

Problema esterno di Dirichlet

Metodo degli Elementi di Contorno

Alice Daldossi, # Matricola 510972
Università di Pavia

Relazione del corso
Metodi Numerici Avanzati per le Equazioni alle Derivate Parziali

Introduzione

Tanto in campo medico quanto in campo industriale risulta importante creare modelli della propagazione delle onde acustiche, ad esempio per terapie non invasive oppure comunicazione subacquea, imaging a ultrasuoni oppure contenimento acustico.
Sia la pressione acustica p_{\approx} , la densità ρ_{\approx} del mezzo in cui l'onda si propaga, la velocità potenziale ϕ che le componenti della velocità $v_{\approx,j}$ soddisfano la stessa equazione delle onde, che allora si può generalizzare per un qualsiasi campo scalare U :

$$\frac{1}{c^2} \frac{\partial^2 U}{\partial t^2} - \Delta U = 0.$$

In questa relazione sono interessata a cercare le soluzioni armoniche nel tempo, quindi quelle del tipo

$$U(\mathbf{x}, t) = \Re\{u(\mathbf{x})e^{-i\omega t}\},$$

con $\omega > 0$ frequenza temporale e u campo a valori complessi dipendente dalla posizione in spazio \mathbf{x} ma non dalla variabile temporale t . Per la risoluzione di questo problema ci si riconduce all'equazione di Helmholtz

$$\Delta u + k^2 u = 0.$$

Infatti vale che, se $U(\mathbf{x}, t)$ è una soluzione armonica in tempo dell'equazione delle onde, allora $u(\mathbf{x})$ è una soluzione dell'equazione di Helmholtz con numero d'onda $k := \frac{\omega}{c} > 0$. Questo implica che l'equazione di Helmholtz descrive tutte le soluzioni armoniche in tempo dell'equazione delle onde.

L'equazione di Helmholtz viene risolta introducendo un'equazione integrale al contorno, ovvero *boundary integral equation* (BIE), a cui viene applicato il Metodo degli Elementi di Contorno, ovvero *Boundary Element Method* (BEM). La peculiarità di questo metodo numerico sta nel fatto che richiede la discretizzazione del dominio solo ai bordi: ciò riduce le dimensioni del problema e semplifica i dati richiesti in ingresso.
In particolare mi concentrerò sulla seguente classe di problemi al valore al bordo.

Definizione 1. Problema di Dirichlet esterno - EDP. Sia Ω_- un dominio aperto, Lipschitz e limitato, Γ il suo bordo e Ω_+ il suo complementare aperto. Siano poi $k > 0$ un parametro e $g_D \in H^{\frac{1}{2}}(\Gamma)$. Diciamo che $u \in H_{\text{loc}}^1(\overline{\Omega_+})$ soddisfa il problema Dirichlet-Helmholtz esterno se

$$\begin{cases} \Delta u + k^2 u = 0 & \text{in } \Omega_+, \\ \gamma^+ u = g_D & \text{su } \Gamma \\ u \text{ è radiativa.} \end{cases} \quad (1)$$

L'obiettivo di questa relazione è illustrare come ho implementato in Matlab il Metodo degli Elementi di Contorno nel caso di funzioni base dello spazio $H^{-\frac{1}{2}}(\Gamma)$ prese lineari a tratti. Inizio trattando alcuni possibili metodi di discretizzazione del bordo: uniforme e adattivo. Successivamente individuo una soluzione approssimata dell'equazione integrale al contorno considerata, usando il metodo di Galerkin e quello di collocazione; entrambi i metodi vengono poi adattati nel caso in cui si vuole operare una *singularity extraction*. Quindi, ricavo i valori dell'onda cercata sul dominio. Ho concluso con uno studio degli errori in funzione della scelta del metodo di discretizzazione oppure della presenza o meno della *singularity extraction*.

Tutti i codici si possono trovare al link <https://github.com/Daldossi/BEM>.

Indice

1 BEM per BIE	2
1.1 Definizione di V_N e partizionamento di Γ	3
1.2 Approssimazione della soluzione ψ di (BIE)	4
1.2.1 Collocazione-BEM	6
1.2.2 Galerkin-BEM	7
1.2.3 Singularity extraction	7
1.3 Formula di rappresentazione	8
1.4 Convergenza	9

1 BEM per BIE

Per risolvere il problema ai valori al bordo (BVP) nel caso del problema esterno di Dirichlet (EDP), lo riformulerò con un'equazione al bordo integrale (BIE) che mi permetterà di trovare un'approssimazione della soluzione tramite il metodo numerico degli elementi di contorno (BEM).

Siano u^{Inc} l'onda incidente, u^{Scat} l'onda riflessa e $u^{\text{Tot}} = u^{\text{Inc}} + u^{\text{Scat}}$ l'onda totale. Siano $\Omega_- \subset \mathbb{R}^2$ un dominio Lipschitz limitato, $\Omega_+ := \mathbb{R}^2 \setminus \overline{\Omega_-}$ e $\Gamma := \partial\Omega_-$ (su questo bordo considererò solo condizioni *sound-soft*¹). Nel mio problema l'onda incidente u^{Inc} su Ω_- rappresenta il dato, mentre l'onda riflessa u^{Scat} è l'incognita che deve soddisfare l'equazione di Helmholtz sul dominio di propagazione Ω_+ e le condizioni sound-soft $\gamma(u^{\text{Tot}}) = 0$ su Γ . Inoltre, impongo che u^{Scat} sia radiativa, cioè che soddisfi la condizione radiativa di Sommerfeld:

$$|\partial_r u^{\text{Scat}} - ik u^{\text{Scat}}| = o(r^{-\frac{1}{2}}) \quad r \rightarrow \infty. \quad (2)$$

Allora il problema si può riscrivere:

$$\text{trovare } u^{\text{Scat}} \in H_{\text{loc}}^1(\overline{\Omega_+}) \text{ tale che} \begin{cases} \Delta u^{\text{Scat}} + k^2 u^{\text{Scat}} = 0 & \text{in } \Omega_+, \\ \gamma^+(u^{\text{Scat}} + u^{\text{Inc}}) = 0 & \text{su } \Gamma, \\ u^{\text{Scat}} \text{ è radiativa.} \end{cases} \quad (3)$$

Nel caso 2-dimensionale, per un valore $\mathbf{y} \in \mathbb{R}^2$ fissato, una soluzione radiativa fondamentale dell'equazione di Helmholtz è:

$$\Phi_k(\mathbf{x}, \mathbf{y}) := \frac{i}{4} H_0^{(1)}(k|\mathbf{x} - \mathbf{y}|), \quad \mathbf{x} \neq \mathbf{y} \in \mathbb{R}^2, \quad (4)$$

dove $H_0^{(1)}$ è la funzione di Hankel di ordine 0. Dal momento che ogni combinazione lineare delle componenti di Φ_k , $\Phi_k(\cdot, \mathbf{y}_j)$ soddisfa ancora l'equazione di Helmholtz in Ω_+

¹Il caso *sound-soft* si verifica quando la pressione acustica sul bordo va a zero, $p_\approx = 0$; quindi, imporre condizioni *sound-soft* su Γ significa che la traccia di Dirichlet dell'onda totale u^{Tot} si azzera su Γ .

