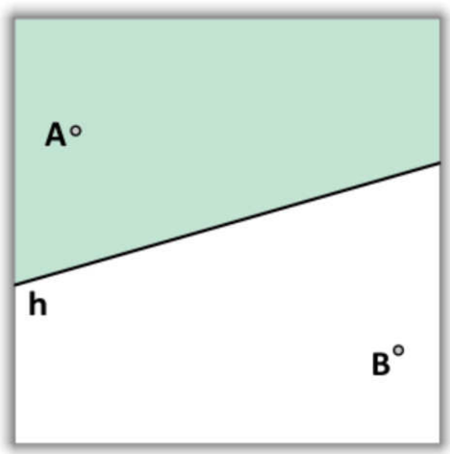
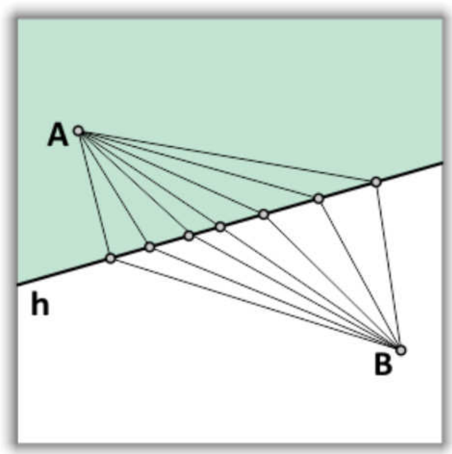
Zadání úkolu

Místo **A** leží v oblasti bez sněhu, ve které se pohybujeme rychlostí **v1**. Místo **B** leží v zasněžené oblasti, ve které se pohybujeme rychlostí **v2**. Hranice mezi nezasněženou a zasněženou oblastí je tvořena přímkou **h**.

Máme najít takovou trasu, po které dojdeme z bodu **A** do bodu **B** za nejkratší čas.

Je zřejmé, že uvnitř oblasti *A* i uvnitř oblasti *B* se budeme pohybovat přímo (po úsečkách). Úkolem je tedy najít takový bod **P** na hraniční přímce **h**, pro který bude čas strávený na sestě **AP** a **PB** nejkratší.





Řešení (matematické)

Pro usnadnění výpočtů použijeme CAS (Computer Algebra System) Maple.

<u>Příkazy systému jsou uvedeny spíše pro zajímavost, pozornost věnujte především modře odlišeným výstupům.</u>

Inicializace systému a nahrání potřebných knihoven.

```
> restart;
  with(plots):
  with(geometry):
```

Nejprve zvolíme polohu bodů **A**, **B** a hraniční přímky **h** v souřadnicovém systému a dále zvolíme matematický popis hraniční přímky, resp. hledaného bodu **P**.

Pokud body **A** a **B** neleží na hranici **h**, lze bez újmy na obecnosti předpokládat, že:

- bod **A** leží na ose *y* a jeho druhá souřadnice je kladná, tedy: *A[0,a2], a2 > 0*;
- hraniční přímka **h** je osa *x*, tedy *h(t)=[t,0]*, $t \in \mathbb{R}$;
- bod **P** je bodem hraniční přímky, má tedy souřadnice *P[t,0]*;
- bod **B** má nezápornou první a zápornou druhou souřadnici, tedy *B[b1,b2], b1* $\geqq$ 0, *b2* < 0;

