École polytechnique de Louvain

LINFO1115
Graph Theory

Année académique :
EPL BAC3

Auteurs :
Félix Gaudin

19 mai 2021

TABLE DES MATIÈRES

1 GRAPHE THEORIE

1.1 Quelques définitions

- **Node :** Point du graphe
- **Edge :** Connection sur le graphe
- **Directed graph :** Y'a des flèches zézoli (ex : $N = \{a, b, c\}, E = \{\{a, b\}, \{b, c\}\}$)
- **Undirected graph :** Y'a pas de flèches du coup on compte que chaque edge c'est une flèche dans les deux sens
- **Path :** Séquence de nodes de tel sorte que y'ait une edge entre chaque pair qui existe sur le graphe
- **Simple path :** Un chemin où y'a max 1 fois la même node
- **Cycle :** un path avec minimum ($\geqslant$) 3 edges tel que la première et dernière nodes sont les mêmes
- **Connected graph :** (=connexe) il existe un chemin entre chaque nodes
- **Component :** C'est un sous-ensemble de nodes tel que :
  - Le sous-ensemble est **connexe**
  - Le sous-ensemble est **maximal** ("Prendre tout le sous graphe")

  Exemple :

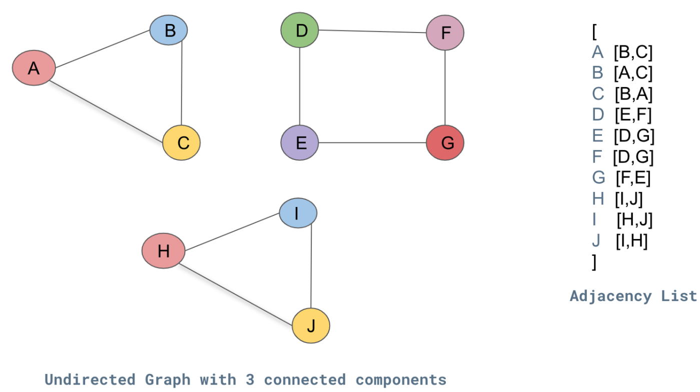


FIGURE 1 – Exemple de component

  Ici on a 3 components ($\{A, B, C\}$, $\{D, F, G, E\}$ et $\{H, I, J\}$). La première propriété c'est genre on peut pas faire un component avec A, B, C, H, I, J par exemple (*y'a un trou entre plusieurs nodes*). Et la seconde propriétés c'est qu'on peut pas prendre un compoent avec D, F et E, il faut que ça soit maximal du coup on doit prendre G.
- **Lenght of a path :** Le nombre d'edges dans un path (Mazette je m'y attendais pas)
- **Distance between two nodes :** Le shortest path entre eux POG

1.2 Breadth-first search (BFS)

BFS c'est pas la Belgium Flying School mais bien un algo pour trouver un shortest path. Bon on va compter que t'as suivi algo3 et que tu sais ce que c'est. Mais bon potit rappel :
Tu prends toi (distance de 0), puis tu regardes tes potes (distance de 1), puis les potes de tes potes (distance de 2), puis les potes des tes potes de tes potes (distance de 3), et ainsi de suite. On compte bien sur que t'es sur un graphe connected sinon la distance est infinie (en pratique t'es quand même sensé avoir des potes sinon pas de chance).

1.3 Triadic closure

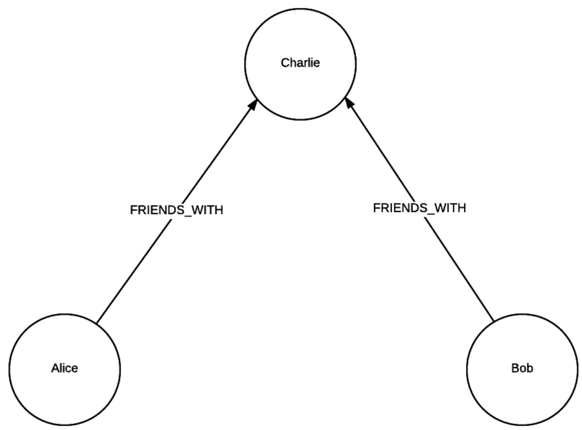


FIGURE 2 – Exemple de triadic closure

C'est super compliqué mais ici vu que Alice est pote avec Charlie et que Bob et aussi pote avec Charlie bah y'a une grosse chance que que Alice et Bob deviennent pote :saperlipopette :

1.4 Clustering coefficient

$$CC(A) = \frac{\text{Nombre de pair d'amis de A qui sont amis entre eux}}{\text{Nombre d'ami de A}}$$

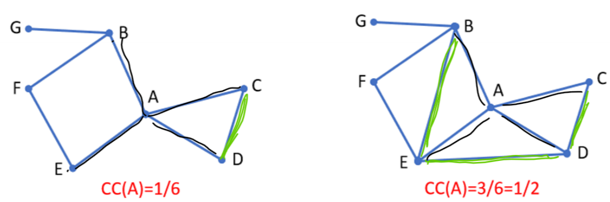


FIGURE 3 – Exemple de clustering coefficient

(Je sais pas quoi dire de plus)

1.5 Bridge

C'est une edge qui, si elle est retirer, ça split le graphe en deux components.

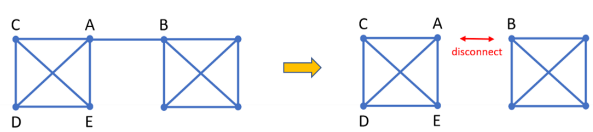


FIGURE 4 – Exemple de bridge

En pratique ça arrive quasi jamais parce qu'on est super connecter du coup on a des local brigdes

1.5.1 *Local bridge*

C'est comme une autoroute à coté des chemins de campagne. L'autoroute c'est un chemin paf t'es là où tu veux, sauf que si l'autoroute est cassée tu dois te taper les chemins de campagne (plus long). On compte que c'est un pont local quand si le pont casse la distance pour rejoindre l'autre noeud est strictement supérieur à 2.

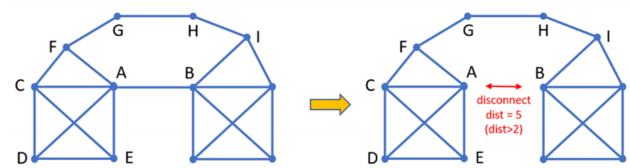


Figure 5 – Exemple de local bridge

1.5.2 *Notion de STRONG*

Bon il sort de nul part dans le cours mais l'idée c'est que sur un triadic closure, si A est fortement lié à B et que C est aussi fortement lié à B. On a donc que A est faiblement lié à C.

1.5.3 *Neighborhood overlap*

C'est un truc qui calcule les locals bridges.

$$overlaping(A, B) = \frac{\text{Nombre de nodes qui sont pote de A ET B}}{\text{Nombre de nodes qui sont pote de A OU B}}$$

— **Si = 0 :** local bridge
— **Si tend vers 0 :** "Almost" local bridges

1.6 Homophily

T'es potes avec des gens qui ont des trucs en communs avec toi.

1.6.1 *Caractéristiques immutables*

Truc que tu peux pas changer (Age, groupe ethnique, sexe, etc)

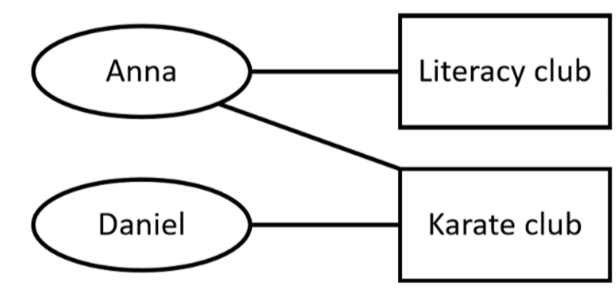
1.6.2 *Caractéristiques mutables*

Truc que tu peux changer (Occupations, croyances, couleur de cheveux (ouais fin déconne pas non plus))

1.6.3 *Mécanismes sous-jasent*

— **Selection :** Tu choisis les gens qui te ressemble
— **Social influence :** Tu te fais influencer par les autres (genre si tout le monde va discuter sur discord bah tu vas aussi aller sur discord)

1.7 Affiliation network



Un affiliation network c'est un *bipartite graph*.
Un *bipartite graph* c'est un graphe avec plusieurs "set de nodes". Ici on a les *individuals* (les gens) et les *foci* (les activités).
Mais du coup ça marche comme avant pour la triadic closure genre ici Anna et Daniel font du karaté du coup ils ont une grande chance d'être pote (sauf que le nom c'est pas le même mais on verra ça après).

