Cursul 8

Tipuri de grafuri. Structuri de date pentru grafuri. Conectivitate: Algoritmul naiv și algoritmul lui Warshall

noiembrie 2016

Reamintim că:

Graf = structură matematică $G = (V, E)$ unde

- V : mulțime de noduri (sau vârfuri)
- E : mulțime de muchii incidente la două noduri, sau la 1 nod

Tipuri de grafuri, în funcție de tipul muchiilor $e \in E$:

- **Neorientate**: Fiecare muchie are 2 capete

  Reprezenare grafică: $a \overset{e}{\text{——}} b$ sau $\overset{e}{\frown} \atop a$ (buclă)

- **Orientate** (sau digrafuri): Fiecare muchie $e \in E$ are o sursă (sau start) și o destinație (sau end)

  Reprezentare grafică: $a \xrightarrow{e} b$ sau $\overset{e}{\circlearrowright} \atop a$ (buclă)

  Muchiile orientate se numesc si arce (singular: arc).

Graf simplu: graf neorientat sau orientat, care are cel mult un arc între orice două noduri și nu are nici o buclă.

$$G = (V, E)$$

$e_1, e_2 \in E$ sunt muchii paralele dacă sunt incidente la aceleași noduri și

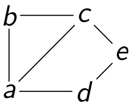
- Dacă G este orientat, atunci $start(e_1) = start(e_2)$ și $end(e_1) = end(e_2)$

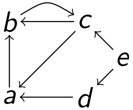
- **Multigraf** orientat sau neorientat: nu are bucle, iar dacă graful este

  neorientat: poate avea muchii paralele
  orientat: pooate avea arce paralele

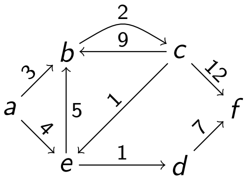
- **Pseudograf**: graf neorientat care poate avea muchii paralele și bucle.
- **Graf ponderat**: fiecare muchie $e \in E$ are o pondere (sau greutate) $w(E)$; de obicei $w(e) \in \mathbb{R}$.

- Grafuri simple: se trasează linii sau săgeți între nodurile conectate

Graf simplu **neorientat**: 

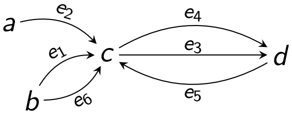
Graf simplu **orientat**: 

- Grafuri simple ponderate: se indică ponderea în dreptul conexiunilor

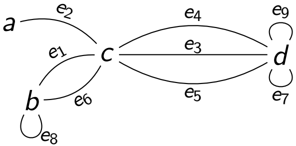


Multigrafuri sau pseudografuri: dacă au muchii paralele pe care vrem să le distingem, se etichetăm muchiile:

multigraf:



pseudograf:

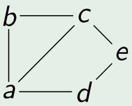


Reprezentări concrete ale grafurilor simple

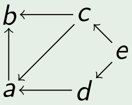
1. Listă de noduri + listă de muchii
2. Liste de adiacență
3. Matrice de adiacență
4. Matrice de incidență
5. Matrice de ponderi

Grafuri simple
Reprezentarea cu listă de noduri + listă de muchii

Exemplu



Listă de noduri $V = [a, b, c, d, e]$
Listă de muchii $E = [\{a, b\}, \{a, c\}, \{a, d\}, \{b, c\}, \{c, e\}, \{d, e\}]$
Observații: $\{a, b\} = \{b, a\}$, $\{a, c\} = \{c, a\}$, etc.
muchie $\leftrightarrow$ mulțimea de noduri adiacente la muchie



Listă de noduri $V = [a, b, c, d, e]$
Listă de arce $E = [(a, b), (c, a), (c, b), (d, a), (e, c), (e, d)]$
Observații: $(a, b) \neq (b, a)$, $(a, c) \neq (c, a)$, etc.
muchie $\leftrightarrow$ pereche (start,end)

Remarcă

Dacă nu există noduri izolate (cu 0 vecini), nu este necesar să fie reținută lista de noduri V:

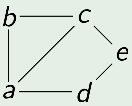
- V se poate calcula din E

Grafuri simple
Reprezentarea cu liste de adiacență

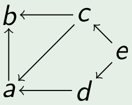
Pentru fiecare nod $u \in V$ se reține lista de noduri adiacente la u

- Dacă G este **neorientat**, v este adiacent la u dacă există o muchie cu capetele u și v.
    - În grafuri neorientate, relația de adiacență este simetrică.
- Dacă G este **orientat**, v este adiacent la u dacă există un arc $e \in E$ de la u la v, adică $start(e) = u$ și $end(e) = v$.