Definice bodů A, B, P.

```
> A:=[0,a2];
  B:=[b1,b2];
  P:=[t,0];
```

$$A := [0, a2]$$
$$B := [b1, b2]$$
$$P := [t, 0] \quad (1)$$

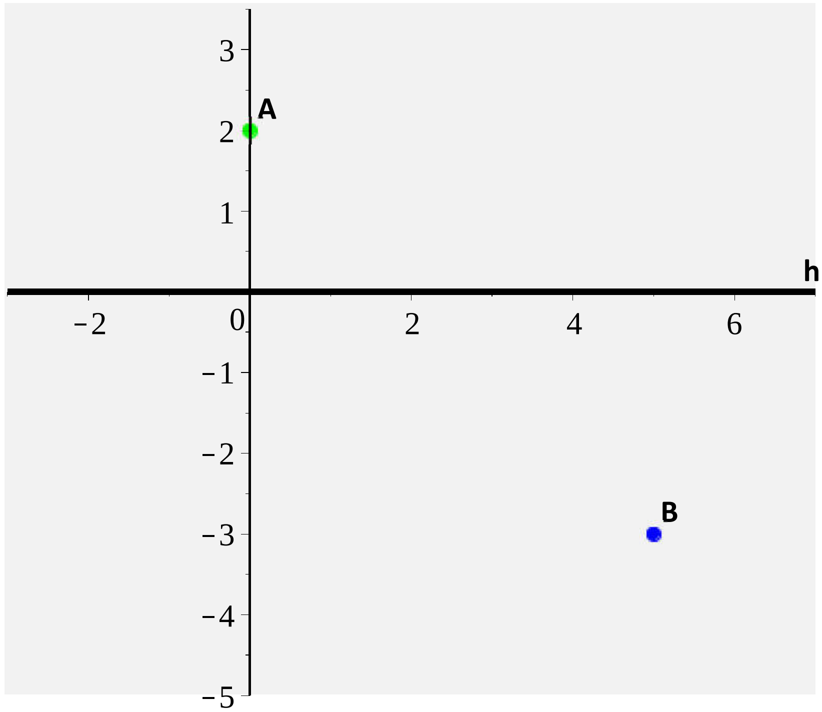
Volba konkrétních hodnot parametrů - pro vykreslení a později pro výpočty.
Předpokládejme např., že vzdálenosti jsou zadávány v kilometrech a rychosli v km/h, čas tedy vychází v hodinách.

```
> volba:={a2=2, b1=5, b2=-3};
```

$$volba := \{a2 = 2, b1 = 5, b2 = -3\} \quad (2)$$

```
> vykresli_zadani:=proc(A,B,P)
  local A_d, B_d, kresli_A, kresli_B, kresli_h, popis;
  point(A_d,subs(volba,A)):
  point(B_d,subs(volba,B)):
  kresli_A:=draw(A_d, symbol=solidcircle, symbolsize=15, color=
  green):
  kresli_B:=draw(B_d, symbol=solidcircle, symbolsize=15, color=
  blue):
  kresli_h:=plot([P[1],P[2],t=-3..7], thickness=3, color=black):
  popis:=textplot({[HorizontalCoord(A_d),VerticalCoord(A_d), "A"],
  [HorizontalCoord(B_d),VerticalCoord(B_d), "B"],[6.75,0, "h"]},
  'align'={'above', 'right'}, font = [Calibri, bold, 12]):

  display({kresli_A,kresli_B,kresli_h,popis},axes=normal,scaling=
  constrained,background=ColorTools:-Color([0.95, 0.95, 0.95]),
  view=[-3..7,-5..3.5]);
  end proc:
> vykresli_zadani(A,B,P);
```



Spočítáme obecné vzdálenosti bodů **AP** a **BP**.
Vzdálenosti označíme jako *d1, d2*.

```
> d1:=sqrt( (A[1]-P[1])^2 + (A[2]-P[2])^2);
  d2:=sqrt( (B[1]-P[1])^2 + (B[2]-P[2])^2);
```

$$d1 := \sqrt{a2^2 + t^2}$$

$$d2 := \sqrt{(b1 - t)^2 + b2^2} \tag{3}$$

Spočítáme čas potřebný k projití úseček **AP** a **BP** - tj. vydělíme vzdálenosti rychlostí chůze (**v1**, **v2**).
Dílčí časy označíme jako *cas1, cas2*.

```
> cas1:=d1/v1;
  cas2:=d2/v2;
```

$$cas1 := \frac{\sqrt{a2^2 + t^2}}{v1}$$

$$cas2 := \frac{\sqrt{(b1 - t)^2 + b2^2}}{v2} \tag{4}$$

Celkový čas pak označíme jako *cas* - je to funkce proměnné *t*: *cas*(*t*).

```
> cas:=cas1+cas2;
```

$$cas := \frac{\sqrt{a2^2 + t^2}}{v1} + \frac{\sqrt{(b1 - t)^2 + b2^2}}{v2} \quad \textbf{(5)}$$

