

Dokumentace k programu Flow Solver

2017

MICHAL MARŠÁLEK

Úvod

Flow Solver je program na řešení úrovně ze hry *Flow Free / Flow Bridges*. Je v něm možné si skrze grafické uživatelské rozhraní naklikat úroveň ze hry a následně si ji nechat vyřešit a řešení vykreslit zpět na obrazovku.

Popis hry

Úroveň ve hře Flow se skládá z obdélníkové (často čtvercové) mřížky. V ní se nachází několik dvojic teček, každá dvojice je identifikována svou barvou. Cílem hráče je propojit každou dvojici teček sérií lomených čar. Ty se za normálních okolností nesmí překrývat ani křížit a úrovně jsou navrženy tak, že řešení je právě jedno a pokryje celou mřížku. Tento standardní formát má několik rozšiřujících prvků:

Mosty: některá pole obsahují most. Takovým polem naopak musí procházet dvě navzájem se křížící větve

Zdi: některá pole mřížky jsou oddělena zdí, která znemožňuje mezi těmito dvěma poli přecházet.

Warpy: některá krajní pole obsahují „portály“ na opačnou stranu mřížky.

Práce s programem

V pravém horním rohu je možno si nastavit rozměry úrovně, poté kliknutím na některé z tlačítek pod výběrem velikosti vytvořit mřížku (prázdnou, se čtyřmi warpy, se čtyřmi mosty s jedním prstencem zdí, respektive se dvěma prstenci zdí). Níže je možno vybrat si nástroj pro práci s mřížkou: *Bridge* pro práci s mosty, *Wall* pro práci se zdmi, warpy se „vkládají“ odstraněním zdi na okraji, barevná kolečka pro vkládání jednotlivých barevných teček. Kliknutím do mřížky se pak vloží zvolený prvek (nebo se odstraní, pokud tam již je). Nástroj lze aplikovat na více polí najednou, bez nutnosti pouštět tlačítko myši. Mezi nástroji je možné také přepínat kolečkem myši. Když je vložena celá úroveň, je poté možné spustit řešení kliknutím na tlačítko *Solve*. Pokud výpočet trvá příliš dlouho, je možné jej kliknutím na tlačítko *Stop* zastavit (pak lze opět pokračovat tam, kde se skočilo). Poté je možno řešení zresetovat kliknutím na tlačítko *Reset*, což mimo jiné umožní provádět další úpravy na mřížce. Tlačítko *Next State* slouží ke sledování průběhu výpočtu. Oknu lze různě měnit velikost.

Konstrukce řešení

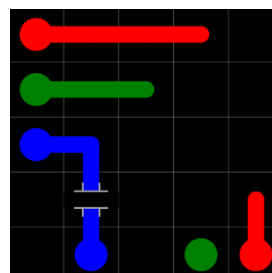
Výpočet řešení probíhá tak, že se postupně generují různé mezistavy. Barevné tečky se program pokouší propojovat současně z obou stran (tím pro každou barvu tvoří dvě „větve“, které se nakonec spojí) a to nejprve podle určitých „chytrých“ pravidel, dále se provádí prohledávání do hloubky, přičemž se snaží prohledávat nejprve ty stavy, které „mají větší šanci na úspěch“. Zároveň program detekuje stavy, které je možno zahodit a neprohledávat je dále.

Vynucené kroky

Vynucené kroky jsou dvou typů:

- 1) Jelikož se předpokládá, že řešením je plně zaplněná mřížka, pokud se konec některé větve dotýká mostu, musí již nutně přes tento most přejít.
- 2) Pokud se některé dva konce větví stejné barvy dotýkají, pak se spojí.
- 3) Pokud má některá nedokončená větev právě jeden směr, kterým může pokračovat, pak tímto směrem nutně pokračovat musí.

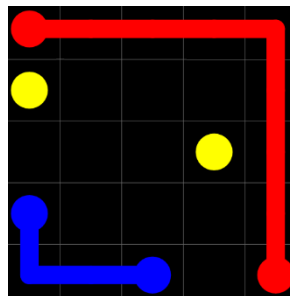
Provedení některého z těchto kroků může způsobit řetězovou reakci (která u některých zadání vede k nalezení výsledku) a proto se provádí opakovaně, dokud je to možné. Na obrázku vpravo proběhlo: krok **1)** a poté **2)** na modré barvě. Poté šestkrát krok **3)** na zelené a červené barvě.



Skoro jisté kroky

Kroky tohoto typu jsou upřednostňované kroky, které obvykle vedou k řešení, ale je třeba případně prohledat i jiná alternativní pokračování.

- 4) Pokud se obě tečky stejné barvy nacházejí na okraji mřížky, a lze je po okraji propojit, a podél tohoto propojení je všude zed', je dobrý nápad je takto spojit, jelikož jde o jednoduché propojení a nedochází k blokování cesty jiným větvím.



Napravo je provedení kroku 4) na červené a modré barvě.

Zde vidím prostor pro zlepšení algoritmu, sice zobecněním kroku 3) a to z „obkreslování“ fyzického okraje mřížky na „obkreslování“ libovolného „okraje“ který vznikne v průběhu výpočtu někde uprostřed, nalézt tak propojení takové, že na jedné straně od něj už je vše rozřešeno.

Pořadí ostatního prohledávání

- 5) Nejprve se určí pořadí, ve kterém se budou prodlužovat větve. Větve se seřadí podle toho, kolik má jejich konec možných směrů, kterým pokračovat, v případě shody podle toho, jak je vzdálen od druhého konce stejné barvy (taxicab metrika na mřížce, nepočítá se přechod přes okraj).
- 6) Následně se pro tuto větev určí pořadí směrů, kterými pokračovat. Směry se seřadí podle toho, kolik možností má větev na pokračování pokud se vydá daným směrem, v případě shody podle vzdálenosti od druhého konce stejné barvy, na kterou se dostane, pokud se vydá daným směrem.

Zahazování stavů

- A) Pokud má některá nevyřešená větev 0 směrů, kterými se může vydat.
- B) Pokud některou dvojici větví již nelze propojit (ve smyslu, že jejich konce jsou odděleny ostatními větvemi).
- C) Pokud je některá větev moc složitá, tj. pokud se dotýká sama sebe a od místa doteku nepřešla přes most, je možná ji zkrátit aby od místa doteku pokračovala přímo na poslední pole, o řešení se tím nepříjde.
- D) Vzhledem k tomu, že se předpokládá, že každé řešení zaplní každé pole, zahazují se stavy, ve kterých existuje oblast, která nesousedí s žádným koncem větve.

Ilustrace případů A), B), C), D):

