the coarsest grid that

will be called level 0, the next finer grid being level 1, etc. The number of gridpoints on grid k is $n = 2^{k+3} + 1$, i.e. its meshsize is $h_k = 2^{-k-2}$.

To monitor the discretization error we calculated:

$$E^h = \frac{1}{n} \sum_{i=1}^{n} |G^{h_k} u^{h_k}(x_i) - Gu(x_i)|. \tag{51}$$

Table 1 shows E^h as a function of the grid level k for three values of r_0, i.e. $r_0 = 1$ (singularity at the boundary), $r_0 = 0.5$ (singularity at an interior gridpoint), and $r_0 = 0.6$ (singularity at an arbitrary location, not coinciding with a gridpoint).

For a given r_0 the ratio between the values of E obtained for k and $k+1$ shows the rate of convergence to the exact continuum solution with decreasing meshsize. If the second derivative of u were bounded, this ratio would be $2^2 = 4$. However, as can be verified from the table, for the present problem we only obtain a factor of about 2.8, which confirms the analysis presented in Sec. 9 where a factor $2^{3/2} = 2.83$ is predicted.

With an (additional) evaluation error that is small compared to this discretization error, the discrete integrals can be computed using the fast evaluation algorithm outlined in [3]. In that case the total work per gridpoint would tend to $2p + 1$ where, for the present problem and for second order discretization, $p = 4$, and the total work to obtain all integrals (on the uniform grid) is $W = O(1/h)$.

So, each time the grid is refined (uniformly) the computational work increases by a factor of 2 and the error decreases by a factor of $2\sqrt{2}$. Thus, as predicted in Sec. 9, we have $E \propto W^{-3/2}$ instead of the regular relation $E \propto W^{-2}$.

10.2 Local refinement

We again use $s = 2$, i.e. on the non-uniform grid too $u(y)$ is approximated by a piecewise linear function. The coarsest grid, denoted by level 0 will extend over the entire domain $[-1, 1]$ with meshsize $h_0 = 1/4$, i.e. it has $(8 + 1)$ points. Subsequent finer grids with meshsizes $h^k = 2^{-k} h_0$ are created using the control parameter λ in the following way. For a given λ the distance from the singularity to which the local patch with meshsize h_k should extend is given by:

$$R_k = \lambda^{\frac{-2}{2s-1}} h_k^{\frac{2(s+1)}{2s-1}} \tag{52}$$

with the requirement that $R_k \geq 2h_{k-1}$. For $s = 2$ we obtain:

$$R_k = \bar{\lambda}^{-2} h_k^2 \tag{53}$$

where $\bar{\lambda} = \lambda^{1/3}$.

If $R_k > r_0$ then grid P_k consists of a single patch extending over $[\max(-1.0, -r_0 - s\ h_k/2), \min(r_0 + s\ h_k/2, 1.0)]$. Otherwise grid P^k consists of two grid patches. The first patch covering $[\max(-1, -r_0 - s\ h_k/2), -r_0 + R_k]$, and the second, its mirror image,

covering $[r_0 - R_k, \min(r_0 + s\ h_k/2, 1)]$. The boundaries of the patches will be rounded in such a way that they coincide with points of the next coarser grid P^{k-1}. For any given λ we thus obtain a composite grid. The maximum number of grids is always limited by the natural stopping explained in Sec. 9, i.e. K is the largest integer k satisfying $R_k \geq 2h_{k-1}$, or.

$$k \leq \frac{\ln(\frac{h_0}{4\bar{\lambda}^2})}{\ln(2)}. \tag{54}$$

We present the results for a sequence of $\bar{\lambda}$ values $\bar{\lambda}_0 > \bar{\lambda}_1 > \bar{\lambda}_2 > \cdots$, chosen by $\bar{\lambda}_i = \bar{\lambda}_{i-1}/2$. To monitor the error we define

$$E^{(K)} = \frac{1}{N} \sum_{i=1}^{N} |Gu(x_i) - G^{(K)}u(x_i)| \tag{55}$$

where the summation is done over all points $\{x_i\}$ of the non-uniform evaluation grid, i.e. over the points $x_i^k \in \bar{P}^k - \bar{P}^{k+1}$ for every $0 \leq k \leq K$.

