Tropical algebra

From shortest path algorithms to Hamilton-Jacobi-Bellman Equation

Nicolas Delanoue
LARIS - Universite d'Angers - France
http://perso-laris.univ-angers.fr/~delanoue/

Medellin - EAFIT http://www.eafit.edu.co/

November 2019

Outline

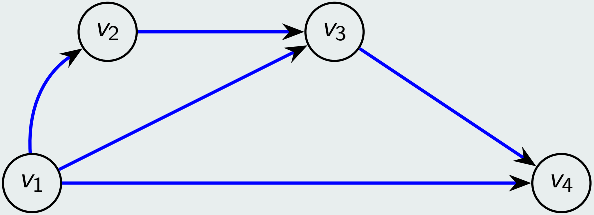
1. Graph theory
   - Bellman Ford Algorithm
   - An example
2. Tropical linear algebra
   - Semi ring
   - Bellman-Ford algorithm with tropical algebra
3. Optimal control - Hamilton Jacobi Bellman

Definition - Graph

A *directed graph* is an ordered pair $G = (V, E)$ where

- V is a set whose elements are called vertices,
- E is a set of ordered pairs of vertices, called directed edges.

Example

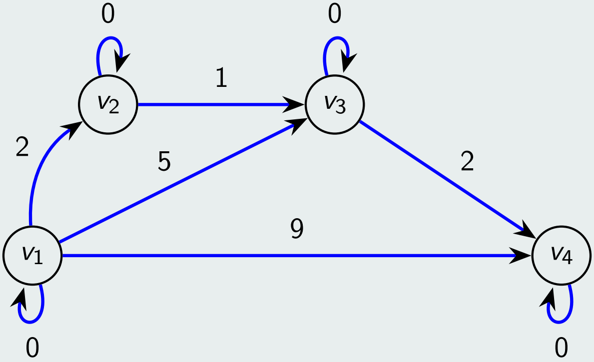


Here, $V = \{v_1, v_2, v_3, v_4\}$ and
$E = \{(v_1, v_2), (v_1, v_3), (v_1, v_4), (v_2, v_3), (v_3, v_4)\}$

Weighted graph

A *weighted directed graph* is a directed graph with weights assigned to their edges, i.e. one has function $h : E \to \mathbb{R}$.

Example



Here, $h(v_1, v_2) = 2, h(v_1, v_3) = 5, \ldots$

Shortest path problem

The *shortest path problem* is the problem of finding a path between two vertices in a graph such that the sum of the weights of its constituent edges is minimized.

Algorithms

- Dijkstra's algorithm solves the single-source shortest path problem with non-negative edge weight.
- Bellman Ford algorithm solves the single-source problem if edge weights may be negative.
- ...

Bellman-Ford Algorithm

Input : A weighted directed graph (V, E, h), a source vertex s

Bellman-Ford Algorithm

Input : A weighted directed graph (V, E, h), a source vertex s
Output: The cost of the shortest path from s to all other nodes :

$$V \ni c \mapsto J(c) \in \mathbb{R}$$

Bellman-Ford Algorithm

Input : A weighted directed graph (V, E, h), a source vertex s
Output: The cost of the shortest path from s to all other nodes :

$$V \ni c \mapsto J(c) \in \mathbb{R}$$

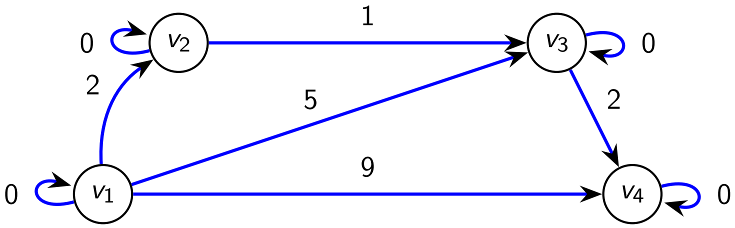
for $c \in V - \{s\}$ **do**
    $J(c, 0) \leftarrow +\infty$;
end
$J(s, 0) \leftarrow 0$;

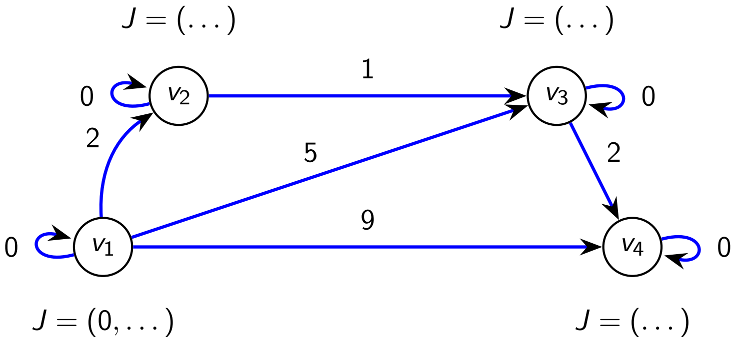
Bellman-Ford Algorithm

Input : A weighted directed graph (V, E, h), a source vertex s
Output: The cost of the shortest path from s to all other nodes :