1.8 Mesurer l'homophilie

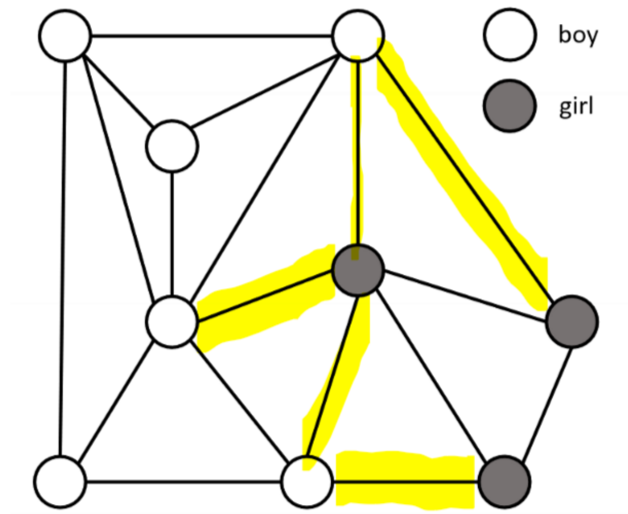


FIGURE 6 – Exemple de calcul d'homophily

Nombre de boys = 6
Nombre de girls = 3
$\Longrightarrow$
$P_b = \frac{6}{9} = \frac{2}{3}$ (proportion boys)
$P_g = \frac{3}{9} = \frac{1}{3}$ (proportion girls)
On a que$P_b + P_g = 1$
On utilise$(P_b + P_g)^2 = P_b^2 + 2P_bPg + P_g^2$
Si il n'y a pas d'homophilie le nombre de liaisons boy-girl doit être $\geqslant$ à $2PbP_g$
Ici $2PbP_g = \frac{8}{18}$
Or le nombre de liaisons boy-girl = $5 \rightarrow \frac{5}{18}$
$\Longrightarrow$ Homophilie

1.9 Type de closures IMPORTANT

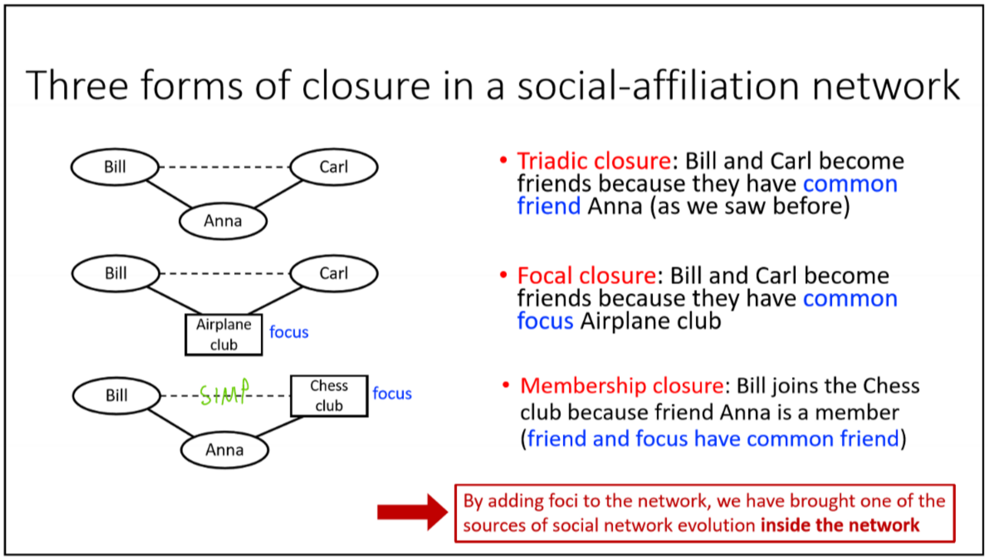


Figure 7 – Types de closures

1.10 Positive/negative relationship

En gros positif = pote, negatif = le type que tu peux pas saquer.
Du coup y'a plusieurs type de "triangles" possibles :

— + + + [balanced] tout le monde est pote du coup ça se passe bien

— + - - [balanced] ton pote et toi vous avec le même type que vous pouvez pas saquer

— - - - [weak] tout le monde se déteste (peut rester comme ça longtemps ou peut disparaitre)

— + + - [unbalanced] t'es pote avec deux personne qui se détestent (tend à disparaître)

Balance theorem

Si le graph est balancé (que + + + et + - -) il y a deux groupes.

Preuve

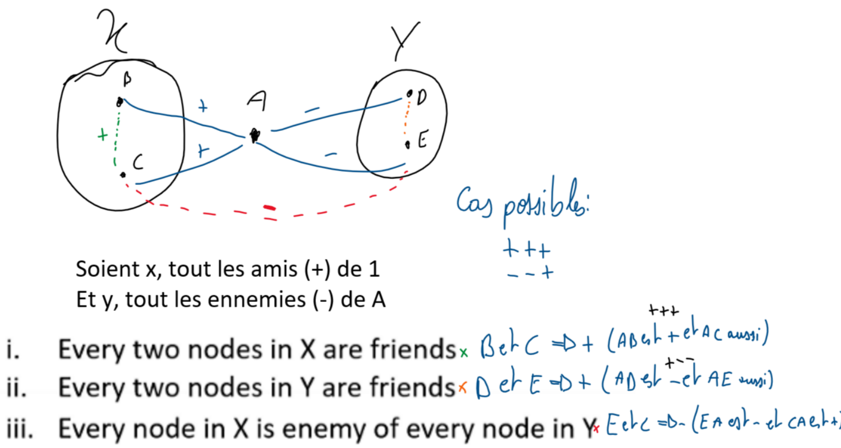


FIGURE 8 – Preuve du balance theorem

Nombre de groupes

— Que des + + + : 1 groupe
— + + + et + - - : 2 groupes
— + + + et + - - et - - - : n groupes
— les 4 : non-structuré

2 GAME THEORIE

Bon cette partie y'a full truc d'éco pol / micro du coup prépare toi à un max de fun !

		Your partner	
		Presentation	*Exam*
You	*Presentation*	90, 90	86, 92
	Exam	92, 86	88, 88

Généralement c'est un tableau comme ça où chaque joueur à plusieurs choix auquel il reçoit un certain *payoff*. Généralement le *payoff* ça va être un nombre et le joueur préfera avoir le plus gros *payoff* possible.

2.1 Termes

— *Stratégie* : Action que le joueurs va faire
— *Best response* : la réaction a un truc qu'il pense que l'autre va faire
— *Dominante strategy* : la meilleur réponse à toutes les stratégies de l'autre
— *Stricly dominante strategy* : le meilleur choix où quoi que fasse l'autre c'est plus avantageux de le prendre

2.2 Hypothèses

Pour garder des jeux équivalents on va poser une hypothèse sur les jeux qu'on va évaluer :

- Les joueurs ne comptent que sur le *payoff* (si le joueur est altruiste et tient compte des autres cela doit être mit dans son *payoff*)
- Tous les joueurs sont au courant de tout ce qui se passe dans le jeu
- Chaque joueur cherche à maximiser son profit (*payoff*)

En utilisant ces hypothèses on peut voir que le *payoff* de choisir *Presentation* est de 88% contre un *payoff* de 90% si on choisit *Exam*. Notre partenaire à le même choix et donc rationnellement les deux vont choisir *Exam* pour maximiser son *payoff*. Cependant on arrive dans une situation où les deux ont 88% au lieu d'avoir 90% (qui est possible). C'est ce qu'on appelle un équilibre de Nash.