Exemplu



$a \mapsto [b, c, d]$ $\quad$ $d \mapsto [a, e]$
$b \mapsto [a, c]$ $\quad$ $e \mapsto [c, d]$
$c \mapsto [a, b, e]$



$a \mapsto [b]$ $\quad$ $d \mapsto [a]$
$b \mapsto []$ $\quad$ $e \mapsto [c, d]$
$c \mapsto [a, b]$

Dacă G are n noduri, $A_G = (m_{ij})$ are dimensiunea $n \times n$ și

$m_{ij} :=$ numărul de muchii de la al i-lea nod la al j-lea nod.

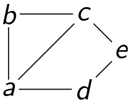
Observații

1. Înainte de a construi M_G din G, trebuie fixată o enumerare a tuturor nodurilor: $[v_1, v_2, \ldots, v_n]$
2. Dacă G este neorientat, A_G este matrice simetrică
3. Dacă G este graf simplu, A_G conține doar 0 și 1

Grafuri neorientate
Matricea de adiacență A_G a unui graf neorientat G

Matricea de adiacență a grafului neorientat

G :



pentru enumerarea de noduri $[a, b, c, d, e]$ este

$$A_G = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Observație: A_G este matrice simetrică.

Grafuri
Digraful corespunzător unei matrici simetrice de adiacență

Dacă A este o matrice simetrică $n \times n$ cu $a_{ij} \in \mathbb{N}$ pentru toți i, j, un digraf G care are matricea de adiacență A se construiește astfel:

1. Se desenează n puncte $v_1, \ldots, v_n$ în plan
2. Pentru orice $i, j \in \{1, \ldots, n\}$, se trasează a_{ij} muchii distincte între v_i și v_j

Exemplu

$$A = \begin{pmatrix} 0 & 1 & 0 & 2 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 2 & 0 & 1 & 0 \end{pmatrix} \Rightarrow G :
\begin{tikzcd}
v_2 \arrow[r, dash] \arrow[d, dash] & v_3 \arrow[d, dash] \\
v_1 \arrow[r, dash, bend left] \arrow[r, dash, bend right] & v_4
\end{tikzcd}$$

Matricea de adiacență a grafului orientat

$$G : \begin{tikzcd} b & c \arrow[l] \arrow[ddl] & \\ & & e \arrow[ul] \arrow[dl] \\ a \arrow[uu] & d \arrow[l] & \end{tikzcd}$$

pentru enumerarea de noduri $[a, b, c, d, e]$ este

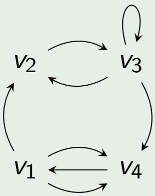
$$A_G = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Dacă A este o matrice $n \times n$ cu $a_{ij} \in \mathbb{N}$ pentru toți i, j, un digraf G care are matricea de adiacență A se construiește astfel:

1. Se desenează n puncte $v_1, \ldots, v_n$ în plan
2. Pentru orice $i, j \in \{1, \ldots, n\}$, se trasează a_{ij} arce distincte de la v_i la v_j

Exemplu

$$A = \begin{pmatrix} 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix} \Rightarrow G :$$



Presupunem date două liste (sau enumerări):

- $V = [v_1, \ldots, v_n]$ a nodurilor lui G
- $L = [e_1, \ldots, e_p]$ a etichetelor de muchii din G

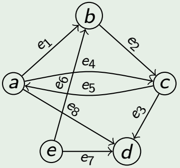
Matricea de incidență $M_G = (m_{ij})$ are dimensiunea $n \times p$ și

$$m_{ij} = \begin{cases} -1 & \text{dacă start}(e_j) = v_i \\ 1 & \text{dacă end}(e_j) = v_i \\ 0 & \text{în toate celelalte cazuri.} \end{cases}$$

Exemplu

Dacă $V = [a, b, c, d, e]$, $L = [e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8]$ și