In Tables 2, 3, and 4 the results are presented for $r_0 = 0$, $r_0 = 0.5$ and $r_0 = 0.6$, respectively. Three columns contain data related to the composite grid that is created, i.e. the number of gridlevels K, the index K_0 of the finest grid that still consists of only one patch, and the total number N of gridpoints $x_i^k \in \bar{P}^k - \bar{P}^{k+1}$ $(0 \leq k \leq K)$. The next column of each of the tables gives $E^{(K)}$ as obtained by straightforward evaluation of the transform on the non-uniform grid, i.e. evaluation of (11) by straightforward summation, using $O(N)$ operations per gridpoint. Then this column displays the L_1 norm of the discretization error on the non-uniform grid. For all three values of r_0 the results confirm the analysis of Sec. 9: With $s = 2$ we should have $E^{(K)} = O(\lambda^{\frac{2}{3}}) = O(\bar{\lambda}^2)$ and $N = O(\lambda^{-\frac{1}{3}}) = O(\bar{\lambda}^{-1})$. Thus, each time we halve $\bar{\lambda}$ the number of nodes N should increase by a factor of 2, and the average error should decrease by a factor 4, as indeed the results clearly show.

The next columns of Table 2, 3 and 4 display results obtained when the evaluation of (11) is replaced by the evaluation of (26) as described in Sec. 6. The following data is presented: The error $E^{(K)}$, the direct evaluation grid K_S, i.e. the value of L for which step (ii) of part (I) of the fast evaluation algorithm is performed using direct evaluation, and the average number of operations per gridpoint invested to obtain the result. For a given composite grid (K) the values of the parameters p and m to be used on each of the grids h_k $(0 \leq k \leq K)$ were obtained from (32) and (33) with $l = 2$, $c_l = 0$, and $p \leq 20$.

The error $E^{(K)}$ of the fast evaluation turns out to be practically the same as in the direct evaluation. This means that the error introduced by the fast evaluation is substantially smaller than the discretization error (and even much more so when $\bar{\lambda}$ gets smaller, i.e. when better accuracy is sought).

The fast evaluation work *per gridpoint* (W/N) can be seen in the tables to increase logarithmically as function of the accuracy $(1/E^{(K)})$. This follows directly from the choice

of $h = h_K$ in using (32) and (33). For the case of a uniform grid with meshsize h we obtained $m = 0$ and $p = p_{min} \geq l + 1$ on the finest grids. As a result the total work per gridpoint tends to a small constant number, see [6]. In the present case of local refinement the work to obtain the integral transform is not the same for each gridpoint, i.e. it is $w_1 p + w_2 m$ but with m and p depending on h_K and x (the *locally* finest (evaluation) grid). For sufficiently small $\bar{\lambda}$ substituting $h = h_K$ and $H = h_k$ in (34) results in $m = 0$ and $p = p_{min}$ on several of the finest patches and, indeed, we obtain the integral transform for these points in the same small amount of work as for a uniform grid. However, for a given $\bar{\lambda}$ (and thereby given K), we have $m = O(\ln(1/h_K))$ on a number of coarser grids. Some of these grids may just be auxilliary, i.e. only serve for fast evaluation purposes, but generally they may contain evaluation points, i.e. cover part of the domain not covered by the next finer grid. In that case it will be the cost of the evaluation on these grids that determines the average work per gridpoint.

Let's for example consider the work per gridpoint invested to obtain the integral transform in the points of the level K_0, i.e. the evaluation points $x_i \in \bar{P}^{K_0} - \bar{P}^{K_0+1}$. Let N_0 denote the number of evaluation points on this grid level. One can show straightforwardly that the ratio N_0/N does not depend on $\bar{\lambda}$. Now, from Table 2, 3, and 4 it can be seen that each time $\bar{\lambda}$ decreases by a factor of 2, K increases by 2 and K_0 increases by 1. So h_K decreases faster than h_{K_0}. Thus p and m used on K_0 are proportional to $O(\ln(1/h_K)) = O(\ln(1/\bar{\lambda}))$, see Table 5. Therefore, the work per gridpoint to obtain the integral transform on $\bar{P}^{K_0} - \bar{P}^{K_0+1}$ is $(w_1 p + w_2 m) = O(\ln(1/\bar{\lambda}))$. With N_0/N constant, the overall average work per gridpoint must also be $O(\ln(1/\bar{\lambda}))$.

Summarizing, with the fast evaluation algorithm as presented here we obtain for the model problem:

$$E = O(\bar{\lambda}^2) = O(\lambda^{\frac{2}{3}}) \tag{56}$$

and

$$W = O(\bar{\lambda}^{-1} \ln(1/\bar{\lambda})) = O(\lambda^{-\frac{1}{3}} \ln(1/\lambda)) \tag{57}$$

where the factor $\ln(1/\lambda)$ results from the dependence of m and p on the meshsize, i.e. the local meshsize on the evaluation grid, that was neglected in our original estimate (36). Thus, with the non-uniform grid and the fast evaluation algorithm we have restored the *regular* situation, where the accuracy of the discrete integral transform follows from the (second) order of the discretization error, rather than depending on the singularity in u. Also for $s = 2$ we showed that, up to a factor $\ln(W)$, the algorithm has the regular work-accuracy relation $E \approx W^{-s}$ (see Sec. 11.1).