K nalezení dráhy s minimálním časem potřebujeme najít minimum funkce *cas*. To je úkol, který lze vyřešit s pomocí derivací.
Spočítáme derivaci *dcas* funkce *cas* podle proměnné *t* a najdeme její nulové body. Řešením je hodnota parametru *t*, tedy poloha bodu **P** na hraniční přímce **h**.

```
> dcas:=diff(cas,t);
```

$$dcas := \frac{t}{\sqrt{a2^2 + t^2}\, v1} + \frac{1}{2} \frac{-2\, b1 + 2\, t}{\sqrt{(b1 - t)^2 + b2^2}\, v2} \quad \textbf{(6)}$$

```
> solve(dcas=0,t);
```

$$RootOf\left((v1^2 - v2^2)\, \_Z^4 + (-2\, b1\, v1^2 + 2\, b1\, v2^2)\, \_Z^3 + (a2^2\, v1^2 + b1^2\, v1^2 - b1^2\, v2^2 - b2^2\, v2^2)\, \_Z^2 - 2\, a2^2\, b1\, v1^2\, \_Z + a2^2\, b1^2\, v1^2 \right) \quad \textbf{(7)}$$

Symbolické řešení rovnice není příliš přehledné - je zřejmé, že to je řešení rovnice čtvrtého stupně.
Vzhledem k zadání - hranicí je <u>přímka</u> **h**, je možno upustit od ověřování, zda je nalezený stacionární bod funkce *cas* opravdu lokálním minimem - tak, jak požadujeme. To je sice v tomto případě zřejmé, ale lze to také znázornit vykreselním grafů funkcí *cas* a *dcas* pro konkrétní volby parametrů.

Najdeme řešení pro konkrétní volby parametrů.

```
> volba:={a2=2, b1=5, b2=-3, v1=5, v2=7/2};
```

$$volba := \left\{ a2 = 2, b1 = 5, b2 = -3, v1 = 5, v2 = \frac{7}{2} \right\} \quad \textbf{(8)}$$

```
> cas_v:=subs(volba,cas);
  dcas_v:=diff(cas_v,t);
```

$$cas\_v := \frac{1}{5} \sqrt{t^2 + 4} + \frac{2}{7} \sqrt{(5 - t)^2 + 9}$$

$$dcas\_v := \frac{1}{5} \frac{t}{\sqrt{t^2 + 4}} + \frac{1}{7} \frac{-10 + 2\, t}{\sqrt{(5 - t)^2 + 9}} \quad \textbf{(9)}$$

```
> reseni:=solve(dcas_v=0,t);
```

$$reseni := RootOf\left(51\, \_Z^4 - 510\, \_Z^3 + 1234\, \_Z^2 - 4000\, \_Z + 10000, index = 1\right) \quad \textbf{(10)}$$

Symbolické řešení opět není příliš názorné, ale při číselném vyjádření je vše jasnější.

```
> reseni_cis:=evalf(reseni);
```

$$reseni\_cis := 2.888722287 \quad \textbf{(11)}$$

Dosazením nalezené hodnoty *t* do funkce *cas_v* získáme požadovaný minimální čas chůze z bodu **A** do bodu **B** pro zvolené vzdálenosti a rychlosti.

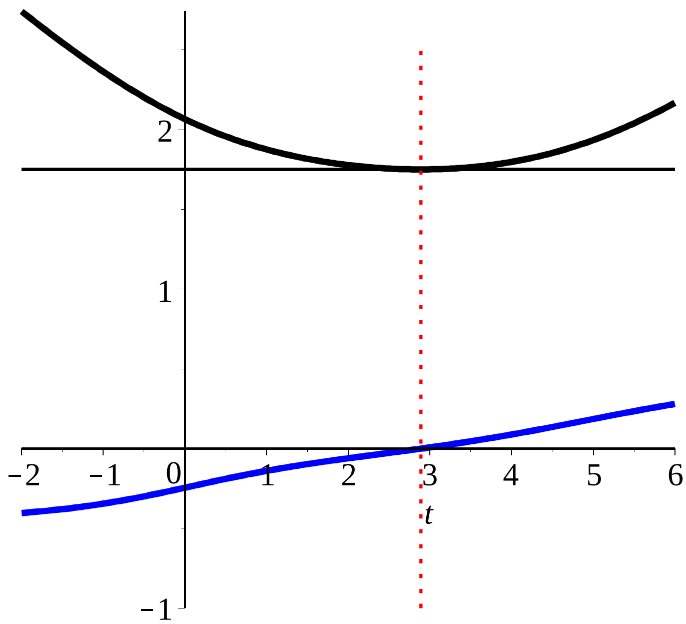
```
> min_cas:=subs(t=reseni_cis,cas_v);
```

$$min\_cas := 1.750828305 \quad \textbf{(12)}$$

Vykreslíme grafy funkcí *cas_v* a *dcas_v* pro ověření existence lokálního minima.

```
> zobraz_funkce:=plot([cas_v,dcas_v],t=-2..6,color=[black,blue],
  thickness=3):
  zobraz_primky:=plot([[reseni_cis,s,s=-1..2.5],[s,min_cas,s=-2..6]
  ],thickness=1,linestyle=["dot","solid"], color=[red,black]):
```