2.3 Exemple d'équilibres

		Suspect 2	
		Not confess	*Confess*
Suspect 1	*Not confess*	-1, -1	-10, 0
	Confess	0, -10	-4, -4

Table 1 – Stratégie dominante (les deux choisissent de confesser)

Formalisation : Best response

		Player 2	
		T'	T
Player 1	S'		$(P'_1, \_)$
	S		$(P_1, \_)$

Table 2 – Best response pour player 1 si $P_1 > P'_1$
Strictly best response pour player 1 si $P_1 \geqslant P'_1$

2.4 Stratégie strictement dominante

		Player 2	
		T'	T
Player 1	S'	0.48, 0.12	0.60, 0.40
	S	0.40, 0.60	0.32, 0.08

Table 3 – Stratégie strictement dominante pour Player 1 (il choisira toujours S' car il se fait d'office plus dessus)

2.5 Nash equilibrium

Un équilibre de Nash est pour une stratégie où chaque joueurs a la meilleur réponse par rapport aux autres

		Player 2	
		C1	*C2*
Player 1	*C1*	a, b	e, f
	C2	c, d	g, h

Table 4 – P(C2, C1) est un équilibre de Nash si $a < c$ et $h < d$

2.6 Mixed Strategies

On ajoute la randomisation

		Player 2	
		Head	*Tail*
Player 1	*Head*	-1, +1	+1, -1
	Tail	+1, -1	-1, +1

Table 5 – Mixed strategy (aucun équilibre de Nash)

Pour le Player 1 on pose p sa probabilité de faire la stratégie H (p est compris entre 0 et 1)
Pour le Player 2 on pose p sa probabilité de faire la stratégie H (q est compris entre 0 et 1)

Le player 1 va donc analyser les choix de son adversaire : $-q + (1 - q) = 2q - 1$ et donc que $q = \frac{1}{2}$.
Le player 2 va pouvoir faire le même raisonement pour trouver que $p = \frac{1}{2}$.

Principe d'indiférence

Si Player 2 a tendence à beaucoup jouer H alors Player 1 aura tendance à jouer plus T (pour gagner)

2.6.1 Autre exemple

		Défense	
		Defendre Pass	*Defendre Run*
Attaquants	*Pass*	0, 0	10, -10
	Run	5, -5	0, 0

Table 6 – Autre exemple ($p = \frac{1}{3}$ et $q = \frac{2}{3}$)

Equilibre paretto optimal

Equilibre ou aucun acteur peut améliorer sa situation sans dégradé celle de l'autre

Attention

Un équilibre pareto optimal n'est pas nécessairement un équilibre de Nash (et l'inverse)

		Player 2	
		C1	*C2*
Player 1	*C1*	90, 90	86, 92
	C2	92, 86	88, 88

Table 7 – (88, 88) est un équilibre de Nash tandis que les 3 autres options sont des équilibres pareto optimaux

2.7 Dynamic Games

Jeux qui se jouent sur le temps

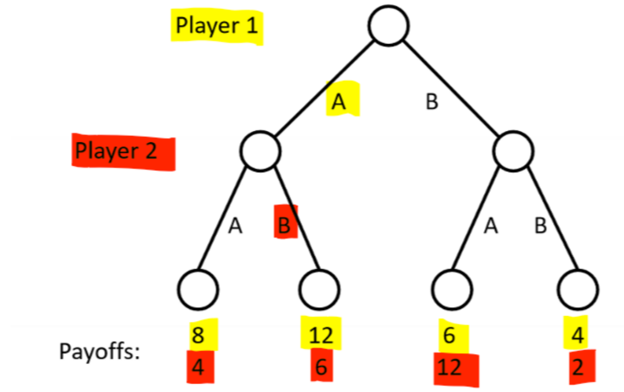


Figure 9 – Exemple de dynamic game

		Firm 2			
		AA,AB	*AA,BB*	*BA,AB*	*BA,BB*
Firm 1	*A*	8, 4	8, 4	12, 6	12, 6
	B	6, 12	4, 2	6, 12	4, 2

Figure 10 – Exemple de dynamic game (forme normale)

Attention la forme normale donne le résultat final et non pas un choix sur l'évolution des états.

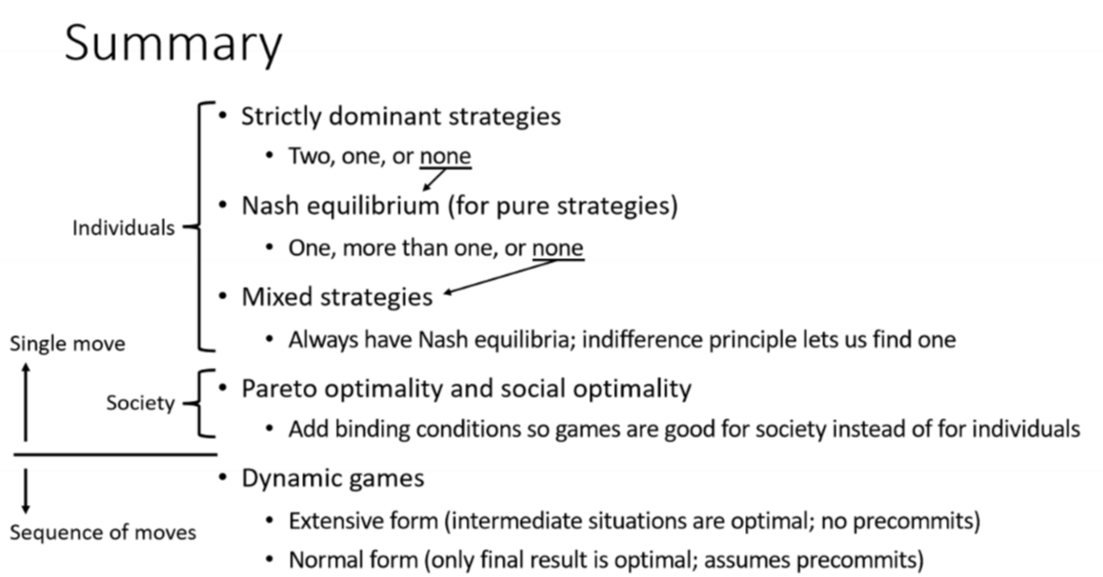


Figure 11 – Résumer de tout

3 NETWORK TRAFFIC

Attention, ça va aller vite.