$$V \ni c \mapsto J(c) \in \mathbb{R}$$

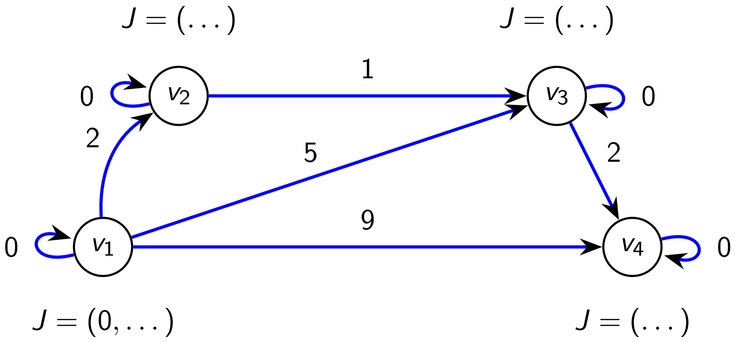
```
for c ∈ V − {s} do
    J(c, 0) ← +∞ ;
end
J(s, 0) ← 0;
for k ← 1 to #V − 1 do
    for c ∈ V do
        J′ ← +∞;
        for (u, c) ∈ E do
            J′ ← min(J′, J(u, k − 1) + h(u, c)) ;
        end
        J(c, k) ← J′;
    end
end
J(·) = min_k J(·, k)
```


0
v_2
1
v_3
0
2
5
2
9
0
v_1
v_4
0



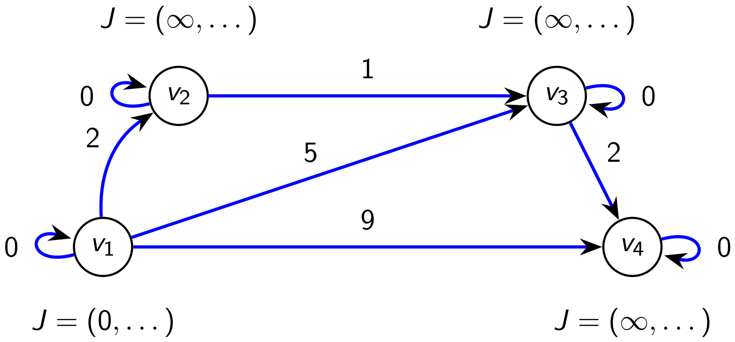
Initialisation steps :

- $J(v_1) = (0, \dots)$ since v_1 is the source,



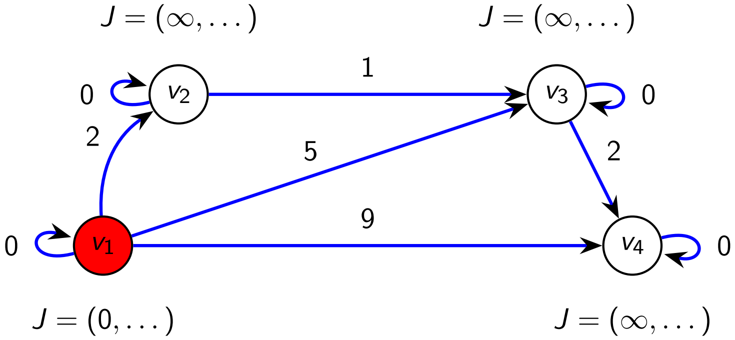
Initialisation steps :

- $J(v_1) = (0, \dots)$ since v_1 is the source,
- $J(v_i) = (\infty, \dots)$ for all other vertices.



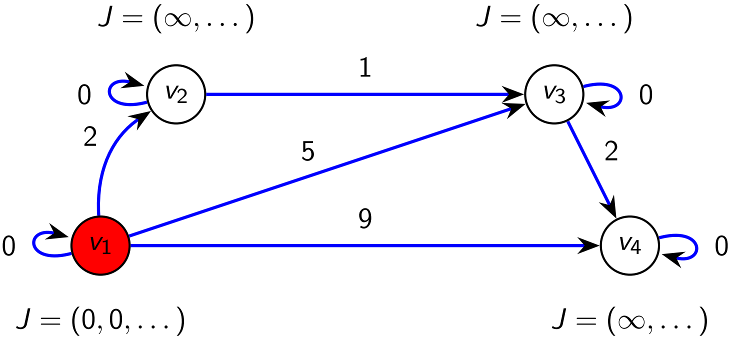
Initialisation steps :

- $J(v_1) = (0, \dots)$ since v_1 is the source,
- $J(v_i) = (\infty, \dots)$ for all other vertices.



