



**OI-OPPA. European Social Fund
Prague & EU: We invest in your future.**

Detekce jádra polygonální oblasti (Detection of a polygon kernel)

AE4M39VG

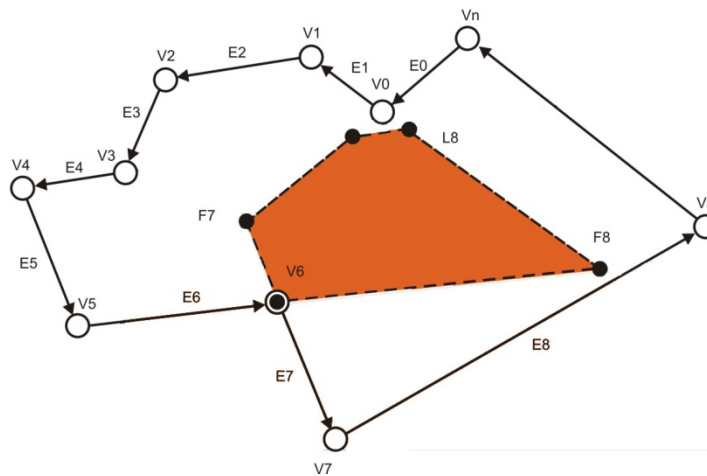
Martin Bláha

Obsah

- Jádro polygonu
- Algoritmy
- Lee - Preperata
 - Představení algoritmu
 - Popis algoritmu
 - Ukázka

Jádro polygonu

- Množina vnitřních bodů, ze které je možno spojit bod z této množiny přímo s jakýmkoliv vrcholem polygonu tak, aby byla celá úsečka součástí polygonu.



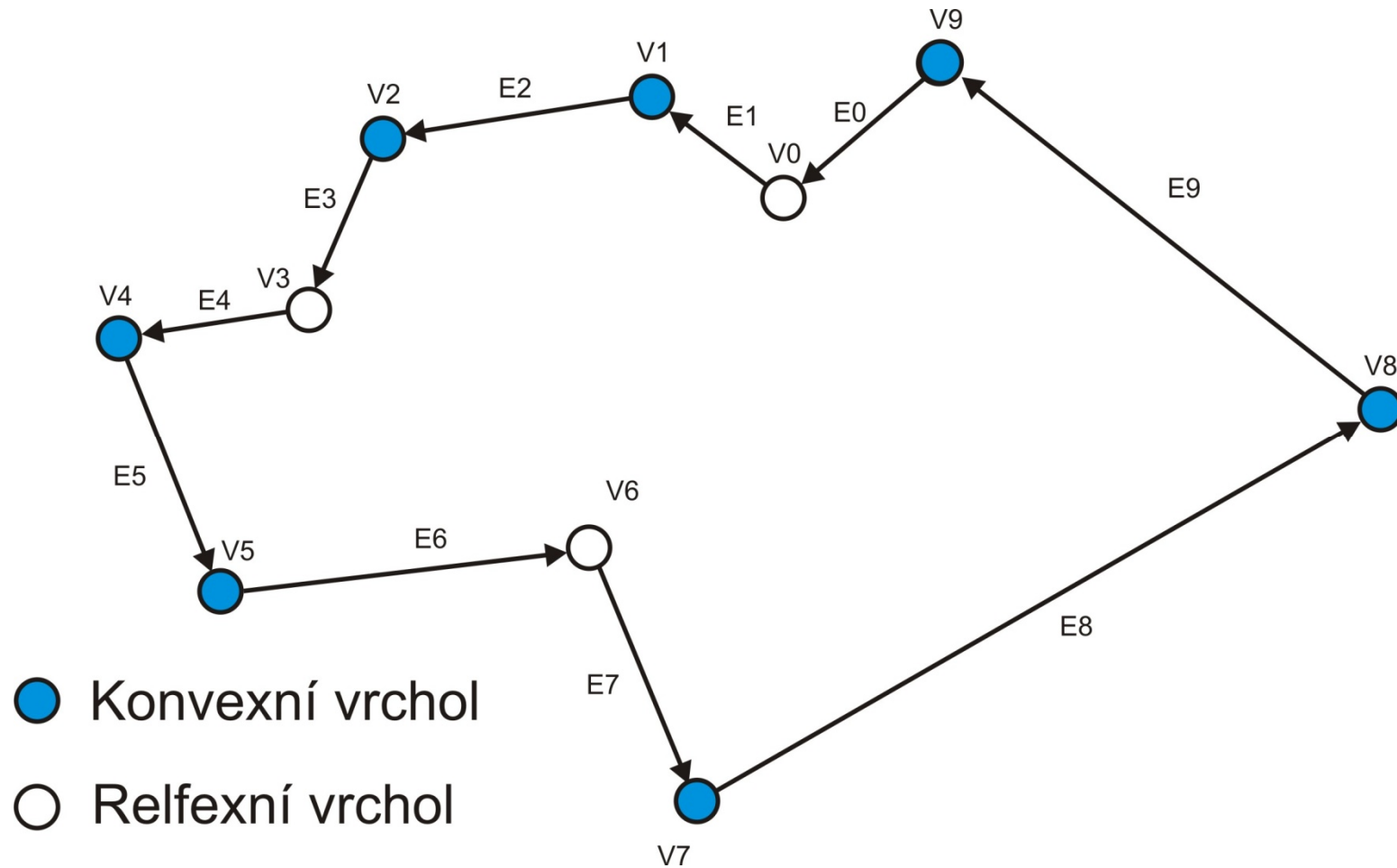
Algoritmy

- Průnik polorovin $O(n^2)$
- Shamos- Hoey $O(n \log n)$ – *rekurze, rozděl a panuj*
- Lee- Preperata $O(n)$ – *postupně vytváří konvexní polygony K_1, K_2, \dots, K_{n-1} tak, že $K_{n-1} = K(P)$*

Lee-Preperata - Představení

- Polygon určený alespoň 4 vrcholy
- Hrany mají přiděleny orientaci proti směru hodinových ručiček (vnitřní oblast se nachází nalevo od hrany)
- Reflexní vrchol – vnitřní úhel je větší než 180°
- Konvexní vrchol – vnitřní úhel je menší než 180°
- Vrchol s úhlem 180° přeskočí → stejná polorovina

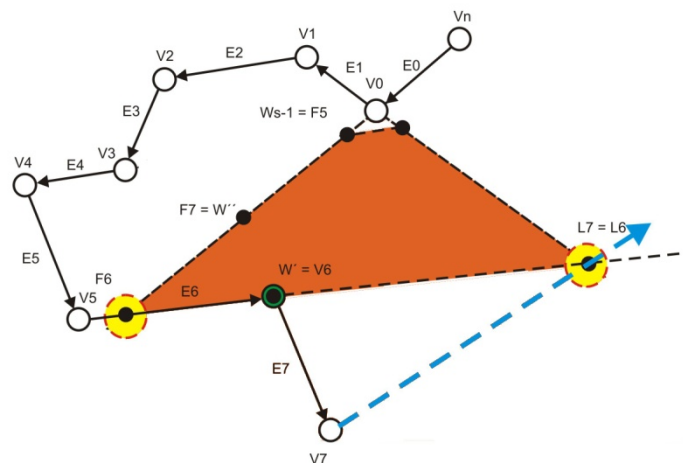
Lee-Preperata - Představení



Lee-Preperata - Představení

- Důležitá funkce dvou podpůrných vrcholů, označených jako L a F (vrcholy, které jsou součástí jádra polygonu) – rozhodnutí, jak dále postupovat, mění se při změně vyšetřovaného vrcholu

1.1



Lee-Preperata - Představení

- Data uložena jako posloupnost vrcholů a hran tak, že každá hrana je uložena mezi svými koncovými vrcholy (...vrchol1-hrana1-vrchol2...)
- Pro uložení této posloupnosti je využit obousměrně zřetězený kruhový seznam (ukazatel na počátek a na konec)

Lee – Preperata – popis algoritmu

Počáteční krok:

- výpočet průniku polorovin určené hranami, které spojuje počáteční vrchol $v_0 = K_1$
- Jádro K_1 ohraničeno polorovinami těchto hran
- Podpůrné vrcholy F a L jsou zvoleny na konci polopřímek určené hranami s počátkem v počátečním vrcholu v_0 (viz ukázka)

Lee – Preperata – popis algoritmu

„Obecný“ krok:

- Následuje od počátečního kroku až do nalezení jádra polygonu
- Rozhoduje se podle typu vrcholu ke kterému dorazil (*konvexní x reflexní*)

Lee-Preperata – Obecný krok

- 1) Vrchol je **reflexní**
- 1.1) F_i leží na nebo vpravo od hrany e_{i+1} -> projdi hrany z K_i proti směru hodinových ručiček z F_i , dokud nenajdeš hranu, která protíná přímku, na které leží e_{i+1} . Pokud průsečík není nalezen, pak $K(P) = 0$, jinak průsečík existuje, označ ho označ w' .

Projdi hrany z K_i po směru hodinových ručiček z F_i , dokud nenajdeš hranu, která protíná přímku, na které leží e_{i+1} . Pokud průsečík nebyl nalezen, je jádro neuzavřený polygon -> otestuj zda bude v dalším průchodu, také neuzavřený tak , že když bude přímka e_{i+1} uzavřena mezi první a poslední hranou jádra, bude i nadále neuzavřený, jinak projdi seznam od posledního prvku (ukazatel na konec) po směru hodinových ručiček a najdi průsečík w'' hrany K_i s přímkou, na které leží e_{i+1} -> seznam se stane cyklický. Pokud byl nalezen průsečík, označ ho w'' . Aktualizuj K_{i+1}

Lee-Preperata – Obecný krok

- Aktualizace F_{i+1} : pokud byl nalezen 1 průsečík F_{i+1} leží v nekonečnu na F_{i+1} , jinak $F_{i+1} = w''$
- Aktualizace L_{i+1} : projdi hrany z K_i po směru hodinových ručiček z L_i , dokud nenajdeš takový uzel w_u , kde jeho levý soused leží vlevo od přímky procházející body v_{i+1} a w_u orientovaný do z v_{i+1} a w_u . Pokud takový bod existuje pak $L_{i+1} = w_u$. Jinak $L_{i+1} = L_i$
- 1.2) F_i leží vlevo od hrany $e_{i+1} \rightarrow K_{i+1} = K_i$
- Aktualizace F_{i+1} : projdi hrany z K_i proti směru hodinových ručiček z F_i , dokud nenajdeš takový uzel w_t , kde jeho levý soused leží vpravo od přímky procházející body v_{i+1} a w_t orientovaný do z v_{i+1} a w_t . Pak $L_{i+1} = w_t$.
- Aktualizace L_{i+1} : stejně jako v případě 1.10

Lee-Preperata – Obecný krok

- 2) Vrchol je **konvexní**
- 2.1) L_i leží na nebo vpravo od hrany e_{i+1} -> projdi hrany z K_i po směru hodinových ručiček z L_i , dokud nenajdeš hranu, která protíná přímku, na které leží e_{i+1} . Pokud průsečík není nalezen, pak $K(P) = 0$, jinak průsečík existuje, označ ho označ w' .

Projdi hrany z K_i proti směru hodinových ručiček z L_i , dokud nenajdeš hranu, která protíná přímku, na které leží e_{i+1} . Pokud průsečík nebyl nalezen, je jádro neuzavřený polygon -> otestuj zda bude v dalším průchodu, také neuzavřený tak , že když bude přímka e_{i+1} uzavřena mezi první a poslední hranou jádra, bude i nadále neuzavřený, , jinak projdi seznam od prvního prvku proti směru hodinových ručiček(ukazatel na počátek) a najdi průsečík w'' hrany K_i s přímkou, na které leží e_{i+1} -> seznam se stane cyklický. Pokud byl nalezen průsečík, označ ho w'' . Aktualizuj K_{i+1}

Lee-Preperata – Obecný krok

2.1.1) Pokud byly nalezeny 2 průsečíky :

- Aktualizace F_{i+1} : a v_{i+1} je prvkem úsečky $v_i w'$, pak se F_{i+1} určí jako v případě 1.2) , jinak $F_{i+1} = w''$
- Aktualizace L_{i+1} : a v_{i+1} je prvkem úsečky $v_i w''$, pak $L_{i+1} = w''$, jinak se L_{i+1} určí jako v případě 1.1)

2.1.2) Pokud byl nalezen jen průsečík w' :

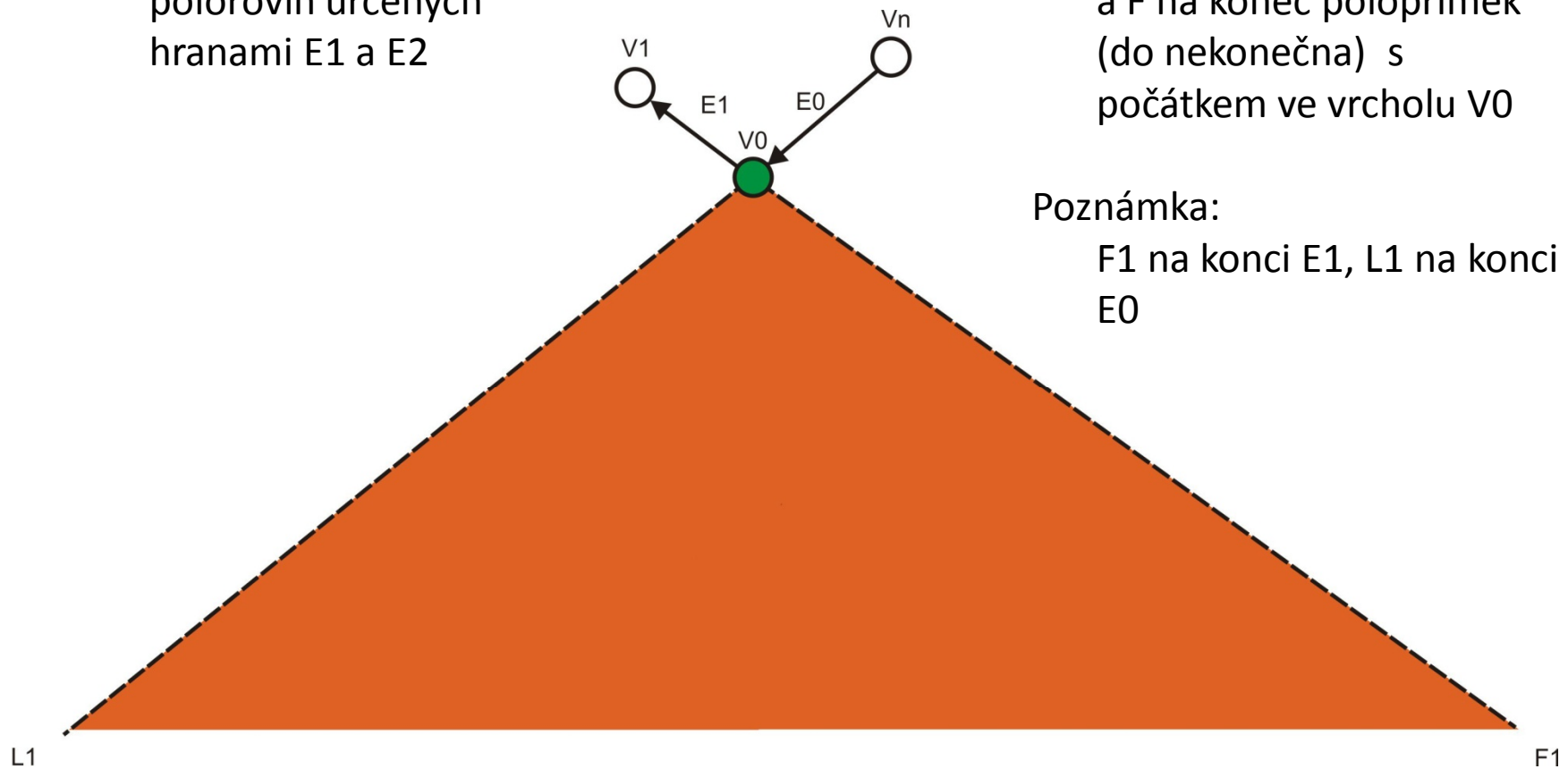
- Aktualizace F_{i+1} : a v_{i+1} je prvkem úsečky $v_i w'$, pak se F_{i+1} určí jako v případě 1.2) , jinak $F_{i+1} = w'$
- Aktualizace L_{i+1} : je nastaven na konec polopřímky určené e_{i+1} s koncovým bodem ve v_i

2.2) L_i leží vlevo od hrany $e_{i+1} \rightarrow K_{i+1} = K_i$, F_{i+1} určí jako v případě 1.2), pokud je K_i uzavřený pak, L_{i+1} určí jako v případě 1.1), $L_{i+1} = L_i$

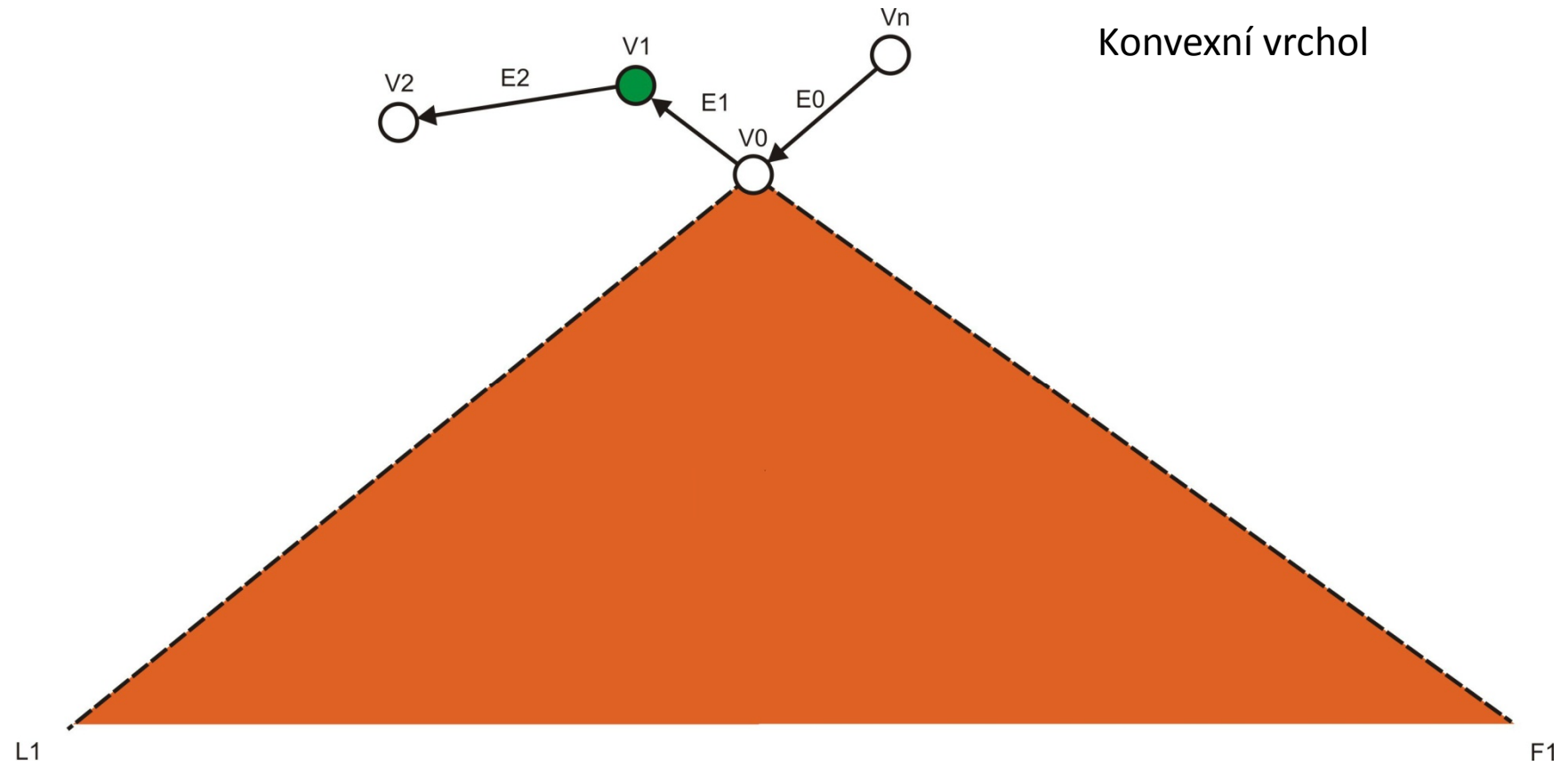
Počáteční krok

1) Výpočet průsečíku polorovin určených hranami E1 a E2

2) Zvolení podpůrných bodů L a F na konec polopřímek (do nekonečna) s počátkem ve vrcholu V0