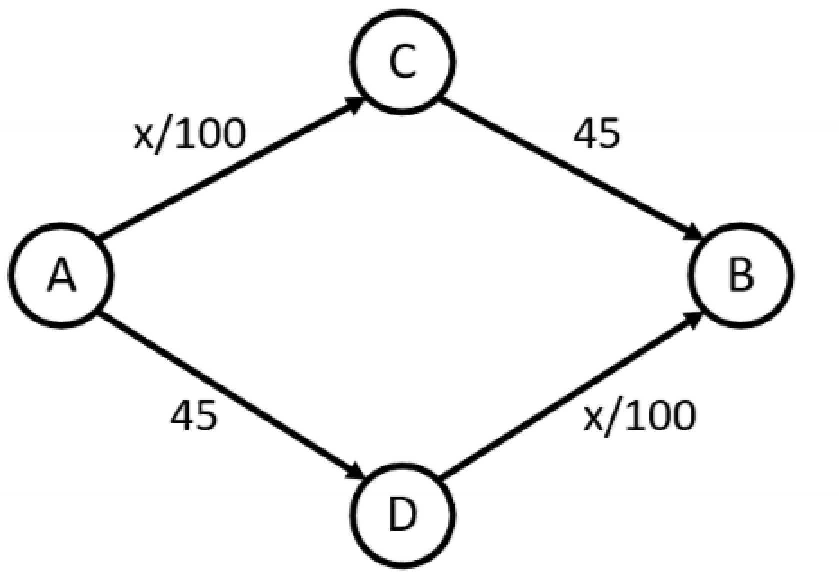


FIGURE 12 – Ici le mieux c'est de faire moite-moite

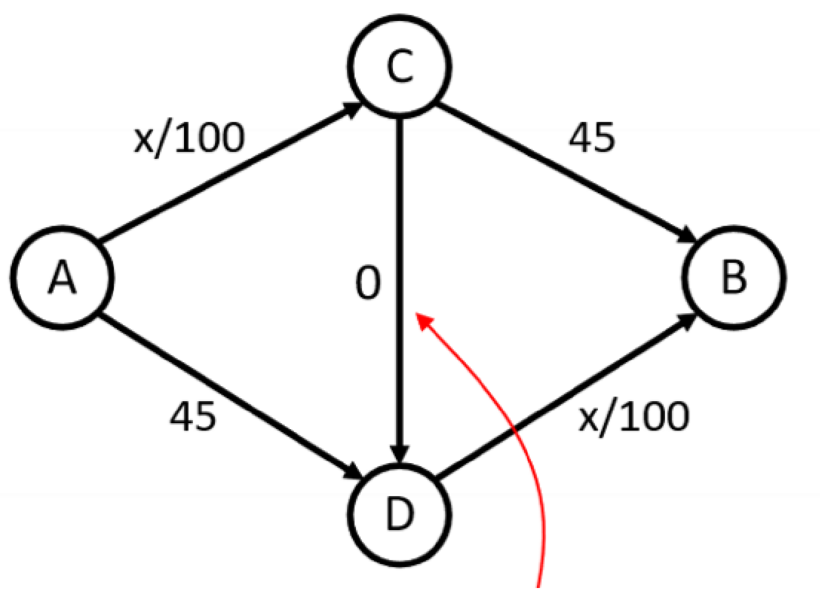


FIGURE 13 – Si on ajoute une route en plus, si elle a pas de coût les gens vont vouloir la prendre alors que ça fait de la merde au niveau général

Fin. (c'est le paradox de Braess si jamais)

4 ENCHÈRES

Bidder's true value est la valeur max que la personne est prête à acheter un bien (noté v_i plus tard avec i qui représente la personne)

4.1 Type d'enchères

- **Ascending-bid auctions (English auctions)** On monte le prix graduellement et les gens se retire au fur et à mesure.
- **Descending-bid auctions (Dutch auctions)** On descend le prix graduellement et le premier qui accepte le prix à le bien.
- **First-place sealed-bid auctions** Les bidders envoi tous leur bid et le plus offrant a le bien (et paye son prix)
- **Second-place sealed-bid auctions** Les bidders envoi tous leur bid et le plus offrant a le bien (et paye le prix du 2^e)

4.2 Similarité entre les types

Descending-bid et First-place auctions

Dans chacun, le bidder achète à son prix maximum le bien.

Ascending-bid et Second-place auctions

Dans chacun, le bidder achète au prix du deuxième.

4.3 Second price auction theorem

Le théorème dit que si tu fais une enchère de second price auction le prix b_i que tu bid doit être égal à v_i. Et donc le payoff est égale à $v_i - b_j$ (avec j qui représente le deuxième). (Si on perd l'énchère, on compte que le payoff est de 0)

Preuve

- Si on mise b_i' t.q. $b_i' > v_i$: $\mathtt{Si}\ \exists b_j' \mathtt{tq}\ b_j' > v_i' \implies \mathtt{payoff} = v_i - b_j' < v_i - b_j (= 0)$
- Si on mise b_i'' t.q. $b_i'' > v_i$ $\mathtt{Si}\ \exists b_j \mathtt{tq}\ b_j > b_j'' \implies \mathtt{payoff} = 0 \leqslant v_i - b_j'$

4.4 Winner's curse

C'est quand tu achète un truc pour le revendre mais que t'arrive pas à le revendre aussi cher que quand tu l'as acheté (bref tu l'as dans le cul sorry)

5 MATCHING MARKETS

Ici le but c'est de match non pas toi sur Tinder (laisse tomber ça sert à rien). Mais a match des buyers avec des sellers.

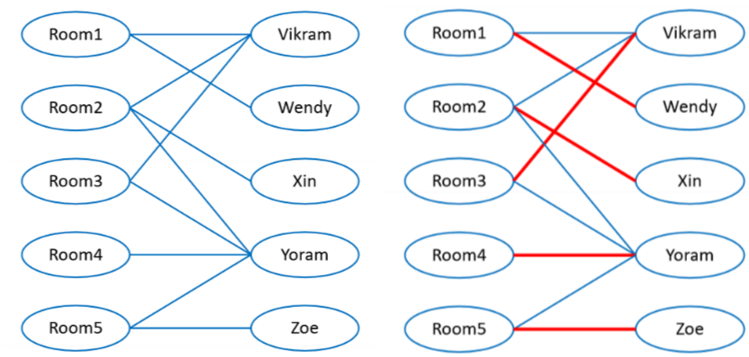


FIGURE 14 – Exemple de perfect matching

Ici on a à gauche les Room (les biens) et à droites les personnes (acheteurs).
Un *perfect match* c'est quand tout le monde match et que personne n'a un sad-cupidon (si t'as la ref courage soldat). Et qu'aucune des nodes à gauche n'est assigné a plusieurs nodes sur la droite (comme en blind date c'est relou si la meuf va voir ailleurs quand t'as match avec) => **Bijective**

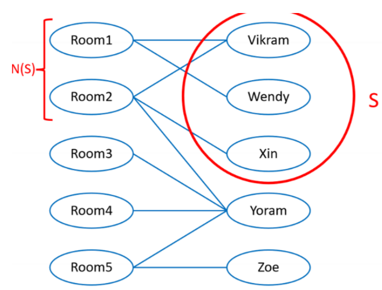


FIGURE 15 – Exemple de constricted set

Un *constricted set* c'est un set qui prouve qu'il ne peut pas y avoir un *perfect match*. Ici on à {Vikram, Wendy, Xin} (3) pour 2 rooms. On appelle le sets de nodes S et le set de neighbor N(S).

$$\text{S est un set constricted si } |N(S)| < |S|$$

Théorème (bullshiting) de Köning : *Si un graph bipartie avec un nombre égal de nodes à gauche et à droite n'a pas de perfect match, alors il doit contenir un constricted set*

5.1 Matching market with prices

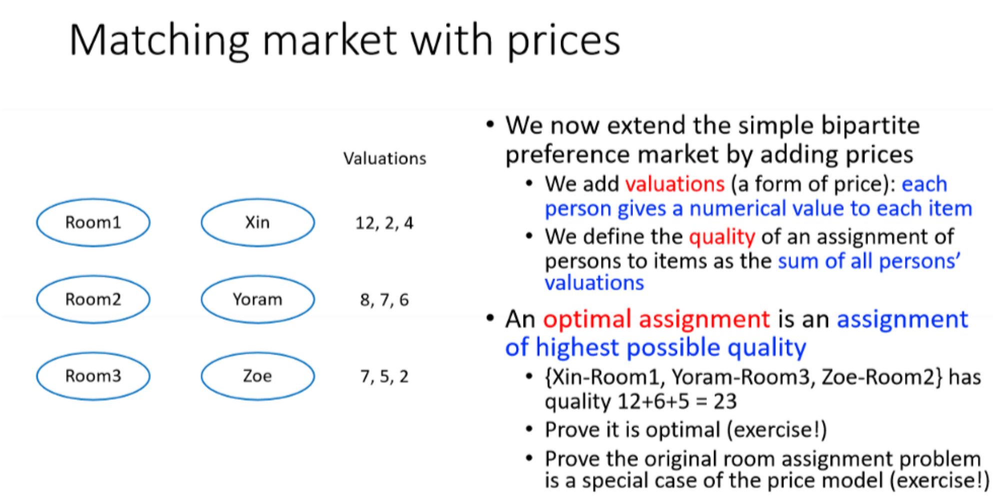


Figure 16 – La slide est bien faite

Maintenant qu'on a ça (ça pose la base), bah va falloir faire le ménage et voir comment on arrange tout ça en pratique ! On va juste introduire qqe notations parce que... Bah... C'est joli je suppose.