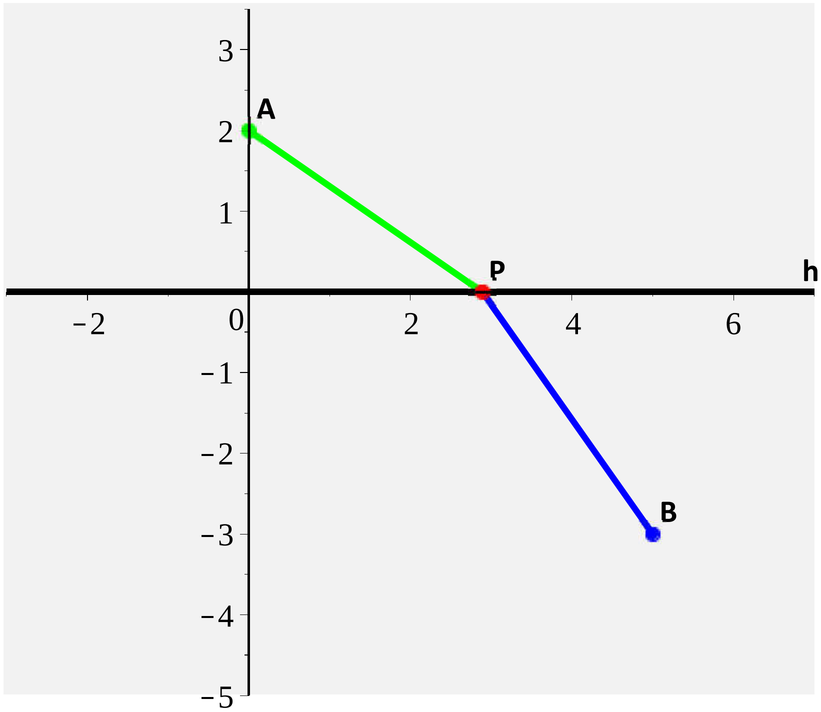
```
display(zobraz_funkce,zobraz_primky);
```



Vykreslení výsledné cesty.

```
> vykresleni:=proc(A,B,P)
  local A_d, B_d, P_d, AP_d, BP_d, kresli_A, kresli_B, kresli_P,
  kresli_h, kresli_AP, kresli_BP, popis;
  point(A_d,subs(volba,A)):
  point(B_d,subs(volba,B)):
  point(P_d,subs(t=reseni_cis,P)):
  kresli_A:=draw(A_d, symbol=solidcircle, symbolsize=15, color=
  green):
  kresli_B:=draw(B_d, symbol=solidcircle, symbolsize=15, color=
  blue):
  kresli_P:=draw(P_d, symbol=solidcircle, symbolsize=15, color=red)
  :
  kresli_h:=plot([P[1],P[2],t=-3..7], thickness=3, color=black):
  segment(AP_d,A_d,P_d):
  segment(BP_d,B_d,P_d):
  kresli_AP:=draw(AP_d, thickness=3, color=green):
  kresli_BP:=draw(BP_d, thickness=3, color=blue):
  popis:=textplot({[HorizontalCoord(A_d),VerticalCoord(A_d), "A"],
  [HorizontalCoord(B_d),VerticalCoord(B_d), "B"],[HorizontalCoord
  (P_d),VerticalCoord(P_d), "P"],[6.75,0, "h"]}, 'align'={'above',
  'right'}, font = [Calibri, bold, 12]):

  display({kresli_A,popis,kresli_B,kresli_P,kresli_h,kresli_AP,
  kresli_BP},axes=normal,scaling=constrained,background=
  ColorTools:-Color([0.95, 0.95, 0.95]),view=[-3..7,-5..3.5]);
  end proc:
> vykresleni(A,B,P);
```



Možná zobecnění úkolu

Úkol lze snadno zobecnit zadáním jiné hraniční křivky. Postup bude obdobný, ale je třeba věnovat mnohem větší pozornost ověřování, zda nalezený stacionární bod funkce *cas* je opravdu lokálním minimem a zda vypočtená dráha nepřekračuje hranici vícekrát.