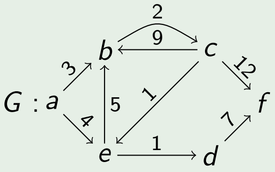
$$M_G = \begin{pmatrix} -1 & 0 & 0 & -1 & 1 & 0 & 0 & -1 \\ 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 \end{pmatrix} \Rightarrow$$



Matricea de ponderi $W_G = (w_{ij})$ a unui graf simplu ponderat G cu n noduri $[v_1, \ldots, v_n]$ are dimensiunea $n \times n$ și

- $w_{ii} = 0$ pentru toți $1 \leq i \leq n$.
- $w_{ij} = w(\{v_i, v_j\})$ pentru orice muchie $\{v_i, v_j\} \in E$, dacă G este neorientat.
- $w_{ij} = w((v_i, v_j))$ pentru orice arc $(v_i, v_j) \in E$, dacă G este orientat.
- $w_{ij} = \infty$ în toate celelalte cazuri.

Exemplu (Digraf ponderat cu enumerare de noduri $[a, b, c, d, e, f]$)



$$\Rightarrow W_G = \begin{pmatrix} 0 & 3 & \infty & \infty & 4 & \infty \\ \infty & 0 & 2 & \infty & \infty & \infty \\ \infty & 9 & 0 & \infty & 1 & 12 \\ \infty & \infty & \infty & 0 & \infty & 7 \\ \infty & 5 & \infty & 1 & 0 & \infty \\ \infty & \infty & \infty & \infty & \infty & 0 \end{pmatrix}$$

- Reprezentarea cu listă de muchii
  - Adecvată pentru reprezentarea grafurilor simple fără noduri izolate, cu $|E| \ll |V|$
  - Complexitate spațială (memorie ocupată): $O(|E|)$
- Reprezentarea cu liste de adiacență
  - Permite enumerarea rapidă a vecinilor unui nod
  - Complexitate spațială (memorie ocupată): $O(|V| + |E|)$
- Reprezentarea cu matrice de adiacență $A_G = (a_{ij})$ sau cu matrice de ponderi $W_G = (w_{ij})$
  - Test rapid de conectivitate directă între 2 noduri: $O(1)$
    - $\nexists(v_i, v_j) \in E$ dacă $a_{ij} = 0$ sau dacă $w_{ij} = \infty$
  - Complexitate spațială (memorie ocupată): $O(|V|^2)$
    - reprezentare neadecvată când $|E| \ll |V|^2$
- Reprezentare cu matrice de incidență M_G
  - Complexitate temporală: $O(|V| \cdot |E|)$

Presupunem că G este (di)graf simplu cu n noduri și matricea de adiacență $A_G = (a_{ij})$

- $a_{ij} = 0$ (sau `false`) dacă $\nexists$ arc $v_i \to v_j$
- $a_{ij} = 1$ (sau `true`) dacă $\exists$ arc $v_i \to v_j$

Definim:

- I_n : matricea identitate $n \times n$
- Operațiile booleene $\odot$ (conjuncție) și $\oplus$ (disjuncție):

$\odot$	0	1
0	0	0
1	0	1

$\oplus$	0	1
0	0	1
1	1	1

Observații:
$a \odot b = \min(a, b)$
$a \oplus b = \max(a, b)$

- Dacă $U = (u_{ij})$, $V = (v_{ij})$ sunt matrici $n \times n$ cu elemente 0 sau 1, definim
  - $U \oplus V = (c_{ij})$ dacă $c_{ij} = u_{ij} \oplus v_{ij}$ pentru toți i, j
  - $U \odot V = (d_{ij})$ dacă $d_{ij} = (u_{i1} \odot v_{1j}) \oplus \ldots \oplus (u_{in} \odot v_{nj})$
  - $U^k = \underbrace{U \odot \ldots \odot U}_{k \text{ ori}}$ pentru orice $k > 0$

