离散数学

Discrete Mathematics

for Computer Science

计算机学院计科系

薛思清 xuesiqing@cug.edu.cn

第6讲 关系 Relation（2）

Good order is the foundation of all things.

—Edmund Burke (1729–1797)

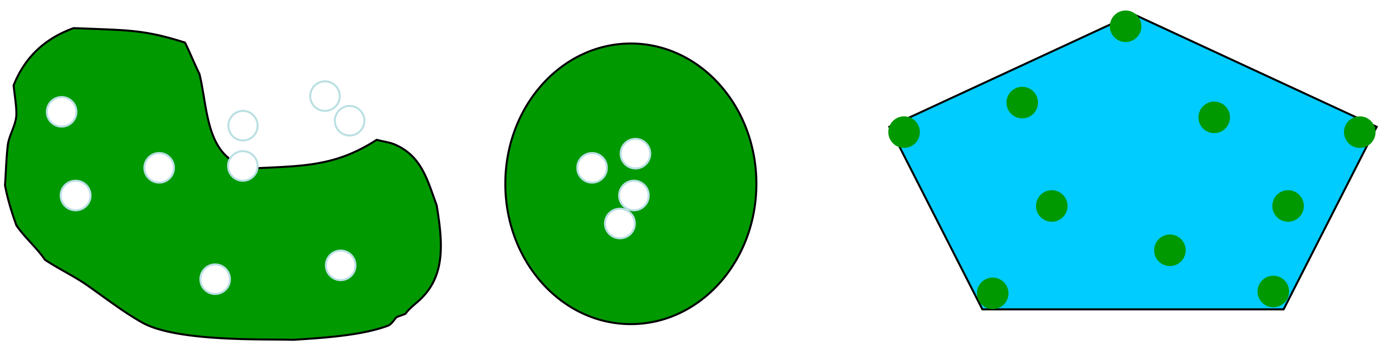
Outline

- 关系闭包

闭包(closure):

包含一些给定对象, 具有指定性质的最小集合“最小”：任何包含同样对象, 具有同样性质的集合, 都包含这个闭包集合。

示例



Computer science

- Closure (computer programming), an abstraction binding a function to its scope
- Clojure, a dialect of the Lisp programming language
- Kleene closure
- Syntactic closure
- Google Closure Tools, a set of JavaScript tools created by Google
- Relational database model: Set-theoretic formulation and Armstrong's axioms for its use in database theory

In graph theory

In logic and computational complexity

In database query languages

Algorithms

自反闭包: 包含给定关系R的最小自反关系, 称为R的自反闭包:

(1) $R \subseteq R'$;

(2) R '是自反的;

(3) $\forall S(\ (R \subseteq S \land S$自反$) \to R' \subseteq S\ )$.

R'记作: r(R)

对称闭包 s(R)

传递闭包 t(R)

设$R\subseteq A\times A$且$A\neq\varnothing$,则

(1) R自反 $\Leftrightarrow$ $r(R) = R$;

(2) R对称 $\Leftrightarrow$ $s(R) = R$;

(3) R传递 $\Leftrightarrow$ $t(R) = R$.

(1) $r(R)$是R的自反闭包,

$R\subseteq R \wedge R$自反 $\Rightarrow$ $r(R)\subseteq R$, 且 $R\subseteq r(R)$,

所以，$r(R) = R$.

如何求闭包?

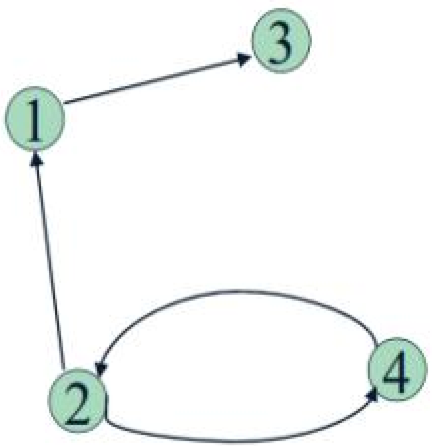
(1) $r(R) = R \cup ?$

(2) $s(R) = R \cup ?$

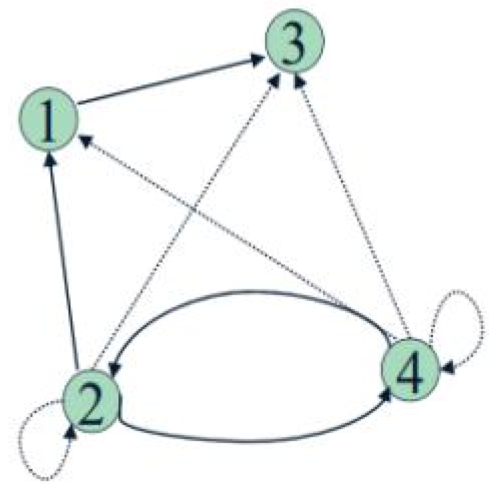
(3) $t(R) = R \cup ?$

- Computes the transitive closure of a relation
- (Alternatively: all paths in a directed graph)