Pro hledání řešení - nulových bodů derivace - bude nutno pro některé složitější zadané hranice použít numerického (ne symbolického) řešení. Tedy místo příkazu *solve* využít příkaz *fsolve*.

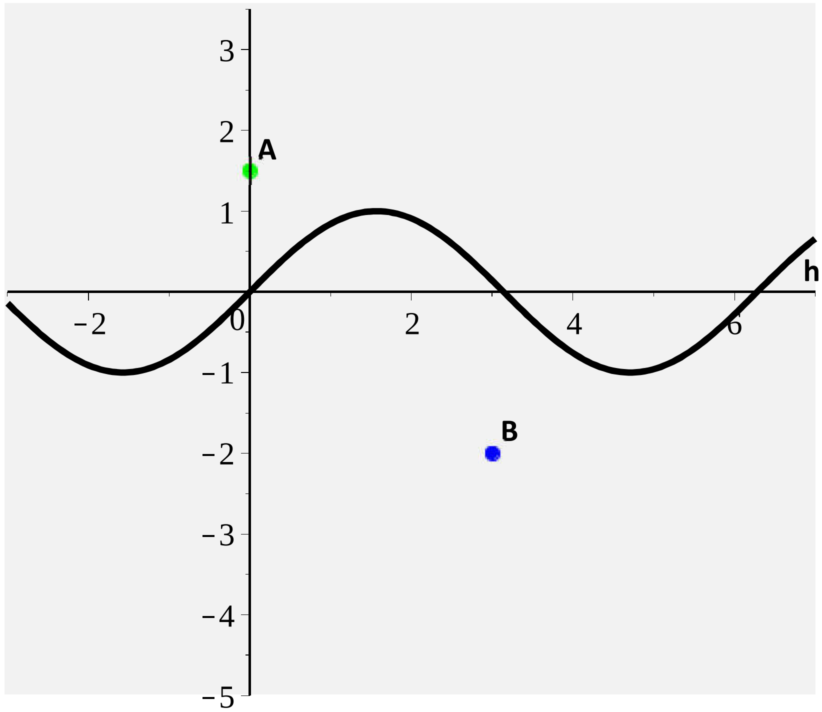
```
> P:=[t,sin(t)];
```

$$P := [t, \sin(t)] \tag{13}$$

Pro vykreslení použijeme dříve definované procedury.

```
> volba:={a2=1.5, b1=3, b2=-2, v1=5, v2=7/2};
  vykresli_zadani(A,B,P);
```

$$volba := \left\{a2 = 1.5, b1 = 3, b2 = -2, v1 = 5, v2 = \frac{7}{2}\right\}$$



Znovu definujeme a napočítáme potřebné proměnné a funkce.

```
> d1:=sqrt( (A[1]-P[1])^2 + (A[2]-P[2])^2):
  d2:=sqrt( (B[1]-P[1])^2 + (B[2]-P[2])^2):
  cas1:=d1/v1:
  cas2:=d2/v2:
  cas:=cas1+cas2;
  dcas:=diff(cas,t);
```

$$cas := \frac{\sqrt{t^2 + (a2 - \sin(t))^2}}{v1} + \frac{\sqrt{(b1 - t)^2 + (b2 - \sin(t))^2}}{v2}$$

$$dcas := \frac{1}{2} \frac{2t - 2(a2 - \sin(t))\cos(t)}{\sqrt{t^2 + (a2 - \sin(t))^2}\; v1} + \frac{1}{2} \frac{-2\, b1 + 2t - 2(b2 - \sin(t))\cos(t)}{\sqrt{(b1 - t)^2 + (b2 - \sin(t))^2}\; v2} \tag{14}$$

```
> cas_v:=subs(volba,cas);
  dcas_v:=diff(cas_v,t);
```

$$cas\_v := \frac{1}{5}\sqrt{t^2 + (1.5 - \sin(t))^2} + \frac{2}{7}\sqrt{(3 - t)^2 + (-2 - \sin(t))^2}$$

$$dcas\_v := \frac{1}{10} \frac{2t - 2(1.5 - \sin(t))\cos(t)}{\sqrt{t^2 + (1.5 - \sin(t))^2}} + \frac{1}{7} \frac{-6 + 2t - 2(-2 - \sin(t))\cos(t)}{\sqrt{(3 - t)^2 + (-2 - \sin(t))^2}} \tag{15}$$

