

## Rette parametriche in $R^2$ proiezione di un punto su una retta

Enrico Bertolazzi  
Esercitazioni Maple di Geometria

```
> # carica la libreria per i plottaggi
```

```
with(plots):
```

```
> # definisco due punti di passaggio come liste di due numeri
```

```
PA := [1,2] ;
```

```
PB := [3,4] ;
```

```
PA := [1, 2]
```

(1)

```
PB := [3, 4]
```

```
> # scrive l'equazione della retta di passaggio in forma parametrica
```

```
P := PA + t*(PB-PA) ;
```

```
P := expand(P) ;
```

```
P := unapply(P,t) ;
```

```
P := [1, 2] + t [2, 2]
```

(2)

```
P := [2 t + 1, 2 t + 2]
```

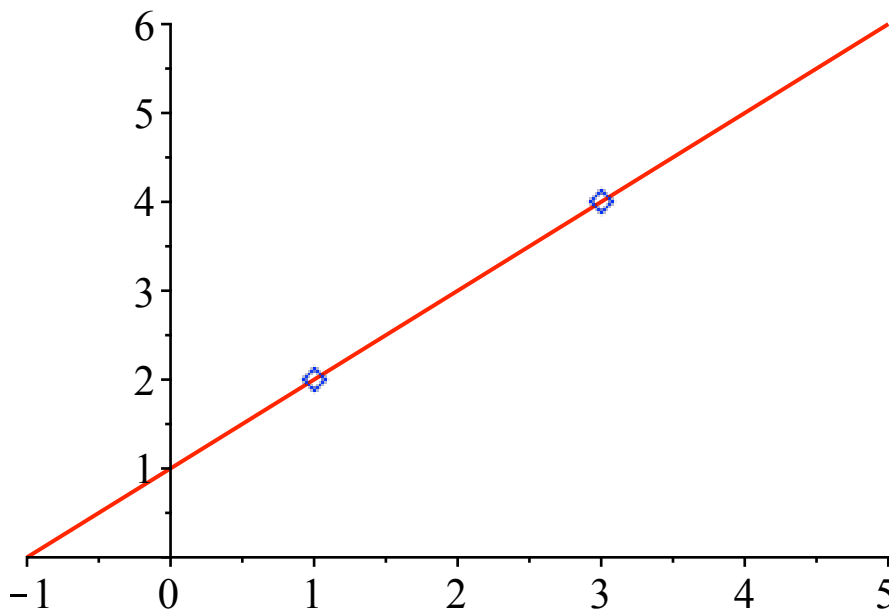
```
P := t → [2 t + 1, 2 t + 2]
```

```
> # stampa la retta parametrica e i suoi punti di passaggio
```

```
A := plot([op(P(t)),t=-1..2]) :
```

```
B := plot([PA,PB],style=point,symbolsize=20,color=blue) :
```

```
display(A,B) ;
```



```
> # trasforma la retta dalla forma parametrica alla forma implicita
```

```
PX := unapply(P(t)[1],t) ; # funzione parametrica x(t)
```

```

PY := unapply(P(t)[2],t) ; # funzione parametrica y(t)
# ricava t in funzione di x e sostituisce il y(t)
R := subs(isolate(x=PX(t),t),y=PY(t)) ;
      PX := t→2t+1
      PY := t→2t+2
      R := y=x+1

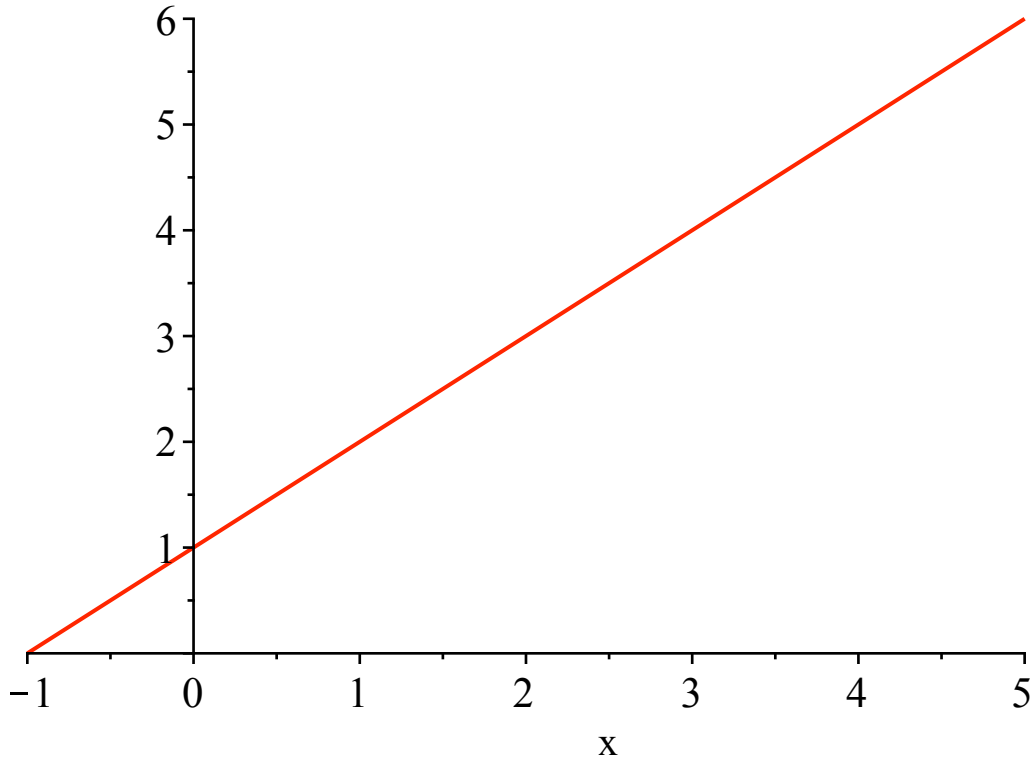
```

(3)

```

> # stampa la retta in forma implicita
plot(rhs(R),x=-1..5) ;

```



```

> # definisce il punto da proiettare
Q := [2,0] ;

```

```

      Q := [2,0]

```

(4)

```

> # funzione distanza del punto Q dal punto (x(t),y(t))
DIST := unapply(sqrt((PX(t)-Q[1])^2 + (PY(t)-Q[2])^2),t) ;

```

```

      DIST := t→sqrt(8t^2 + 4t + 5)

```

(5)

```

> # cerca il parametro t che minimizza la distanza
tmin := solve(D(DIST)(t)=0) ;

```

```

      tmin := -1/4

```

(6)

```

> # calcolo il punto sulla retta parametrica corrispondente a tmin
PMIN := P(tmin) ;

```

(7)

$$PMIN := \left[ \frac{1}{2}, \frac{3}{2} \right]$$

(7)

```
> # disegna la retta il punto Q e la sua proiezione  
C1 := plot([PMIN,Q],style=point, color=green, symbolsize=20) :  
C2 := plot([PMIN,Q],style=line, linestyle=3, color=green) :  
display(A,B,C1,C2,scaling=CONSTRAINED) ;
```