11 Advanced insights

11.1 Spatially varying m and p.

To obtain the numerical results presented in Sec. 10.2, for a given $\bar{\lambda}$ the values of m and p used on a coarser grid h_k depend only on h_k, not on x. They are determined by the accuracy needed in the interpolation to the finest grid patches present. In fact, the *discretization* is adaptive, but the *fast evaluation* process is not. It can be made adaptive too in the following way. On a given grid h_k a high accuracy is in fact only needed in the region from which these grid h_K patches, via the intermediate grids, actually receive values. In regions further away, e.g. in regions were the locally finest patch is much coarser than h_K, this may not be necessary. One should therefore be able to further increase the efficiency of the algorithm by introducing $m = m(x)$ and $p = p(x)$. This option would give the full work-to-accuracy relation $E = O(W^{-s})$. Practically, however, it does not seem advisable to use this option, since it very substantially complicates the algorithm, for quite a marginal gain. The current, simpler algorithm yields only $E = O(W^{-s} \ln(W))$, but it will both be simpler and more profitable to improve this relation by increasing s, rather than using spatially varying m and p.

11.2 Refinement strategy

In the model problem considered here the behavior of u near the singularity is known a priori, and thus can be used to create the required local refinement. However, in general this may not be the case, e.g. when the evaluation is part of solving for u an integral or integro-differential equation containing $Gu(x)$. In that case the relative local truncation error τ_h^{2h} (the defect correction of the approximate solution on the current grid h to similar discrete equations on a coarser grid $2h$; see [1] or [3]) should be used instead of τ in (39) for controlling the refinement. (The values of τ_h^{2h} are anyway calculated in multigrid solvers of the equations.)

This refinement strategy is even more natural if our purpose is indeed the solution of u. The composite grids created by our sequence of λ values would then each serve to obtain a first approximation for the solution process on the next composite grid. This is the so-called λ *FMG algorithm*; see [1, 3].

12 Conclusion

In [6] we presented a new algorithm for the fast evaluation of integral transforms. The novelty of the algorithm was that for its efficiency it only relied on the smoothnes of the kernel. Thereby it facilitates local grid refinement wherever needed. This has been demonstrated in the present paper. First we have presented the formulation of the algorithm for the case of adaptive grids or local refinement. The non-uniform resolution

is effected in the natural way characteristic for multilevel algorithms, i.e. by means of uniform finer grids extending over smaller and smaller parts of the domain.

Subsequently numerical results were presented for a problem from contact mechanics where $u(y)$ has a singularity. When evaluated on a uniform grid the accuracy of the result is dictated by the singularity instead of by the order of discretization, unlike regular case (when the derivatives of u are bounded). To restore the regular relation between work and accuracy local grid refinement is needed. It was explained how the required refinement can be derived. Subsequently we demonstrated that with the presented fast evaluation algorithm on a non-uniform grid one can essentially restore the regular behavior, i.e. obtain the integral transform with a work to accuracy efficiency as for the case without a singularity.

13 Acknowledgement

The authors gratefully acknowledge the support of the Royal Netherlands Academy of Arts and Sciences, the Feinberg Graduate School of the Weizmann Institute of Science, and the Air Force Office of Scientific Research, United States Air Force (Grant F 49620-92-J-0439).

k	$r_0 = 1$	$r_0 = 0.5$	$r_0 = 0.6$
0	$3.876\ 10^{-2}$	$8.164\ 10^{-2}$	$2.327\ 10^{-2}$
1	$1.272\ 10^{-2}$	$3.073\ 10^{-2}$	$1.008\ 10^{-2}$
2	$4.084\ 10^{-3}$	$1.116\ 10^{-2}$	$1.357\ 10^{-3}$
3	$1.318\ 10^{-3}$	$3.991\ 10^{-3}$	$1.204\ 10^{-3}$
4	$4.318\ 10^{-4}$	$1.416\ 10^{-3}$	$4.616\ 10^{-4}$
5	$1.440\ 10^{-4}$	$5.012\ 10^{-4}$	$1.583\ 10^{-4}$
6	$4.877\ 10^{-5}$	$1.771\ 10^{-4}$	$1.667\ 10^{-5}$
7	$1.672\ 10^{-5}$	$6.259\ 10^{-5}$	$2.021\ 10^{-5}$
8	$5.786\ 10^{-6}$	$2.211\ 10^{-5}$	$7.533\ 10^{-6}$
9	$2.016\ 10^{-6}$	$7.813\ 10^{-6}$	$2.426\ 10^{-6}$