ed è radiativa, allora posso anche considerare la seguente combinazione lineare continua delle soluzioni fondamentali

$$(\mathcal{S}\psi)(\mathbf{x}) := \int_{\Gamma} \Phi_k(\mathbf{x}, \mathbf{y})\psi(\mathbf{y})ds(\mathbf{y}) \quad \mathbf{x} \in \Omega_+, \quad (5)$$

che chiamerò *single-layer potential*. Inoltre definisco il *single-layer operator* come

$$(S\psi)(\mathbf{x}) := \int_{\Gamma} \Psi_k(\mathbf{x}, \mathbf{y})\psi(\mathbf{y})ds(\mathbf{y}) \quad \mathbf{x} \in \Gamma. \quad (6)$$

Osservo poi che vale la relazione

$$S\psi = \gamma^+(\mathcal{S}\psi). \quad (7)$$

L'idea del metodo degli elementi di contorno (BEM) consiste nello svolgimento dei seguenti passaggi:

1. scegliere uno spazio $V_N \subset H^{-\frac{1}{2}}(\Gamma)$ N -dimensionale e le sue funzioni base;
2. partizionare la curva Γ in una mesh $\mathcal{T}_N(\Gamma)$ di N segmenti $K_1, \dots, K_N \subset \Gamma$;
3. trovare $\psi_N \in V_N$ definita su Γ tale che approssimi la soluzione dell'equazione integrale al contorno (BIE)

$$S\psi = g_D \quad \text{su } \Gamma, \quad (8)$$

dove g_D è il dato di Dirichlet per il (EDP) e ψ è detta densità;

4. ricavare u tramite la formula di rappresentazione

$$u = \mathcal{S}\psi \quad \text{in } \Omega_+. \quad (9)$$

Si può affermare che la u così identificata risulta essere una soluzione radiativa di Helmholtz in Ω_+ con $\gamma^+ u = \gamma^+ \mathcal{S}\psi = S\psi = g_D$, quindi u è la soluzione del problema di Dirichlet esterno (EDP) che si cercava.

1.1 Definizione di V_N e partizionamento di Γ

Scelgo V_N come lo spazio delle funzioni costanti a tratti su $\mathcal{T}_N(\Gamma)$. Prendo come base di V_N $\{\varphi_j\}_{j=1}^N$, dove, per ogni j , la funzione di base φ_j è definita nel seguente modo

$$\varphi_j(\mathbf{x}) := \begin{cases} 1 & \text{se } \mathbf{x} \in K_j \\ 0 & \text{se } \mathbf{x} \in \Gamma \setminus K_j \end{cases}.$$

Per partizionare Γ implemento i seguenti due metodi.

1. **Uniforme.** Nella funzione funzione `BE_u.m` (*BoundaryElements_uniform*) sono dati in input: `ver` vettore in cui ogni riga rappresenta un vertice (come valore complesso) del poligono che si sta studiando (in senso antiorario), `h` ampiezza massima di ogni elemento della discretizzazione e `i_pol` matrice a due colonne in cui la riga j -esima indica la posizione (indice) nel vettore `ver` del vertice di inizio e quella del vertice di fine del poligono corrispondente alla riga. Ci sono due output: `PT` è il vettore contenente tutte le coordinate dei punti della discretizzazione, `i_POL` invece è una matrice a due colonne in cui ogni riga indica il vertice di inizio e quello di fine di ogni elemento creato. L'algoritmo consiste nel trattare un lato l_i del poligono alla volta e decidere il suo numero di elementi n_i come l'approssimazione per eccesso di $length(l_i)/h$. Si veda [BEu.m](#).

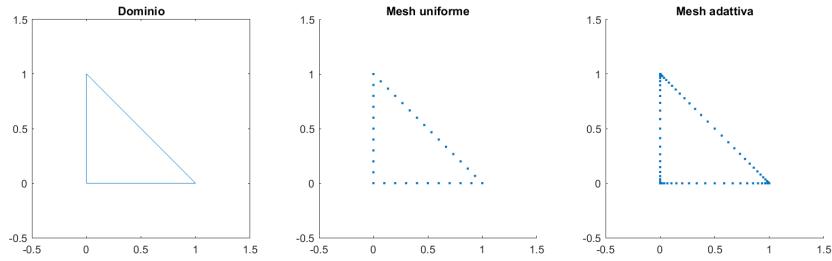


Figura 1: [script: [BoundaryElementsTest.m](#).] Confronto tra i metodi di discretizzazione per $h = 0.1$ e $z = 2$ sul triangolo di vertici $(0;0)$, $(1;0)$ e $(0;1)$.

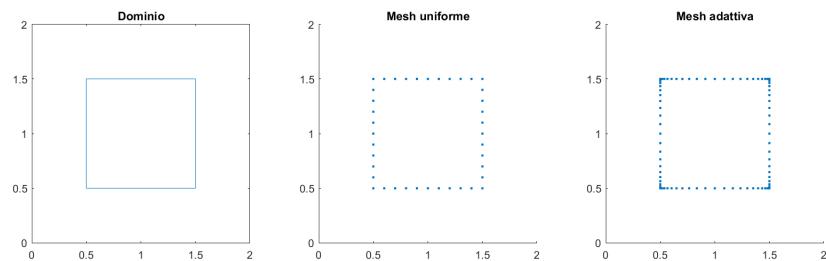


Figura 2: [script: [BoundaryElementsTest.m](#).] Confronto tra i metodi di discretizzazione per $h = 0.1$ e $z = 2$ sul quadrato di centro $(0.5;0.5)$ e lato 1.

2. **Adattivo.** La funzione `BE_a` (*BoundaryElements_adaptive*) ha output è analogo a quello del metodo precedente; l'input richiede ancora `ver` e `i_pol` che mantengono il ruolo di prima, `h` è l'ampiezza che determinerà il numero di elementi n per lato e `z` è il parametro che rappresenta la velocità con cui i punti da individuare verranno addensati verso un'estremità. L'idea dell'algoritmo è la seguente: si considera un lato l_i alla volta, si prende $n_i := \text{length}(l_i)/h$ e sull'intervallo $[0, 1]$ si scelgono n_i punti con la formula $\{\left(\frac{i}{n_i+1}\right)^z\}_{i=1}^{n_i}$ (più z è grande, più la discretizzazione concentrerà gli elementi verso zero); questi poi si trasferiscono sugli intervalli che definiscono le due metà del lato l_i . Si vedano `BEa.m` e `adaptive.m`.

Osservo che, nel caso in cui Γ non sia poligonale ma abbia una forma circolare, allora l'informazione geometrica necessaria per descriverlo deve comprendere il centro e il raggio del cerchio (o eventualmente dei cerchi), non ci saranno più vertici. Per questo motivo è stata creata la funzione `BE_u_c` (*BoundaryElements_uniform_circular*) in cui il vettore `ver` in input contiene appunto le informazioni del centro e del raggio che descrive Γ , mentre `h` e `i_pol` mantengono i loro significati di prima. Con un procedimento analogo a quello svolto per la discretizzazione adattiva, i punti sulla mesh circolare vengono individuati tramite una parametrizzazione di quelli sull'intervallo $[0, 1]$. Si veda `BEuc.m`. Nelle figure 1-5 riporto 5 diversi esempi di dominio con la loro rispettiva discretizzazione.