— Chaque vendeur i a une maison à vendre et chaque acheteur j veut acheter une maison

— Chaque acheteur j évalue la maison de i a un prix v_{ij} ($v_{ij} > 0$)

— Chaque vendeur i met un prix p_i a sa maison ($p_i \geqslant 0$)

— Le payoff de l'acheteur j est $v_{ij} - p_i$ (si il achète, sinon 0)

— Le payoff du vendeur i est p_i

— Tous les agents cherchent à maximiser leur payoff

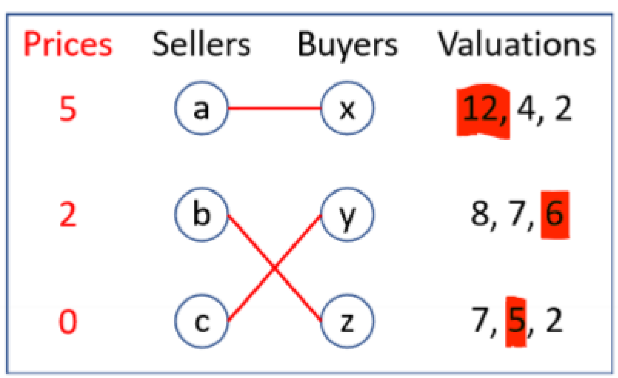


Figure 17 – Exemple d'un marché avec prix

Ici les payoffs :

— X

  — A : 12-5 = 7

  — B : 4-2 = 2

  — A : 2-0 = 2

— Y
  — A : 8-5 = 3
  — B : 7-2 = 5
  — A : 6-0 = **6**
— X
  — A : 7-5 = 2
  — B : 5-2 = **3**
  — A : 2-0 = 2

Ici chaque buyer à une maison, on dit donc que c'est market cleaning.
Bon ici il parle pendant 2/3 slides que bah c'est cool mais brasser du vent wala mais non. Du coup tout ce qu'il faut retenir c'est que si c'est market cleaning => socialement optimal. Càd si tu sommes tous les payoff t'auras le truc le plus élevé.

5.2 Construire les prix (pour avoir un truc market clean)

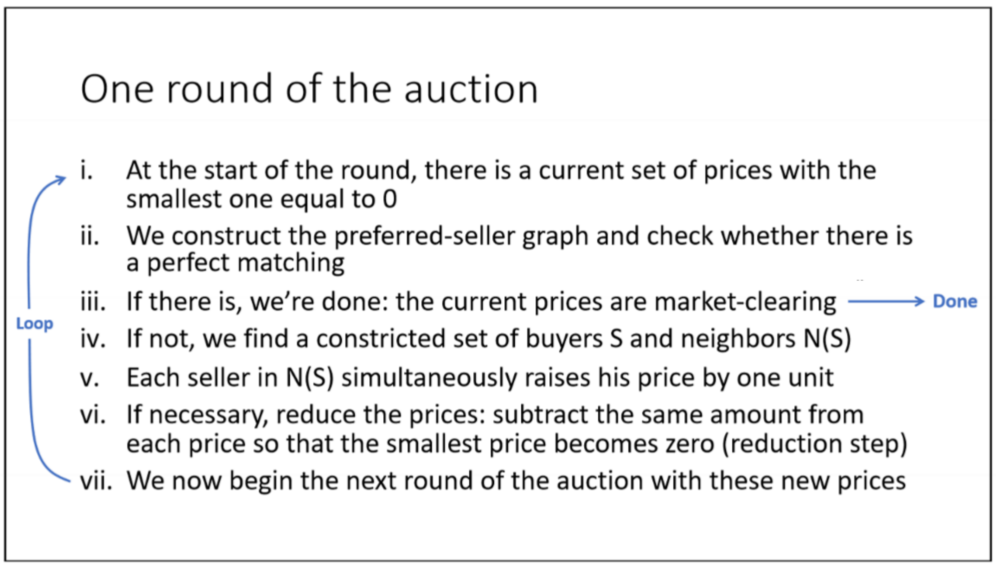


Figure 18 – Pseudo code du cours, il est bien fait pourquoi le changer ?
(y'a un bon exemple dans les slides, j'ai pas la fois de le copier coller ici)

Un autre truc à remarquer, c'est que celui qui paye le plus paye au prix du deuxième pour les trucs.

6 MARKETS AVEC DES INTERMÉDIAIRES

C'est comme le truc d'avant, sauf que pour "faciliter" (se masturber le cuir chevelu) le truc on va mettre un agent au milieu. Le point positif du truc c'est que le vendeur doit référé qu'a peu de gens pour vendre son bien et l'intermédiaire s'occupera de revendre (en se faisant de la moula). Le point négatif c'est que c'est vachement chiant et que l'intermédiaire va se faire full moulax.

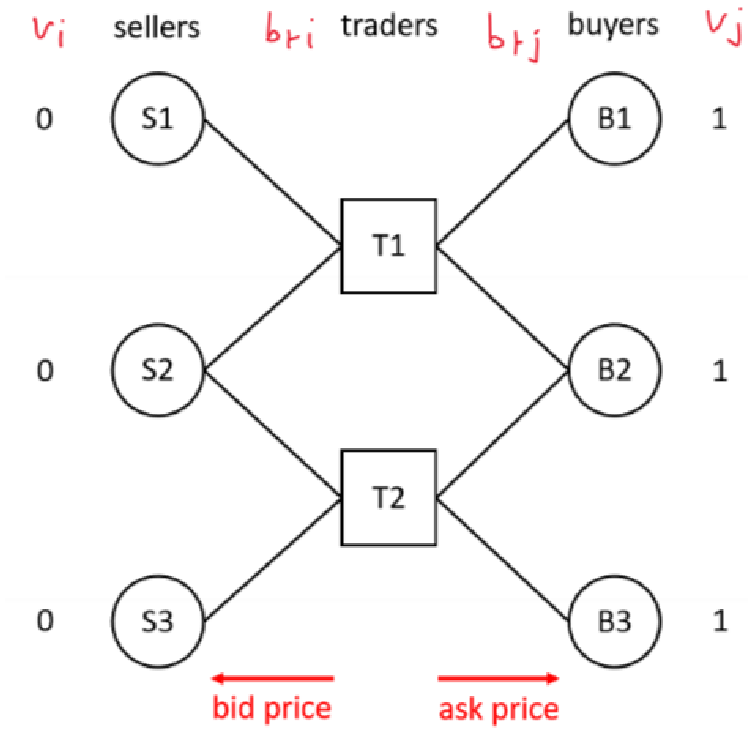


FIGURE 19 – Exemple d'un graph avec un intermédiaire

Eeeeeeeet salut à tous les amis, c'est le pavé notation qui est là !

— Chaque vendeurs i peut vendre son bien (une unité) à un prix $\geqslant v_i$

— Chaque acheteur j peut acheteur un bien (une unité) à un prix $\leqslant v_j$

— Chaque intermédiaire t peut placer un bid price (prix d'achat) au vendeur i a un prix b_{ti}

— Chaque intermédiaire t peut placer un ask price (offre) à l'acheteur j a un prix b_{tj}

Y'a aussi deux autres notions importantes, c'est le monopole et la concurrence parfaite. En gros ici T1 il a le monopole sur S1 du coup il va lui proposer le prix minimum (on va dire 0 par simplicité mais on est d'accord que c'est $0 + \epsilon$). T1 a aussi le monopole sur B1 et va donc lui vendre le bien au prix maximum (ici 1). Pour la concurrence parfaite, ici on aura S2 par rapport à T1 et T2 (et par rapport à B2). On avoir que le prix d'achat doit être égal a celui de revente pour essayer d'avoir le bien et de pouvoir le vendre. Les deux intermédiaires auront la même offre. Ils vont chacun proposer un prix x à S2 et proposer le bien à un prix x pour B2. (avec $0 \leqslant x \leqslant 1$).