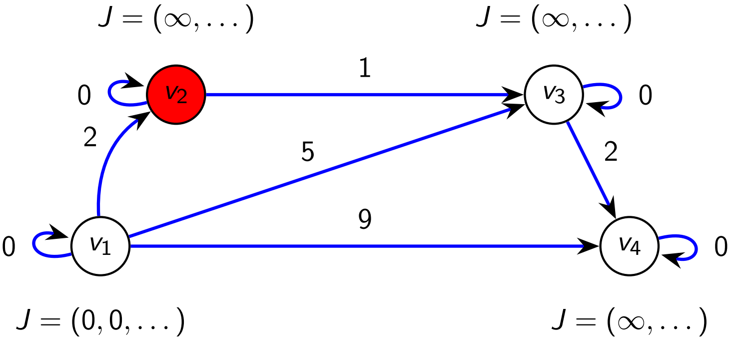
Iteration k = 1 :

- v_1 has only v_1 as predecessors, therefore $J(v_1) = (0, 0, \dots)$

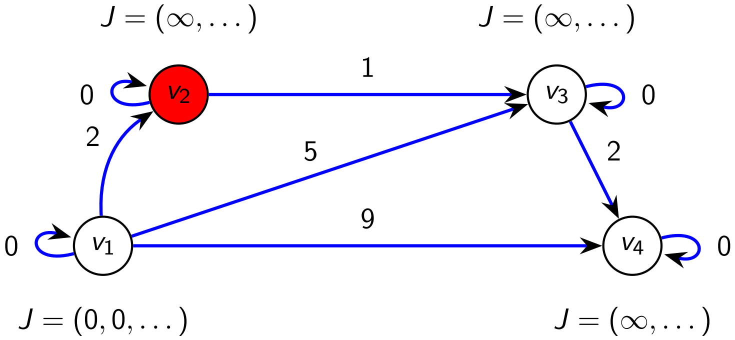


Iteration k = 1 :

- v_1 has only v_1 as predecessors, therefore $J(v_1) = (0, 0, \dots)$

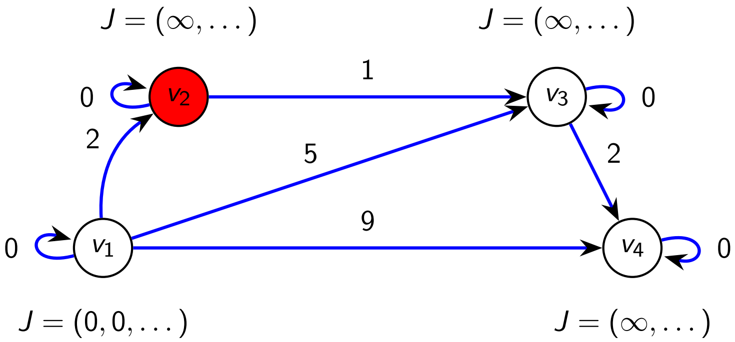


Iteration k = 1 :



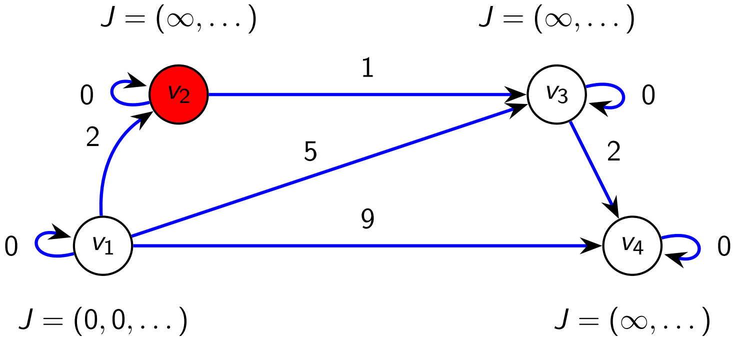
Iteration k = 1 :

- v_2 has two predecessors : v_1 and v_2, therefore



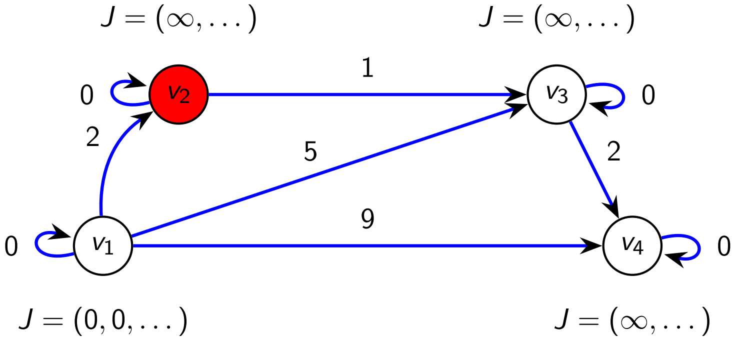
Iteration k = 1 :

- v_2 has two predecessors : v_1 and v_2, therefore
  $J(v_2, k) = \min\{J(v_1, 0) + h(v_1, v_2), J(v_2, 0) + h(v_2, v_2)\}$



Iteration k = 1 :

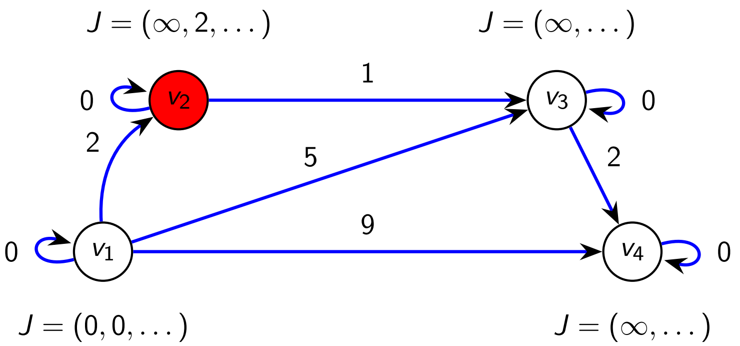
- v_2 has two predecessors : v_1 and v_2, therefore
  $J(v_2, k) = \min\{J(v_1, 0) + h(v_1, v_2), J(v_2, 0) + h(v_2, v_2)\}$
  $= \min\{\ 0 \quad + \quad 2 \quad , \ \infty \quad + \quad 0 \quad \}$



