

Assignment 2 - DFA

Iacopo Ruzzier

Ultimo aggiornamento: 10 aprile 2025

per ciascuno dei 3 problemi di analisi, definire dominio direzione fz di trasf meet op e condizioni di boundary e iniziali

inoltre per il cfg di esempio fornito bisogna compilare una tabella per descrivere le iterazioni che servono a raggiungere convergenza

per validare la formulazione, fate domande del tipo es. liveness analysis: questa variabile è viva in questo punto? eccetera

0.1 very busy expressions

very busy se indipendentemente dal percorso preso, l'espr. viene usata prima che uno dei suoi operandi venga definito (viene usata) ci interessa conoscere le espressioni disponibili .. ?

utile perche permette code hoisting (tecnica usata in loop invariant code motion, vedi inizio, recupera)

sposto prima del loop una certa espressione: posso farlo solo se sono sicuro che qualsiasi espr che usa il valore dell'espr da spostare ... rec

capire se posso evitare di ripetere un'op in maniera ridondante eseguendola una volta sola in un punto dove il control flow è comune a tutti i path?

in ogni caso ci ritorneremo, per ora ci limitiamo a ragionare in termini di dfa relativamente al sottoproblema delle vbexpr

0.2 dominator analysis

analisi dei dominatori - inizieremo a parlarne quando inizieremo a parlare dei loop, fondamentale per riconoscere l'esistenza di un loop

riconoscere l'esistenza di un loop significa in primis riconoscere la presenza di un ciclo

un blocco domina un altro blocco se il primo appare sempre in ogni percorso prima del secondo - in questo caso x domina y

questa era la condizione di dominanza; unendola alla condizione di "direzione" dell'arco, posso rilevare la presenza di cicli (ne parleremo in dettaglio)

come impostare il framework di dfa per capire quali blocchi sono dominanti rispetto agli altri?

da esempio: i blocchi che dominano f sono a c f, non d ed e perche sono opzioni "mutualm escl" ovvero non necessariamente devo passare attraverso uno dei due; inoltre f perche un blococ è sempre dominatore di se stesso

0.3 constant propagation

per poter trovare i punti del programma in cui le var hanno valore costante bisogna usare un dominio speciale: devo associare sia nome variabile che valore della costante - se una var viene nel frattempo riassegnata l'informazione di constant pr cambia

dominio fatto da coppie del tipo var,val-costante (x,c)

dunque garantisco che per coppie x,c ad ogni uso di x sia associata c (il suo valore)

questa analisi riesce a determinare il valore costante di espr binarie dove 1 o entrambi gli operandi sono costanti note - nel determinare le equazioni possiamo tenere conto di questa info, e quindi eventualmente fare constant folding (non obbligatorio?)