Bon si t'as pas tout compris c'est normal mais faut juste réfléchir ça passe +-.

6.1 Second-price auction

Il montre que ça peut exister avec ce modèle.

6.2 Ripple effects

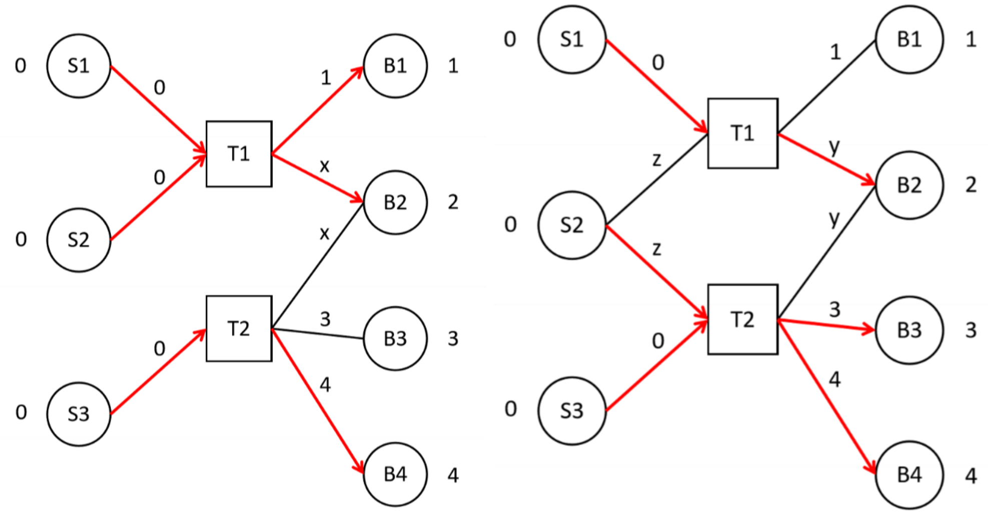


Figure 20 – Ripple effect

Ici on voit qu'en ajoutant un lien ça peut (très) vite partir en couilles. Dans ce cas T2 qui a des gros client pourra proposer un meilleur offre à S2 que pourrait le faire T1. On a donc T1 qui avait un monopole qui se fait toz et préfera vendre qu'a B2 à un prix de 2. Et B1 qui a rien a faire dans l'histoire pourra simplement rien acheté et gardera juste son seum.

Le *Social welfare* (bien être social) qui vaut la somme des tous les payoffs est égal à $(b_{ti} - v_i) + (a_{tj} - b_{ti}) + (v_j - a_{tj}) = (v_j - v_i)$ (le changement dans le ripple effect a passé la satisfactions de 7 à 9). On a donc qu'au plus le graphs à des edges, au plus la satisfactions générale est haute.

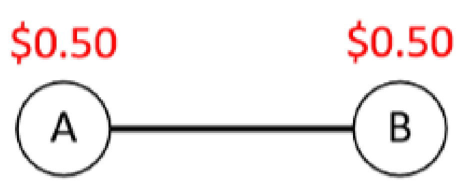
7 BARGAINING AND POWER IN NETWORK

Tu as toujours voulu dominer le monde et te faire full moulax ? Bah cette section sert à rien !

7.1 OMG EXPERIENCE SOCIAL CA TOURNE MAL EXPLICATIONS

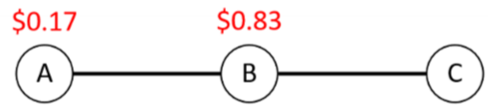
Ici on va poser des hypothèses super théorique et comparer ça a des expériences réalisées sur des humains. L'idée : on a un graph où chaque node est un humain qui, chaque tour, peut se partager uniquement avec un de ses voisins 1$.

7.1.1 *Two node path*



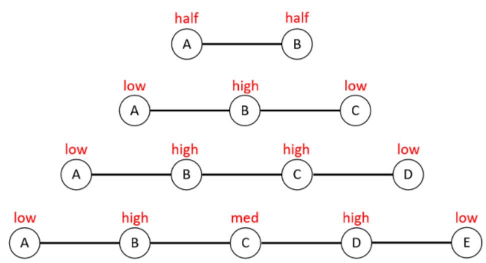
Rien de compliquer, les gens n'ont pas de supériorité par rapport aux autres du coup ils vont faire moite-moite.

7.1.2 *Three node path*



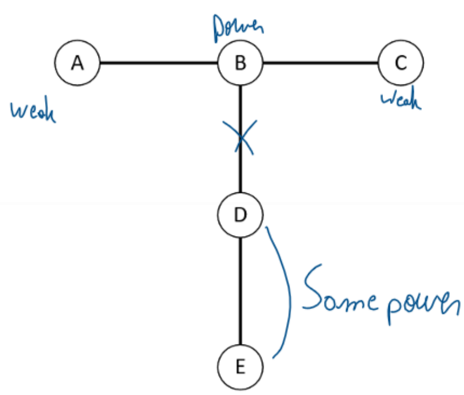
Ici, B doit choisir d'exclure un de ses voisins. En théorie chaque tour, celui avec qui il a pas tradé va lui proposer moins jusqu'à atteindre une limite de 0 ($+\epsilon$). Mais en pratique à partir d'un seuil l'autre ne va pas proposer moins.

7.1.3 *Résumé*



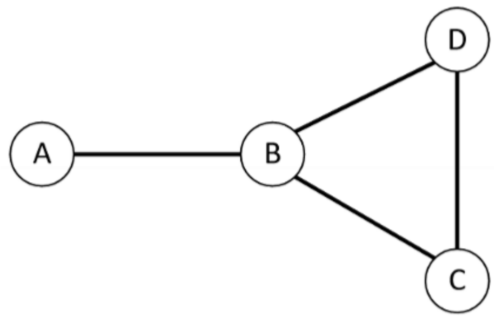
C'est cool.

7.1.4 *Graph hanté*



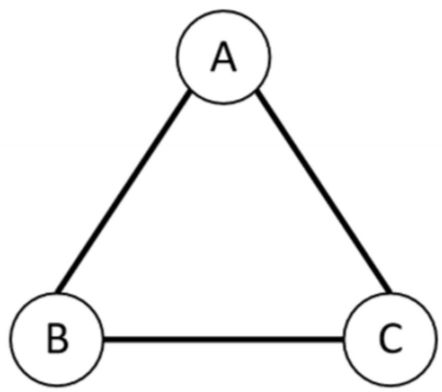
B domine A et C. Du coup D il a 0 intéret de négocier avec lui et B non plus. On aura donc D qui a la même puissance que E.

7.1.5 *Stem Graph*



C et D même poweer et B légèrement plus puissant que A.

7.1.6 *Triangle*



Juste le zbeul, y'a pas de convergence ni rien.