Iteration k = 1 :

- v_2 has two predecessors : v_1 and v_2, therefore

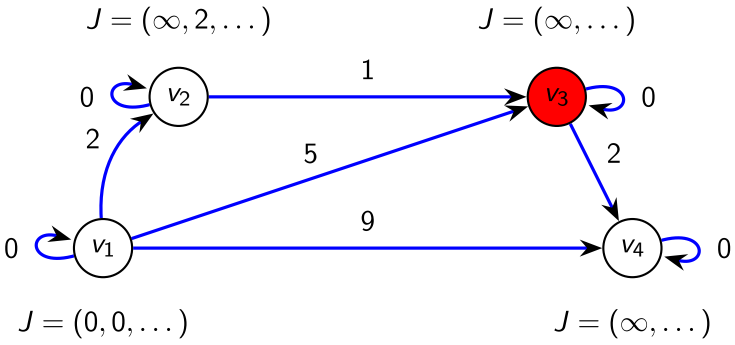
$$\begin{aligned} J(v_2, k) &= \min\{J(v_1, 0) + h(v_1, v_2), J(v_2, 0) + h(v_2, v_2)\} \\ &= \min\{\ 0 \quad + \quad 2 \quad , \ \infty \quad + \quad 0 \quad \} \\ &= 2 \end{aligned}$$



Iteration k = 1 :

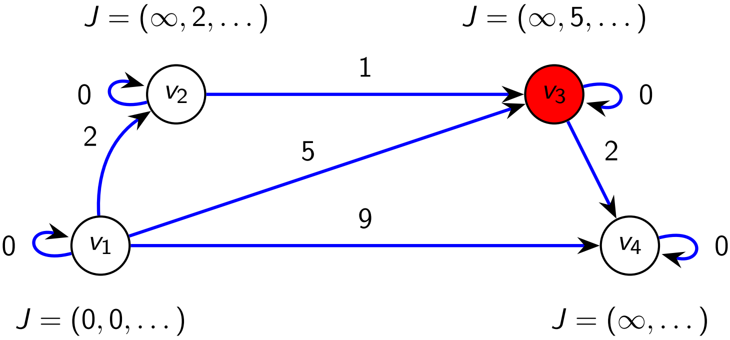
- v_2 has two predecessors : v_1 and v_2, therefore

$$
\begin{aligned}
J(v_2, k) &= \min\{J(v_1, 0) + h(v_1, v_2), J(v_2, 0) + h(v_2, v_2)\} \\
&= \min\{\ 0 \quad + \quad 2 \quad , \ \infty \quad + \quad 0 \quad \} \\
&= 2
\end{aligned}
$$



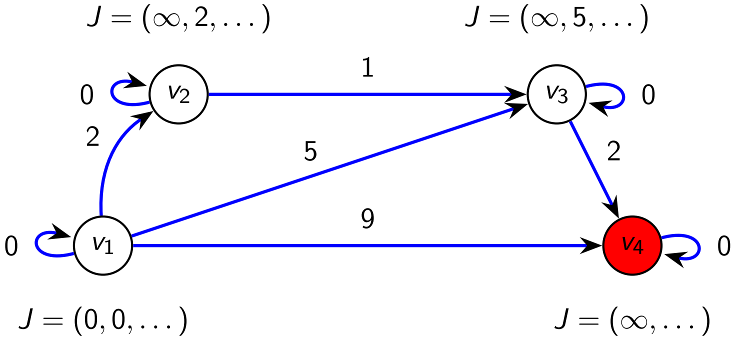
Iteration k = 1 :

- $J(v_3, k) = 5$



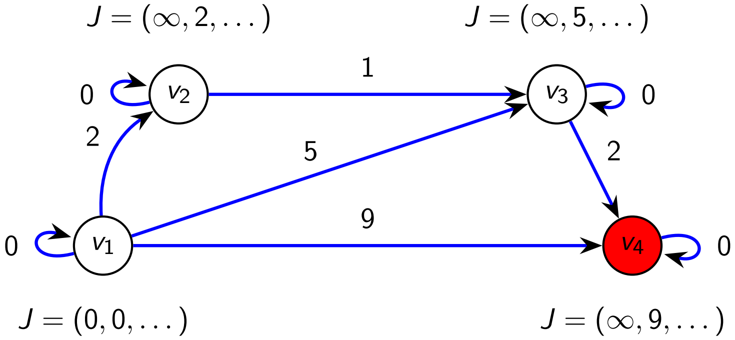
Iteration k = 1 :

- $J(v_3, k) = 5$



Iteration k = 1 :