Table 1: L_1 norm of the discretization error E^h as a function of the grid level (meshsize) for a uniform discretization ($s = 2$).

	grid data			Simple evaluation	Fast Evaluation		
$\bar{\lambda}$	K	K_0	N	$E^{(K)}$	$E^{(K)}$	K_S	W/N
2^{-3}	2	1	21	$6.766\ 10^{-3}$	$7.221\ 10^{-3}$	0	20
2^{-4}	4	2	45	$1.311\ 10^{-3}$	$2.057\ 10^{-3}$	1	29
2^{-5}	6	3	93	$2.740\ 10^{-4}$	$5.867\ 10^{-4}$	2	36
2^{-6}	8	4	189	$6.106\ 10^{-5}$	$1.027\ 10^{-4}$	2	45
2^{-7}	10	5	381	$1.426\ 10^{-5}$	$2.171\ 10^{-5}$	3	52
2^{-8}	12	6	765	$3.435\ 10^{-6}$	$5.620\ 10^{-6}$	3	60
2^{-9}	14	7	1533	$8.419\ 10^{-7}$	$1.134\ 10^{-6}$	3	70
2^{-10}	16	8	3069	$2.083\ 10^{-7}$	$2.651\ 10^{-7}$	3	75
2^{-11}	18	9	6141	$5.181\ 10^{-8}$	$6.796\ 10^{-8}$	3	84
2^{-12}	20	10	12285	$1.292\ 10^{-8}$	$1.517\ 10^{-8}$	4	93
2^{-13}	22	11	24573	$3.226\ 10^{-9}$	$3.671\ 10^{-9}$	4	101
2^{-14}	24	12	49149	$8.053\ 10^{-10}$	$9.372\ 10^{-10}$	4	107
2^{-15}	26	13	98301		$2.395\ 10^{-10}$	4	116
2^{-16}	28	14	196605		$6.013\ 10^{-11}$	4	125

Table 2: Non uniform grid results for $r_0 = 1.0$ as a function of the control parameter $\bar{\lambda} = \lambda^{1/3}$ ($s = 2$).

	grid data			Simple evaluation	Fast Evaluation		
$\bar{\lambda}$	K	K_0	N	$E^{(K)}$	$E^{(K)}$	K_S	W/N
2^{-3}	2	1	17	$1.596\ 10^{-2}$	$2.356\ 10^{-2}$	0	18
2^{-4}	4	2	33	$3.167\ 10^{-3}$	$4.369\ 10^{-3}$	1	31
2^{-5}	6	3	65	$6.637\ 10^{-4}$	$9.812\ 10^{-4}$	2	44
2^{-6}	8	4	129	$1.476\ 10^{-4}$	$1.955\ 10^{-4}$	3	56
2^{-7}	10	5	257	$3.439\ 10^{-5}$	$4.192\ 10^{-5}$	3	60
2^{-8}	12	6	513	$8.267\ 10^{-6}$	$9.774\ 10^{-6}$	3	69
2^{-9}	14	7	1025	$2.024\ 10^{-6}$	$2.243\ 10^{-6}$	3	76
2^{-10}	16	8	2049	$5.007\ 10^{-7}$	$5.316\ 10^{-7}$	3	79
2^{-11}	18	9	4097	$1.245\ 10^{-7}$	$1.305\ 10^{-7}$	4	84
2^{-12}	20	10	8193	$3.104\ 10^{-8}$	$3.190\ 10^{-8}$	4	91
2^{-13}	22	11	16385		$7.870\ 10^{-9}$	4	95
2^{-14}	24	12	32769		$1.962\ 10^{-9}$	4	102
2^{-15}	26	13	65537		$4.864\ 10^{-10}$	4	110
2^{-16}	28	14	131073		$1.243\ 10^{-10}$	4	118

Table 3: Non uniform grid results for $r_0 = 0.5$ as a function of the control parameter $\bar{\lambda} = \lambda^{1/3}$ ($s = 2$).

