Projekt SFC 2017/2018

104 - Demonstrace fuzzy implikací (C/C++)

Autor: **Ondřej Valeš** **Datum vytvoření: 27. 10. 2017**

Login: **xvales03**

Fuzzy logika

Fuzzy logika je rozšířením klasické dvouhodnotové logiky. Na rozdíl od klasické logiky, která umožnuje odpovědět na otázku jestli je něco pravda nebo ne, fuzzy logika umožňuje odpovědět na otázku jak moc je něco pravdivé. Pracuje tedy s celou škálou možných hodnot, která popisuje míru pravdivosti.

Analogicky k běžným množinám, kde je možné rozhodnout, zda daný prvek do množiny patří nebo ne, jsou zavedeny fuzzy množiny, jejichž prvky do množiny patří s nějakou mírou. To, do jaké míry prvek do fuzzy množiny patří, popisuje funkce příslušnosti:

$$m_A(x) \in \langle 0, 1 \rangle$$

která říká, že prvek *x* patří do množiny *A* s mírou $m_A(x)$. Právě nad příslušnostmi prvků do fuzzy množin je potom možné provádět logické operace.

Fuzzy logika je tedy nástroj, který umožňuje pracovat s nepřesnými vstupními údaji (reprezentovaných fuzzy množinami) a vyvozovat z nich závěry, což by s běžnou logikou bylo u nepřesných vstupních údajů přinejmenším komplikované.

Klasická implikace a fuzzy implikace

Klasická implikace je definována následující tabulkou:

x	y	$x \to y$
0	0	1
0	1	1
1	0	0
1	1	1

Ve fuzzy logice existuje pro výpočet implikace (stejně jako pro ostatní logické spojky) mnoho různých přístupů. Ty základní vychází z přepisu implikace pomocí ostatních logických spojek, například:

$$x \to y \equiv \neg x \vee y \,.$$

Pokud jako negaci vezmeme Sugeno (s parametrem *s = 0*):

$$\neg m(x) = 1 - m(x)$$

a jako disjunkci S-normu maxima:

$$m(x) \vee m(y) = \max(m(x), m(y))$$

a dosadíme do přepisu implikace dostaneme:

$$m(x \to y) = \max(1 - m(x), m(y))$$

což je jeden z možných způsobů, jak ve fuzzy logice počítat implikace (konkrétně se jedná implikaci Kleene-Dienes). Další příklady fuzzy implikací dostaneme použitím jiných S-norem, T-norem a fuzzy negací nebo využitím jiného přepisu klasické implikace. Mezi další možnosti získání předpisu fuzzy implikace patří residuum nad T-normami, případně vymyšlení předpisu nezávisle na klasické implikaci (což může vést k předpisu, který z matematického hlediska není implikací.

Ekvivalent implikace ve fuzzy logice by měl pro hraniční hodnoty dávat stejné výsledky jako klasická implikace (i když se ukazuje, že existují i příklady fuzzy implikací, které toto pravidlo porušují a přesto jsou hojně využívány).

Fuzzy implikace využité při demonstraci

Pro demonstraci byly vybrány předpisy implikací Gödel, Kleene-Dienes, Kleene-Dienes-Łukasievicz, Standart Strict, Larsen a Mamdani. A to protože reprezentují základní přístupy k realizaci fuzzy implikace vycházející ze základních T-norem a S-norem.

Konkrétně výskyt násobení v Kleene-Dienes a Larsen (vycházející ze součinové T-normy), maxima a minima v Kleene-Dienes-Łukasievicz a Mamdani (vycházející z minimové T-normy a maximové S-normy) a porovnání v Gödel a Standart Strict (vycházející z drastického součinu).

Dalším cílem je prezentovat rozdíly mezi implikacemi využívající stejný princip a zdůraznit, že některé verze fuzzy implikací nejsou z matematického hlediska validní implikace.

Gödel

Předpis: $m(x \ \to y) = \begin{cases} 1 & m(x) \ < m(y) \\ m(y) & otherwise \end{cases}.$

Splňuje podmínky klasické implikace: ano.

Kleene-Dienes

Předpis: $m(x \ \to y) = \max(1 - m(x), m(y)).$

Splňuje podmínky klasické implikace: ano.

Kleene-Dienes-Łukasievicz

Předpis: $m(x \ \to y) = 1 - \mathrm{m(x)} + m(x) \times m(y).$

Splňuje podmínky klasické implikace: ano.

Standart Strict

Předpis: $m(x \ \to y) = \begin{cases} 1 & m(x) \ < m(y) \\ 0 & otherwise \end{cases}.$

Splňuje podmínky klasické implikace: ano.

Larsen

Předpis: $m(x \ \to y) = m(x) \times m(y).$

Splňuje podmínky klasické implikace: ne.

Mamdani

Předpis: $m(x \ \to y) = \min(m(x), m(y)).$

Splňuje podmínky klasické implikace: ne.

Demonstrační program

Demonstrační program je napsaný v jazyce C++ a pro zobrazování grafiky používá knihovnu Qt s rozšířením QCustomPlot pro vykreslování barevných 2D map. Jeho cílem je vizualizovat závislost výsledku fuzzy implikace na vstupních hodnotách a rozdíly mezi jednotlivými předpisy implikace.