Proprietăți

1. Dacă $A_G^k = (a_{ij}^{(k)})$ pentru $k \geq 1$ atunci $a_{ij}^{(k)} = 1$ dacă și numai dacă există o cale cu lungimea k de la nodul v_i la v_j.
2. Fie $A_G^* = I_n \oplus A_G \oplus A_G^2 \oplus \ldots \oplus A_G^{n-1} = (\bar{a}_{ij})$. Atunci
   - $\bar{a}_{ij} = 1$ dacă și numai dacă există o cale de lungime $j \in \{1, \ldots, n-1\}$ de la nodul v_i la v_j.
   - A_G^* se poate calcula în $O(n^4)$.
   - A_G^* se numește închidere reflexivă și tranzitivă a lui A_G.
3. v_i și v_j sunt conectate $\Leftrightarrow \exists$ o cale simplă $v_i \rightsquigarrow v_j \Leftrightarrow \bar{a}_{ij} = 1$.
4. $I_n \oplus A_G \oplus A_G^2 \oplus \ldots \oplus A_G^{k+1} = I_n \oplus (I_n \oplus A_G \oplus A_G^2 \oplus \ldots \oplus A_G^k) \odot A_G$

   $\Rightarrow A^*$ se poate calcula în $O(n^4)$.

Proprietăți

1. Dacă $A_G^k = (a_{ij}^{(k)})$ pentru $k \geq 1$ atunci $a_{ij}^{(k)} = 1$ dacă și numai dacă există o cale cu lungimea k de la nodul v_i la v_j.
2. Fie $A_G^* = I_n \oplus A_G \oplus A_G^2 \oplus \ldots \oplus A_G^{n-1} = (\bar{a}_{ij})$. Atunci
   - $\bar{a}_{ij} = 1$ dacă și numai dacă există o cale de lungime $j \in \{1, \ldots, n-1\}$ de la nodul v_i la v_j.
   - A_G^* se poate calcula în $O(n^4)$.
   - A_G^* se numește închidere reflexivă și tranzitivă a lui A_G.
3. v_i și v_j sunt conectate $\Leftrightarrow \exists$ o cale simplă $v_i \rightsquigarrow v_j \Leftrightarrow \bar{a}_{ij} = 1$.
4. $I_n \oplus A_G \oplus A_G^2 \oplus \ldots \oplus A_G^{k+1} = I_n \oplus (I_n \oplus A_G \oplus A_G^2 \oplus \ldots \oplus A_G^k) \odot A_G$
   $\Rightarrow A^*$ se poate calcula în $O(n^4)$.

Corolar

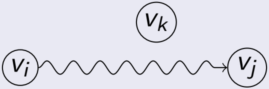
Conectivitatea într-un digraf simplu se poate detecta în $O(n^4)$.

Algoritmul lui Warshall calculează A_G^* în $O(n^3)$.

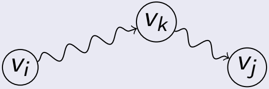
Idee de bază a lui Warshall

Dacă $V = [v_1, \ldots, v_n]$ este o enumerare a nodurilor lui G și $v_k \in V$, atunci orice cale simplă $\pi : v_i \rightsquigarrow v_j$ are una din următoarele forme:

1. v_k nu apare în π ca nod intermediar între v_i și v_j



2. v_k apare exact o dată în π ca nod intermediar între v_i și v_j



Presupunem că $A_G = (a_{ij})$ are dimensiune $n \times n$

- Se calculează recursiv $C^{[n]} = (c_{ij}^{[n]})$ unde

$$c_{ij}^{[k]} := \begin{cases} a_{ij} & \text{dacă } k = 0 \\ c_{ij}^{[k-1]} \oplus (c_{ik}^{[k-1]} \odot c_{kj}^{[k-1]}) & \text{dacă } k \geq 1 \end{cases}$$

Proprietăți

1. $C^{[0]} = A_G$
2. $c_{ij}^{[k]} = 1$ dacă și numai dacă există o cale $\pi : v_i \rightsquigarrow v_j$ în care toate nodurile intermediare sunt din submulțimea $\{v_1, \ldots, v_k\}$
3. $C^{[n]} = A_G^*$
4. $C^{[n]}$ se calculează în $O(n^3)$.