- Example of transitive closure:



$$\begin{matrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{matrix}$$



$$\begin{matrix} 0 & 0 & 1 & 0 \\ 1 & \mathbf{1} & \mathbf{1} & 1 \\ 0 & 0 & 0 & 0 \\ \mathbf{1} & 1 & \mathbf{1} & \mathbf{1} \end{matrix}$$

设 $R \subseteq A \times A$ 且 $A \neq \varnothing$, 则

(1) $r(R) = R \cup I_A$;

(2) $s(R) = R \cup R^{-1}$;

(3) $t(R) = R \cup R^2 \cup R^3 \cup \ldots$ (记为R^+, R的自反传递闭包记为R^*)

>>>

证明: (1)

$R \subseteq R \cup I_A \wedge R \cup I_A$自反 $\Rightarrow r(R) \subseteq R \cup I_A$;

$R \subseteq r(R) \wedge r(R)$自反 $\Rightarrow R \subseteq r(R) \wedge I_A \subseteq r(R) \Rightarrow R \cup I_A \subseteq r(R)$

于是, $r(R) = R \cup I_A$

(3) $t(R) = R \cup R^2 \cup R^3 \cup \ldots$

1) $R \subseteq R \cup R^2 \cup R^3 \cup \ldots$;

2) $(R \cup R^2 \cup R^3 \cup \ldots)^2 = R^2 \cup R^3 \cup \ldots \subseteq R \cup R^2 \cup R^3 \cup \ldots$

$\Leftrightarrow R \cup R^2 \cup R^3 \cup \ldots$传递

$\boxed{R\text{传递} \Leftrightarrow R^2 \subseteq R}$

3) 若有R', $R \subseteq R' \wedge R'$ 传递

$\Rightarrow R \subseteq R' \wedge R^2 \subseteq R' \wedge R^3 \subseteq R' \wedge \ldots$

$\Rightarrow R \cup R^2 \cup R^3 \cup \ldots \subseteq R'$

$\therefore t(R) = R \cup R^2 \cup R^3 \cup \ldots$.

$\boxed{R \subseteq G \wedge G\text{传递} \Rightarrow R^n \subseteq G}$

$$|A| = n,\ t(R) = \bigcup_{i=1}^{n} R^i$$

示例 $A=\{a, b, c\}$, $R=\{<a, b>,<b, c>,<c, a>\}$,求r(R),S(R),t(R).

解：$r(R)=R\cup I_A=\{<a, b>, <b, c>, <c, a>, <a, a>, <b, b>, <c, c>\}$

$s(R)= R\cup R^{-1} =\{<a, b>,<b, a>,<b, c>,<c, b>,<c, a>,<a, c>\}$

为求t(R)先求R^2，R^3，R^4

即$R^2=\{<a, c>,<b, a>,<c, b>\}$

$R^3=\{<a, a>,<b, b>,<c, c>\}$

$R^4=\{<a, b>,<b, c>,<c, a>\}$

可见$R=R^4=R^{3n+1}$，$R^2= R^5= R^{3n+2}$，$R^3= R^6= R^{3n+3}$

故$t(R) = R\cup R^2\cup R^3$

$= \{<a, a>, <b, b>, <c, c>, <a, b>, <b, c>, <c, a>, <a, c>, <b, a>, <c, b>\}$

Warshall's Algorithm (pseudocode and analysis)

ALGORITHM *Warshall*($A[1..n, 1..n]$)

//Implements Warshall's algorithm for computing the transitive closure
//Input: The adjacency matrix A of a digraph with n vertices
//Output: The transitive closure of the digraph
$R^{(0)} \leftarrow A$
for $k \leftarrow 1$ **to** n **do**
  for $i \leftarrow 1$ **to** n **do**
    for $j \leftarrow 1$ **to** n **do**
      $R^{(k)}[i, j] \leftarrow R^{(k-1)}[i, j]$ **or** $(R^{(k-1)}[i, k]$ **and** $R^{(k-1)}[k, j])$
return $R^{(n)}$

Time efficiency: $\Theta(n^3)$

Space efficiency: Matrices can be written over their predecessors