7.2 Nash bargaining solution

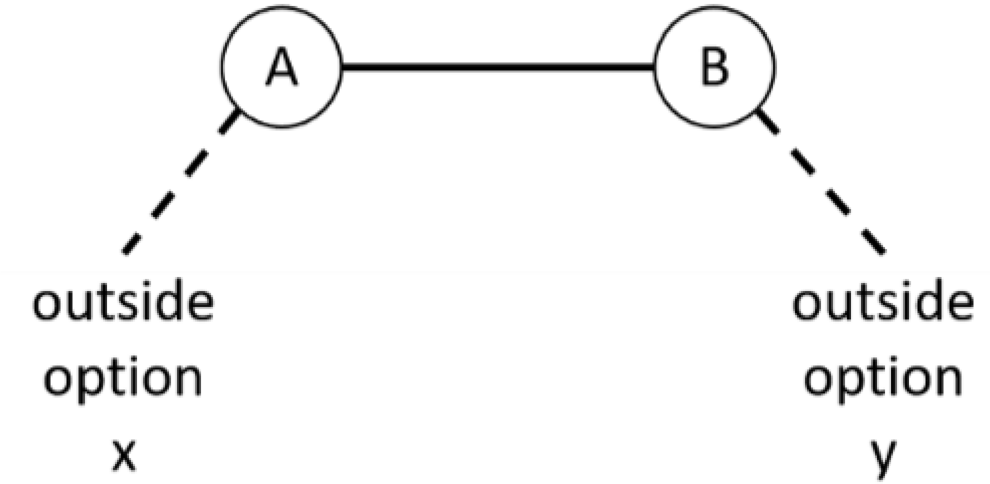


FIGURE 21 – Nash bargaining

Ici A et B se négocient 1\$. A a un choix externe x et B un choix externe y. On compte que $x + y \leqslant 1$ (sinon y'a pas de trade c'est sûr).
A doit avoir $(x + 1 - y)/2$
B doit avoir $(y + 1 - x)/2$

7.2.1 *Ultimatum Game*

Proposer "je prend 1\$ et toi t'as rien" en modèle ça peut marcher, dans la vrai vie nope.

7.2.2 *Stable outcomes*

- Definition of a stable outcome: An outcome of a network exchange is stable if and only if it contains no instabilities
- Definition of an instability: Given an outcome consisting of a matching and set of values, an instability in this outcome is an edge not in the matching, joining two nodes X and Y, such that (X's value) + (Y's value) < 1

(ça se voit que j'ai eu la flemme ?)

7.2.3 *Balanced outcomes*

Balanced outcomes = Nash bargaining solution.

8 STRUCTURES OF THE WEEBS

Si tu veux charger ton tel tu le branche à une mini éolienne que tu mets dans ta chambre en mattant le cours. C'est digne de la LSM niveau brassage de vent. Mais les trucs utiles (je parle dans le cadre d'un exam de Van Roy hn pas de la vie normale).

Le web => directed graph

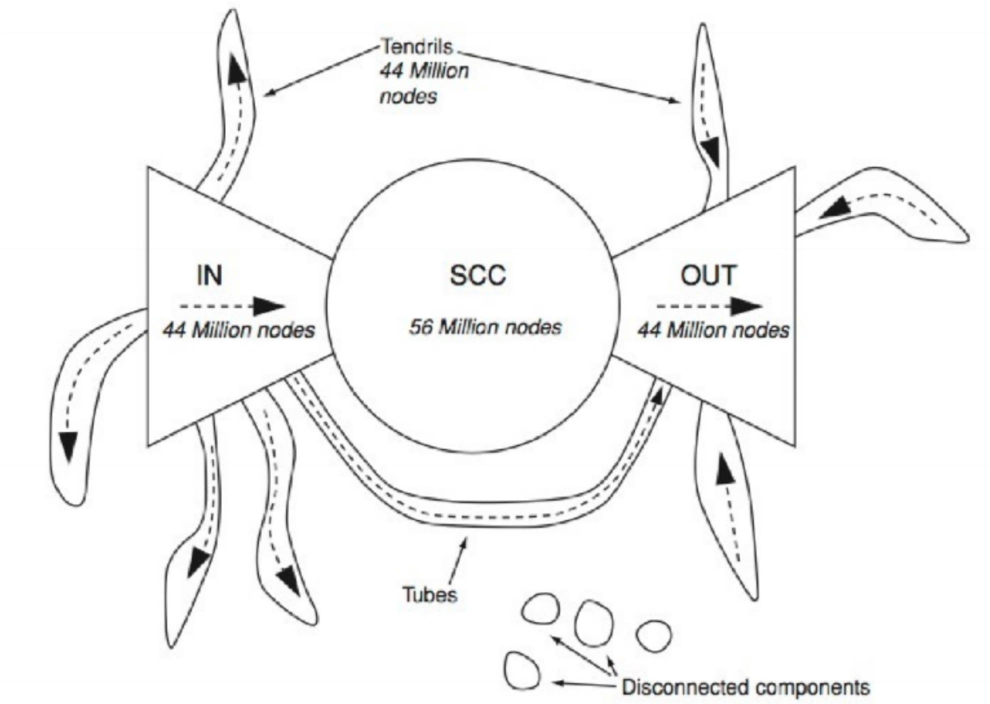


Figure 22 – Strucure du web avant 1999 (ok boomer)

Ce truc qui ressemble plus à un monstre d'un mauvais hentai (fais pas genre t'y as pas pensé) reprénsente le web. Il est assez explicite et son petit nom c'est le *Bow-tie*.

8.1 weebs2.0

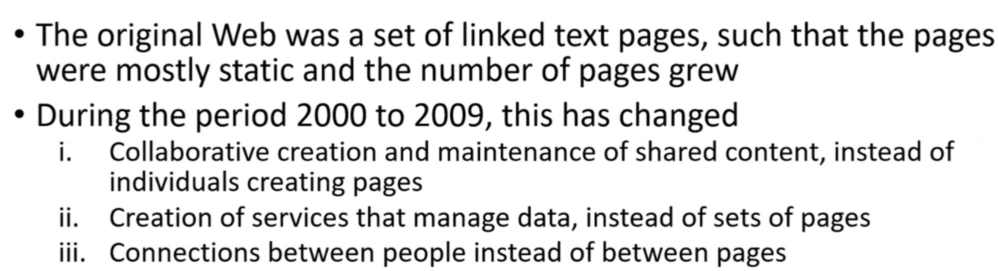


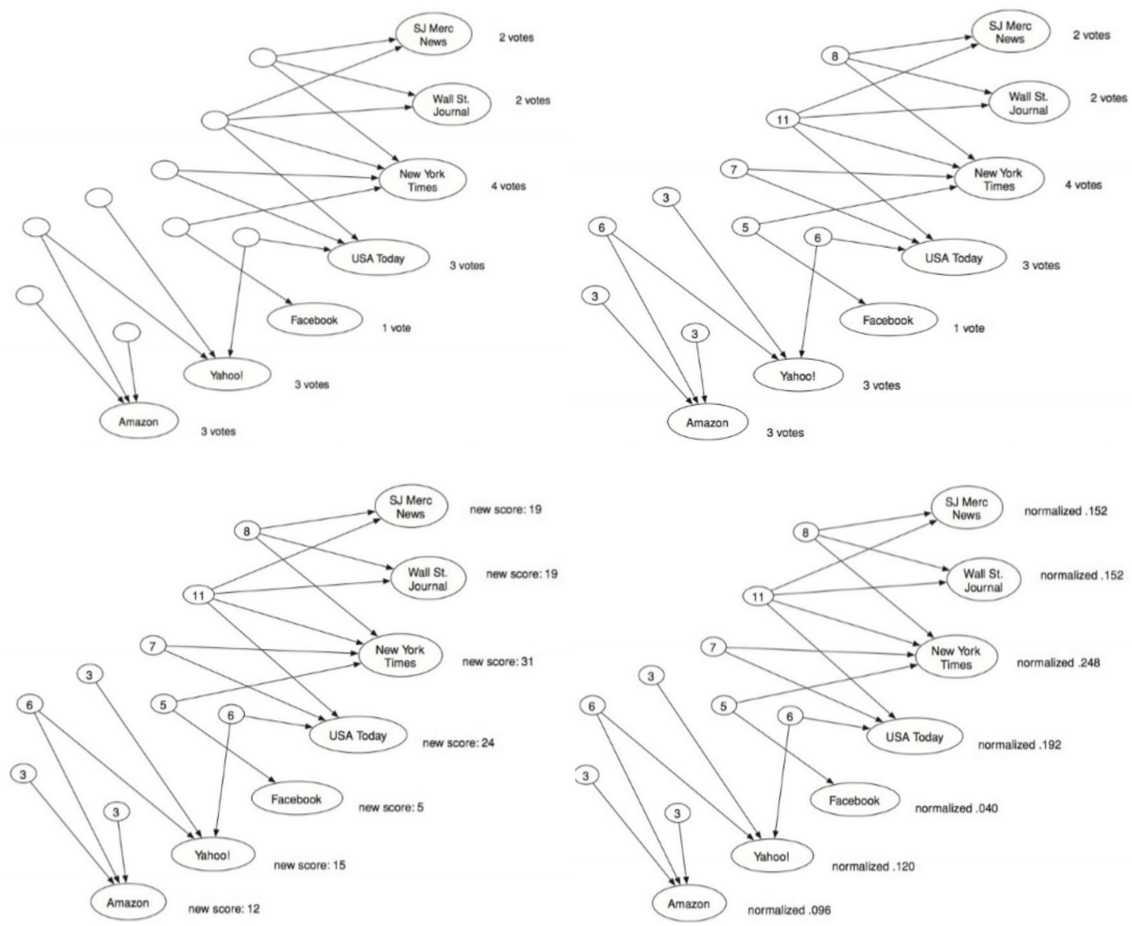
Figure 23 – Je suppose tu l'as bien compris mais balek