1.2 Approssimazione della soluzione ψ di (BIE)

Per risolvere (8), quindi trovare $\psi_N \in V_N$, si possono implementare due diversi metodi:

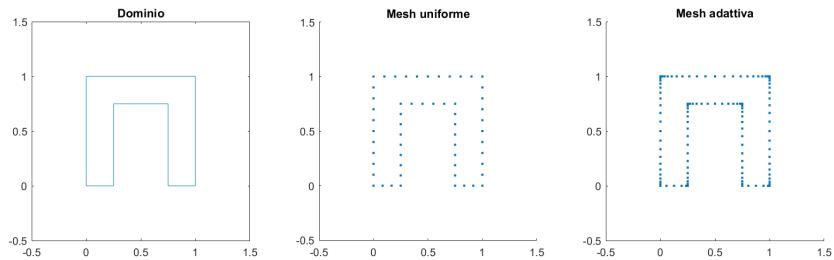


Figura 3: [script: [BoundaryElementsTest.m](#).] Confronto tra i metodi di discretizzazione per $h = 0.1$ e $z = 2$ sulla "U rovesciata".

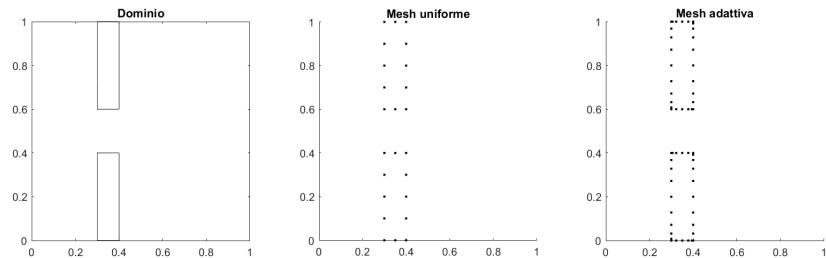


Figura 4: [script: [BoundaryElementsTest.m](#).] Confronto tra i metodi di discretizzazione per $h = 0.1$ e $z = 2$ sulla fenditura ampia 0.2 e muro spesso 0.1.

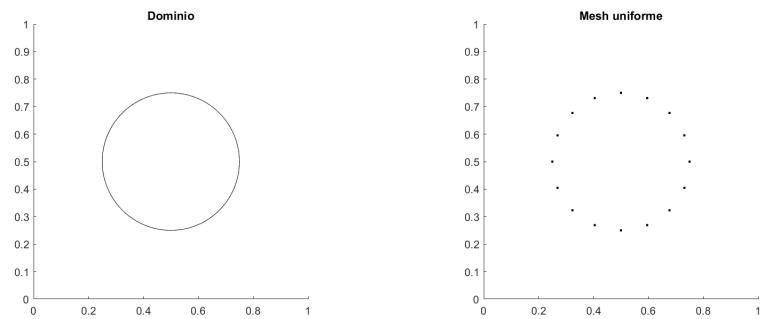


Figura 5: [script: [BoundaryElementsTest.m](#).] Confronto tra i metodi di discretizzazione per $h = 0.1$ sul cerchio di centro $(0.5; 0.5)$ e raggio 0.25.

- **Collocazione-BEM:** si scelgono N punti $x_1, \dots, x_N \in \Gamma$ e si cerca

$$\psi_N \in V_N \text{ tale che } (S\psi_N)(\mathbf{x}_j) = g_D(\mathbf{x}_j), \quad j = 1, \dots, N; \quad (10)$$

- **Galerkin-BEM:** si cerca

$$\psi_N \in V_N \text{ tale che}$$

$$\mathcal{A}(\psi_N, \xi_N) = \int_{\Gamma} (S\psi_N) \overline{\xi_N} ds = \int_{\Gamma} g_D \overline{\xi_N} ds = \mathcal{F}(\xi_N), \quad \forall \xi_N \in V_N. \quad (11)$$

Osservo che espandendo la soluzione approssimata ψ_N in coordinate ho

$$\psi_N = \sum_{j=1}^N \Psi_j \varphi_j,$$

dove $\Psi_j = (\Psi)_j$ è il j -esimo elemento del vettore $\Psi \in \mathbb{C}^n$; cioè,

$$\psi_N(\mathbf{x}) = \Psi_j \text{ se } \mathbf{x} \in K_j.$$

1.2.1 Collocazione-BEM

Presi lo spazio N -dimensionale V_N e la sua base $\{\varphi_j\}_{j=1}^N$ definiti come sopra, il metodo di collocazione prevede la definizione di N nodi \mathbf{x}_j tali che $\mathbf{x}_j \in K_j \forall j = 1, \dots, N$ in cui collochiamo l'equazione differenziale. In questo modo si risolve il sistema lineare $\underline{\underline{\mathbf{A}}}^{\text{col}} \Psi = \mathbf{F}^{\text{col}}$, dove

$$A_{j,m}^{\text{col}} := (S\varphi_m)(\mathbf{x}_j) = \int_{\Gamma} \Phi_k(\mathbf{x}_j, \mathbf{y}) \varphi_m(\mathbf{y}) ds(\mathbf{y}) = \int_{K_m} \Phi_k(\mathbf{x}_j, \mathbf{y}) ds(\mathbf{y}), \quad (12)$$

$$F_j^{\text{col}} := g_D(\mathbf{x}_j). \quad (13)$$

Ora si supponga che Ω_- sia un poligono convesso e che ogni elemento della mesh sia un segmento dritto. Per $j = 1, \dots, N$, l'elemento K_j ha estremità \mathbf{p}_j e \mathbf{p}_{j+1} , lunghezza $h_j := |\mathbf{p}_{j+1} - \mathbf{p}_j|$ ed è parametrizzato da $\mathbf{X}_j : (0, h_j) \rightarrow K_j$, $\mathbf{X}_j(s) := \mathbf{p}_j + s\boldsymbol{\tau}_j$, dove $\boldsymbol{\tau}_j := \frac{\mathbf{p}_{j+1} - \mathbf{p}_j}{|\mathbf{p}_{j+1} - \mathbf{p}_j|}$ è il vettore tangente unitario a K_j . Le equazioni (12) e (13) che definiscono il sistema sono svolte con formule di quadratura:

$$A_{j,m}^{\text{col}} = \frac{i}{4} \int_0^{h_m} H_0^{(1)}(k|\mathbf{p}_m + s\boldsymbol{\tau}_m - \mathbf{x}_j|) ds(\mathbf{y}) = \frac{i}{4} h_m \int_0^1 H_0^{(1)}(k|\mathbf{p}_m + h_m \boldsymbol{\tau}_m t - \mathbf{x}_j|) dt, \quad (14)$$

$$F_j^{\text{col}} = g_D(\mathbf{x}_j). \quad (15)$$

Dal momento che l'integrale che si ottiene per $m = j$ è singolare, allora:

$$A_{j,j}^{\text{col}} = \int_{K_j} \Phi_k(\mathbf{x}_j, \mathbf{y}) ds(\mathbf{y}) = \frac{i}{2} \int_0^{\frac{h_j}{2}} H_0^{(1)}(ks) ds = \frac{ih_j}{4} \int_0^1 H_0^{(1)}\left(k \frac{h_j}{2} t\right) dt. \quad (16)$$