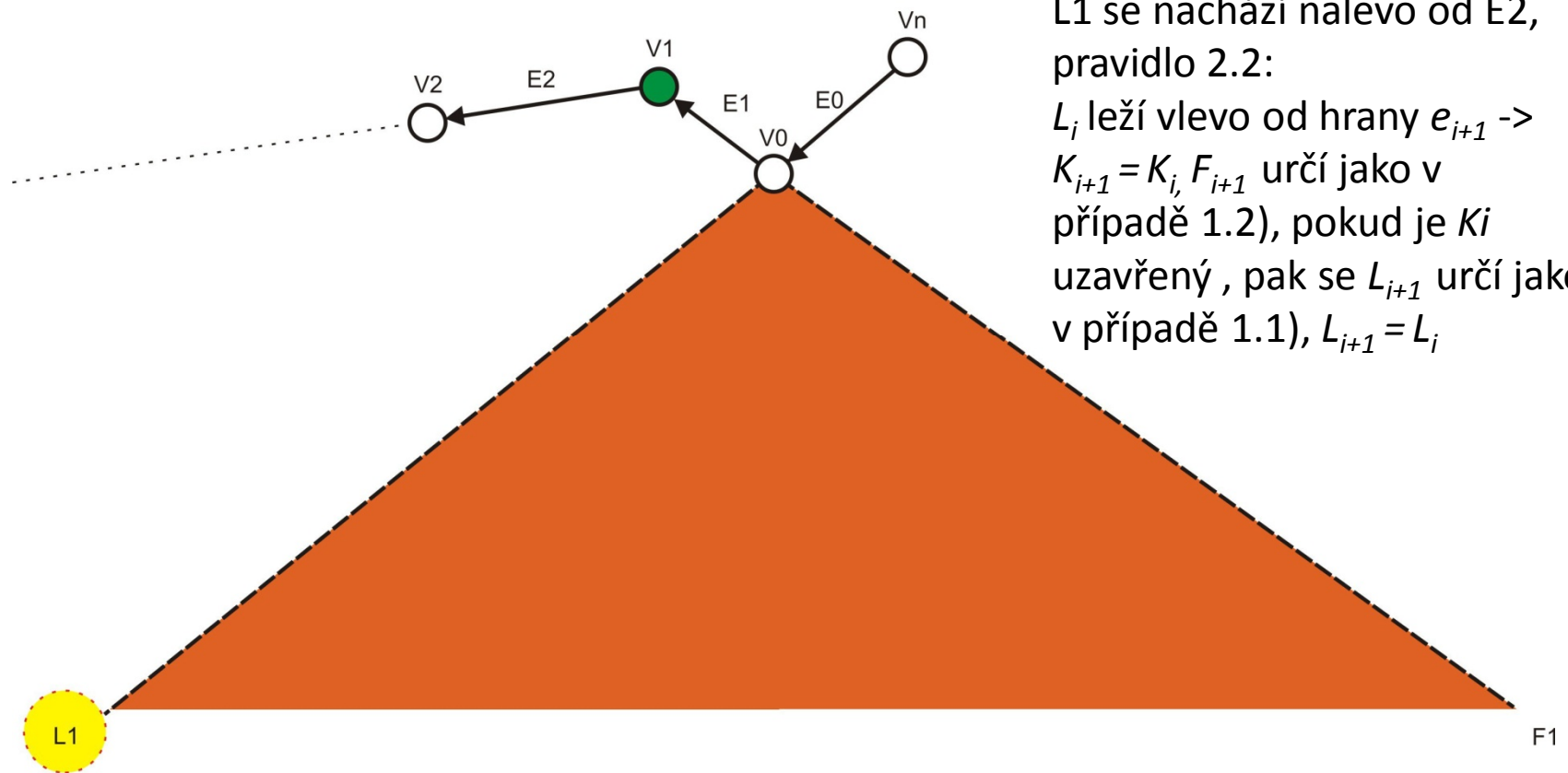


Obecný krok



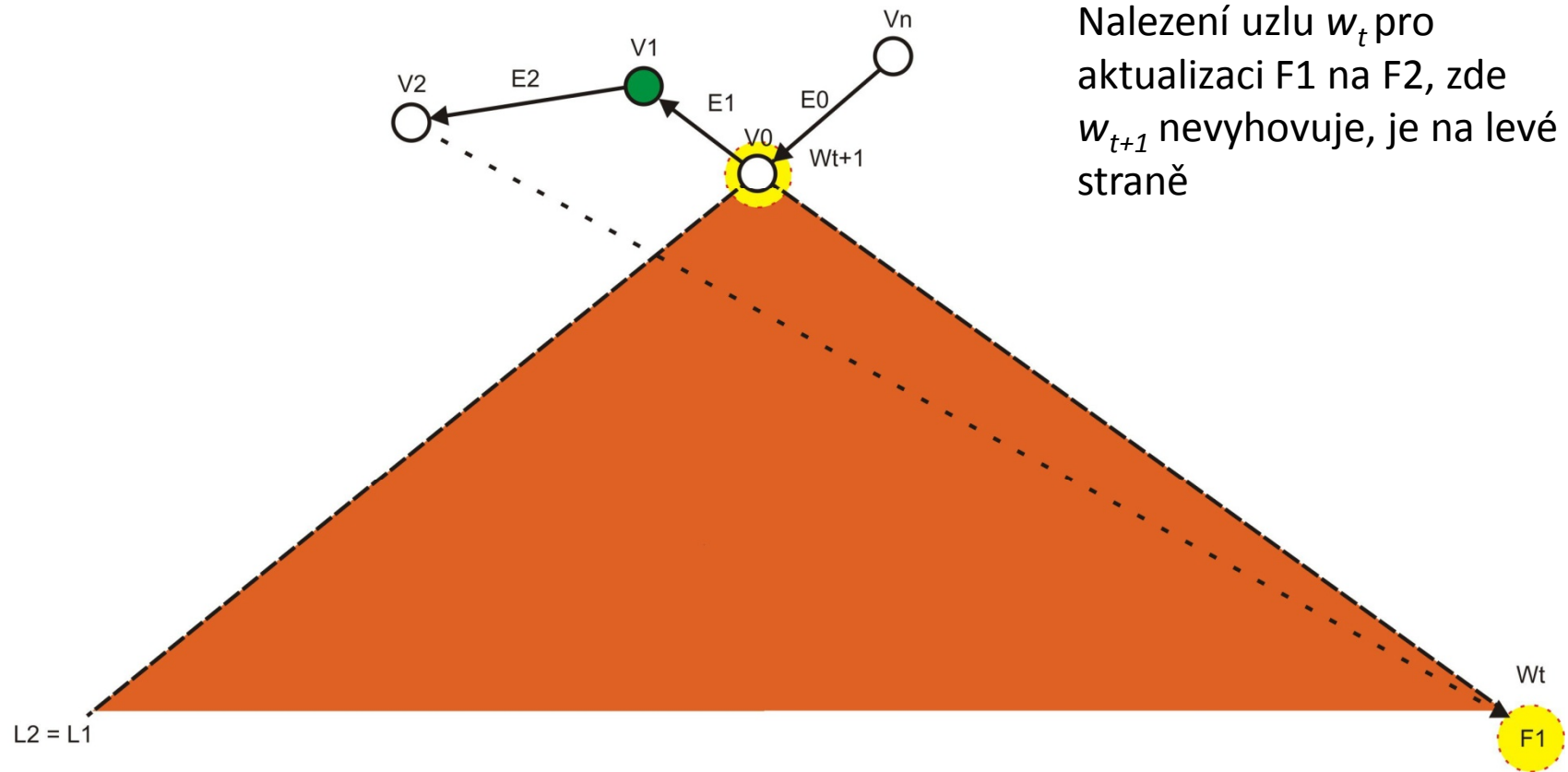
Obecný krok

2.2



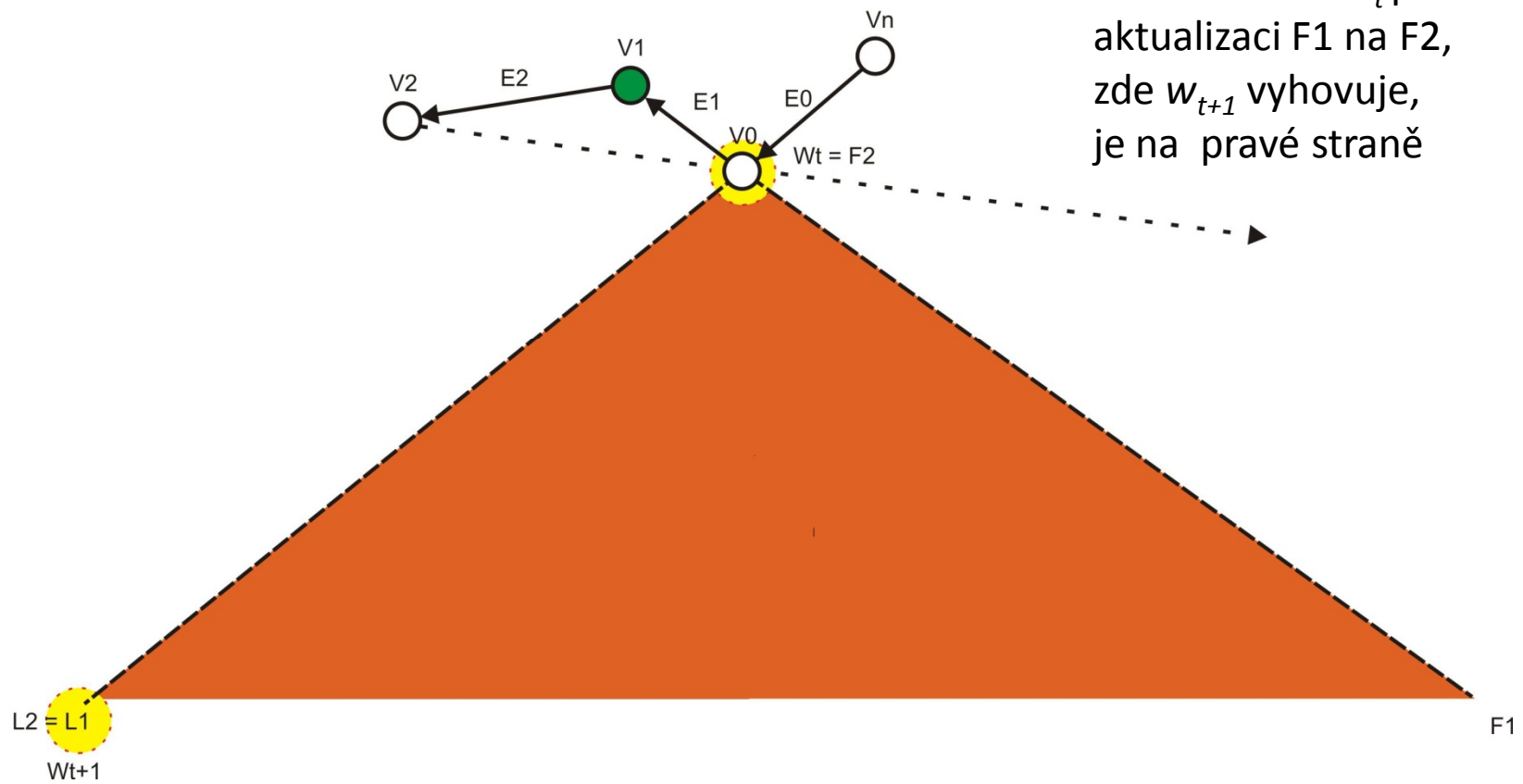
Obecný krok

2.2



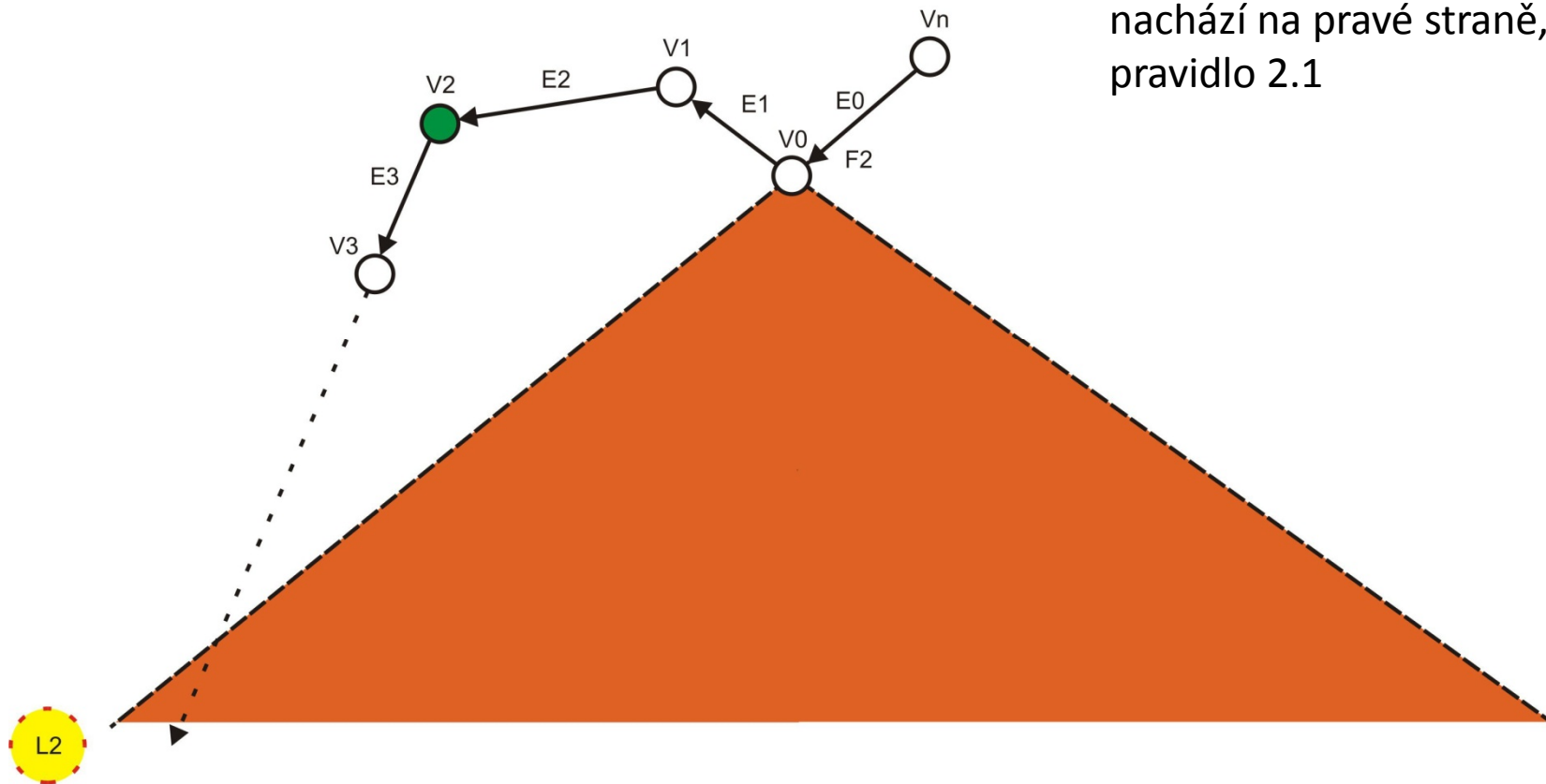
Obecný krok

2.2



Obecný krok

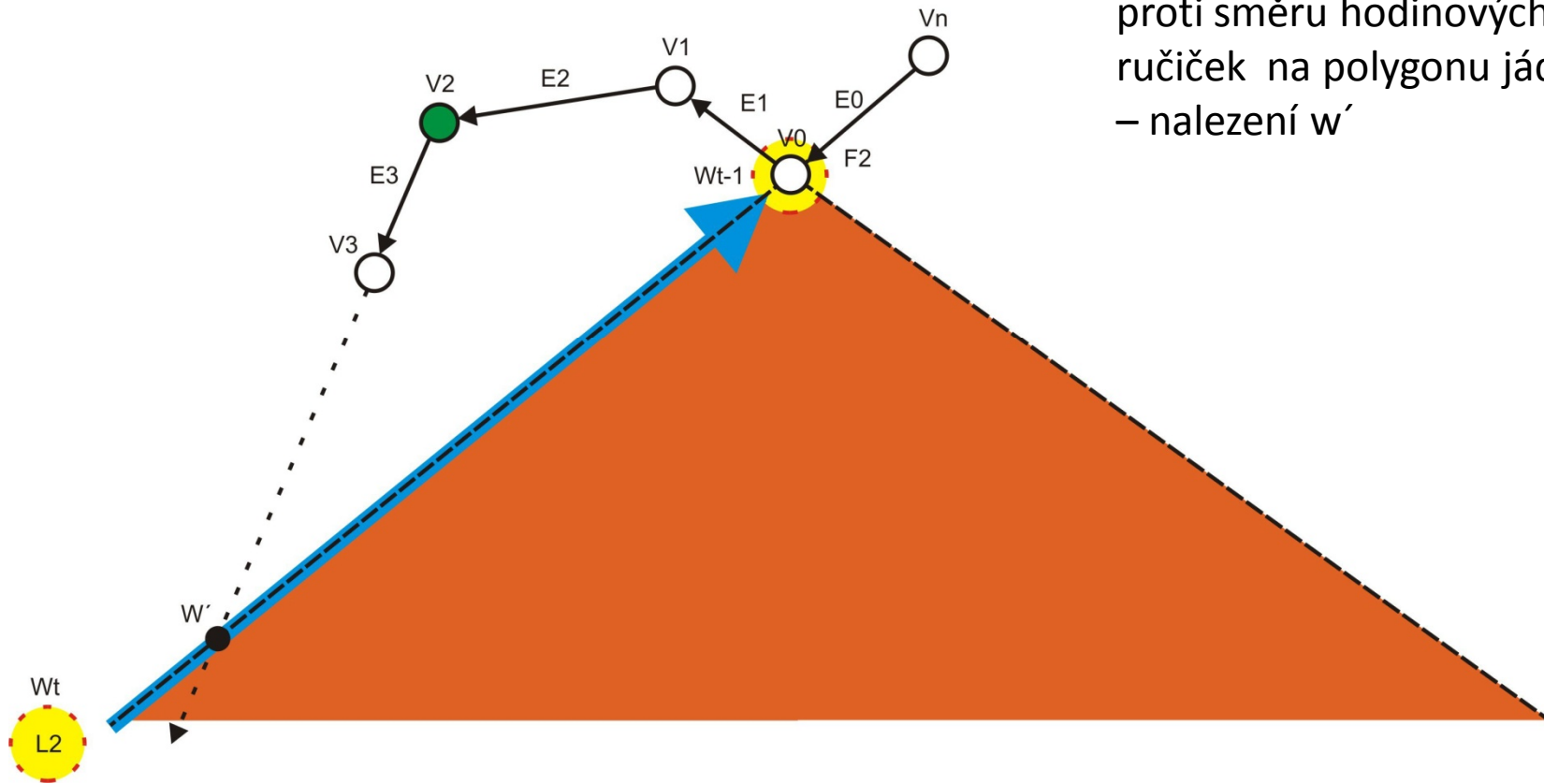
2.1



Obecný krok

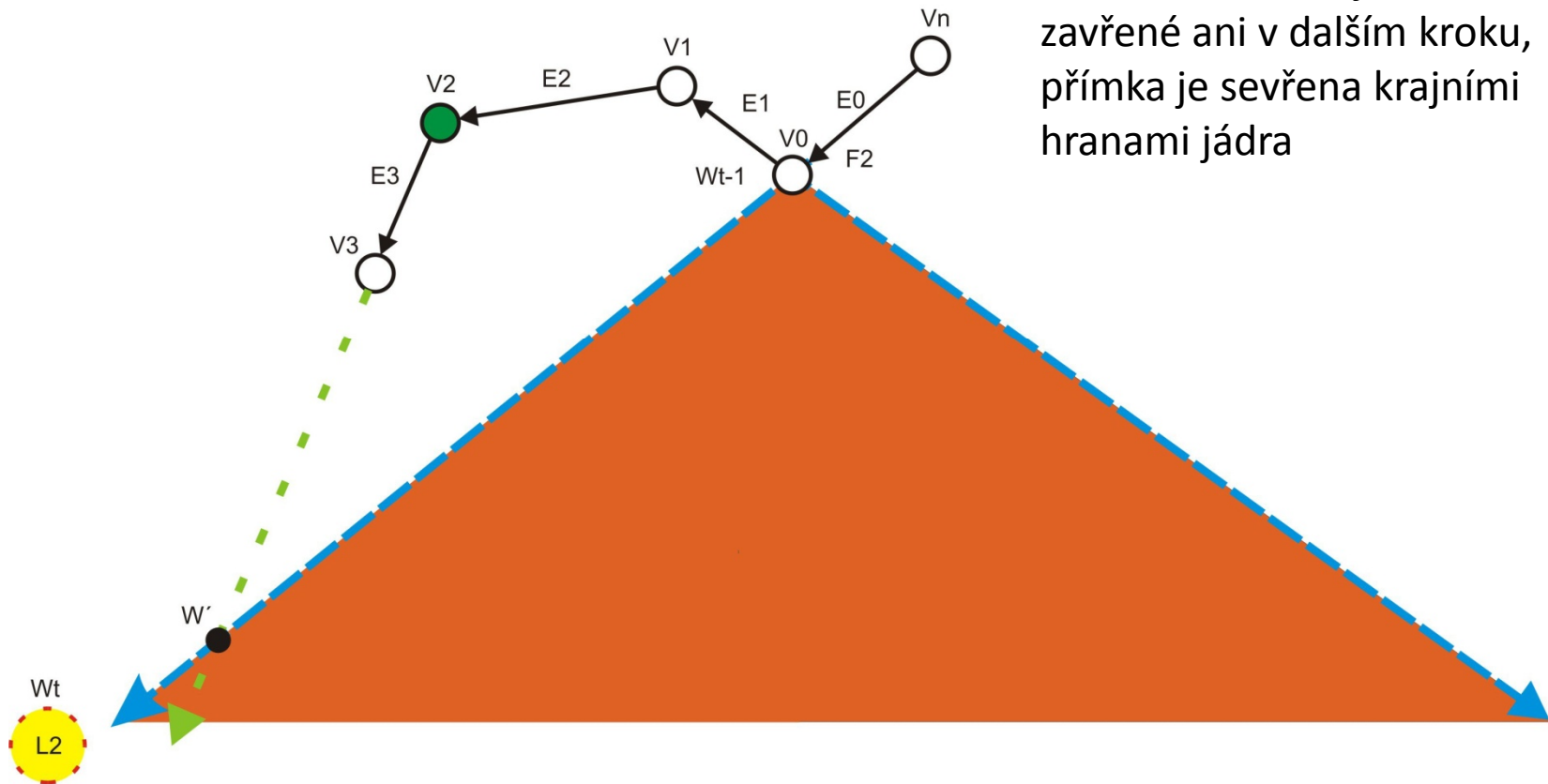
2.1

Hledání od L_i průsečíku
proti směru hodinových
ručiček na polygonu jádra
– nalezení w'