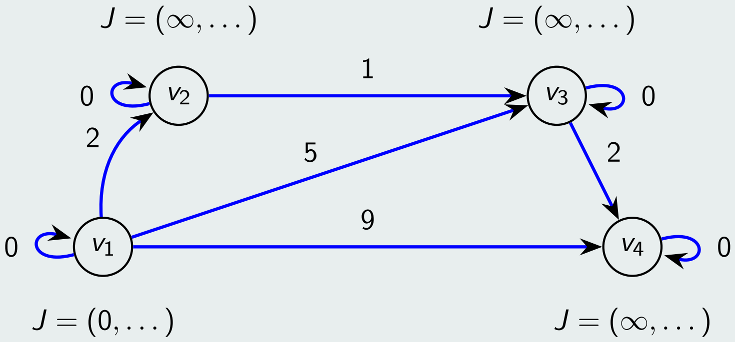
- $J(v_4, k) = 9$



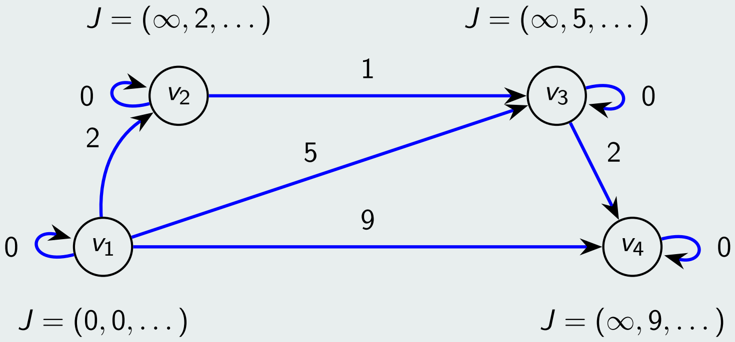
Iteration k = 1 :

- $J(v_4, k) = 9$

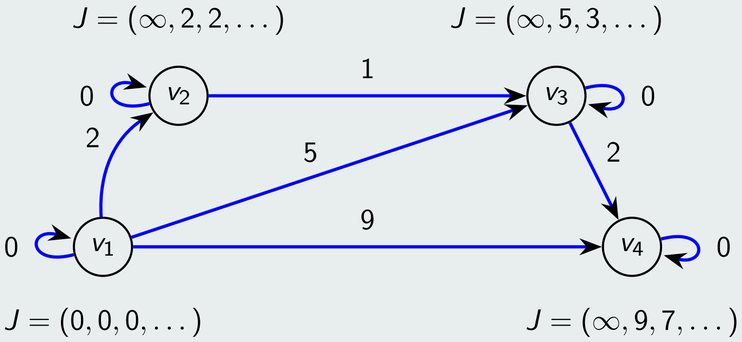
Initialisation



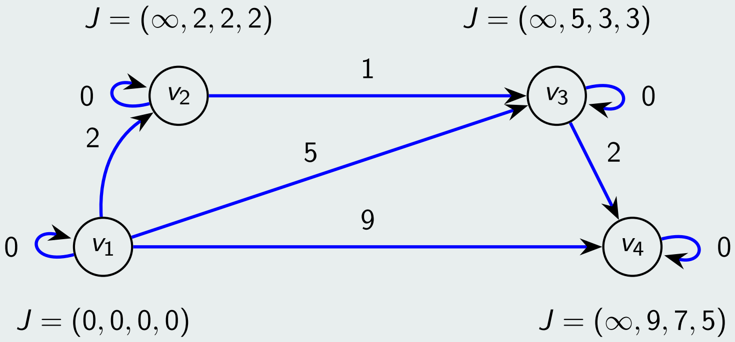
$k = 1$



$k = 2$


$J = (\infty, 2, 2, \dots)$
$J = (\infty, 5, 3, \dots)$
1
0
v_2
v_3
0
2
5
2
9
0
v_1
v_4
0
$J = (0, 0, 0, \dots)$
$J = (\infty, 9, 7, \dots)$

$k = 3$



Initialisation

	0
v_1	
v_2	
v_3	
v_4	

Initialisation

	0
v_1	0
v_2	∞
v_3	∞
v_4	∞

$k = 1$

	0	1
v_1	0	0
v_2	∞	2
v_3	∞	5
v_4	∞	9

$k = 2$

	0	1	2
v_1	0	0	0
v_2	∞	2	2
v_3	∞	5	3
v_4	∞	9	7

$k = 3$

	0	1	2	3
v_1	0	0	0	0
v_2	∞	2	2	2
v_3	∞	5	3	3
v_4	∞	9	7	5

$k = 3$

	0	1	2	3
v_1	0	0	0	0
v_2	∞	2	2	2
v_3	∞	5	3	3
v_4	∞	9	7	5

$$J_0 = \begin{pmatrix} 0 \\ \infty \\ \infty \\ \infty \end{pmatrix}, \quad \text{and } J_{k+1} = f(J_k).$$