Si veda il codice **BEMcol.m**. Qui ho preso in input: **PT** è il vettore tale che ogni entrata è un valore complesso che indica le coordinate di un punto della mesh, **POL** è una matrice a due colonne tale che ogni riga indica gli indici sul vettore **PT** del vertice di inizio e di quello di fine del poligono corrispondente a quella riga, **k** è il parametro dell'equazione di Helmholtz, **g_D** è la funzione dato sul bordo Γ e prende in input un valore complesso ², **x_val** è il dominio riportato in numeri complessi sui cui si valuta l'approssimazione dell'onda scatterata (si veda §1.3), **flag** è una stringa che indica se svolgere ('se') o no ('no') la *singularity extraction* (si veda §1.2.3). L'output: **u_scat** è l'approssimazione dell'onda scatterata calcolata tramite il metodo di collocazione.

²In questa relazione userò sempre le onde piane, quindi $e^{ik\mathbf{x} \cdot \mathbf{d}}$, con \mathbf{d} direzione dell'onda. Con piccoli adattamenti ai codici, si potrebbe sostituire con una funzione di Herglotz $\int_0^{2\pi} g(\phi) e^{ik(x_1 \cos \phi + x_2 \sin \phi)} d\phi$, dove $g \in L^2(0, \pi)$ è il kernel.

1.2.2 Galerkin-BEM

Il metodo di Galerkin si presenta come la “restrizione” del problema variazionale

$$\mathcal{A}(\psi, \xi) := \langle S\psi, \xi \rangle_{H^{\frac{1}{2}}(\Gamma) \times H^{-\frac{1}{2}}(\Gamma)} = \langle g_D, \xi \rangle_{H^{\frac{1}{2}}(\Gamma) \times H^{-\frac{1}{2}}(\Gamma)} =: \mathcal{F}(\xi) \quad \forall \xi \in H^{-\frac{1}{2}}(\Gamma). \quad (17)$$

a un sottospazio finito dimensionale V_N con base $\{\varphi_j\}_{j=1}^N$, dove V_N e $\{\varphi_j\}_{j=1}^N$ sono presi come sopra. Quindi l’equazione (11) che chiedo per ogni elemento di V_N si riduce a N equazioni lineari che creano il sistema lineare $\underline{\underline{\mathbf{A}}}^{\text{Gal}} \Psi = \mathbf{F}^{\text{Gal}}$ in cui

$$A_{j,m}^{\text{Gal}} := \mathcal{A}(\varphi_m, \varphi_j) = \langle S\varphi_m, \varphi_j \rangle_{\Gamma} = \int_{\Gamma} \int_{\Gamma} \Phi_k(\mathbf{x}, \mathbf{y}) \varphi_m(\mathbf{y}) \overline{\varphi_j}(\mathbf{x}) d\mathbf{s}(\mathbf{x}), \quad (18)$$

$$F_j^{\text{Gal}} := \mathcal{F}(\varphi_j) = \langle g_D, \varphi_j \rangle_{\Gamma} = \int_{\Gamma} g_D(\mathbf{x}) \overline{\varphi_j}(\mathbf{x}) d\mathbf{s}(\mathbf{x}). \quad (19)$$

Similmente a quanto svolto per il metodo di collocazione si ricavano le formule

$$F_j^{\text{Gal}} = \int_{K_j} g_D(x) ds(x) = h_j \int_0^1 g_D(\mathbf{p}_j + h_j \boldsymbol{\tau}_j t) dt; \quad (20)$$

$$A_{j,m}^{\text{Gal}} = \int_{K_j} \int_{K_m} \Phi_k(\mathbf{x}, \mathbf{y}) ds(x) ds(y) = \frac{i}{4} h_j h_m \int_0^1 \int_0^1 H_0^{(1)}(k|\mathbf{p}_j + h_j \boldsymbol{\tau}_m \eta|) d\xi d\eta. \quad (21)$$

In questo caso ci sono più situazioni che portano a singolarità: oltre che lungo la diagonale principale $j = m$, anche lungo quella secondaria. Infatti i problemi sono generati anche dal fatto che due elementi possono condividere un estremo. Quindi gli integrali diventano:

$$A_{j,j}^{\text{Gal}} = i \int_0^{\frac{h_j}{\sqrt{2}}} \left(\frac{h_j}{\sqrt{2}} - \xi \right) H_0^{(1)}(\sqrt{2}k\xi) d\xi = i \left(\frac{h_j}{\sqrt{2}} \right)^2 \int_0^1 (1 - \eta) H_0^{(1)}(k \frac{h_j}{\sqrt{2}} \eta) d\eta \\ \text{con } \xi = \frac{s-t}{\sqrt{2}} \text{ e } \eta = \frac{s+t}{\sqrt{2}}, \quad (22)$$

$$A_{j,j+1}^{\text{Gal}} = \int_0^{h_j} \left(\int_0^{h_{j+1}} \frac{1}{4} H_0^{(1)}(k|h_j - s + t|) dt \right) ds, \quad (23)$$

da risolvere ancora una volta con la regola di Gauss.

Si veda il codice [BEMgal.m](#). In questo caso i valori richiesti in input sono gli stessi della funzione [BEM_gal.m](#). L’output [u_scat](#) è l’approssimazione dell’onda scatterata calcolata tramite il metodo di Galerkin.

1.2.3 Singularity extraction

Un metodo alternativo per risolvere il problema della singolarità negli integrali è la tecnica chiamata *singularity extraction* che sfrutta la seguente formula asintotica per $z \searrow 0$

$$H_0^{(1)}(z) = i \frac{2}{\pi} \ln z + 1 + i \frac{2}{\pi} (\gamma - 2) + O(z^2), \quad (24)$$

con $\gamma \approx 0.57721$ costante di Eulero. Infatti, dato che per gli elementi sulla diagonale si ha $|x - y| \rightarrow 0$, allora lì si può usare che

$$\Phi_k(x, y) = -\frac{1}{2\pi} \ln |x - y| + \frac{i}{4} - \frac{1}{2\pi} \ln \frac{k}{2} - \frac{\gamma}{2\pi} + O(|x - y|^2 \ln |x - y|). \quad (25)$$