Obecný krok

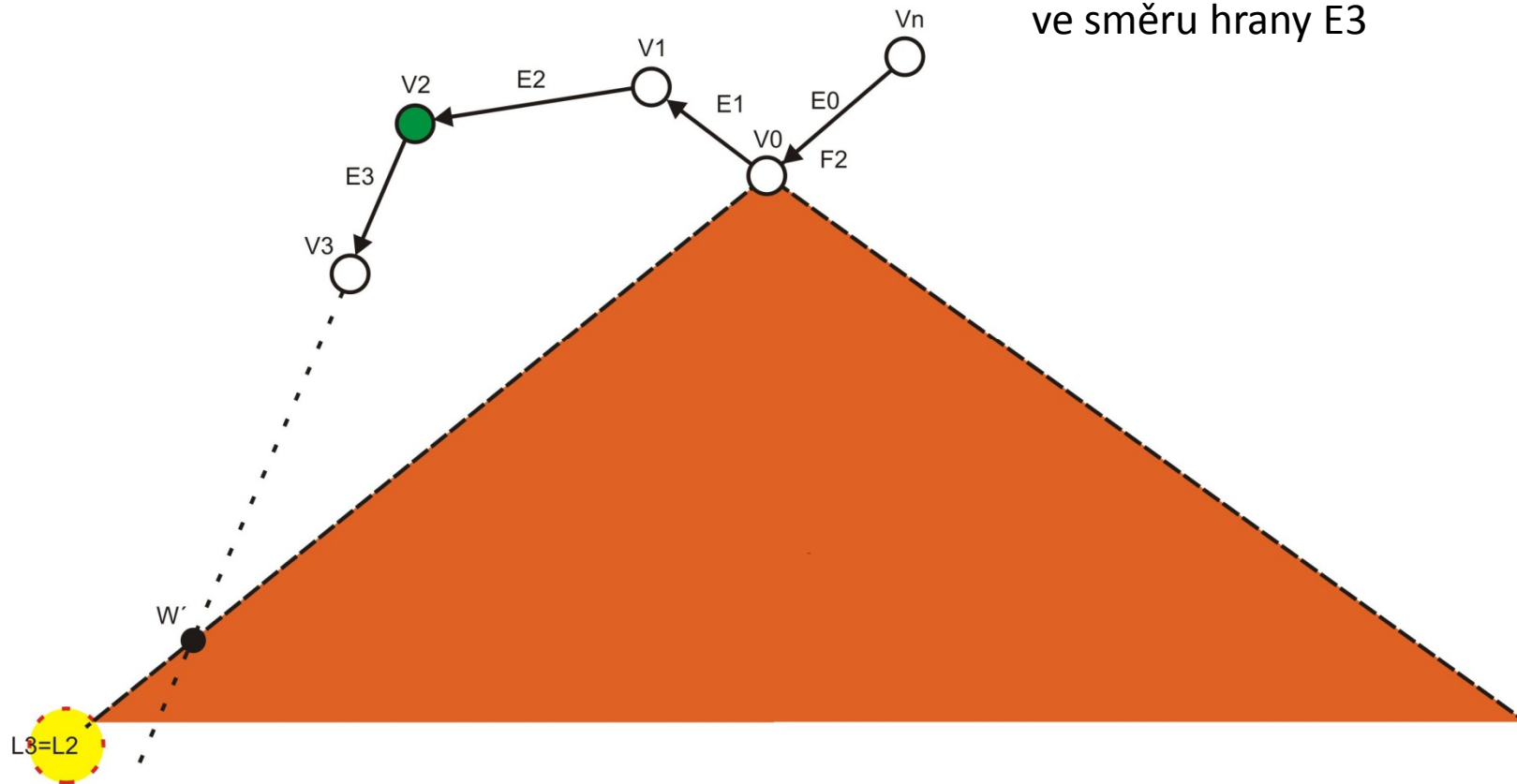
2.1



Obecný krok

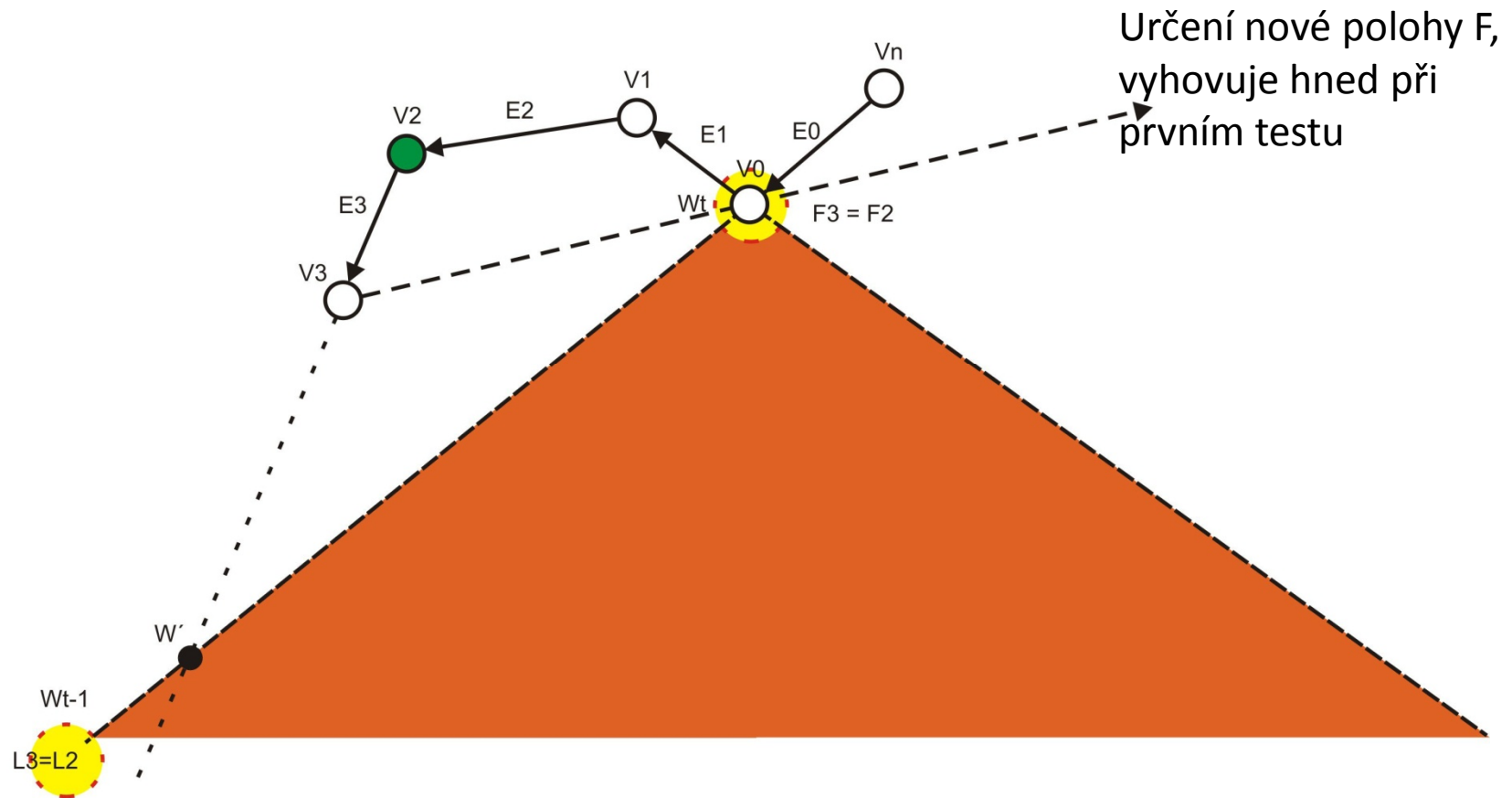
2.1

Posun L3 do nekonečna
ve směru hrany E3



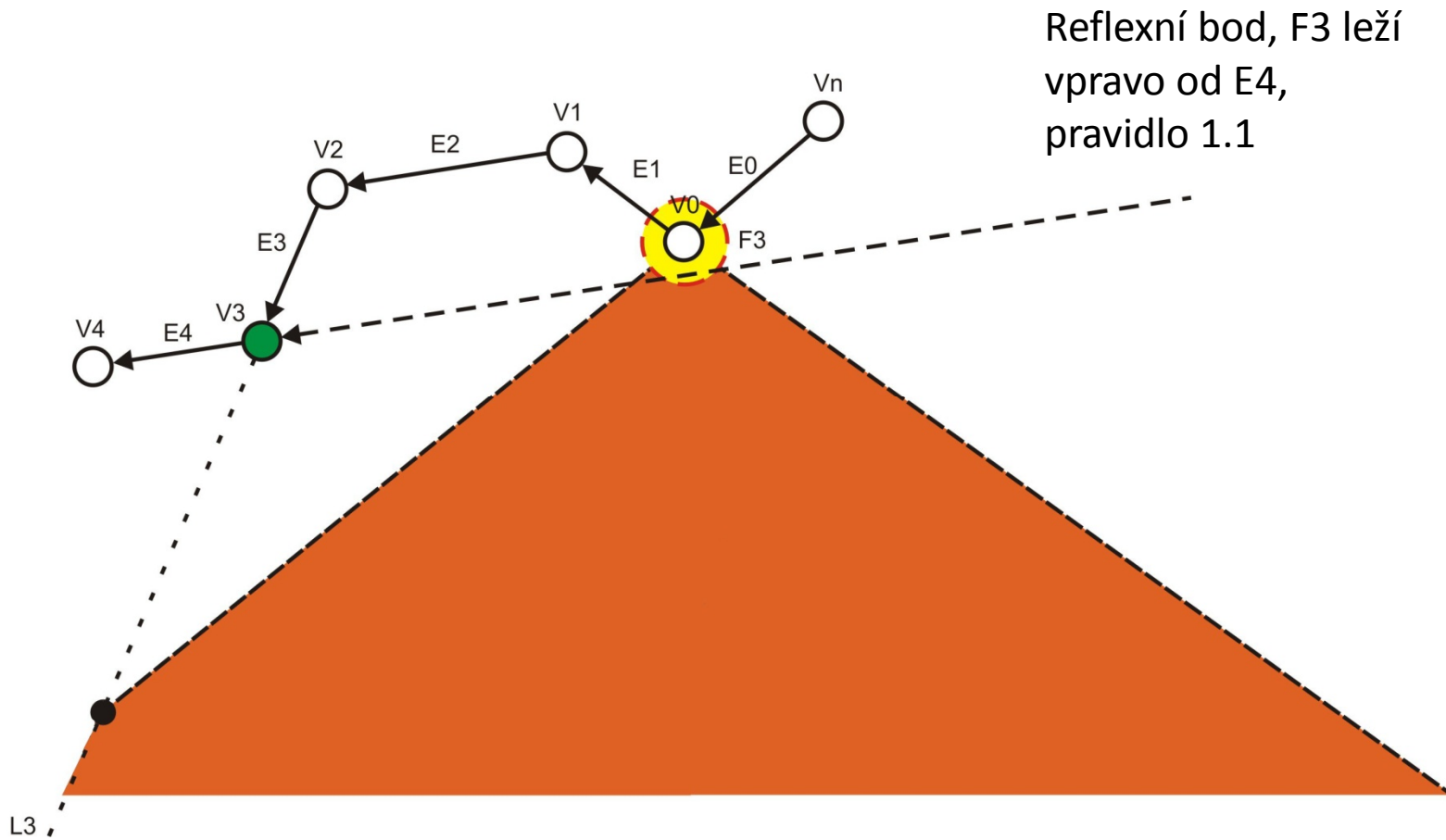
Obecný krok

2.1



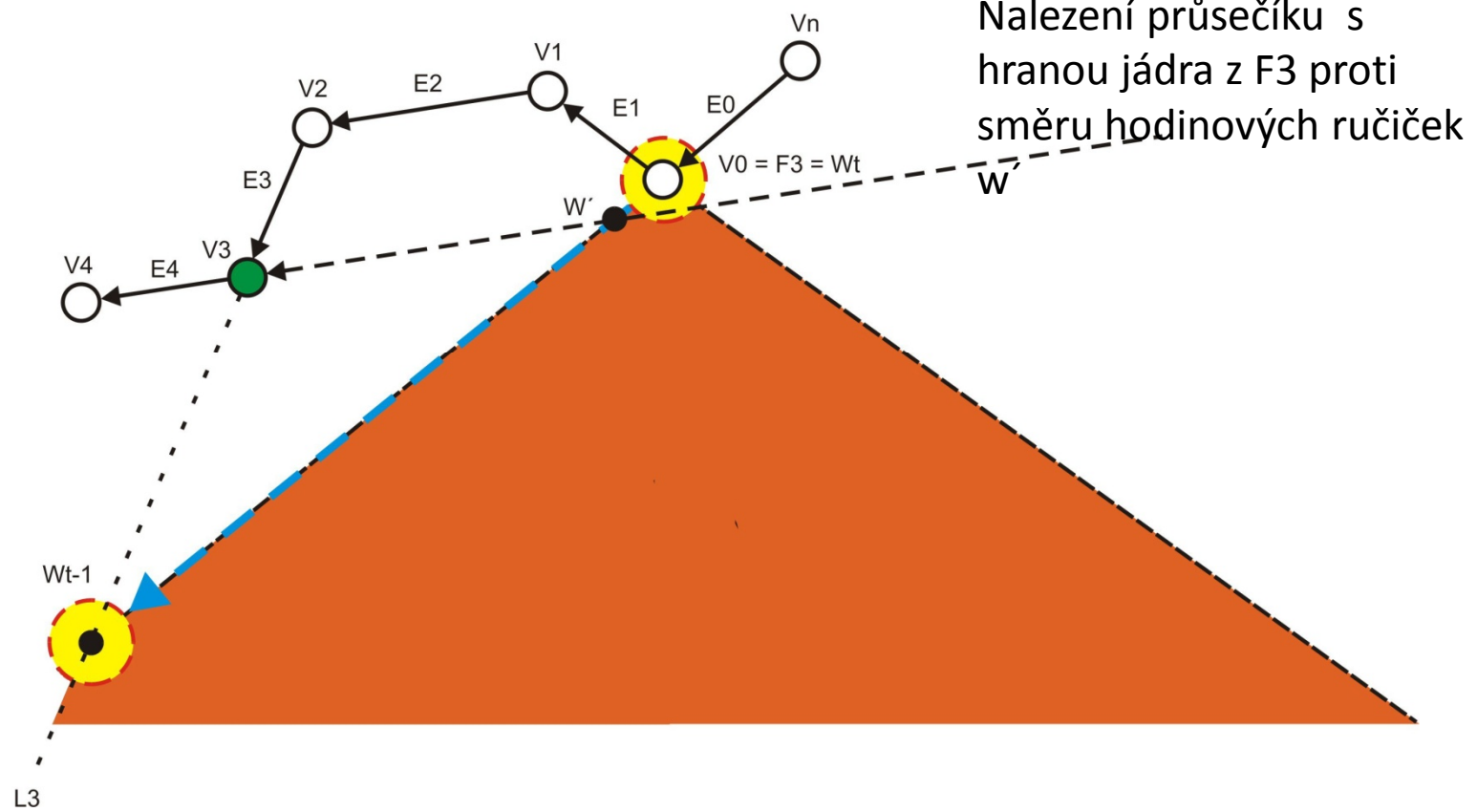
Obecný krok

1.1



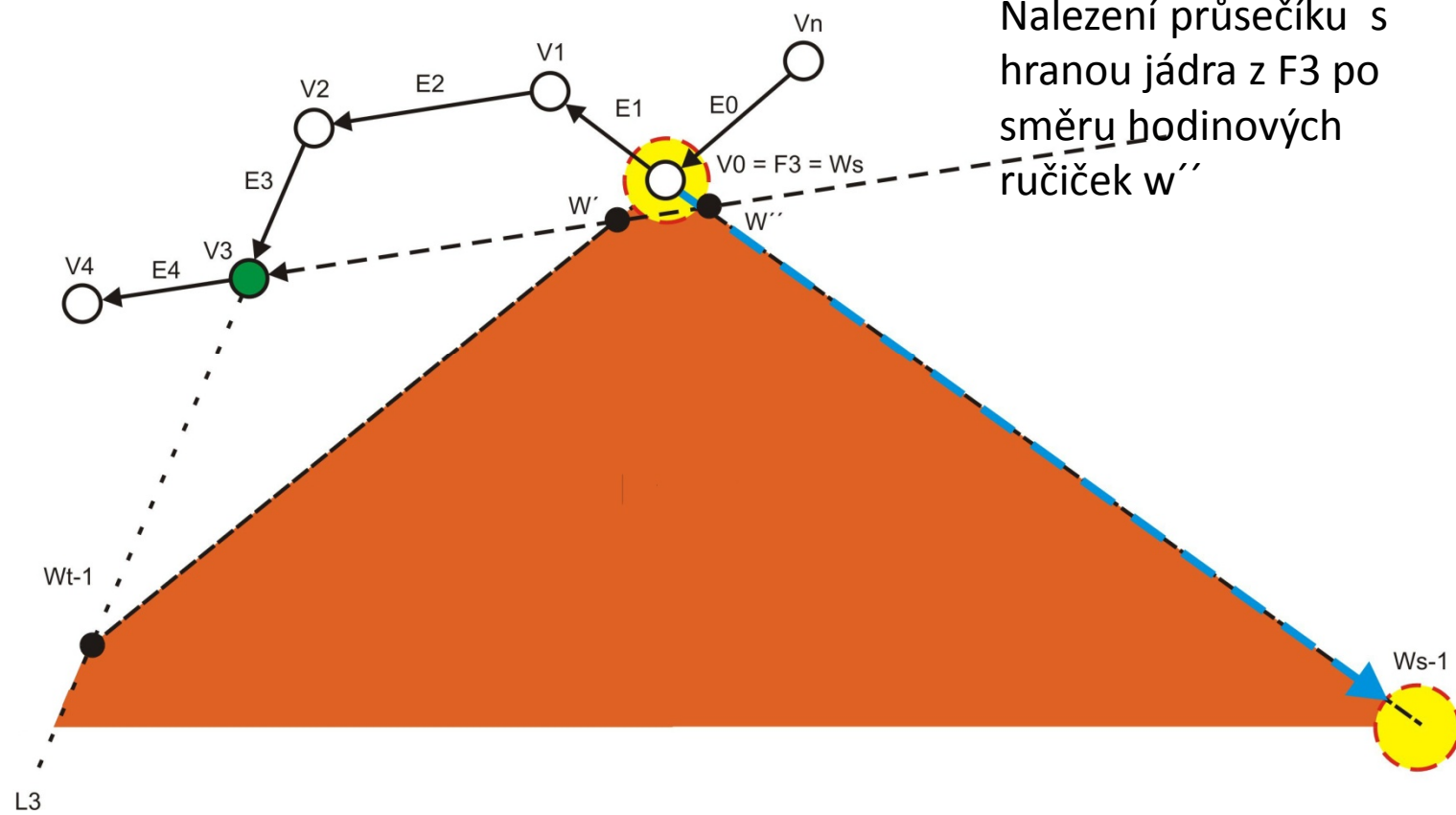
Obecný krok

1.1



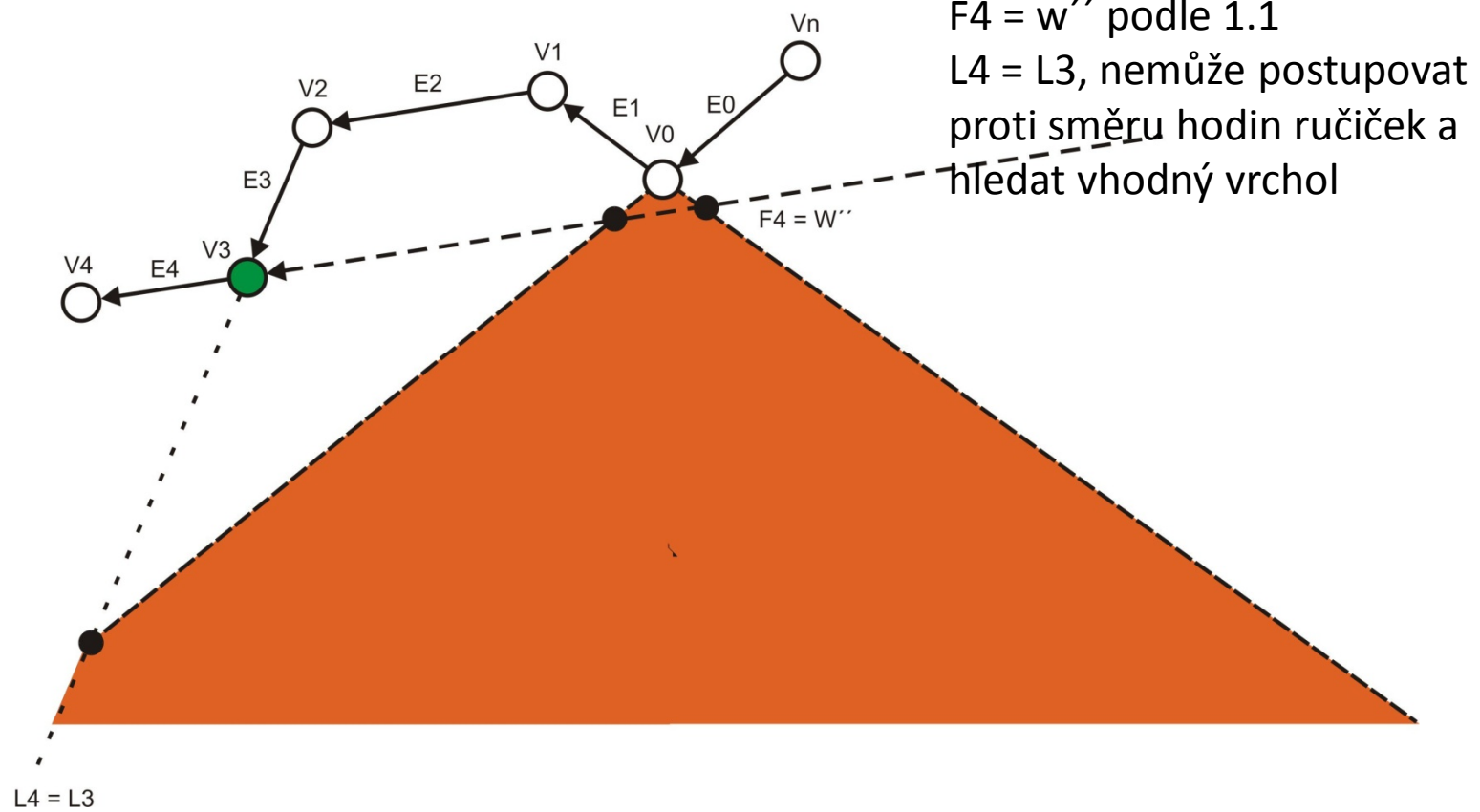