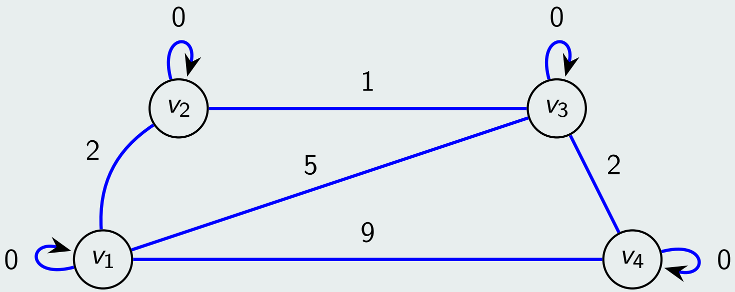
Definition

Let (V, E, h) be a weighted directed graph with $V = \{v_1, \ldots, v_n\}$, we define the square matrix $A = (a_{ij})_{i,j \in 1,\ldots,n}$ with

$$a_{ij} = h(v_i, v_j).$$

I call the matrix A the HJB matrix.

Example 2



$$A = \begin{pmatrix} 0 & 2 & 5 & 9 \\ 2 & 0 & 1 & \infty \\ 5 & 1 & 0 & 2 \\ 9 & \infty & 2 & 0 \end{pmatrix}$$

Remark

The coefficient i, j of A is the cost from node v_i to v_j using one edge.

Recall - Matrix multiplication

Let $A = (a_{ij})_{i,j \in 1,\dots,n}$ be a square matrix then the i, j coefficient of A^2, c_{ij} is given by

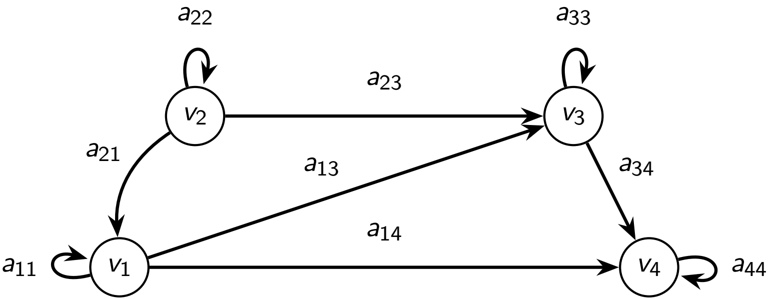
$$c_{ij} = \sum_{k=1}^{n} a_{ik} a_{kj}$$

Example

$$c_{23} = a_{21} a_{13} + a_{22} a_{23} + a_{23} a_{33} + a_{24} a_{43}$$

Interpretation

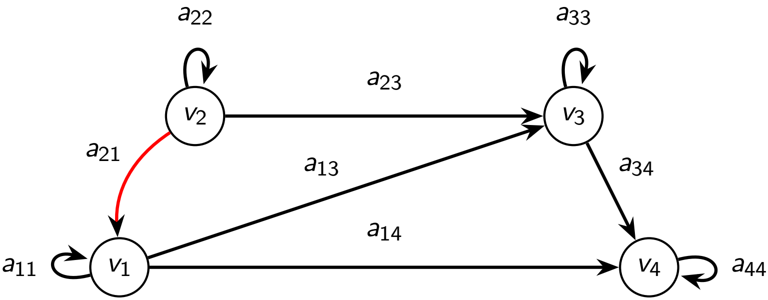
The coefficient c_{ij} is composed of all paths from i to j with two edges.



$c_{23} =$

Interpretation

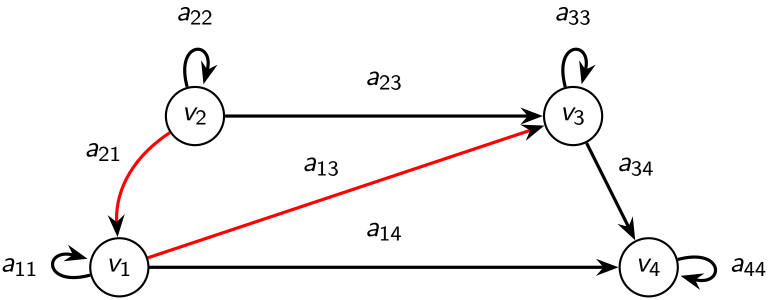
The coefficient c_{ij} is composed of all paths from i to j with two edges.



$$c_{23} = a_{21}$$

Interpretation

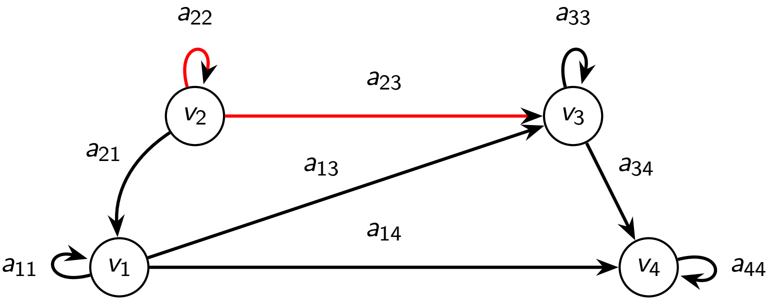
The coefficient c_{ij} is composed of all paths from i to j with two edges.