Quindi le formule per ottenere $A_{j,j}^{\text{Col}}$ e $A_{j,j}^{\text{Gal}}$ si possono scrivere anche sfruttando la *singularity extraction*, ottenendo:

$$\begin{aligned}
A_{j,j}^{\text{Col}} &= \int_{K_j} \phi_k(x_j, y) ds(y) = 2 \int_0^{\frac{h_j}{2}} \frac{i}{4} H_0^{(1)}(ks) ds = \\
&\approx 2 \int_0^{\frac{h_j}{2}} \left(-\frac{1}{2\pi} \ln s + \frac{i}{4} - \frac{1}{2\pi} \ln \frac{k}{2} - \frac{\gamma}{2\pi} \right) ds = \\
&= 2 \left[\left(\frac{i}{4} - \frac{1}{2\pi} \ln \frac{k}{2} - \frac{\gamma}{2\pi} \right) \frac{h_j}{2} - \frac{1}{2\pi} \int_0^{\frac{h_j}{2}} \ln s ds \right] = \\
&\stackrel{(1)}{=} 2 \frac{h_j}{2} \left(\frac{i}{4} - \frac{1}{2\pi} \ln \frac{k}{2} - \frac{\gamma}{2\pi} - \frac{1}{2\pi} \ln \frac{h_j}{2} + \frac{1}{2\pi} \right) = \\
&= \frac{h_j}{2\pi} \left(\frac{\pi i}{2} - \ln \frac{k}{2} - \gamma - \ln \frac{h_j}{2} + 1 \right);
\end{aligned}$$

$$\begin{aligned}
A_{j,j}^{\text{Gal}} &= i \int_0^{\frac{h_j}{\sqrt{2}}} \left(\frac{h_j}{\sqrt{2}} - \xi \right) H_0^{(1)}(\sqrt{2}k\xi) d\xi = \\
&\approx 4 \int_0^{\frac{h_j}{\sqrt{2}}} \left(\frac{h_j}{\sqrt{2}} - \xi \right) \left(-\frac{1}{2\pi} \ln(\sqrt{2}\xi) + \frac{i}{4} - \frac{1}{2\pi} \ln \frac{k}{2} - \frac{\gamma}{2\pi} \right) d\xi = \\
&= 4 \left(\frac{i}{4} - \frac{1}{2\pi} \ln \frac{k}{2} - \frac{\gamma}{2\pi} \right) \int_0^{\frac{h_j}{\sqrt{2}}} \left(\frac{h_j}{\sqrt{2}} - \xi \right) d\xi - \frac{4}{2\pi} \int_0^{\frac{h_j}{\sqrt{2}}} \left(\frac{h_j}{\sqrt{2}} - \xi \right) \ln(\sqrt{2}\xi) d\xi = \\
&= \frac{h_j^2}{4} 4 \left(\frac{i}{4} - \frac{1}{2\pi} \ln \frac{k}{2} - \frac{\gamma}{2\pi} \right) - \frac{4}{2\pi} \frac{h_j}{\sqrt{2}} \int_0^{\frac{h_j}{\sqrt{2}}} \ln(\sqrt{2}\xi) d\xi + \frac{4}{2\pi} \int_0^{\frac{h_j}{\sqrt{2}}} \xi \ln(\sqrt{2}\xi) d\xi = \\
&\stackrel{(2)}{=} h_j^2 \left[\frac{i}{4} - \frac{1}{2\pi} \ln \frac{k}{2} - \frac{\gamma}{2\pi} - \frac{1}{2\pi} \left(2 \ln \frac{h_j}{\sqrt{2}} - 2 + \ln 2 \right) \right] + \frac{4}{2\pi} \int_0^{\frac{h_j}{\sqrt{2}}} \xi \ln(\sqrt{2}\xi) d\xi = \\
&\stackrel{(3)}{=} h_j^2 \left[\frac{i}{4} - \frac{1}{2\pi} \ln \frac{k}{2} - \frac{\gamma}{2\pi} - \frac{1}{2\pi} \left(2 \ln \frac{h_j}{\sqrt{2}} - 2 + \ln 2 \right) + \frac{1}{4\pi} \left(2 \ln \frac{h_j}{\sqrt{2}} - 1 + \ln 2 \right) \right] = \\
&= \frac{h_j^2}{2\pi} \left(\frac{i\pi}{2} - \ln \frac{k}{2} - \gamma - 2 \ln \frac{h_j}{\sqrt{2}} + 2 - \ln 2 + \ln \frac{h_j}{\sqrt{2}} - \frac{1}{2} + \frac{1}{2} \ln 2 \right) = \\
&= \frac{h_j^2}{2\pi} \left(\frac{i\pi}{2} - \ln \frac{k}{2} - \gamma - \ln \frac{h_j}{\sqrt{2}} - \frac{1}{2} \ln 2 + \frac{3}{2} \right) = \\
&= \frac{h_j^2}{2\pi} \left(\frac{i\pi}{2} - \ln \frac{k}{2} - \gamma - \ln h_j + \frac{3}{2} \right).
\end{aligned}$$

In (1) ho usato la regola generale $\int_0^a \ln x dx = a(\ln a - 1)$; in (2) invece $\int_0^a \ln(\sqrt{2}\xi) d\xi = \frac{a}{2}(2 \ln a - 2 + \ln 2)$ e in (3) $\int_0^a \xi \ln(\sqrt{2}\xi) d\xi = \frac{a^2}{4}(2 \ln a - 1 + \ln 2)$.

Si vedano ancora i codici `BEM_gal.m` e `BEM_col.m`, guardando il caso di `flag = 'se'`.

1.3 Formula di rappresentazione

Dopo aver ottenuto il valore di Ψ si ricava l'approssimazione della soluzione u :

$$u_N(\mathbf{x}) := (\mathcal{S}\psi_N)(\mathbf{x}) = \int_{\Gamma} \Phi_k(\mathbf{x}, \mathbf{y}) \psi_N(\mathbf{y}) ds(\mathbf{y}) = \sum_{j=1}^N \Psi_j \int_{K_j} \Phi_k(\mathbf{x}, \mathbf{y}) ds(\mathbf{y}), \quad \mathbf{x} \in \Omega_+.
\tag{26}$$

Tramite le formule di quadratura si giunge all'espressione seguente

$$\begin{aligned}
u_N^{\text{Scat}}(\mathbf{x}) &= (\mathcal{S}\psi)(\mathbf{x}) = \left(\mathcal{S} \sum_{j=1}^n \psi_j \varphi_j \right) (\mathbf{x}) = \int_{\Gamma} \left(\sum_{j=1}^n \psi_j \varphi_j(\mathbf{y}) \right) \phi_k(\mathbf{x}, \mathbf{y}) d\mathbf{s}(\mathbf{y}) = \\
&= \sum_{j=1}^n \psi_j \int_{K_j} \phi_k(\mathbf{x}, \mathbf{y}) d\mathbf{s}(\mathbf{y}) = \\
&= \frac{i}{4} \sum_{j=1}^n \psi_j \int_0^{h_j} H_0^{(1)}(k|\mathbf{p}_j + h_j \boldsymbol{\tau}_j t - \mathbf{x}|) dt = \\
&= \frac{i}{4} \sum_{j=1}^n \psi_j h_j \int_0^1 H_0^{(1)}(k|\mathbf{p}_j + h_j \boldsymbol{\tau}_j t - \mathbf{x}|) dt = \\
&= \frac{i}{4} \int_0^1 \sum_{j=1}^n \psi_j h_j H_0^{(1)}(k|\mathbf{p}_j + h_j \boldsymbol{\tau}_j t - \mathbf{x}|) dt.
\end{aligned}$$

Per il calcolo di u_N ho implementato la funzione `ScatteredWave.m` in cui in input sono richiesti: `PT` vettore in cui ogni entrata è un valore complesso che indica le coordinate di un punto della mesh, `POL` matrice a due colonne tale che ogni riga indica gli indici sul vettore `PT` del vertice di inizio e di quello di fine del poligono corrispondente a quella riga, `x_val` dominio riportato in numeri complessi sui cui si valuta l'approssimazione dell'onda scatterata, `psi` vettore dei coefficienti che rappresenta la densità, `k` coefficiente dell'equazione di Helmholtz. L'output: `u_scat` è l'approssimazione dell'onda scatterata. Si veda il codice `ScatteredWave.m`.