Obecný krok

1.1



Obecný krok

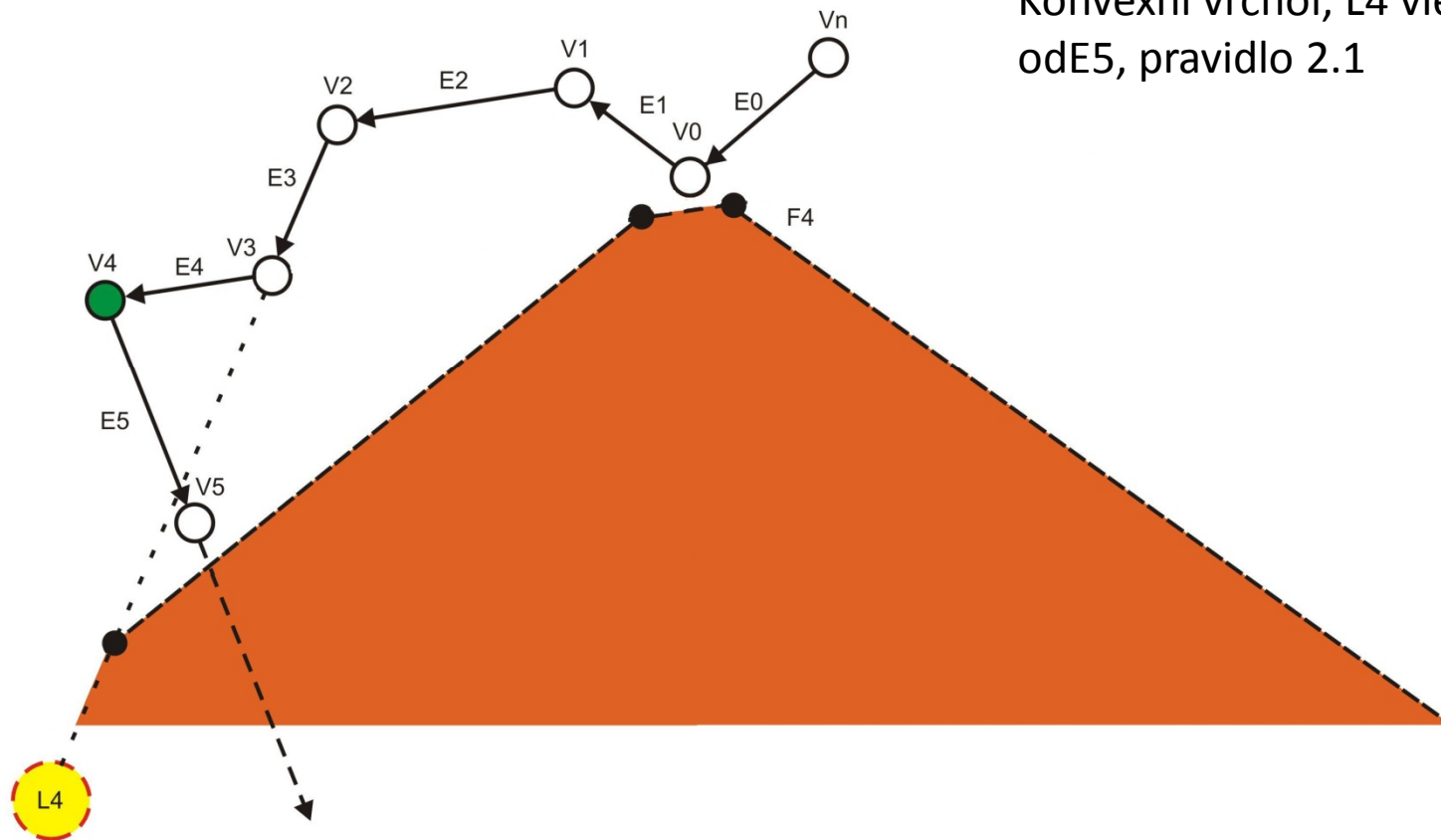
1.1



Obecný krok

2.1

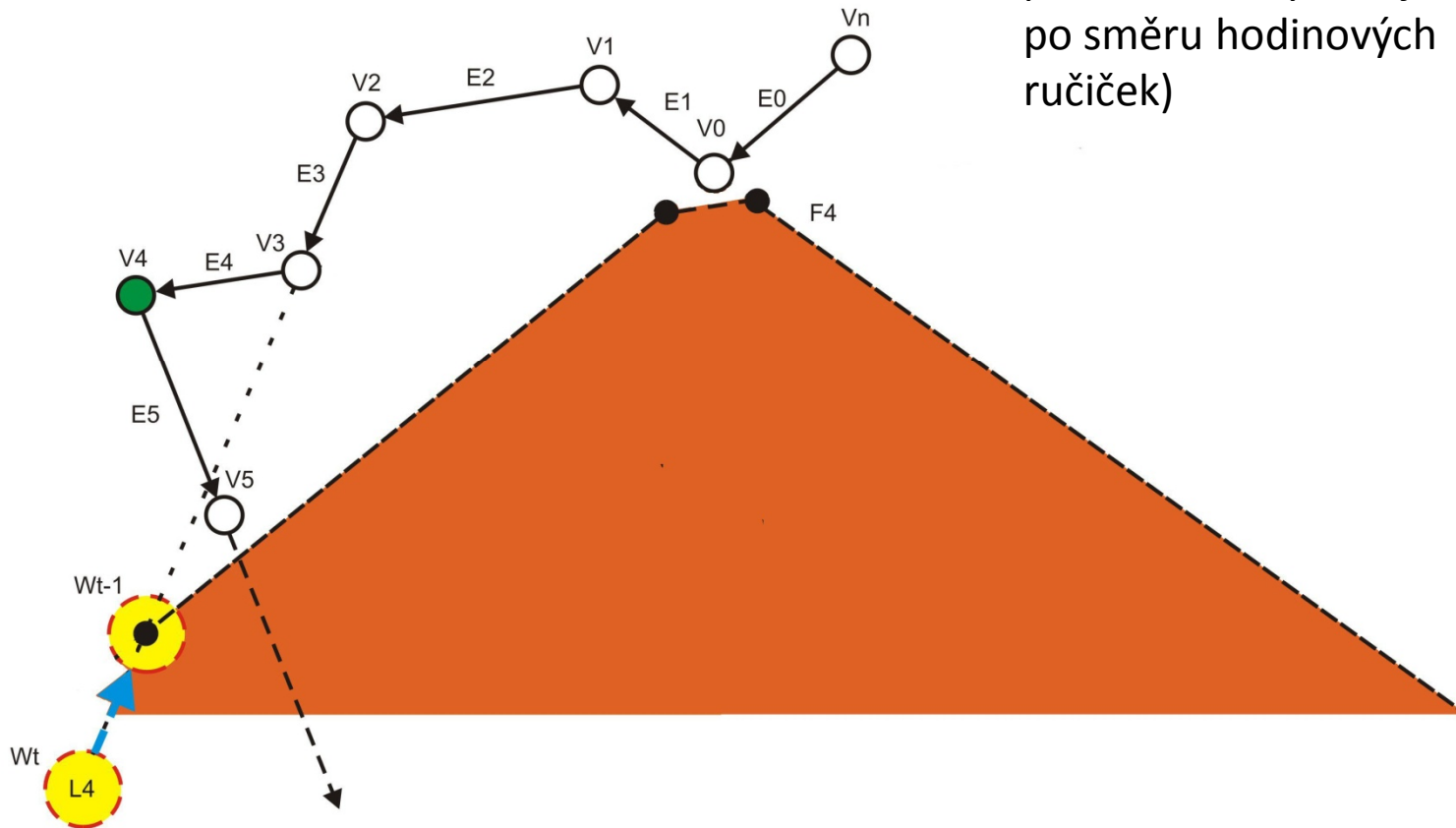
Konvexní vrchol, L4 vlevo
od E5, pravidlo 2.1



Obecný krok

2.1

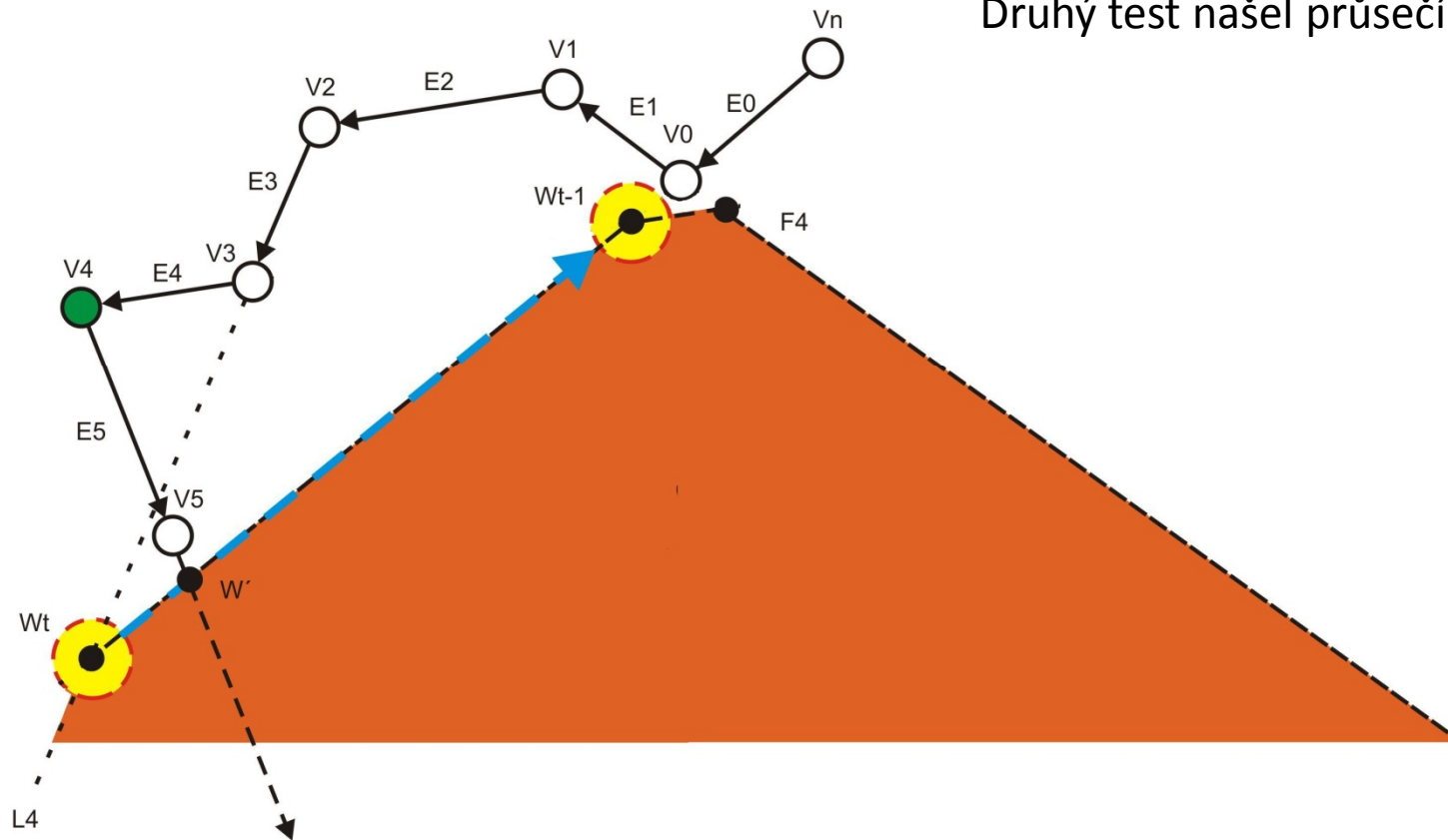
Hledání průsečíku přímky E5 a hrany jádra polygonu, první test nevyhovuje (Z L4 po směru hodinových ručiček)