$$c_{23} = a_{21} a_{13}$$

Interpretation

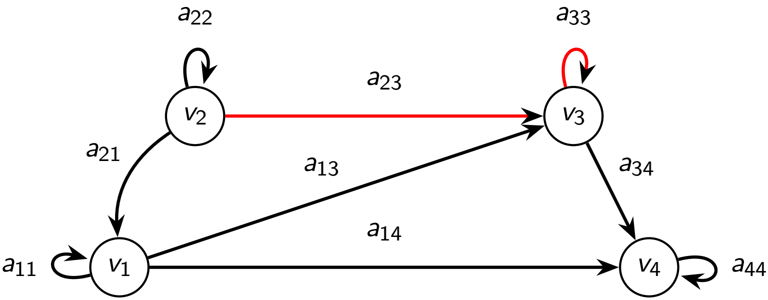
The coefficient c_{ij} is composed of all paths from i to j with two edges.



$$c_{23} = a_{21}a_{13} + a_{22}a_{23}$$

Interpretation

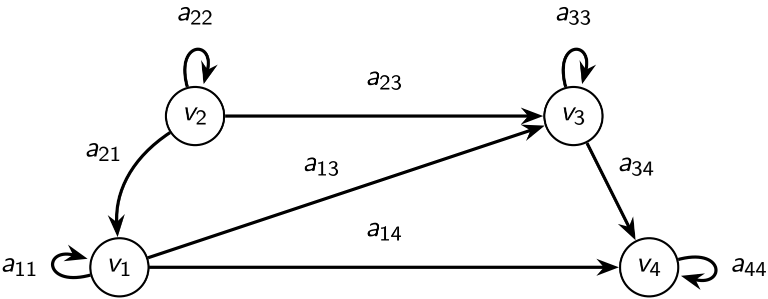
The coefficient c_{ij} is composed of all paths from i to j with two edges.



$$c_{23} = a_{21}a_{13} + a_{22}a_{23} + a_{23}a_{33}$$

Interpretation

The coefficient c_{ij} is composed of all paths from i to j with two edges.



$$c_{23} = a_{21}a_{13} + a_{22}a_{23} + a_{23}a_{33} + a_{24}a_{43}$$

Definition

The min tropical semiring is the semiring $(\mathbb{R} \cup \{\infty\}, \oplus, \otimes)$, with the operations :

- $x \oplus y = \min\{x, y\}$,
- $x \otimes y = x + y$.

Definition

The min tropical semiring is the semiring $(\mathbb{R} \cup \{\infty\}, \oplus, \otimes)$, with the operations :

- $x \oplus y = \min\{x, y\}$,
- $x \otimes y = x + y$.

Example

- $2 \oplus 3 = 2$,

Definition

The min tropical semiring is the semiring $(\mathbb{R} \cup \{\infty\}, \oplus, \otimes)$, with the operations :

- $x \oplus y = \min\{x, y\}$,
- $x \otimes y = x + y$.

Example

- $2 \oplus 3 = 2$,
- $2 \otimes 3 = 5$.

Remarks

- The operations $\oplus$ and $\otimes$ are referred to as tropical addition and tropical multiplication respectively,
- The unit for $\oplus$ is ∞,
- the unit for $\otimes$ is 0.

Linear tropical algebra

Let $A = (a_{ij})_{i,j \in 1,\dots,n}$ be a square matrix then the i, j coefficient of A^2, c_{ij} is given by

$$c_{ij} = \bigoplus_{k=1}^{n} a_{ik} \otimes a_{kj}$$

Example with HJB matrix

$$c_{2,3} = (a_{2,1} \otimes a_{1,3}) \oplus (a_{2,2} \otimes a_{2,3}) \oplus (a_{2,3} \otimes a_{3,3}) \oplus (a_{2,4} \otimes a_{4,3})$$

Lemma

The real value c_{ij} is the smallest cost of paths from v_i to v_j following by two edges.

Proposition

Let $k \in \mathbb{N}$, with the min tropical semi ring, coefficient i, j of the matrix A^k contains the smallest cost of all paths from v_i to v_j using k edges.

Proposition

Let $k \in \mathbb{N}$, with the min tropical semi ring, coefficient i, j of the matrix A^k contains the smallest cost of all paths from v_i to v_j using k edges.

Bellman-Ford algorithm from the tropical point of view :

$$\begin{cases} J_0 & = & H, \\ J_{n+1} & = & AJ_n, \end{cases} \tag{1}$$

with $H = (\infty, \dots, \infty, 0, \infty, \dots, \infty)^T$.

Proposition

Let $k \in \mathbb{N}$, with the min tropical semi ring, coefficient i, j of the matrix A^k contains the smallest cost of all paths from v_i to v_j using k edges.

Bellman-Ford algorithm from the tropical point of view :

$$\begin{cases} J_0 & = & H, \\ J_{n+1} & = & AJ_n, \end{cases} \tag{1}$$

with $H = (\infty, \ldots, \infty, 0, \infty, \ldots, \infty)^T$.

Solution :

$$J_n = A^n J_0$$

Remarks

Due to tropical linearity, i.e. superposition property :

$$A^k(H_1 \oplus H_2) = A^k H_1 \oplus A^k H_2.$$

- $A^k(0, 0, \infty, \dots, \infty)^T$ is the smallest cost to reach any nodes from one of the two sources v_1 and v_2.
- $(0, \infty, \dots, \infty)A^k$ is the cost from any nodes to the target v_1.