Nelle figure 6-11 sono riportati gli scatter delle onde nel caso di diversi domini.

1.4 Convergenza

Al termine ho verificato la convergenza. Al variare dell'ampiezza della mesh $h = 2^{-i}$ con $i = 4, \dots, 10$ ho stimato l'errore in norma L^2 prendendo come soluzione di riferimento dell'onda totale quella descritta da `MPSpackBenchmarkSquareScatt.m` e ho considerato solo la parte esterna al cerchio unitario. Lo studio è stato svolto sia utilizzando collocazione che Galerkin, confrontando, sia nel caso di mesh adattiva che uniforme, la soluzione ricavata con la *singularity extraction* con quella ricavata senza.

Dalla figura 12 si evince che la discretizzazione adattiva è migliore sia col metodo di collocazione che con quello di Galerkin.

Il codice `Error.m` prende in input: `flag1` è una stringa che indica se approssimare l'onda scatterata con il metodo di collocazione ('`col`') oppure con Galerkin ('`gal`'), `flag2` è un'altra stringa che dice se usare la *singularity extraction* ('`se`') oppure no ('`no`''), `PT` è il vettore in cui ogni entrata è un valore complesso che indica le coordinate di un punto della mesh, `POL` è una matrice a due colonne tale che ogni riga indica gli indici sul vettore `PT` del vertice di inizio e di quello di fine del poligono corrispondente a quella riga, `k` è il coefficiente dell'equazione di Helmholtz, `PWave` è una funzione a una variabile complessa che descrive il dato sul bordo Γ , `Omega` sono i punti (descritti come numeri complessi) su cui si valuta l'approssimazione dell'onda scatterata, `u` è l'onda scatterata e totale di riferimento, `ui` è l'onda incidente di riferimento, `X` l'ascissa della griglia di valutazione, `Y` l'ordinata della griglia di valutazione, `R` è il raggio del cerchio al cui esterno si calcola l'errore. L'output è il valore `E` dell'errore in L^2 tra l'onda approssimata e quella di riferimento.

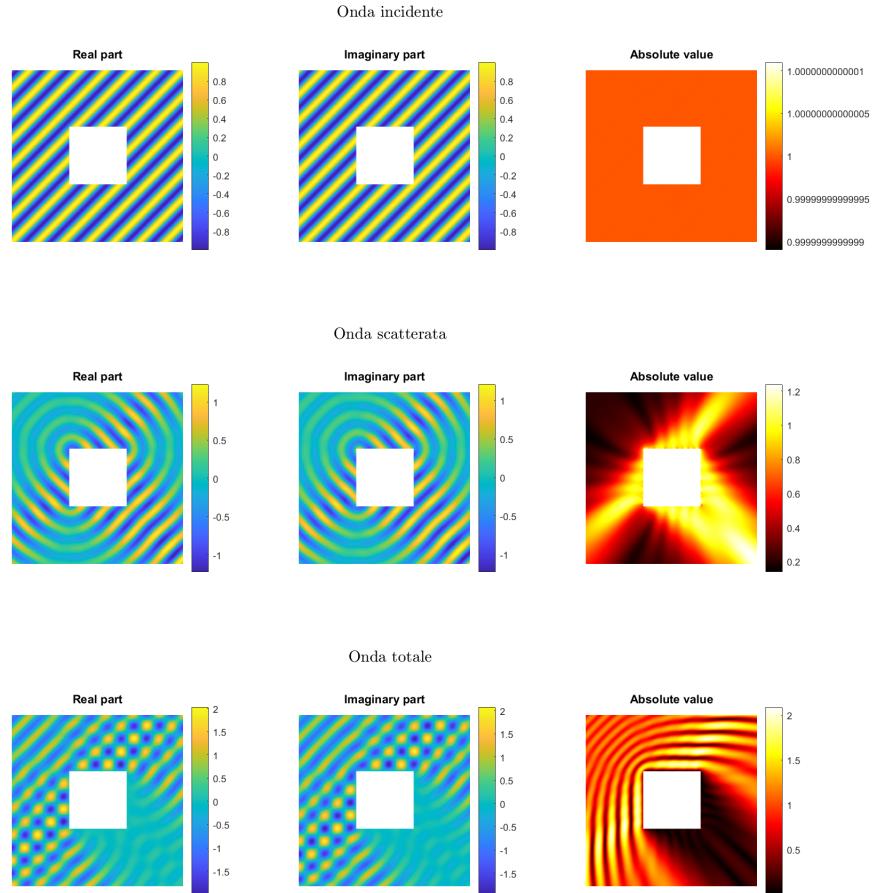


Figura 6: [script: [solution.m](#).] Plot su $[-1.5, 1.5]^2$ di un'onda piana con direzione $\alpha = -\frac{\pi}{4}$ e parametro $k = 20$ su una mesh di 150 punti con un vincolo quadrato. Ho usato il metodo di collocazione senza *singularity extraction* con discretizzazione uniforme e prendendo l'ampiezza $h = 0.1$.