Zkusíme vypočítat řešení a poté ověříme na grafech.

```
> reseni:=solve(dcas_v=0,t);
```

$$reseni := 3.131008940,\ 2.078778703 \qquad \textbf{(16)}$$

Lze využít i numerický výpočet (většinou je rychlejší, ale většinou nenajde všechna řešení).

```
> reseni:=fsolve(dcas_v=0,t);
```

$$reseni := 2.078778703 \qquad \textbf{(17)}$$

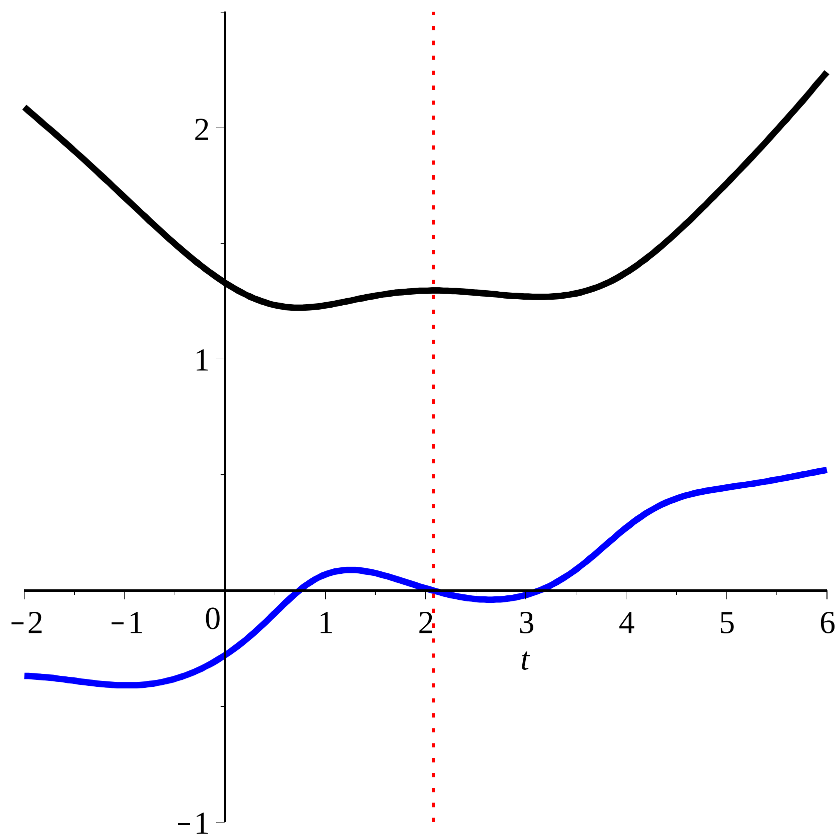
```
> reseni_cis:=evalf(reseni);
  min_cas:=evalf(subs(t=reseni_cis,cas_v));
```

$$reseni\_cis := 2.078778703$$

$$min\_cas := 1.296434799 \qquad \textbf{(18)}$$

Vykreslíme grafy funkcí *cas_v* a *dcas_v* pro ověření existence minima.