	grid data			Simple evaluation	Fast Evaluation		
$\bar{\lambda}$	K	K_0	N	$E^{(K)}$	$E^{(K)}$	K_S	W/N
2^{-3}	2	1	21	$2.350\ 10^{-2}$	$1.375\ 10^{-2}$	0	17
2^{-4}	4	2	41	$6.913\ 10^{-4}$	$1.035\ 10^{-3}$	1	31
2^{-5}	6	3	79	$2.873\ 10^{-4}$	$5.271\ 10^{-4}$	2	42
2^{-6}	8	4	153	$7.818\ 10^{-5}$	$1.336\ 10^{-4}$	3	55
2^{-7}	10	5	297	$2.383\ 10^{-5}$	$2.511\ 10^{-5}$	3	61
2^{-8}	12	6	581	$6.166\ 10^{-6}$	$7.103\ 10^{-6}$	3	70
2^{-9}	14	7	1147	$1.618\ 10^{-6}$	$1.808\ 10^{-6}$	3	78
2^{-10}	16	8	2277	$4.078\ 10^{-7}$	$4.122\ 10^{-7}$	3	82
2^{-11}	18	9	4533	$1.032\ 10^{-7}$	$1.066\ 10^{-7}$	4	88
2^{-12}	20	10	9041	$2.587\ 10^{-8}$	$2.659\ 10^{-8}$	4	94
2^{-13}	22	11	18055	$6.490\ 10^{-9}$	$6.506\ 10^{-9}$	4	100
2^{-14}	24	12	36081		$1.636\ 10^{-9}$	4	106
2^{-15}	26	13	72129		$4.075\ 10^{-10}$	4	115
2^{-16}	28	14	144221		$1.031\ 10^{-10}$	4	123

Table 4: Non uniform grid results for $r_0 = 0.6$ as a function of the control parameter $\bar{\lambda} = \lambda^{1/3}$ ($s = 2$).

$\bar{\lambda}$	K	K_0	p_{K_0}	m_{K_0}
2^{-3}	2	1	4	0
2^{-4}	4	2	4	0
2^{-5}	6	3	4	1
2^{-6}	8	4	4	1
2^{-7}	10	5	6	2
2^{-8}	12	6	6	2
2^{-9}	14	7	6	4
2^{-10}	16	8	6	4
2^{-11}	18	9	8	5
2^{-12}	20	10	8	6
2^{-13}	22	11	8	6
2^{-14}	24	12	10	7
2^{-15}	26	13	10	7
2^{-16}	28	14	10	9

Table 5: p and m used on grid K_0 as a function of the control parameter $\bar{\lambda} = \lambda^{1/3}$, for the results shown in Table 2. ($s = 2$, $r_0 = 1.0$).

References

[1] D. Bai and A. Brandt, *Local mesh refinement multilevel techniques*, SIAM J. Sci. Stat. Comput., 8, 2 (1986), pp. 109–134.

[2] A. Brandt, *Multi-level adaptive solutions to boundary value problems*, Math. Comp., 31 (1977), pp. 333–390.

[3] A. Brandt, *Guide to Multigrid develpment*, In: Multigrid Methods (Proc. Köln-Porz, 1981), edited by W. Hackbusch and U. Trottenberg, Lecture Notes in Math., 960, Springer Verlag, New York, 1982, pp. 220–312, also included in *GMD Studien No. 85, Gesellschaft für Mathematik und Datenverarbeitung MBH, Bonn, Germany* (1984).

[4] A. Brandt, and A. A. Lubrecht, *Multilevel matrix multiplication and the fast solution of integral equations*, J. Comput. Phys., 90, 2 (1990), pp. 348–370.

[5] A. Brandt, *Multilevel computation of integral transforms and particle interactions with oscillatory kernels*, Computer Physics Communications, 65 (1991), pp. 24–38.

[6] A. Brandt, C.H. Venner, *Multilevel evaluation of integral transforms with asymptotically smooth kernels*, Report WI/GC-2, Gauss Center for Scientific Computation (1995), SIAM *J. of Sci. Comp.*, in press.

[7] K.L. Johnson, K. Kendall, and A.D. Roberts, *Surface energy and the contact of elastic solids*, Proc. R. Soc. Lond. Series A., 324 (1971), pp. 301-313.

[8] J. Bos, *Frictional heating of tribological contacts*, PhD Thesis, University of Twente, Enschede, The Netherlands, ISBN 90-9008920-9 (1995).

[9] J.A. Greenwood, *Adhesion of elastic spheres*, Report, University of Cambridge, Dep. of Engineering (1996), Proc. R. Soc. Lond. Series A., in press.

[10] C.H. Venner, *Multilevel solvers for the EHL line and point contact problems* PhD Thesis, University of Twente, Enschede, The Netherlands, ISBN 90-9003974-0 (1991).

[11] C.H. Venner and A.A. Lubrecht, *Numerical simulation of a transverse ridge in a circular EHL contact under rolling/sliding,* ASME J. of Tribology, 116, pp. 751-761, (1994).