Obecný krok

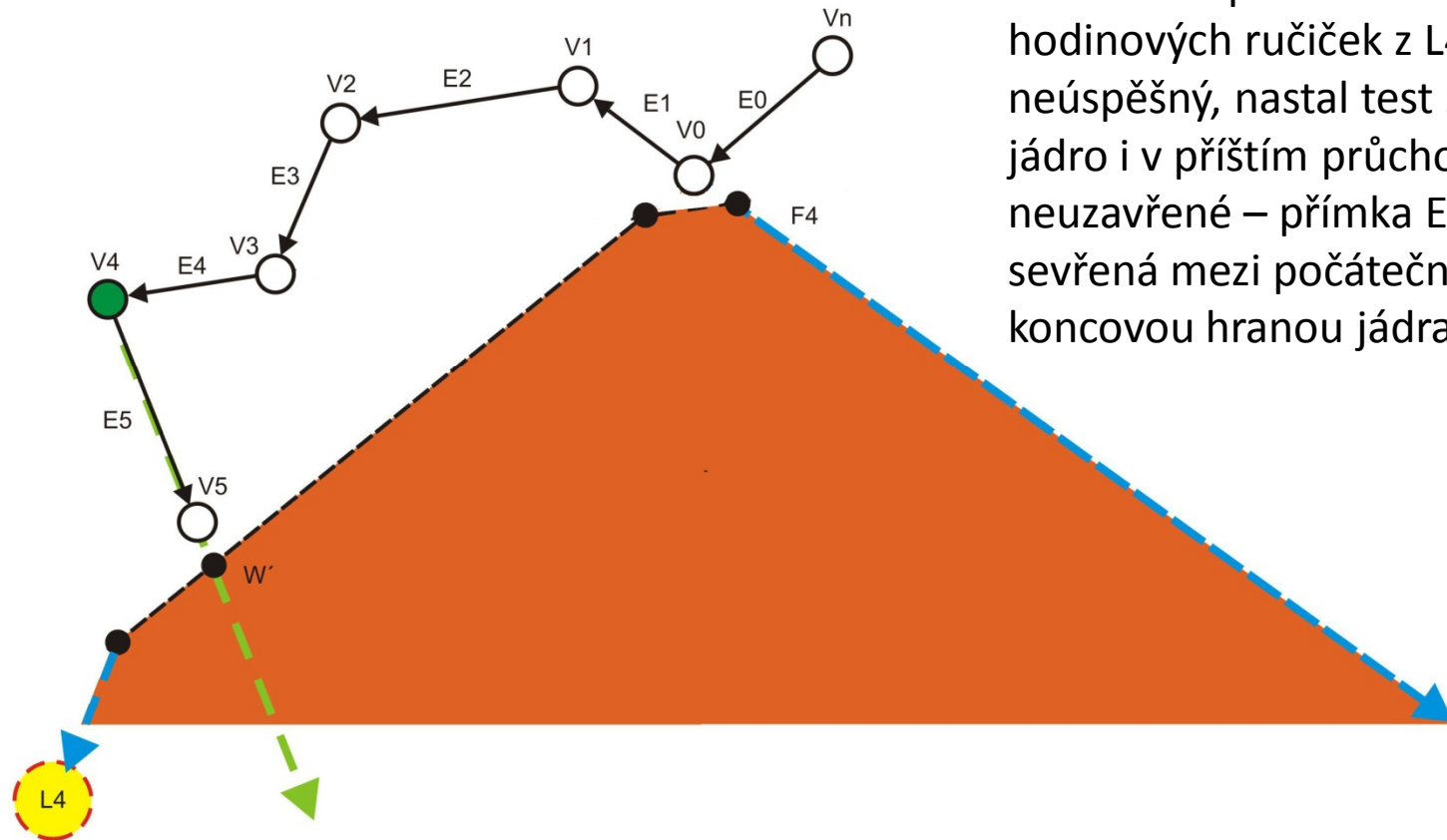
2.1

Druhý test našel průsečík w'



Obecný krok

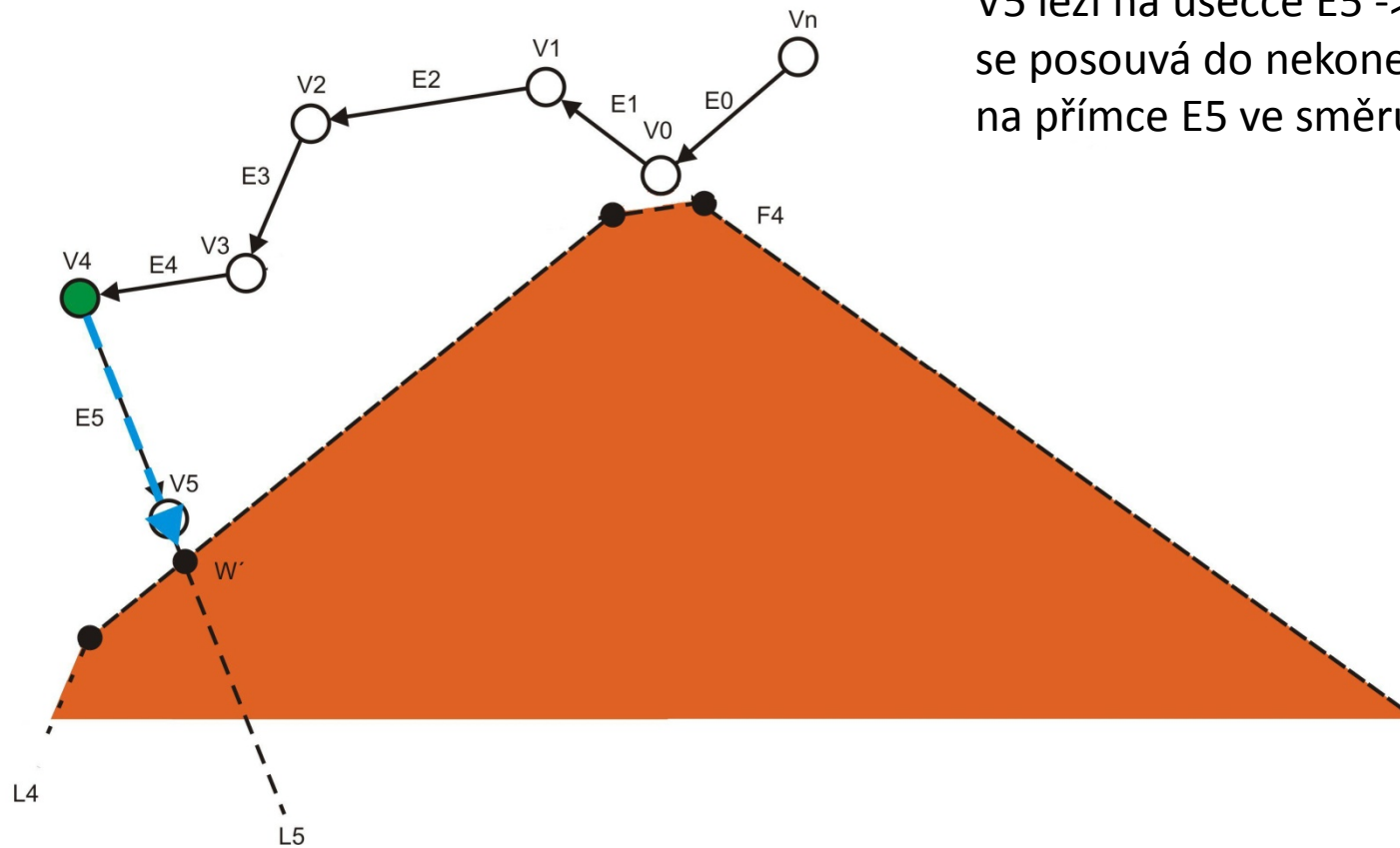
2.1



Test na w'' proti směru hodinových ručiček z L_4 byl neúspěšný, nastal test zda bude jádro i v příštím průchodu neuzavřené – přímka E_5 bude, sevřená mezi počátečním a koncovou hranou jádra

Obecný krok

2.1

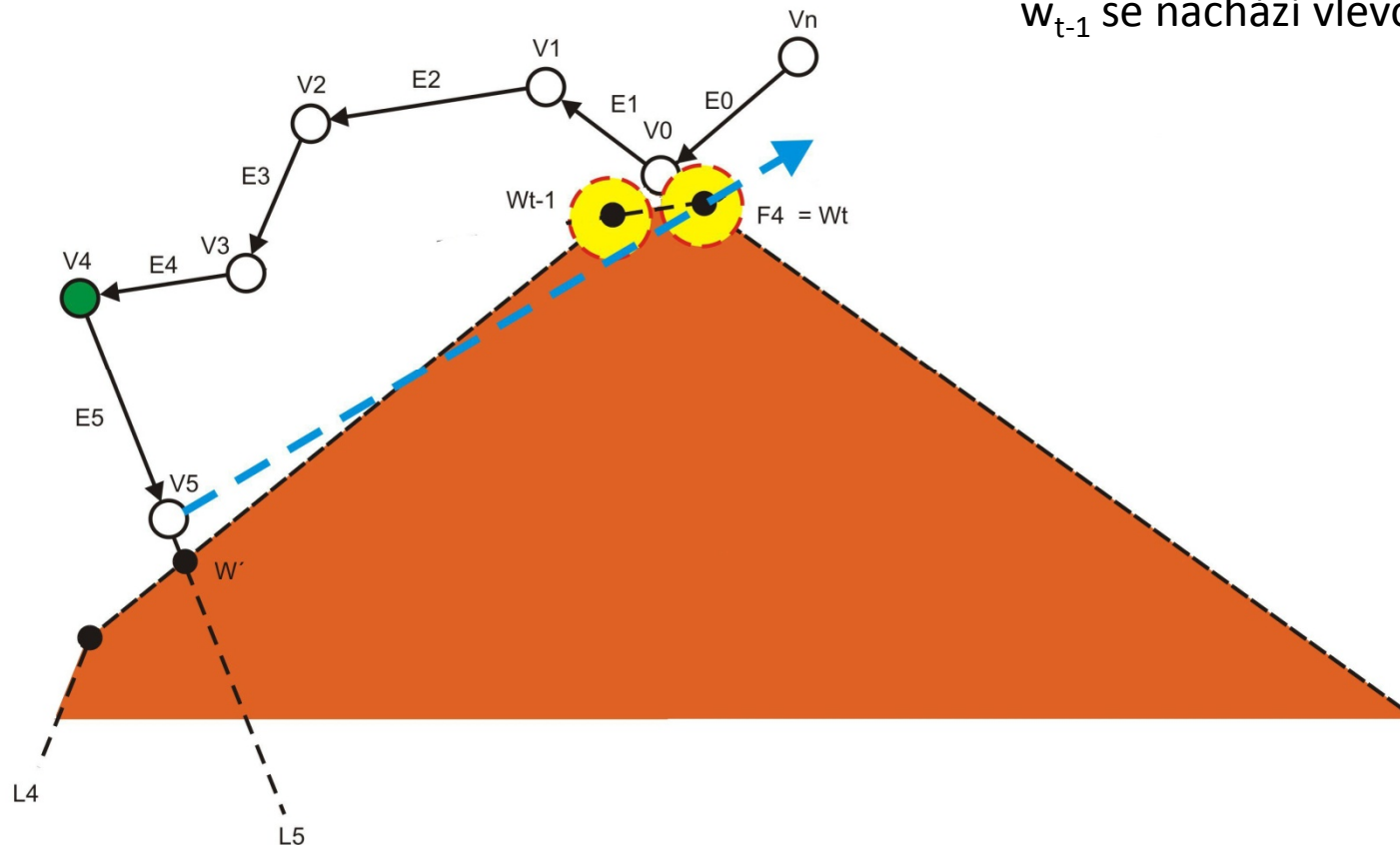


V_5 leží na úsečce $E_5 \rightarrow L_5$
se posouvá do nekonečna
na přímce E_5 ve směru E_5