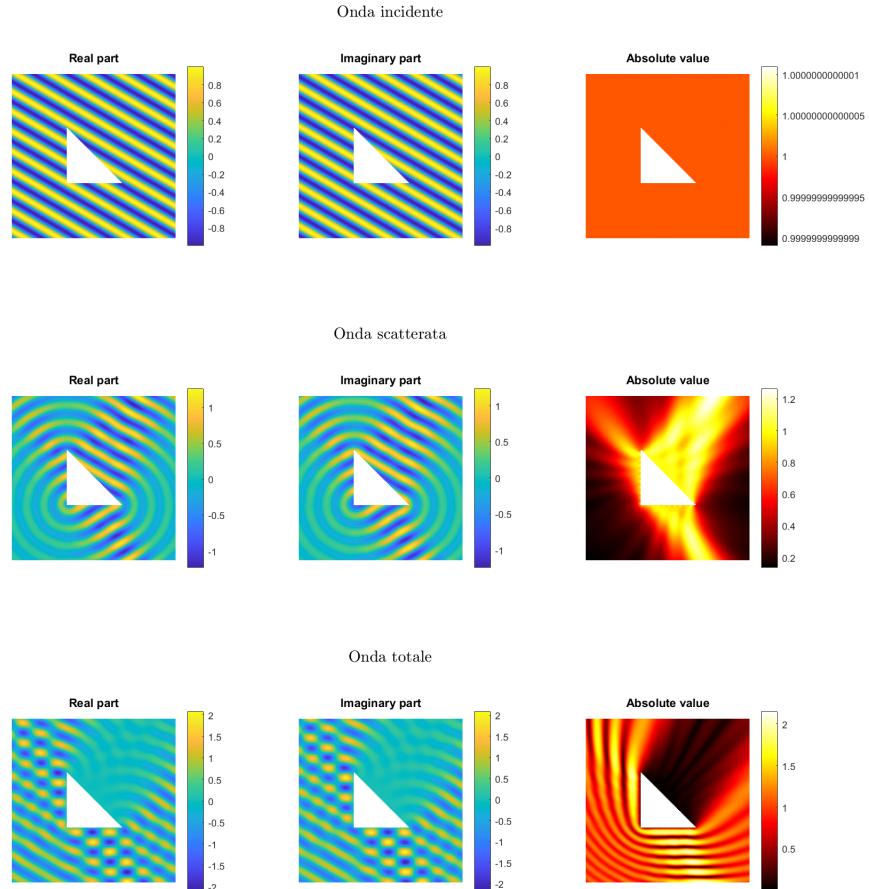


Figura 7: [script: [solution.m](#).] Plot su $[-1, 2]^2$ di un'onda piana con direzione $\alpha = \frac{\pi}{3}$ e parametro $k = 20$ su una mesh di 150 punti con un vincolo triangolare. Ho usato il metodo di collocazione senza *singularity extraction* con discretizzazione uniforme e prendendo l'ampiezza $h = 0.1$.

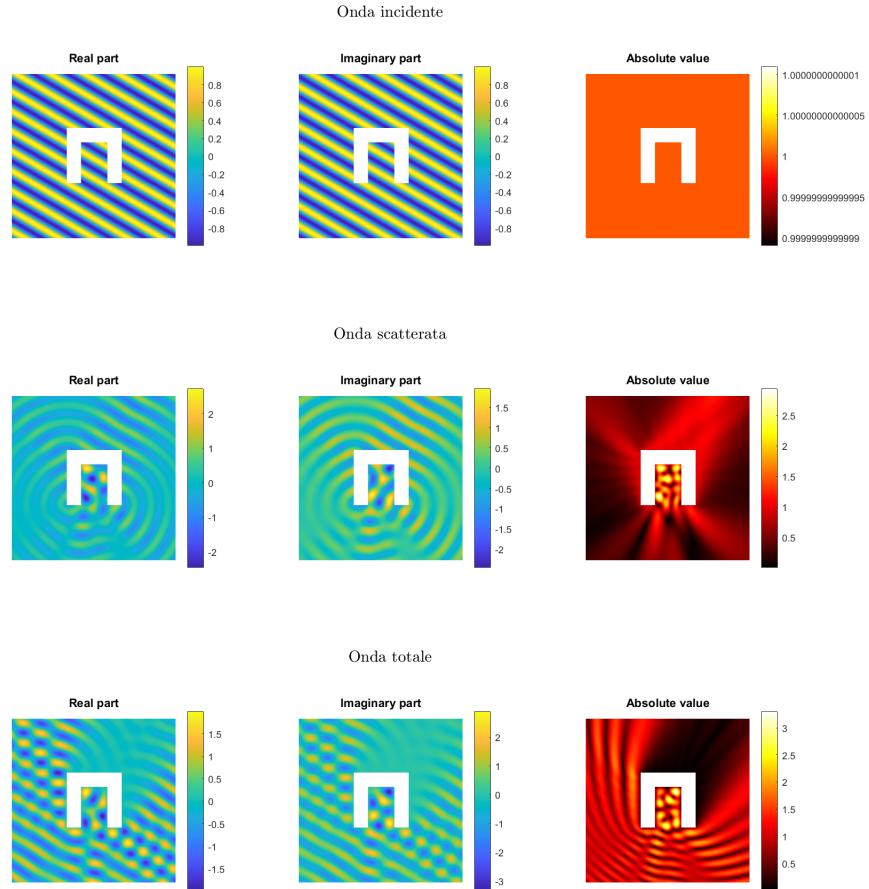


Figura 8: [script: [solution.m](#).] Plot su $[-1, 2]^2$ di un'onda piana con direzione $\alpha = \frac{\pi}{3}$ e parametro $k = 20$ su una mesh di 150 punti con un vincolo a "U" rovesciata. Ho usato il metodo di collocazione senza *singularity extraction* con discretizzazione uniforme e prendendo l'ampiezza $h = 0.1$.

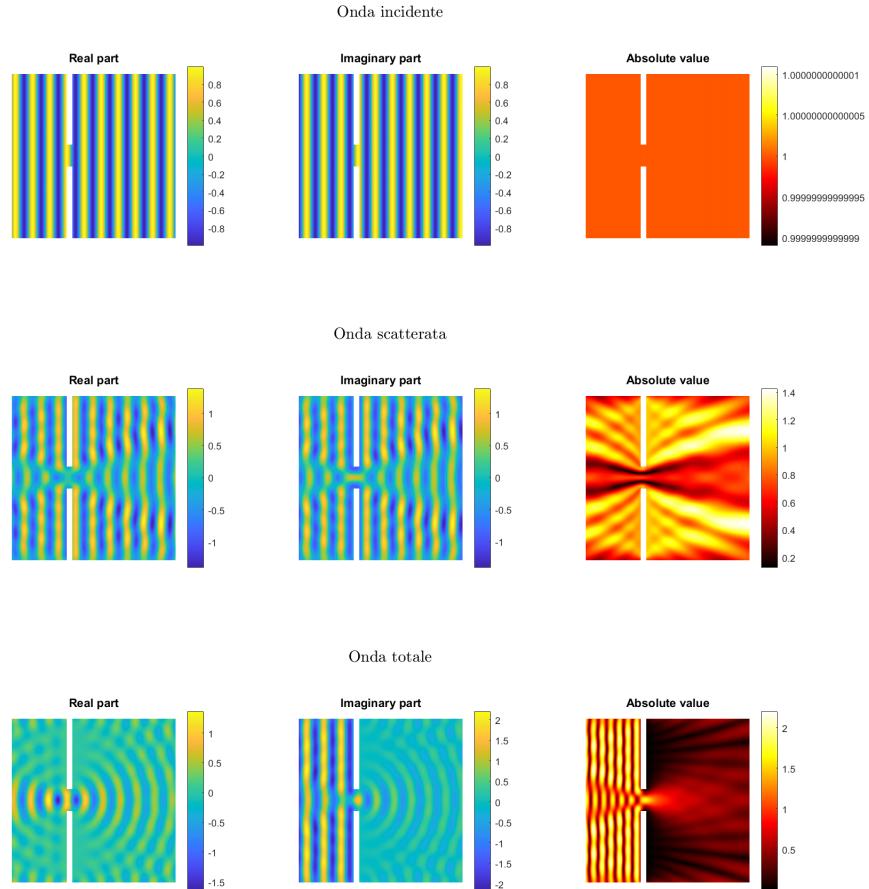


Figura 9: [script: [solution.m](#).] Plot su $[-1, 2]^2$ di un'onda piana con direzione $\alpha = 0$ e parametro $k = 20$ su una mesh di 150 punti con un vincolo con una fenditura. Ho usato il metodo di collocazione senza *singularity extraction* con discretizzazione uniforme e prendendo l'ampiezza $h = 0.1$.