Se consideră digraful simplu $G = (V, E)$ cu $V = \{1, \ldots, n\}$ și funcția de greutate $w : E \to \mathbb{R}^+$

- În G, greutatea unei căi $\pi : v_1 \to v_2 \to \ldots \to v_p$ este
  - $w(\pi) = \sum_{i=1}^{p-1} w((v_i, v_{i+1}))$
  - (se adună greutățile tuturor arcelor din π)
- Pentru orice pereche de noduri (i, j) din V, se dorește găsirea
  - celei mai ușoare căi de la nodul i la nodul j (pot fi mai multe căi cele mai ușoare)
  - greutatea celei mai ușoare căi

Reamintim că matricea de ponderi a lui G este $W_G = (w_{ij})$ unde

$$w_{ij} = \begin{cases} 0 & \text{dacă } j = i, \\ w((i,j)) & \text{dacă} (i,j) \in E, \\ \infty & \text{dacă } (i,j) \notin E \end{cases}$$

Generalizare a ideii de calcul a celor mai mai scurte căi: Fie $k \in V = \{1, \ldots, n\}$, și $\pi : i \rightsquigarrow j$ o cale cea mai ușoară de la i la j. Distingem două cazuri:

1. π nu trece prin nodul k
2. π trece prin k. Atunci $\pi = i \overset{\pi_1}{\rightsquigarrow} k \overset{\pi_2}{\rightsquigarrow} j$ și $w(\pi) = w(\pi_1) + w(\pi_2)$.

Generalizare a ideii de calcul a celor mai mai scurte căi: Fie $k \in V = \{1, \ldots, n\}$, și $\pi : i \rightsquigarrow j$ o cale cea mai ușoară de la i la j. Distingem două cazuri:

1. π nu trece prin nodul k
2. π trece prin k. Atunci $\pi = i \overset{\pi_1}{\rightsquigarrow} k \overset{\pi_2}{\rightsquigarrow} j$ și $w(\pi) = w(\pi_1) + w(\pi_2)$.

Structură de date auxiliară de calcul:

- Matrice de căi ușoare $P^{[k]} = (p_{ij}^{[k]})$ în care fiecare

$$p_{ij}^{[k]} := \begin{cases} \bullet & \text{valoare specială: } \nexists \text{ cale } i \rightsquigarrow j \text{ prin noduri} \\ & \text{intermediare din submulțimea } \{1, \ldots, k\} \\ \pi & \text{o cea mai ușoară cale de la } i \text{ la } j \text{ prin noduri} \\ & \text{intermediare din submulțimea } \{1, \ldots, k\} \end{cases}$$

Presupunem că matricea de ponderi a lui G este $W_G = (w_{ij})$.
Pentru orice $0 \leq k \leq n$ se calculează recursiv, începând de la $k = 0$, următoarele valori:

- $p_{ij}^{[k]}$: o cale cea mai ușoară $p : i \rightsquigarrow j$, în care toate nodurile intermediare sunt din mulțimea $\{1, 2, \ldots, k\}$.
- $w_{ij}^{[k]}$: greutatea căii $p_{ij}^{[k]}$

OBSERVAȚII:

- Dacă nu există o astfel de cale de la i al j, definim $p_{ij}^{[k]} = \bullet$ și $w_{ij}^{[k]} = \infty$.
- $w_{ij}^{[0]} := w_{ij}$ și
- $p_{ij}^{[0]} := \begin{cases} [i] & \text{dacă } i = j \text{ (în acest caz, } w_{ij} = w_{ii} = 0) \\ [i, j] & \text{dacă } i \neq j \text{ și } w_{ij} < \infty \\ \bullet & \text{dacă } w_{ij} = \infty \end{cases}$

Pentru $0 < k \leq n$:

$$p_{ij}^{[k]} := \begin{cases} p_{ij}^{[k-1]} & \text{dacă } w_{ij}^{[k-1]} \leq w_{ik}^{[k-1]} + w_{kj}^{[k-1]} \\ p_{ik}^{[k-1]} \asymp p_{kj}^{[k-1]} & \text{în caz contrar} \end{cases}$$

unde $p_{ik}^{[k-1]} \asymp p_{kj}^{[k-1]}$ denotă concatenarea căilor $p_{ik}^{[k-1]}$ și $p_{kj}^{[k-1]}$.