Obecný krok

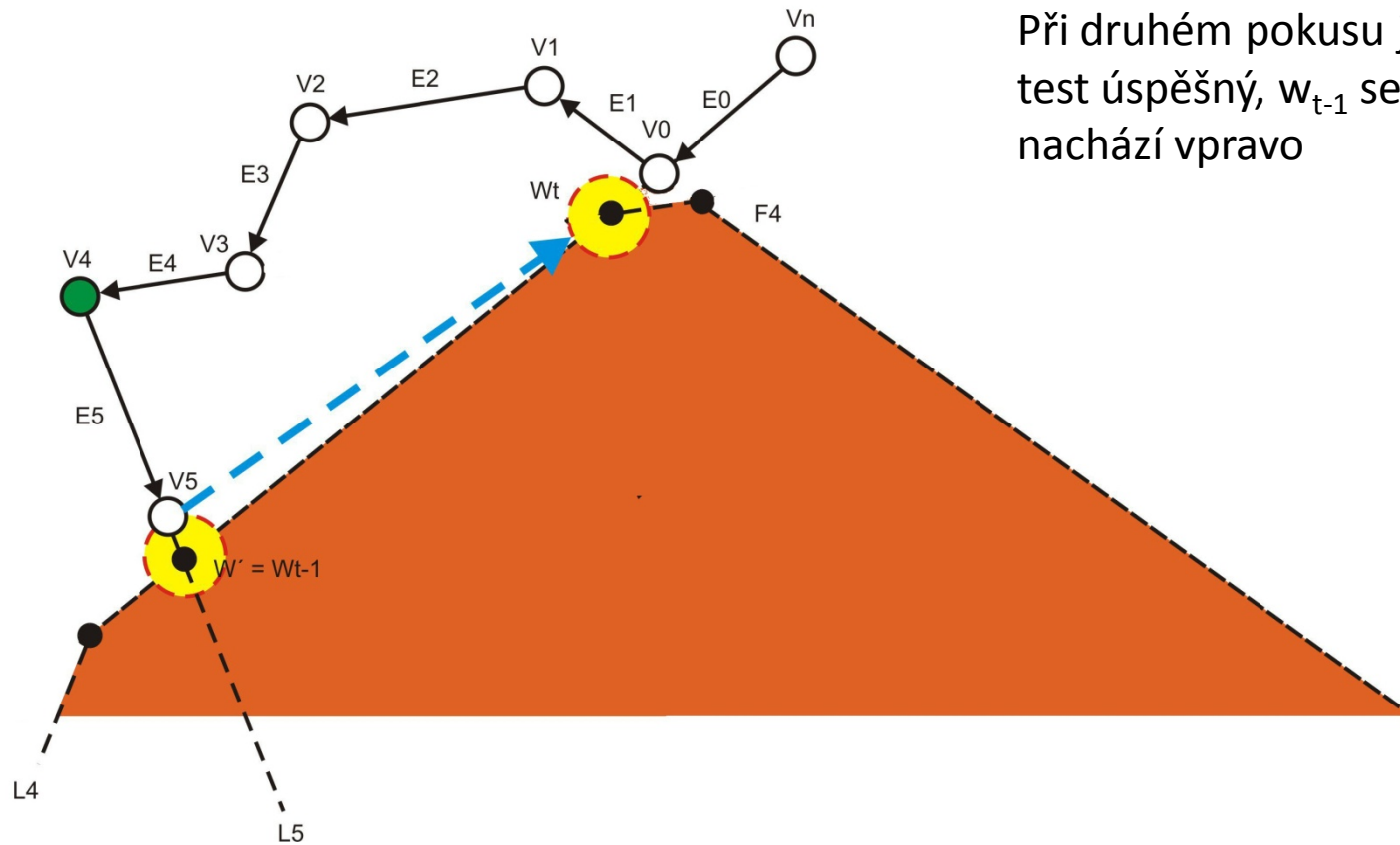
2.1

Test na aktualizaci F5, v tomto případě neprojde, w_{t-1} se nachází vlevo



Obecný krok

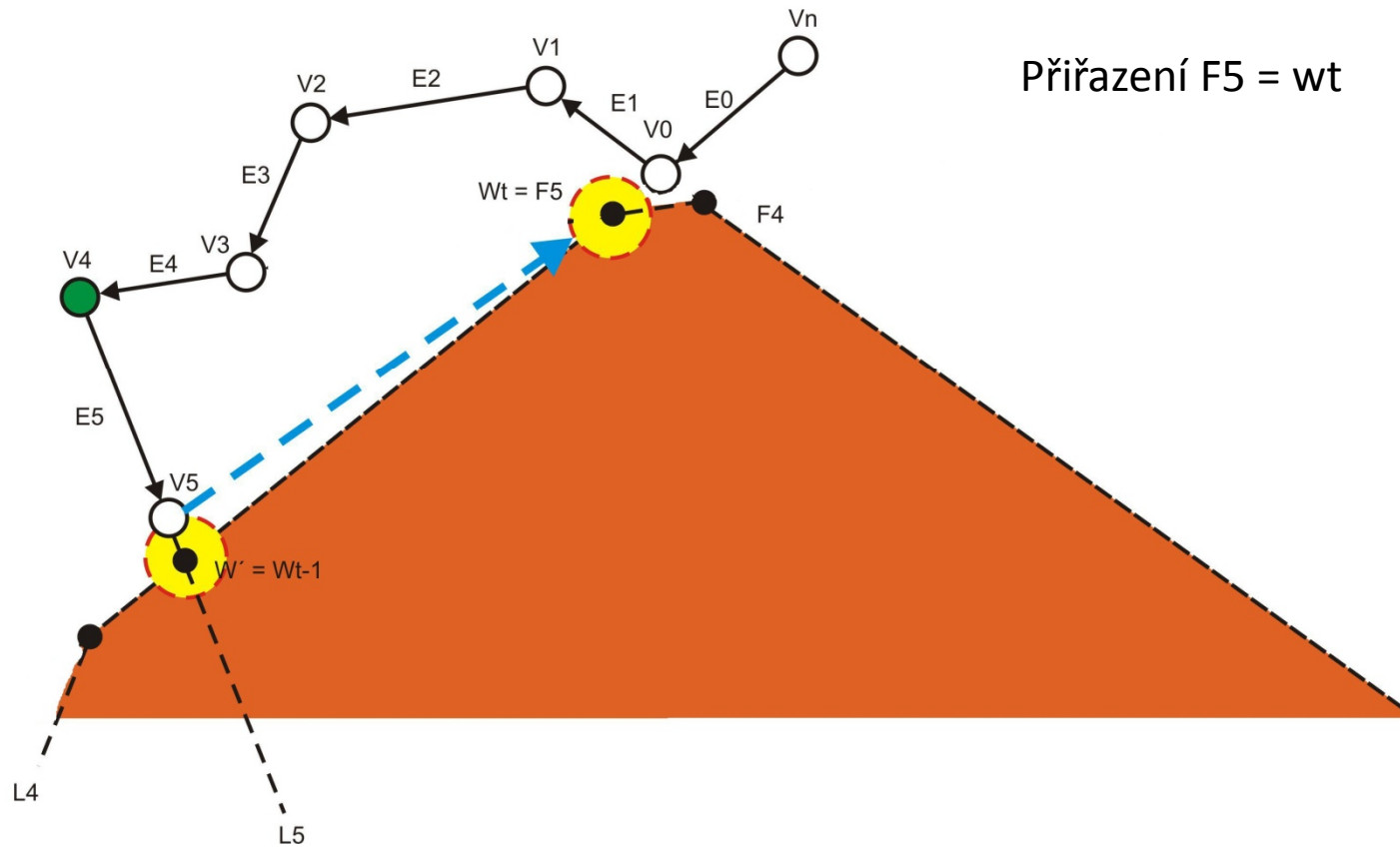
2.1



Při druhém pokusu je test úspěšný, w_{t-1} se nachází vpravo

Obecný krok

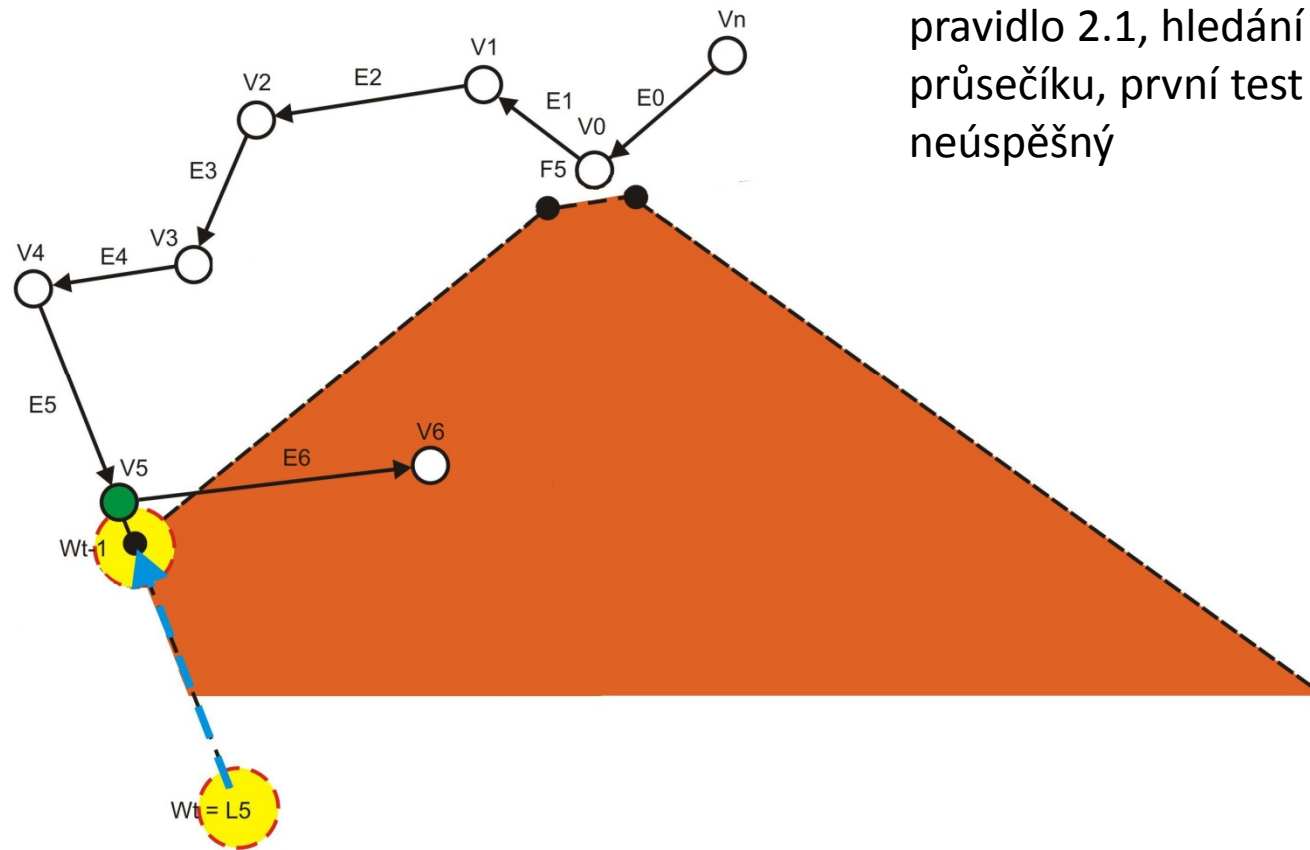
2.1



Obecný krok

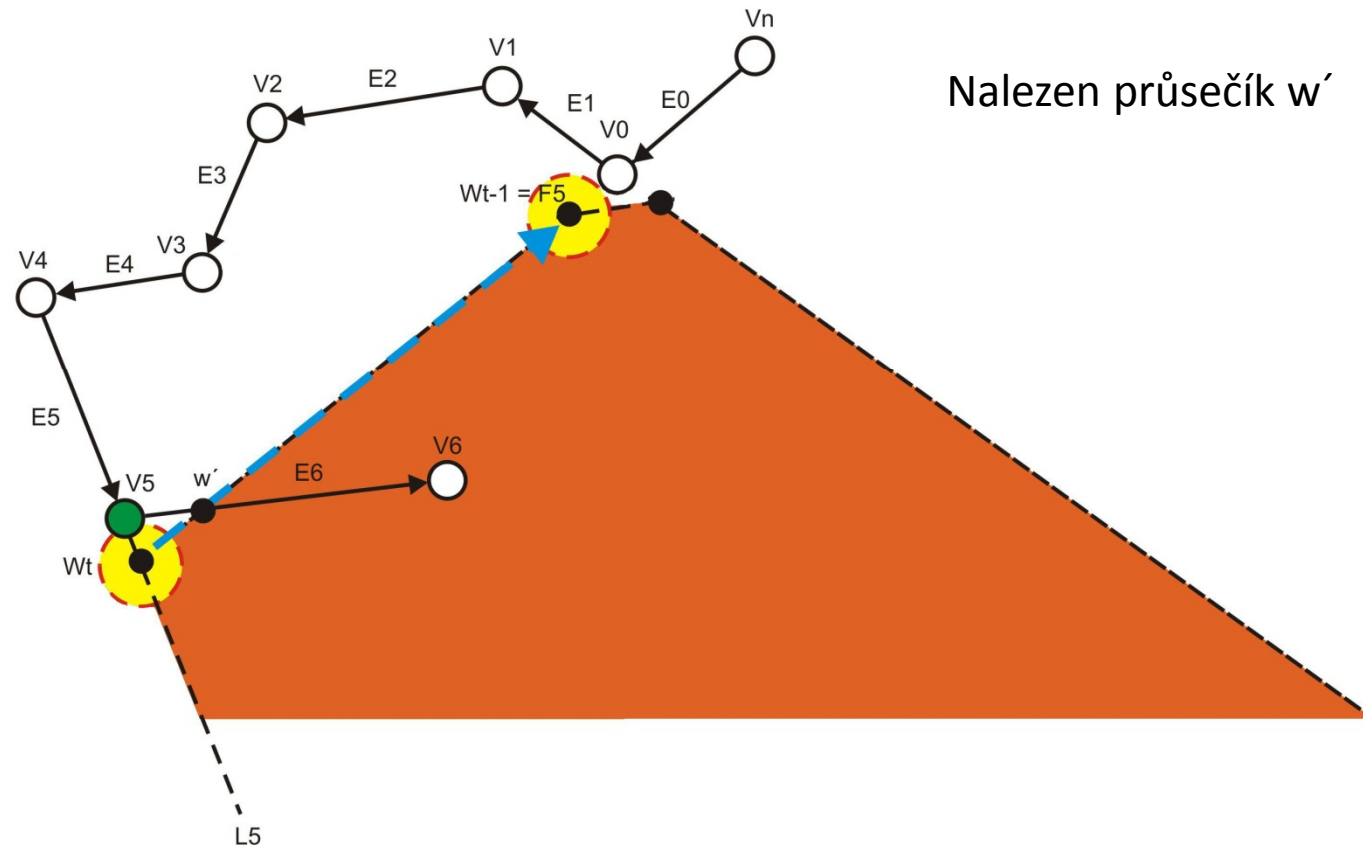
2.1

Konvexní vrchol L5 vlevo,
pravidlo 2.1, hledání
průsečíku, první test
neúspěšný



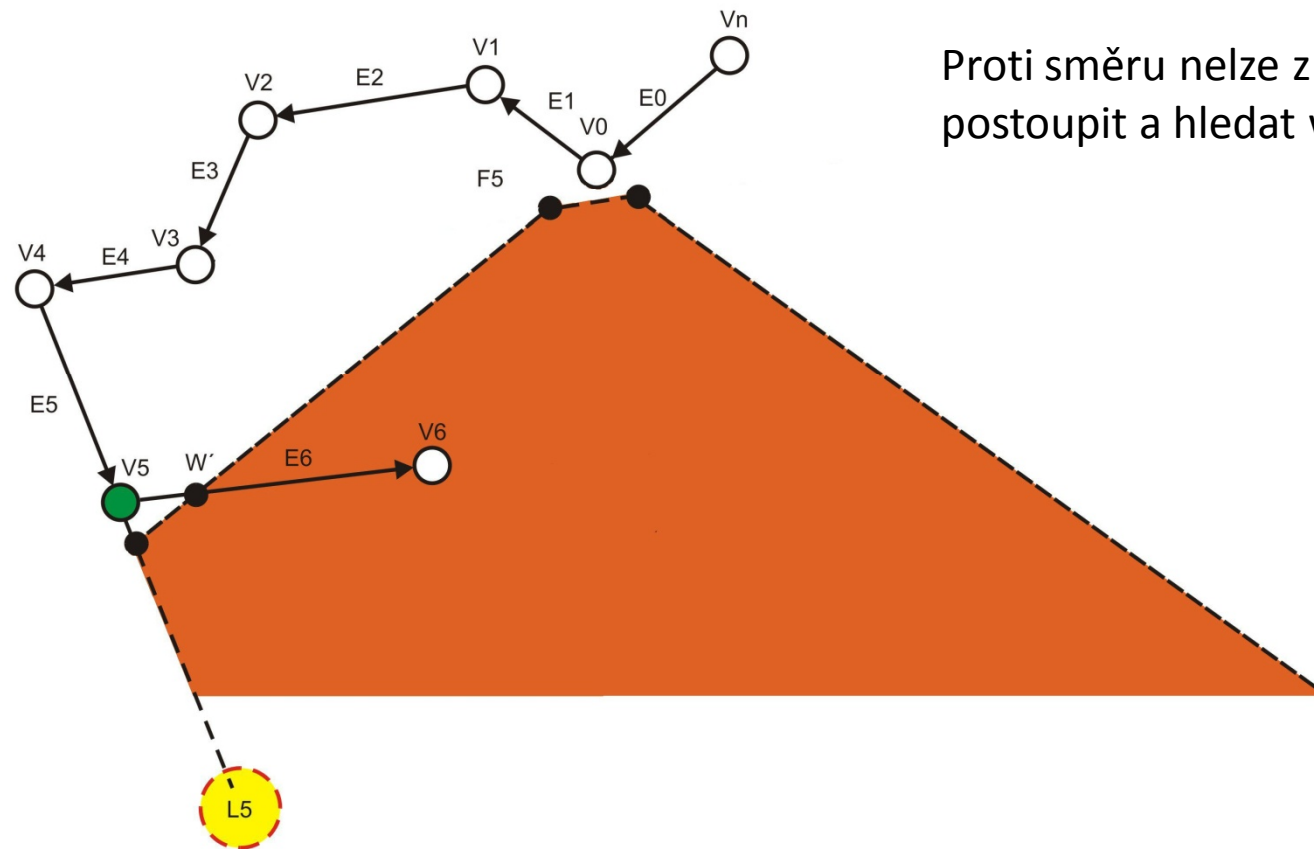
Obecný krok

2.1



Obecný krok

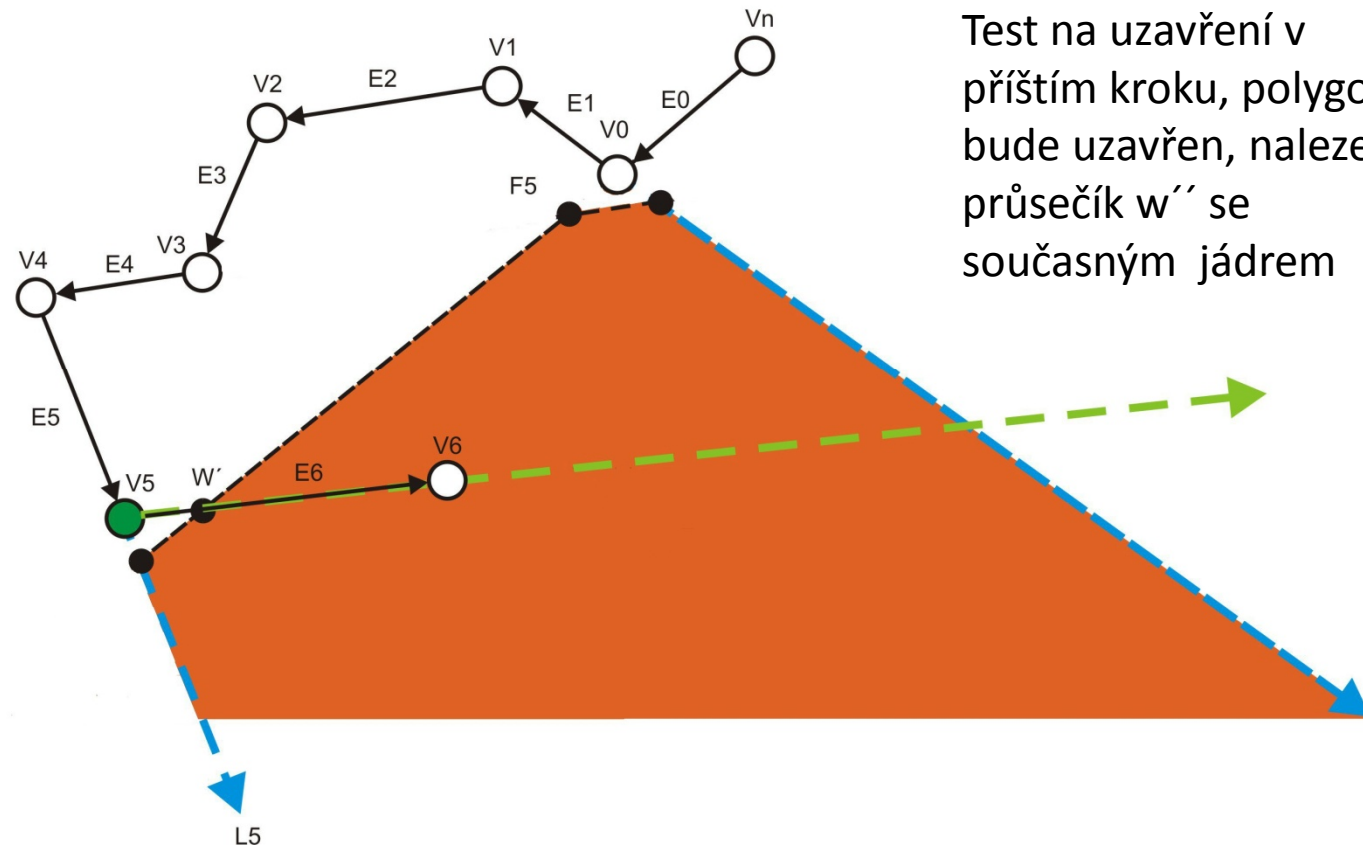
2.1



Proti směru nelze z L5
postoupit a hledat w''

Obecný krok

2.1

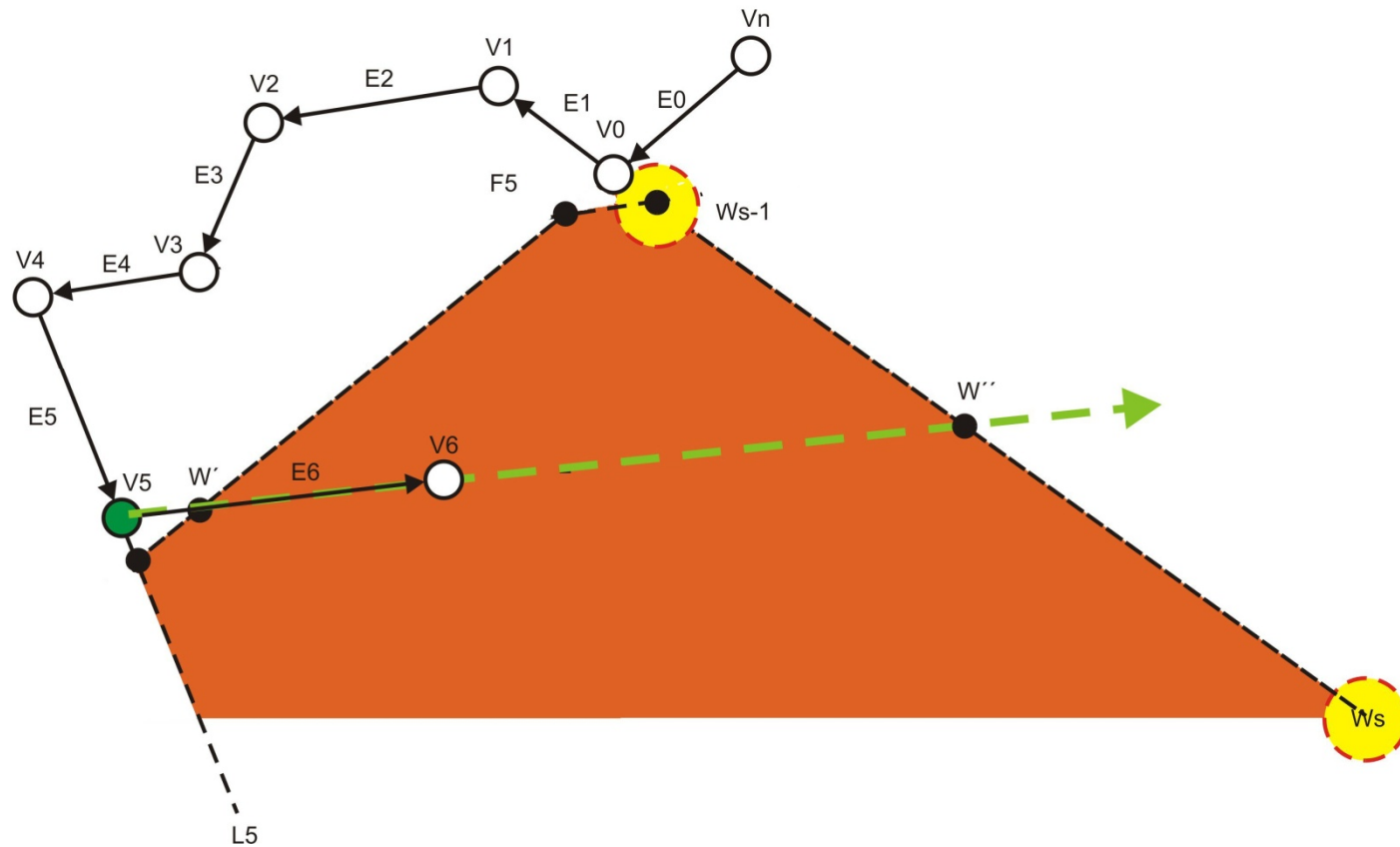


Test na uzavření v
příštím kroku, polygon
bude uzavřen, nalezen
průsečík w'' se
současným jádrem