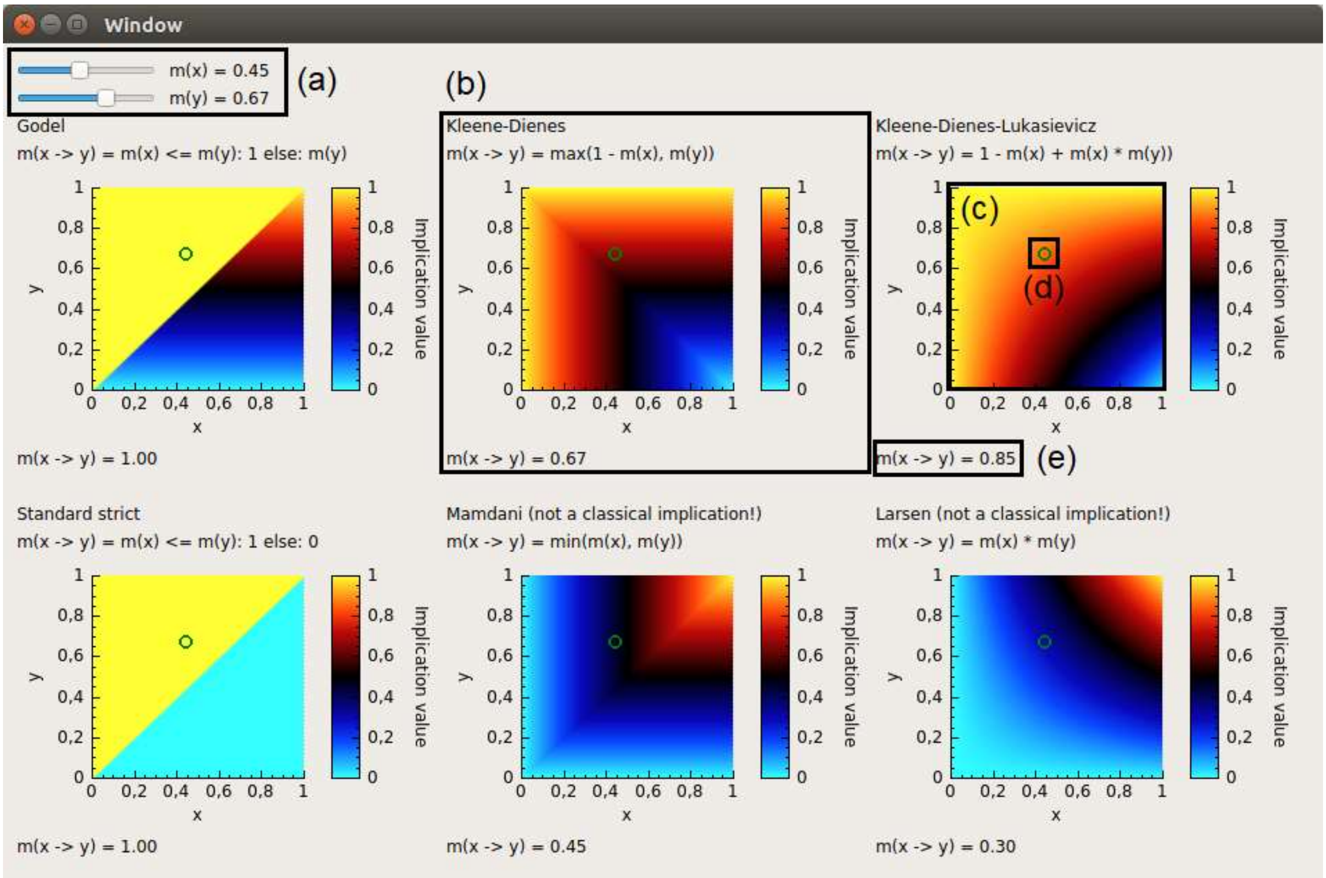
Struktura a přeložení

V odevzdaném souboru se nachází podadresář `resources/` obsahující zdrojové kódy rozšíření QCustomPlot (jedná se o volně dostupné rozšíření Qt), které jsou přiloženy pro zjednodušení překladu na serveru merlin. Autor této zprávy tímto explicitně uvádí, že si nepřivlastňuje autorství kódu v tomto podadresáři.

Přeložení programu se provede příkazem `make`, který vytvoří generovaný makefile pro Qt a spustí samotný překlad. Spuštění programu je možné pomocí `make run` nebo spuštěním binárního souboru `./xvales03` (pokud je využívám předpřipravený přeložený program pro server merlin, je nutno nastavit oprávnění pro spouštění). Program je spouštěn bez parametrů

Ovládání

Program se skládá z jediné obrazovky obsahující dva ovládací prvky (a) umožňující nastavovat hodnoty fuzzy proměnných x a y. Dále grafy (b) pro každou verzi implikace. Každý graf obsahuje barevnou mapu zobrazující výsledek implikace pro celý rozsah hodnot x a y (c) a vyznačenou současnou pozici odpovídající aktuálnímu nastavení proměnných (d). Pod každým grafem je dále zobrazena číselná hodnota výsledku implikace pro aktuální nastavení proměnných (e).



Závěr

V rámci tohoto projektu byl vytvořen program pro vizualizaci několika verzí fuzzy implikace, který umožňuje snadné porovnání jednotlivých fuzzy implikací. Déle umožnuje interaktivně zkoumat závislost mezi nastavením vstupních proměnných a výsledné hodnoty implikace.