

ESERCITAZIONE 1

**Trasformazioni nel piano complesso**

**1. Introduzione: numeri complessi e rappresentazioni grafiche**

L'obiettivo di questa sezione è di acquisire familiarità con l'uso dei numeri complessi in Matlab.

- L'unità immaginaria si indica in Matlab con `1i`. Provate a dare i seguenti comandi e assicuratevi di capire che cosa succede in ciascun caso.

```
i
j
sqrt(-1)
x=2+1i;
y=1+3i;
z=x+y
```

- Le funzioni `conj`, `abs`, `real`, `imag` calcolano rispettivamente il coniugato, il modulo, la parte immaginaria e la parte reale di un numero complesso. Esempio: sia  $x = 1 + 3i$  e verifichiamo sperimentalmente che  $|x| = (x\bar{x})^{1/2}$ .

```
x=1+3i           % definisco x
lhs=abs(x)        % left-hand side dell'uguaglianza
rhs=sqrt(x*conj(x)) % right-hand side dell'uguaglianza
abs(lhs-rhs)      % vale 0
```

- Sia  $\theta = 6$ . Verificate sperimentalmente che  $e^{i\theta} = \cos(\theta) + i\sin(\theta)$ . Potete usare le funzioni `exp`, `cos`, `sin`.
- Le funzioni `abs` e `angle` permettono di passare dalla rappresentazione cartesiana di un numero complesso a quella polare. Provate a dare i comandi seguenti:

```
x=-3+2i          % definisco x
r=abs(x)          % modulo di x
theta=angle(x)    % argomento di x
y=r*exp(i*theta)  % osservo che y=x
```

- Passiamo alla rappresentazione grafica. Provate ad applicare i suggerimenti seguenti a punti, vettori e matrici definiti da voi.

Un numero complesso  $x$  si rappresenta sul piano per mezzo del comando

```
plot(real(x),imag(x),'bo')
```

dove abbiamo usato un piccolo cerchio blu. Un'altra possibilità è scrivere

```
plot(x,'bo')
```

purché  $x$  abbia parte immaginaria non nulla. Se invece  $x$  è dato in coordinate polari, si può usare il comando

```
polar(theta,r,'bo')
```

- Dato un vettore complesso  $v$ , si possono rappresentare i suoi elementi nel piano con i comandi

```
plot(v,'r*-')
```

```
plot(v,'bo')
```

Nel primo caso ciascun punto è collegato al seguente da un segmento (utile per disegnare poligoni), nel secondo caso no.

- Data una matrice (array) complessa  $A$ , il comando

```
plot(A)
```

traccia un grafico sul piano complesso per ogni colonna di  $A$ .

**Esercizio 1** Scrivere una function `cerchio(c,r)` che accetta in input un numero complesso  $c$  e un numero reale positivo  $r$  e traccia sul piano la circonferenza di centro  $c$  e raggio  $r$ . Scrivere una function `triangolo(z,a)` che accetta in input un numero complesso  $z$  e un numero reale positivo  $a$  e traccia sul piano un triangolo equilatero di centro  $z$  e lato  $a$ .

## 2. Traslazioni e rotazioni

Traslazioni e rotazioni sul piano si possono effettuare usando l'aritmetica complessa.

- Traslazione di vettore  $w \in \mathbb{C}$ :

$$z \longrightarrow z + w.$$

- Rotazione di angolo  $\theta \in \mathbb{R}$  e centro l'origine:

$$z \longrightarrow ze^{i\theta}.$$

Esempio: costruiamo un quadrilatero.

```
z=[0 1 1+2i 3i 0];
plot(z,'*-')
axis([1 4 1 4],'equal')
```

Ora vogliamo ruotare il quadrilatero di  $\pi/6$  radianti intorno al suo baricentro. Per farlo, applichiamo dapprima al quadrilatero una traslazione per portarne il baricentro nell'origine, poi applichiamo una rotazione di  $\theta = \pi/6$  e poi trasliamo nuovamente per riportare il baricentro nella posizione iniziale.

```
mu=mean(z(1:end-1))    % calcolo le coordinate del baricentro
```

```

theta=pi/6;
omega=exp(i*theta);
y=omega*(z-mu)+mu;      % traslazione, rotazione, traslazione
hold on      % scriviamo sul grafico precedente
plot(real(y),imag(y),'r-')

```

Osserviamo che in Matlab è consigliabile usare il più possibile le operazioni vettoriali (anziché cicli for).

### 3. Trasformazioni di Möbius

Le funzioni di variabile complessa del tipo

$$z \longrightarrow \frac{az + b}{cz + d},$$

con  $ad - bc \neq 0$ , sono note come *trasformazioni di Möbius*. (Rotazioni e traslazioni sono casi particolari di queste trasformazioni).

**Esercizio 2** Scrivere uno script Matlab che esegua i compiti seguenti:

- definire un vettore  $v$  di numeri complessi che rappresentino “tanti” punti equispaziati sulla circonferenza di centro 1 e raggio 1,
- disegnare la circonferenza usando il vettore  $v$ ,
- applicare a ciascun elemento di  $v$  la trasformazione di Möbius

$$z \longrightarrow \frac{az + b}{z + 1},$$

dove  $a$  e  $b$  sono rispettivamente la penultima e l'ultima cifra del vostro numero di matricola (se sono uguali prendete invece  $a = 1$  e  $b = 0$ ), ottenendo un nuovo vettore  $w$ ,

- rappresentare i punti di  $w$  sullo stesso grafico usato in precedenza.

Qual è la figura che ne risulta? Che cosa congetturate riguardo alle proprietà della trasformazione di Möbius che avete usato? Rispondete commentando lo script.

### 4. Logaritmo

La funzione `log` calcola il logaritmo del numero complesso  $z$  nel modo seguente. Se  $z$  è reale positivo, allora `log(z)` è il consueto logaritmo naturale reale. Altrimenti, dette rispettivamente  $x$  e  $y$  la parte reale e immaginaria di  $z$ , si ha `log(z)=log(abs(z))+i*atan2(y,x)`, dove la funzione `atan2` calcola l'arcotangente di  $y/x$  nell'intervallo  $(-\pi, \pi)$ .

**Esercizio 3** Scrivere uno script in Matlab che esegua i compiti seguenti:

- Verificare graficamente che la funzione `log` mappa la circonferenza di centro 0 e raggio 1 nel segmento  $(-i\pi, i\pi)$ , usando la stessa tecnica dell'esercizio precedente.

- Disegnare sul piano complesso le circonferenze di centro 0 e raggio  $1, 2, \dots, 10$ , applicare la funzione  $\log(z)$  ai punti utilizzati per disegnare le circonferenze, e disegnare i nuovi punti sul piano complesso.

#### 4. Esercizi facoltativi

(Da fare se avete finito tutta la parte precedente).

- La matrice della trasformata discreta di Fourier di ordine  $n$  si costruisce in Matlab con il comando `F=fft(eye(n))`.

Sia  $n = 4$ : costruite la matrice e verificate che i suoi elementi sono radici dell'unità (quali?).

Che grafico produce il comando `plot(F)`? Sperimentate vari valori di  $n$ .

- I *tiling puzzles* sono giochi in cui alcuni tasselli poligonali vengono disposti sul piano per creare forme diverse. Due fra i più noti sono il *T-puzzle* e il *tangram*; potete cercare più dettagli su Wikipedia o con Google. Potete provare a usare i risultati della sezione 2 per simulare il vostro *tiling puzzle* preferito.