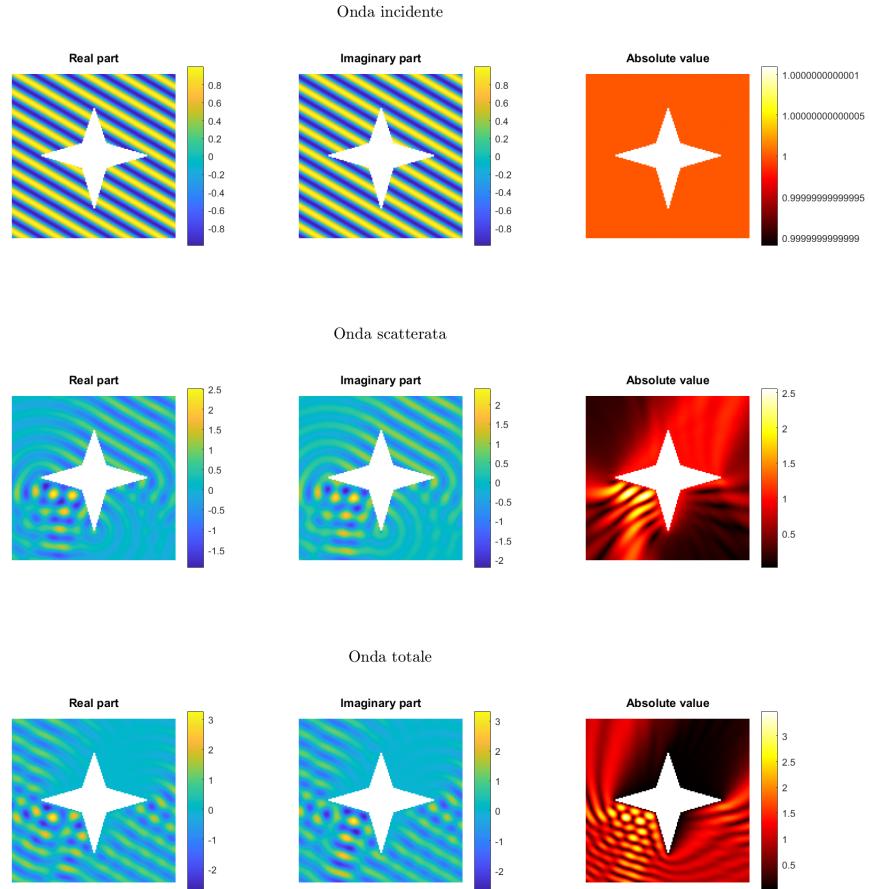


Figura 10: [script: [solution.m](#).] Plot su $[-1, 2]^2$ di un'onda piana con direzione $\alpha = \frac{\pi}{3}$ e parametro $k = 20$ su una mesh di 150 punti con un vincolo a forma di stella a quattro punte. Ho usato il metodo di collocazione senza *singularity extraction* con discretizzazione uniforme e prendendo l'ampiezza $h = 0.001$.

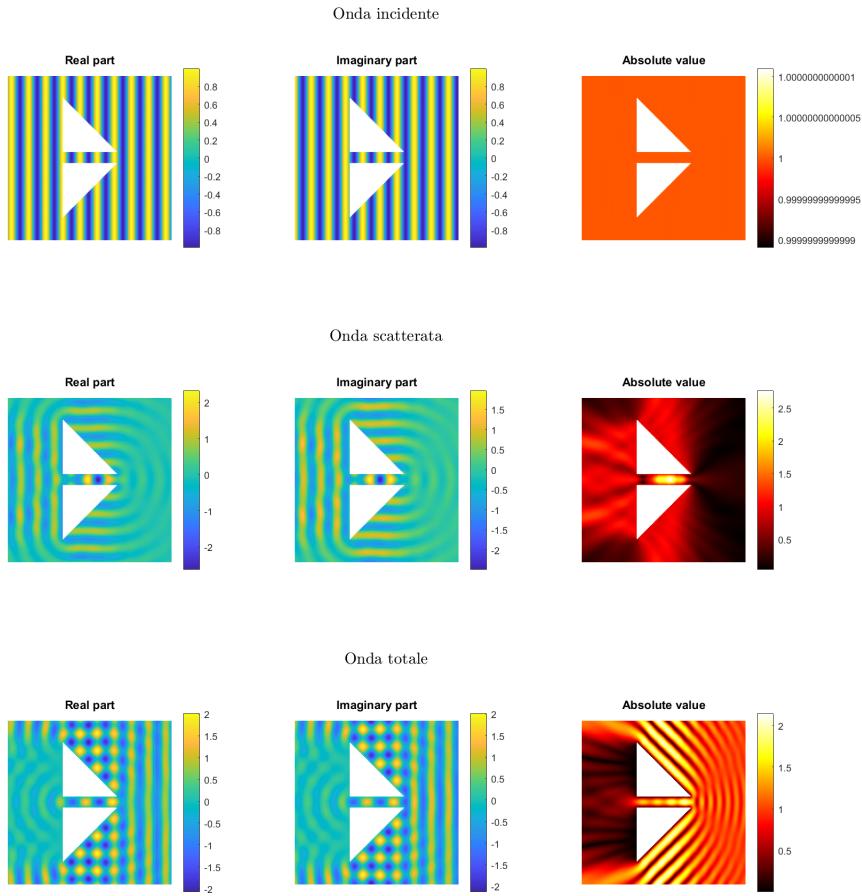


Figura 11: [script: [solution.m](#).] Plot su $[-1, 2]^2$ di un'onda piana con direzione $\alpha = \frac{\pi}{3}$ e parametro $k = 20$ su una mesh di 150 punti con un vincolo a forma di due triangoli. Ho usato il metodo di collocazione senza *singularity extraction* con discretizzazione uniforme e prendendo l'ampiezza $h = 0.1$.

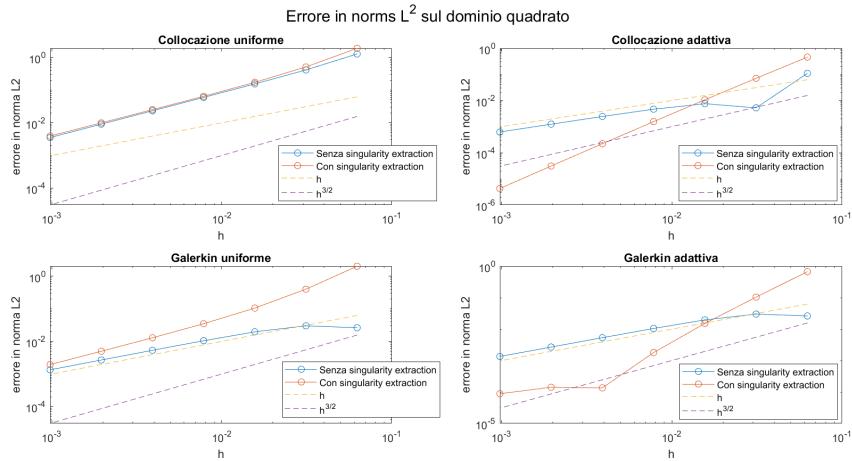


Figura 12: [script: [ERRORtest.m](#)] Errore nel caso di dominio quadrato, per $h = 2^{-4}, \dots, 2^{-10}$, con dimensione del sistema pari a 150.