Obecný krok

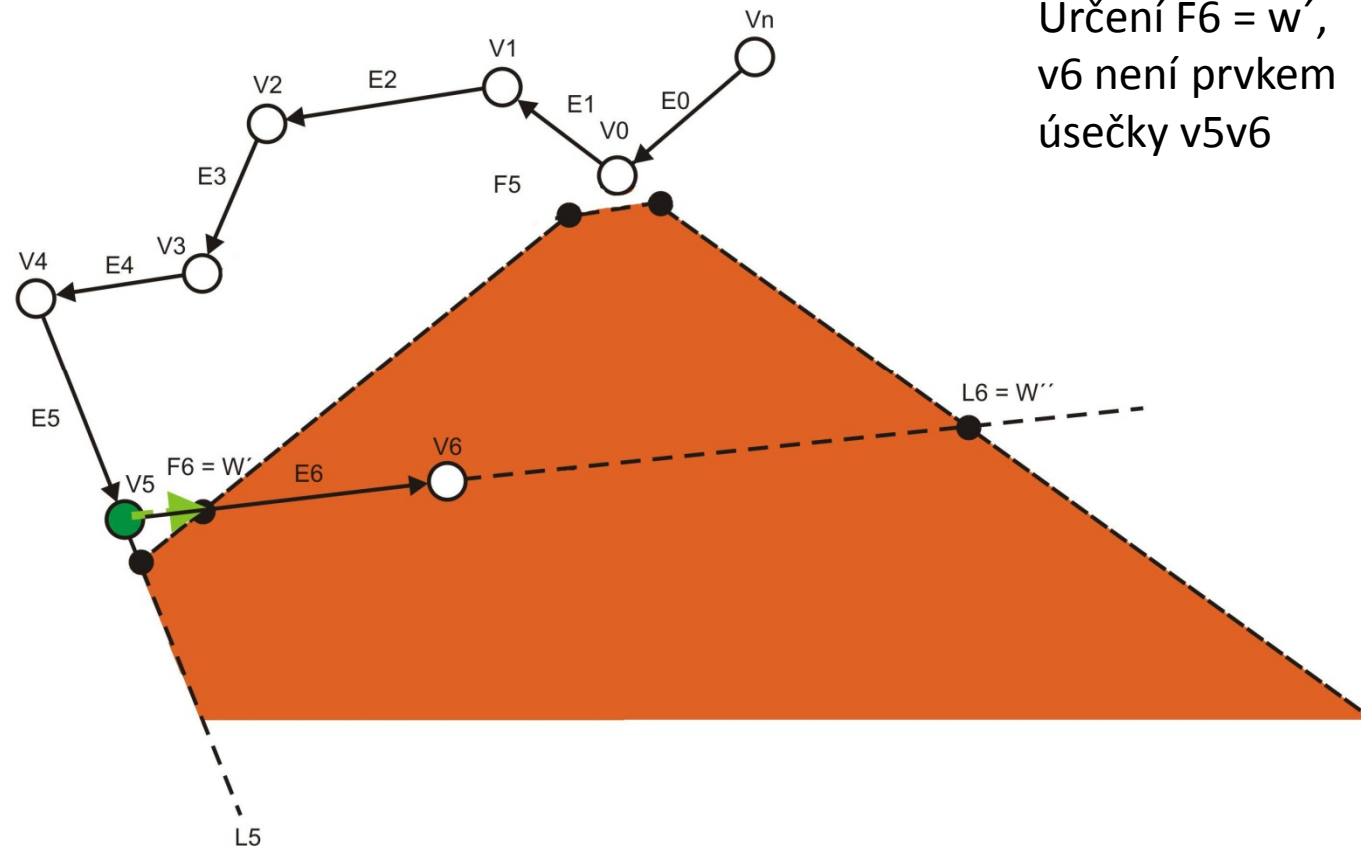
2.1

V6 je součástí úsečky
 $v5w'' \rightarrow F = w'$



Obecný krok

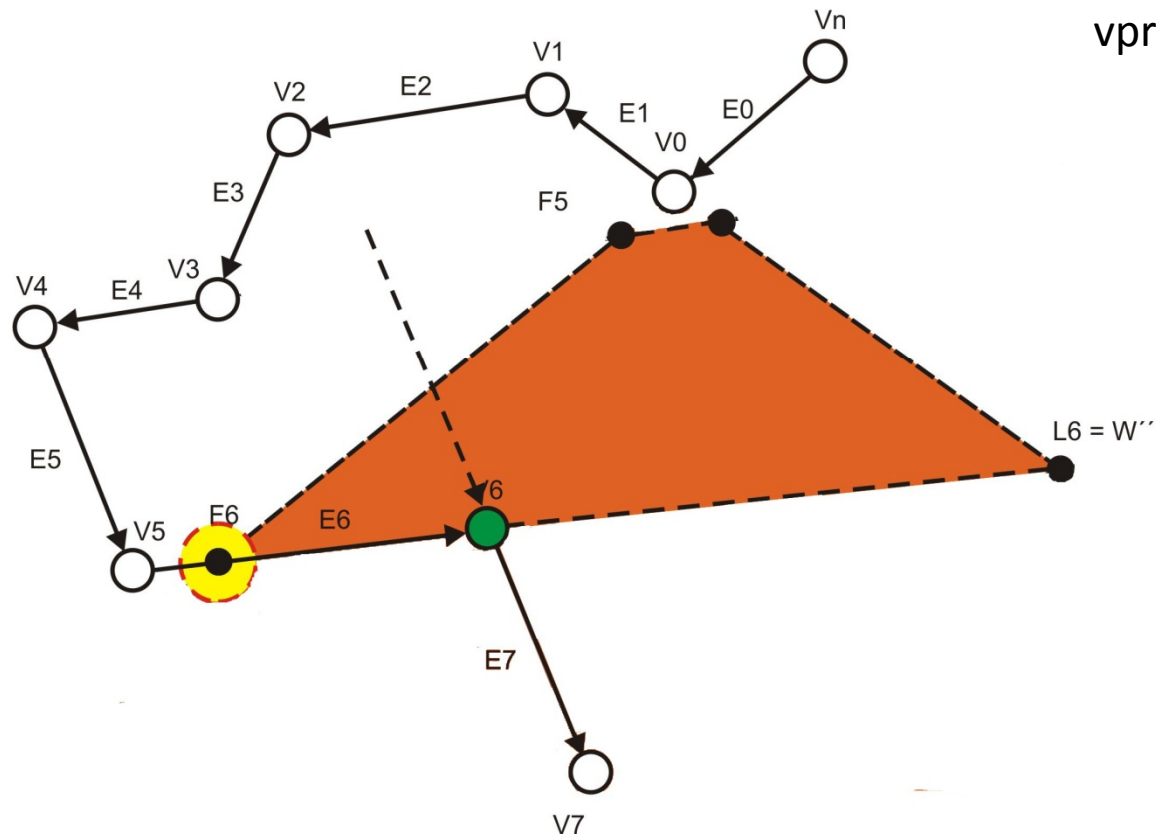
2.1



Obecný krok

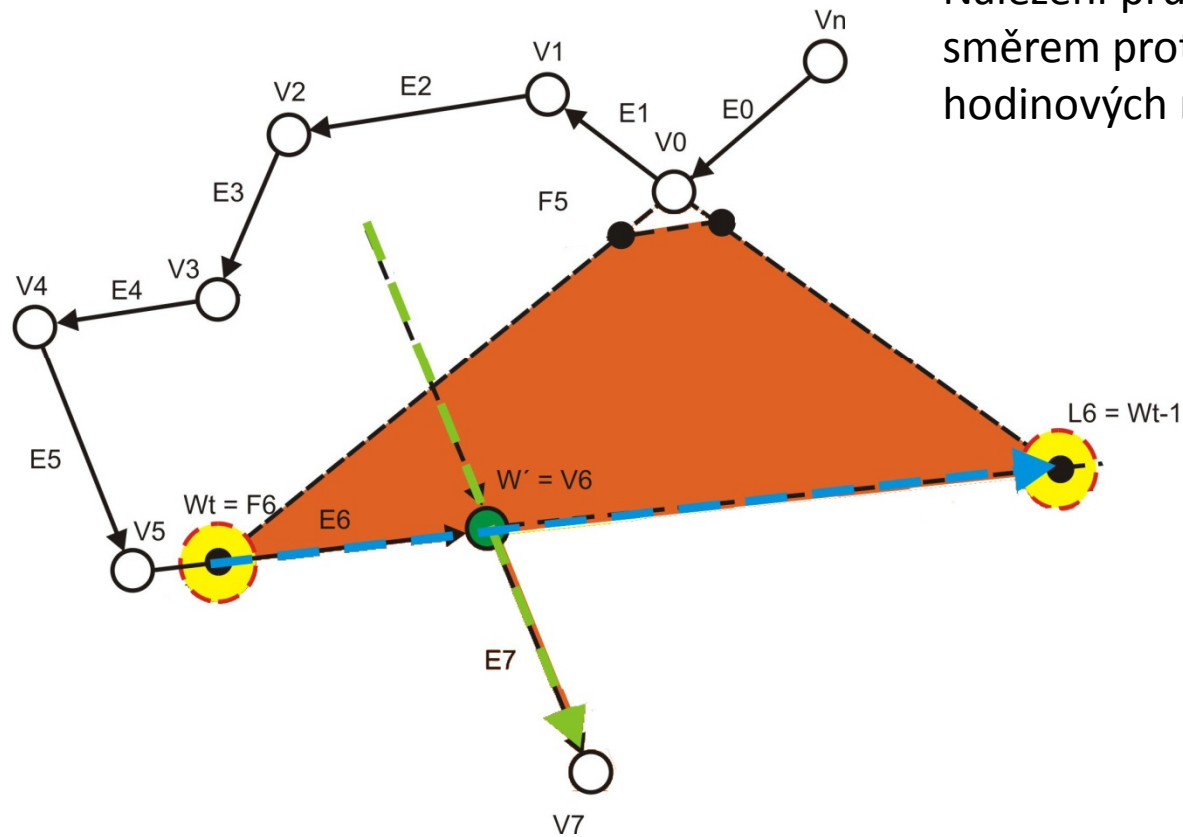
1.1

Reflexní vrchol, F6
vpravo, pravidlo 1.1



Obecný krok

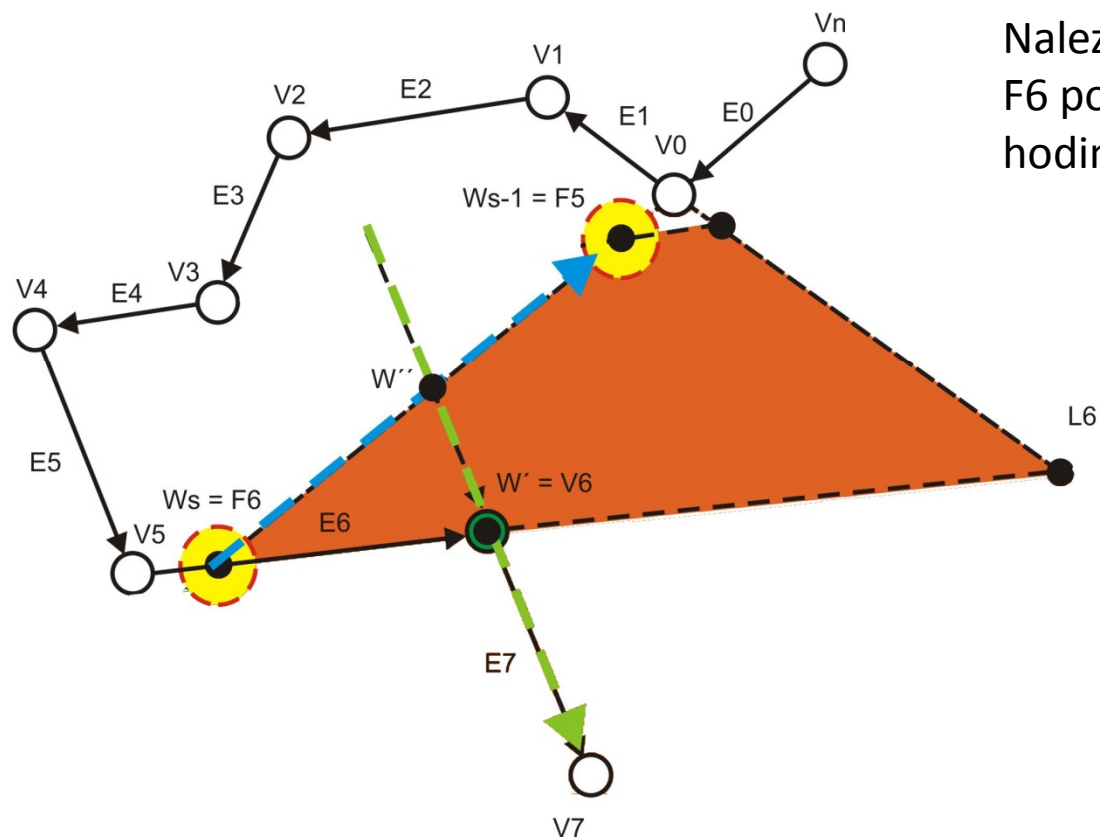
1.1



Nalezení průsečíku z F_6
směrem proti pohybu
hodinových ručiček $w' = v_6$

Obecný krok

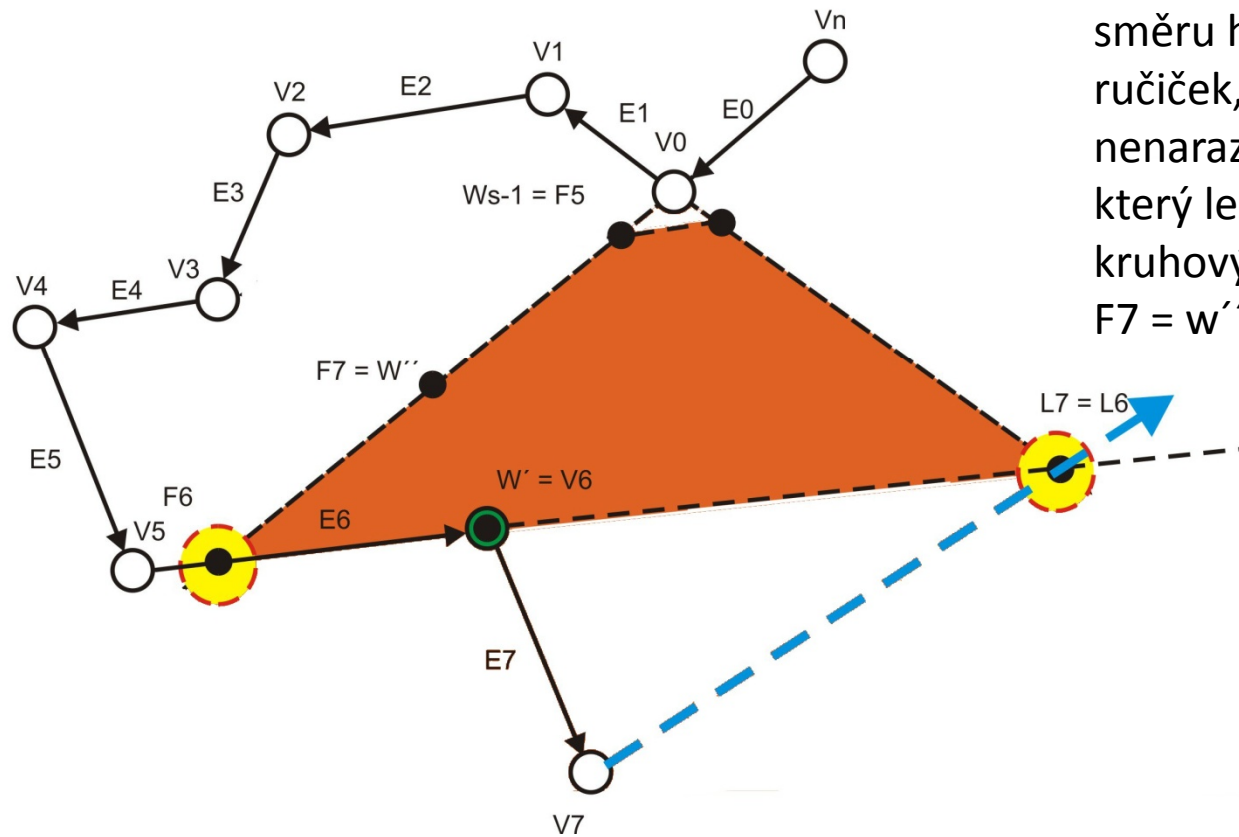
1.1



Nalezení průsečíku w'' z F_6 po směru hodinových ručiček

Obecný krok

1.1

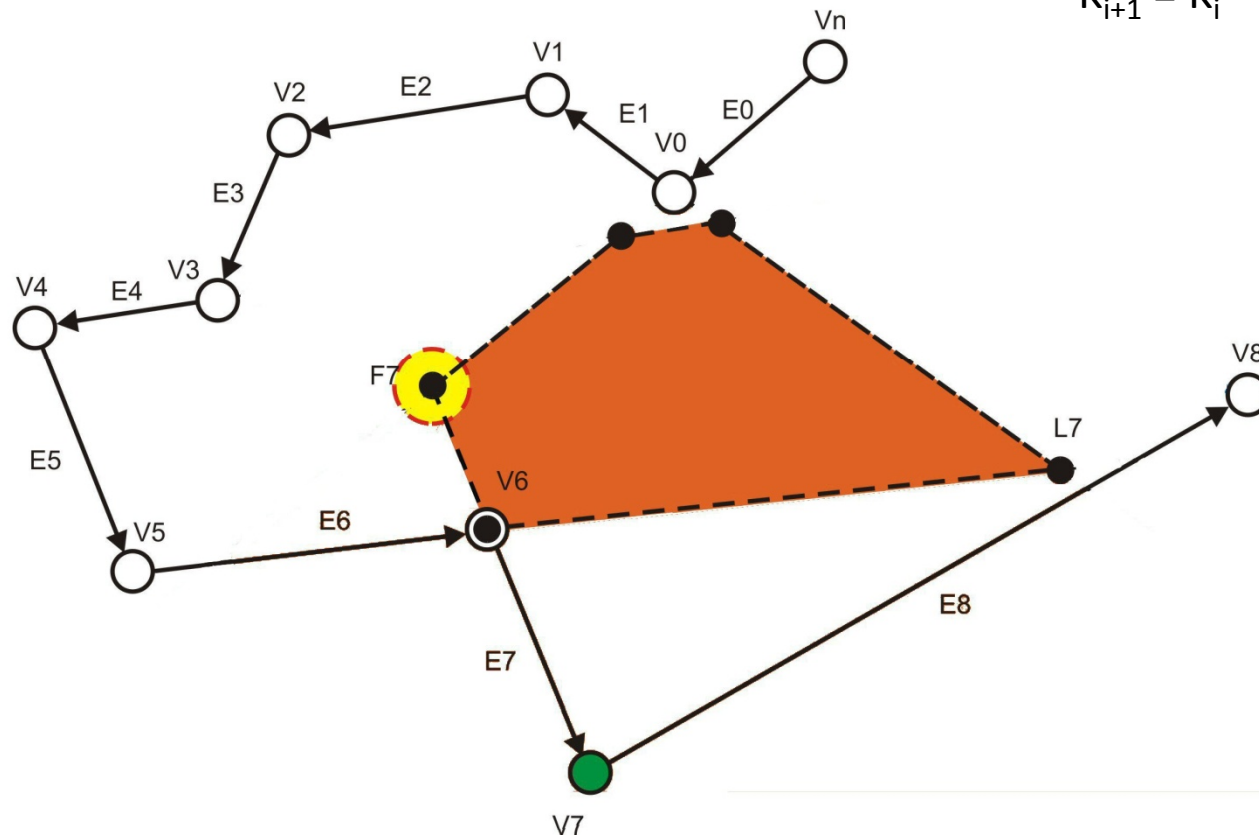


Hledání L_7 proti směru hodinových ručiček, dokud nenarazí na souseda, který leží vlevo, kruhový seznam $F_7 = w''$