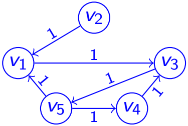
$$w_{ij}^{[k]} := \begin{cases} w_{ij}^{[k-1]} & \text{dacă } w_{ij}^{[k-1]} \leq w_{ik}^{[k-1]} + w_{kj}^{[k-1]} \\ w_{ik}^{[k-1]} + w_{kj}^{[k-1]} & \text{în caz contrar} \end{cases}$$

Observații

- Pentru orice $1 \leq i, j \leq n$, dacă $p_{ij}^{[n]} \neq \bullet$ atunci $p_{ij}^{[n]}$ este o cea mai ușoară cale de la i la j, iar $w(p_{ij}^{[n]}) = w_{ij}^{[n]}$.
- Calculul matricilor $W^{[n]}$ și $P^{[n]}$ se efectuează în $O(n^3)$.

Cele mai scurte căi într-un digraf simplu

- Lungimea unei căi $p = [x_1, \ldots, x_n]$ este $n - 1$, adică numărul de arce ce la x_1 la x_n de-a lungul căii p.
- Dacă presupunem că fiecare arc are greutatea 1, atunci lungimea unei căi coincide cu greutatea ei.
  - $\Rightarrow$ cele mai scurte căi dintre orice pereche de noduri pot fi calculate cu algoritmul lui Warshall (vezi slide-ul următor.)



$$PW_G^{[0]} = \begin{pmatrix} [v_1]_0 & \bullet & [v_1,v_3]_1 & \bullet & \bullet \\ [v_2,v_1]_1 & [v_2]_0 & \bullet & \bullet & \bullet \\ \bullet & \bullet & [v_3]_0 & \bullet & [v_3,v_5]_1 \\ \bullet & \bullet & [v_4,v_3]_1 & [v_4]_0 & \bullet \\ [v_5,v_1]_1 & \bullet & \bullet & [v_5,v_4]_1 & [v_5]_0 \end{pmatrix},$$

$$PW_G^{[1]} = \begin{pmatrix} [v_1]_0 & \bullet & [v_1,v_3]_1 & \bullet & \bullet \\ [v_2,v_1]_1 & [v_2]_0 & [v_2,v_1,v_3]_2 & \bullet & \bullet \\ \bullet & \bullet & [v_3]_0 & \bullet & [v_3,v_5]_1 \\ \bullet & \bullet & [v_4,v_3]_1 & [v_4]_0 & \bullet \\ [v_5,v_1] & \bullet & [v_5,v_1,v_3]_2 & [v_5,v_4]_1 & [v_5]_0 \end{pmatrix},$$

$$PW_G^{[2]} = PW_G^{[1]},\ PW_G^{[3]} =$$

$$\begin{pmatrix} [v_1]_0 & \bullet & [v_1,v_3]_1 & \bullet & [v_1,v_3,v_5]_2 \\ [v_2,v_1]_1 & [v_2]_0 & [v_2,v_1,v_3]_2 & \bullet & [v_2,v_1,v_3,v_5]_3 \\ \bullet & \bullet & [v_3]_0 & \bullet & [v_3,v_5]_1 \\ \bullet & \bullet & [v_4,v_3]_1 & [v_4]_0 & [v_4,v_3,v_5]_2 \\ [v_5,v_1]_1 & \bullet & [v_5,v_1,v_3]_2 & [v_5,v_4]_1 & [v_5]_0 \end{pmatrix},\ PW_G^{[4]} = PW_G^{[3]},$$

$$PW_G^{[5]} = \begin{pmatrix} [v_1]_0 & \bullet & [v_1,v_3] & [v_1,v_3,v_5,v_4]_3 & [v_1,v_3,v_5]_2 \\ [v_2,v_1]_1 & [v_2]_0 & [v_2,v_1,v_3]_2 & [v_2,v_1,v_3,v_5,v_4]_4 & [v_2,v_1,v_3,v_5]_3 \\ [v_3,v_5,v_1]_2 & \bullet & [v_3]_0 & \bullet & [v_3,v_5]_1 \\ [v_4,v_3,v_5,v_1]_3 & \bullet & [v_4,v_3]_1 & [v_4]_0 & [v_4,v_3,v_5]_2 \\ [v_5,v_1]_1 & \bullet & [v_5,v_1,v_3]_2 & [v_5,v_4]_1 & [v_5]_0 \end{pmatrix}$$