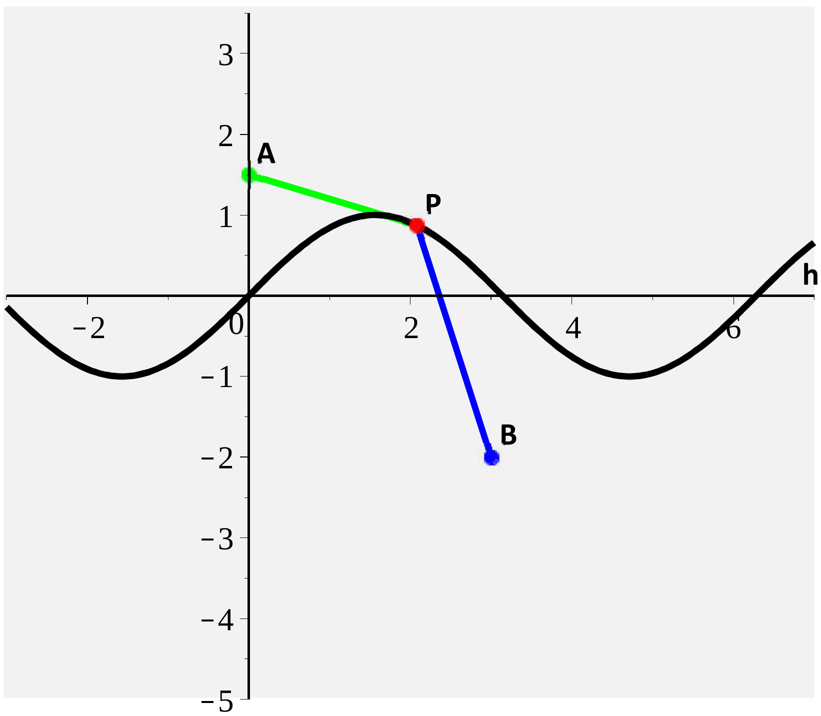
```
> zobraz_funkce:=plot([cas_v,dcas_v],t=-2..6,color=[black,blue],
  thickness=3):
  zobraz_primky:=plot([[reseni_cis,s,s=-1..2.5]],thickness=1,
  linestyle="dot", color=red):
  display(zobraz_funkce,zobraz_primky);
```



Z grafů je zřejmé, že ani symbolickým, ani numerickým výpočtem jsme nezískali všechny stacionární body (modrý graf derivace má tři nulové body).
Vypočtený stacionární bod odpovídá **lokálnímu maximu** funce *cas_v*.

Nalezené řešení odpovídá této cestě:

```
> vykresleni(A,B,P);
```



Pro nalezení správného stacionárního bodu lze omezit interval hledání kořenů u příkazu *fsolve*. Ze dvou lokálních minim je nutno vybrat minimum globální.

```
> reseni:=fsolve(dcas_v=0,t=0..1);
```

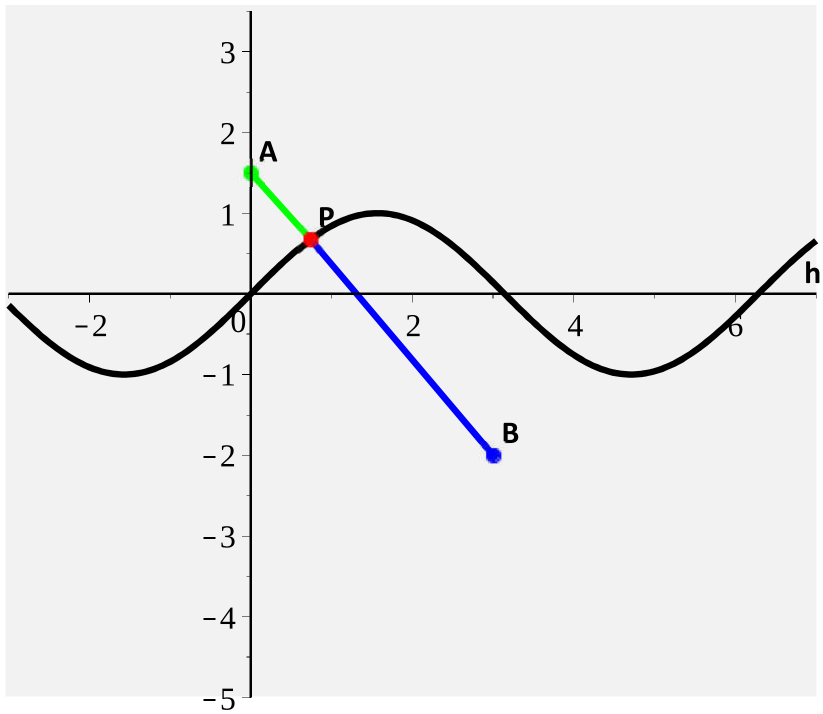
$$reseni := 0.7393649508 \tag{19}$$

```
> reseni_cis:=evalf(reseni);
  min_cas:=evalf(subs(t=reseni_cis,cas_v));
```

$$reseni\_cis := 0.7393649508$$
$$min\_cas := 1.222140839 \tag{20}$$

Nejrychlejší cesta je tentokrát téměř přímá.

```
> vykresleni(A,B,P);
```



Ověření lokálního minima lze provádět i pomocí derivací vyšších řádů, ne jen "vizuálně".
Kromě více stacionárních bodů je nutno hlídat i to, zda se cesta nevrací zpátky do zasněžené oblasti.