Controlled dynamical system

$$\begin{cases} x(0) & = & x_0 \\ \dot{x}(\tau) & = & f(x(\tau), u(\tau)), \forall \tau \in [0, T], \end{cases}$$

where

- τ is the time,
- x is the state,
- f is a vector field (the dynamics),
- u is the control.

Optimal control problem

$$
\begin{aligned}
J^* = \min_{u:[0,T]\to U} \quad & \int_0^T h(\tau, x(\tau), u(\tau))d\tau + H(x(T)) \\
\text{subject to} \quad & x(0) = x_0 \\
& \dot{x}(\tau) = f(x(\tau), u(\tau)), \forall \tau \in [0, T], \\
& x(\tau) \in X, \forall \tau \in [0, T], \\
& x(T) \in K.
\end{aligned}
$$

where

- U is the set of admissible control,
- h and H are real valued functions.

Definition - Optimal cost

$$J^*(x,t) = \min_{u:[t,T]\to U} \int_t^T h(x(\tau),u(\tau))d\tau + H(x(T))$$

such that $x : t \mapsto X$ satisfies

$$\begin{cases} \dot{x}(\tau) = f(x(\tau),u(\tau)) \\ x(t) = x \end{cases}$$

Hamilton Jacobi Bellman Theorem

The value function $(t, x) \mapsto J^*(t, x)$ satisfies the partial differential equation :

$$\frac{\partial J^*}{\partial t} = - \min_{u(t) \in U} \left\{ h(x, u(t)) + \frac{\partial J^*}{\partial x} f(x, u(t)) \right\} \quad (2)$$

with final condition $J^*(T, x) = H(x)$.

Hamilton Jacobi Bellman Theorem

The value function $(t,x) \mapsto J^*(t,x)$ satisfies the partial differential equation :

$$\frac{\partial J^*}{\partial t} = -\min_{u(t)\in U}\left\{h(x,u(t)) + \frac{\partial J^*}{\partial x} f(x,u(t))\right\} \tag{2}$$

with final condition $J^*(T,x) = H(x)$.

Remark

- Equation (2) is a infinite dimensional dynamical system, indeed, the state space is the set of real value fonction $\varphi : X \to \mathbb{R}$.

Proposition

Let us denote by $S^T(H)$ the solution of optimal control problem with final cost H. One has :

- $S^0 = Id$,
- $S^{t_1+t_2} = S^{t_1} S^{t_2}$,
- $S^t(\alpha \otimes H) = \alpha \otimes S^t H_1$.
- $S^t(H_1 \oplus H_2) = S^t H_1 \oplus S^t H_2$.

To finish

Suppose the function J is solution of the following Hamilton-Jacobi equation

$$\frac{\partial J}{\partial t} = H(x, \frac{\partial J}{\partial x}) \text{ and } J(0, \cdot) = \varphi(\cdot)$$

with

$$H(x,p) = \min_u \left(h(u,x) + p \cdot u\right)$$

Hopf formula gives :

$$J(t,x) = \min_y t \cdot h(\frac{x-y}{t}) + \varphi(y)$$

To finish

Suppose the function J is solution of the following Hamilton-Jacobi equation

$$\frac{\partial J}{\partial t} = H(x, \frac{\partial J}{\partial x}) \text{ and } J(0, \cdot) = \varphi(\cdot)$$

with

$$H(x, p) = \min_u \left(h(u, x) + p \cdot u\right)$$

Hopf formula gives :

$$J(t, x) = \min_y t \cdot h(\frac{x - y}{t}) + \varphi(y) = \bigoplus_y t \cdot h_t(x - y)\varphi(y)dy$$

To finish

Suppose the function J is solution of the following Hamilton-Jacobi equation

$$\frac{\partial J}{\partial t} = H(x, \frac{\partial J}{\partial x}) \text{ and } J(0, \cdot) = \varphi(\cdot)$$

with

$$H(x, p) = \min_u \left(h(u, x) + p \cdot u \right)$$

Hopf formula gives :

$$J(t, x) = \min_y t \cdot h(\frac{x - y}{t}) + \varphi(y) = \bigoplus_y t \cdot h_t(x - y)\varphi(y)dy$$

Note that in this case h is the Legendre transform of H.

1. Graphs, Dioids and Semirings : New Models and Algorithms, Gondran, M. and Minoux, M., Springer Science, 2008
2. Max-plus approximations : from optimal control to template methods, Gaubert 2014
3. Dower, P.M. and McEneaney, W.M. A max-plus based fundamental solution for a class of infinite dimensional Riccati equations.

1. Graphs, Dioids and Semirings : New Models and Algorithms, Gondran, M. and Minoux, M., Springer Science, 2008
2. Max-plus approximations : from optimal control to template methods, Gaubert 2014
3. Dower, P.M. and McEneaney, W.M. A max-plus based fundamental solution for a class of infinite dimensional Riccati equations.

Gracias por su atención.