Obecný krok

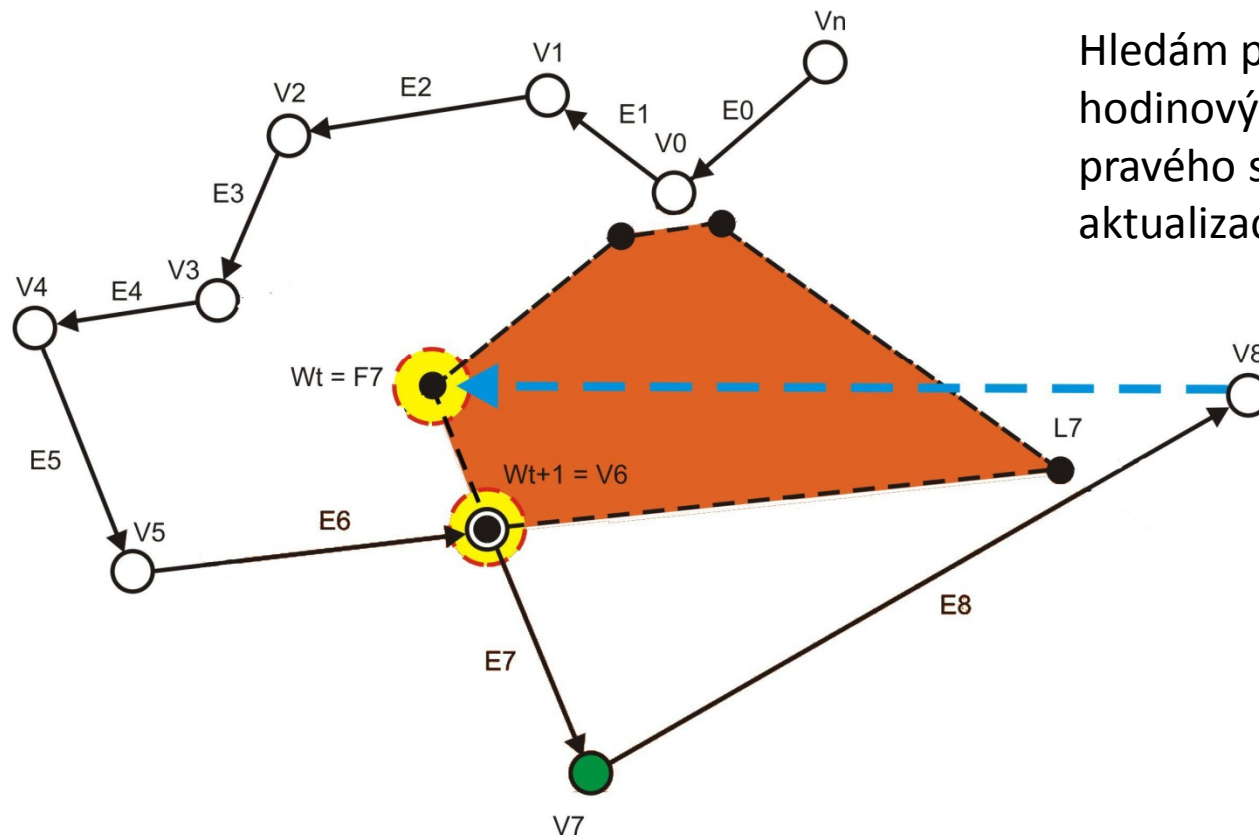
1.2

Konvexní vrchol, F7
vlevo, pravidlo 1.2,
 $K_{i+1} = K_i$



Obecný krok

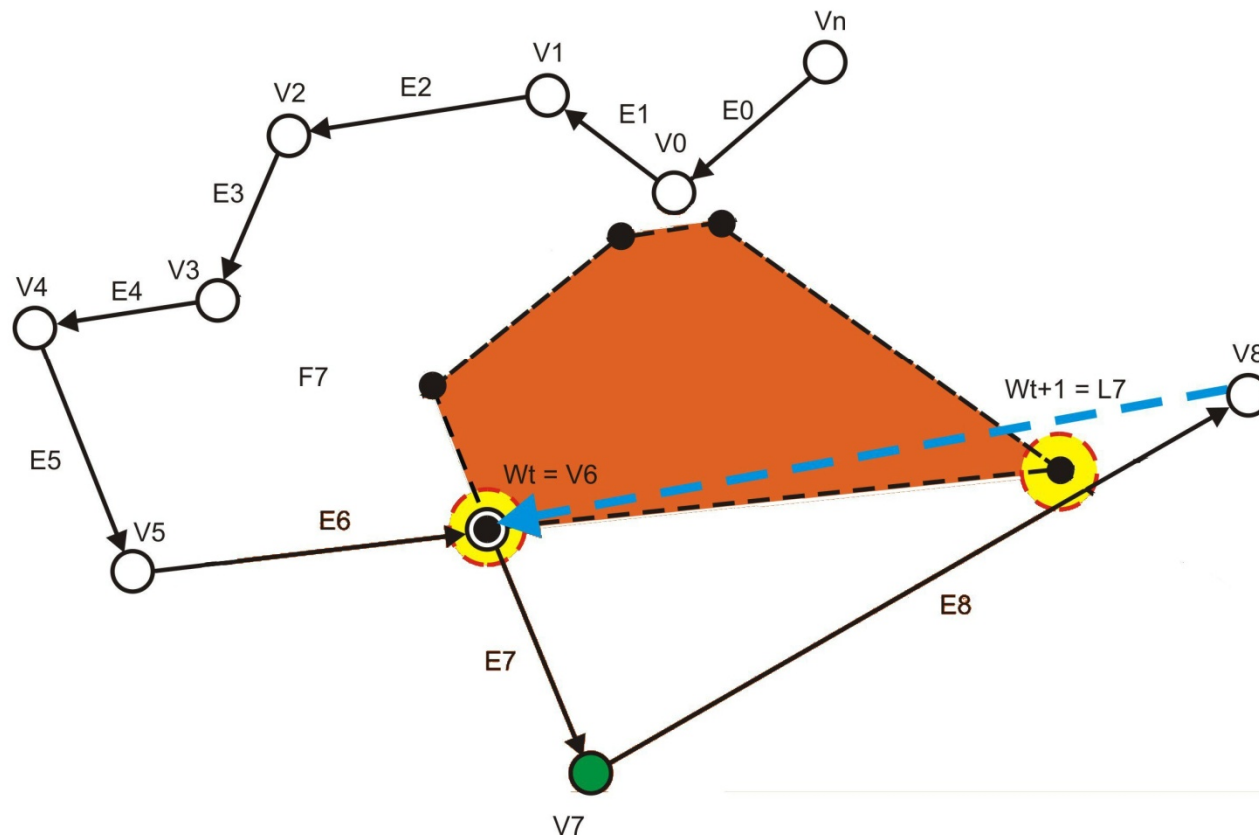
1.2



Hledám proti směru
hodinových ručiček
pravého souseda pro
aktualizaci F8

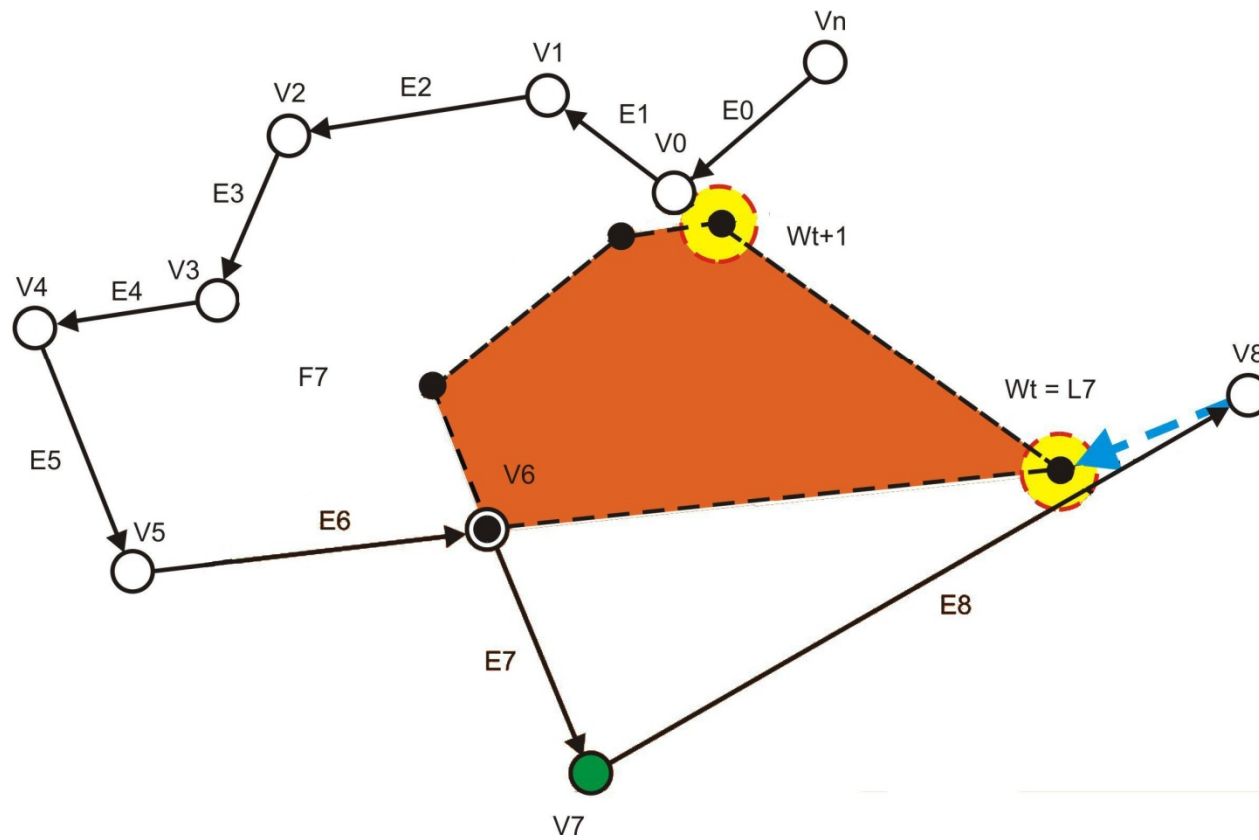
Obecný krok

1.2



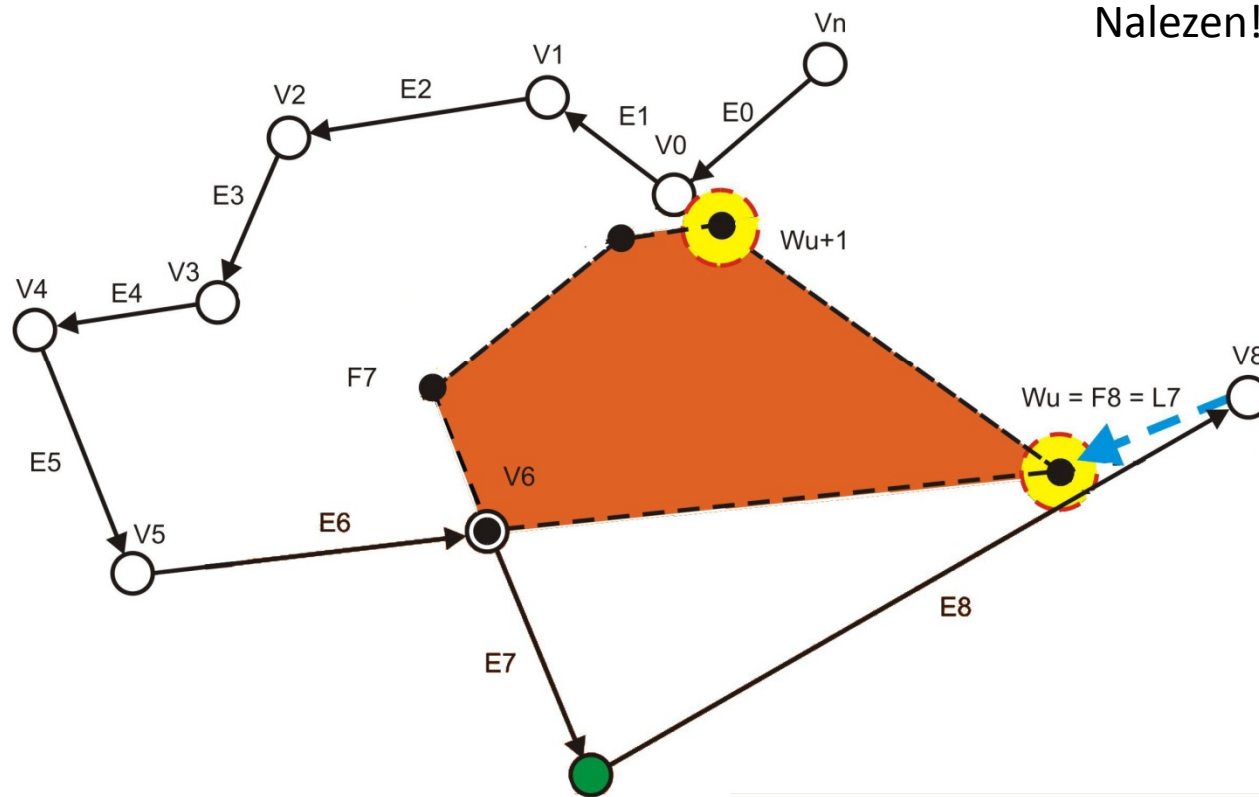
Obecný krok

1.2



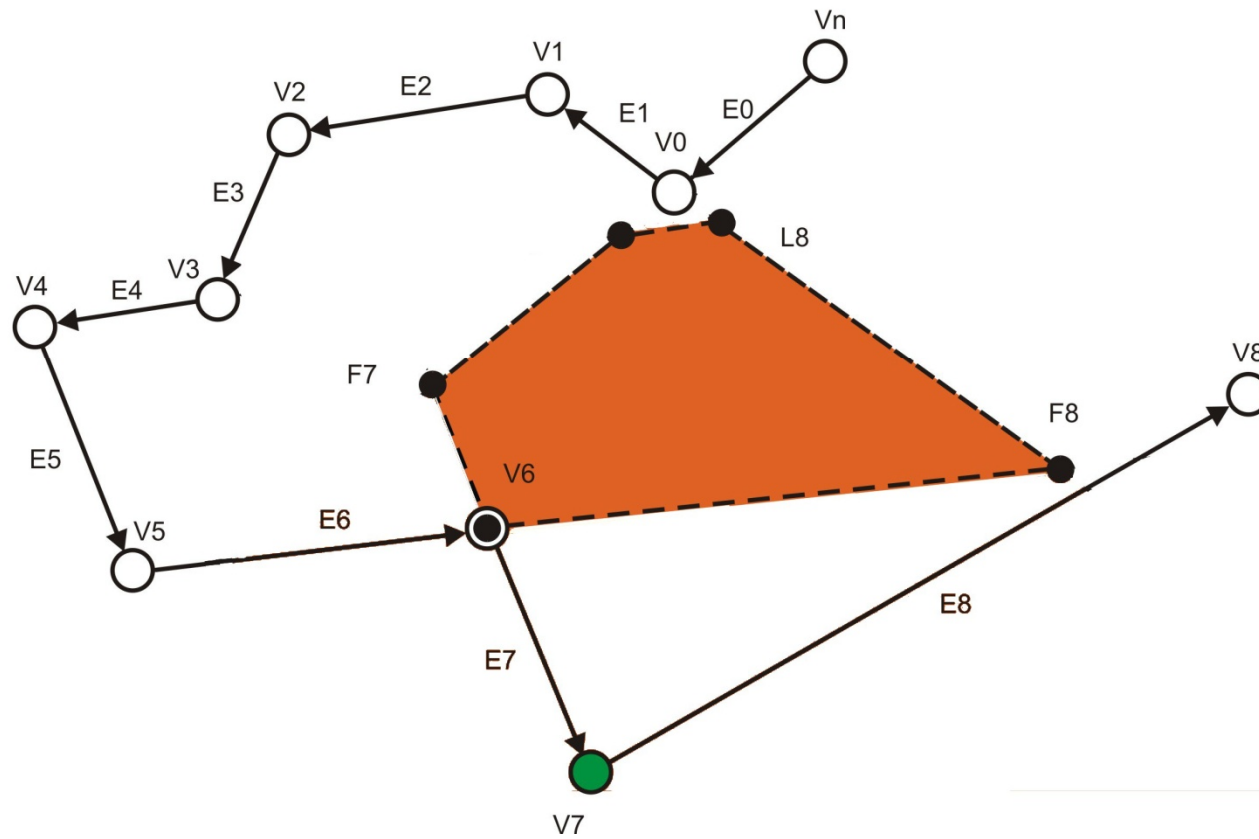
Obecný krok

1.2



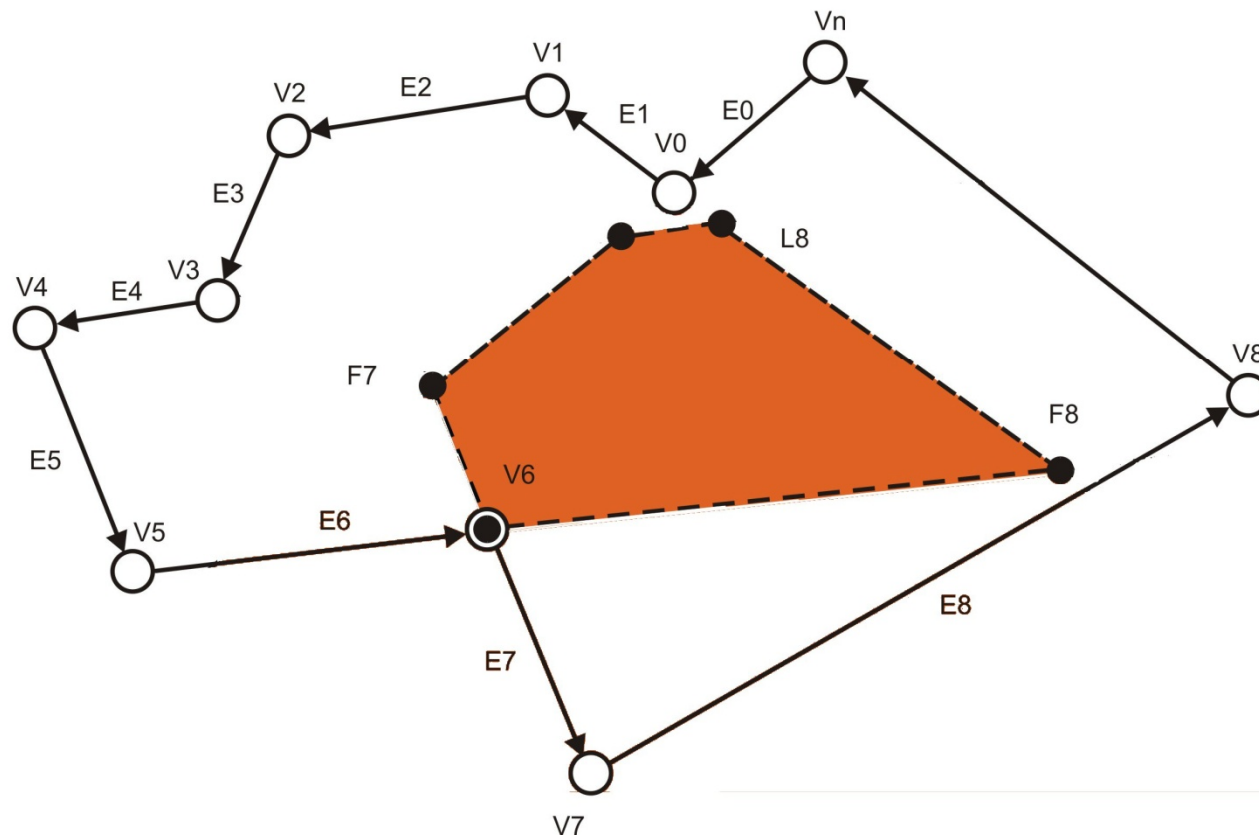
Obecný krok

1.2



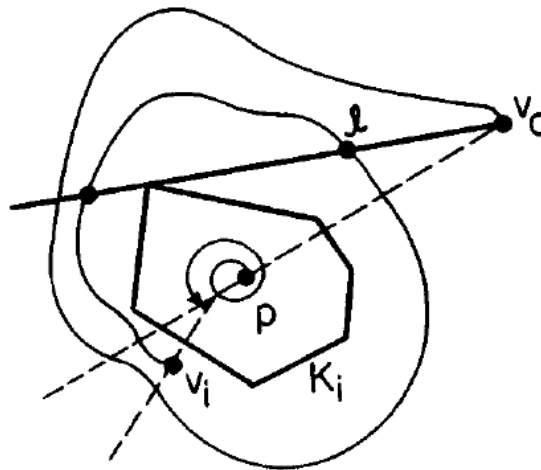
Obecný krok

1.2



Optimalizace pro zvláštní případy polygonů

- Při otočení kteréhokoliv bodu i jen částečně vytvořeného jádra o více než 540° se program ukončí a polygon je určen s $K(P) = 0$



Obrázek převzat z [1]

Otázky

Zdroje

- [1] Lee, D. T. a Preperata F. P. - *An Optimal Algorithm for Finding the Kernel of a Polygon*, *Journal of the ACM*, Volume 26, Issue 3, červenec 1979., str. 415-421
- [2] Preperata F.P.- M.I.Shamos: *Computational Geometry An Introduction*. Berlin, Springer-Verlag,1985.



**OI-OPPA. European Social Fund
Prague & EU: We invest in your future.**