8.2 Problème de la recherche

(blablabla)

8.2.1 Solution naïve

Mieux vaut sortir les slides du cours crop que de me faire chier à expliquer.



Wala, t'as pas compris ? Bah tant pis !
En bref c'est une idée basique qui permet d'aller vers PageRank POG !

8.2.2 PageRank

Avis personnel : Le truc était au projet du coup il va pas le poser à l'exam

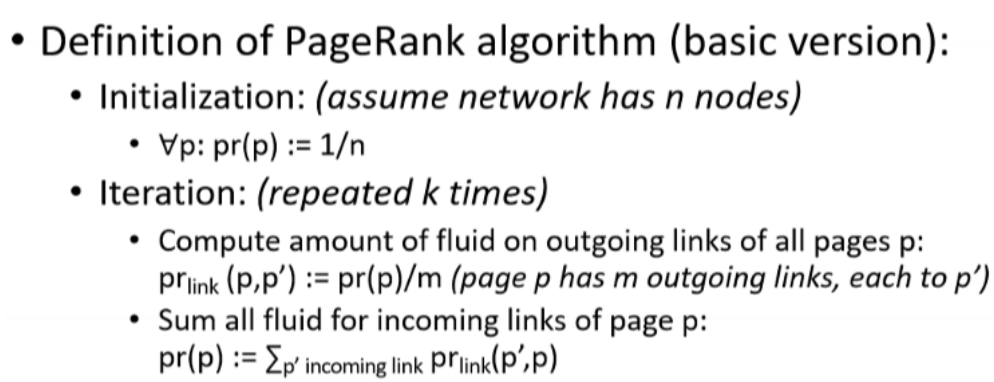


FIGURE 24 – Pourquoi je me fais chier à expliquer quand y'a une slide claire déjà ?

Dans le cours il parle que t'as une quantité d'eau qui tu distribue à chaque itération. Bah ça résume bien en fait.

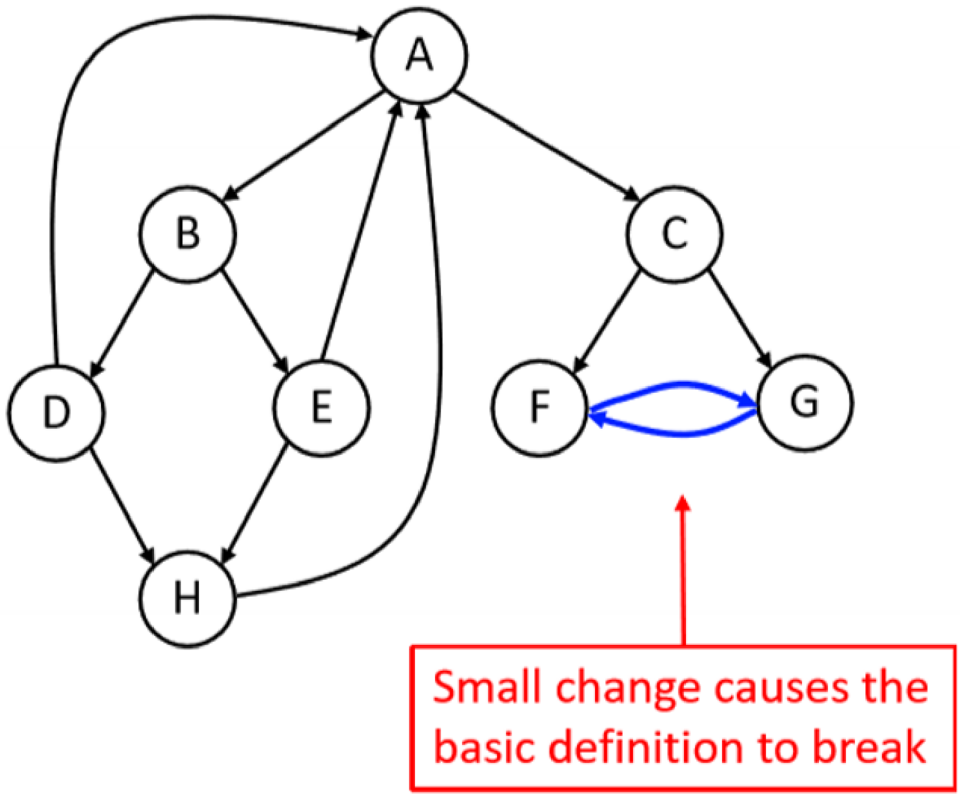


FIGURE 25 – Problème de la cuvette

Quand on verse de l'eau à F ou G ils vont jamais la rendre et juste l'amasser un MAX. Solution ? Faire évaporer la flotte. On va donc faire tout comme avant sauf que $\forall p : pr(p) := s * pr(p) + (1-s)/n$ avec $0 \leqslant S \leqslant 1$. (généralement S est entre 0.8 et 0.9)

8.2.3 Random Walk

Allez Billy, mets tes chaussures de random on va faire une *petite* balade.
(Pour cette partie on va commencer à voir avec un graph non dirigé parce que le prof le fait alors je le fais) L'idée c'est qu'on va mettre un type qui se balade de façon aléatoire sur le graph et c'est lui qui va déterminer le pagerank. A chaque node il a une change 1/d(v) d'aller chez un voisin. Un moment à force d'itérations il va converger vers un résultat. Il va converger en

$$\pi(v) = \frac{d(v)}{2m}$$

Avec :

— *d(v)* : le nombre de voisin de (v)

— *m* : le nombre d'edges

On a donc : Au plus t'as de voisins, au plus t'as de chance d'être visité (logique).
Types de matrices

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix} \qquad D = \begin{pmatrix} 1/2 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 1/3 & 0 & 0 \\ 0 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 \end{pmatrix}$$

$$M = \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 \\ 1/2 & 0 & 0 & 1/2 & 0 \\ 1/3 & 0 & 0 & 1/3 & 1/3 \\ 0 & 1/3 & 1/3 & 0 & 1/3 \\ 0 & 0 & 1/2 & 1/2 & 0 \end{pmatrix}$$

A : Adjency matrix
D : Diagonal matrix with $D_{i,j} = 1/d(i)$
M : Transition (random walk) matrix

Propriété de l'adjency matrix
A_{ij} vaut 1 s'il existe un lien entre i et j (sinon 0). On a donc un 1-hop path. Pour avoir un 2-hop paths il suffit de prendre A^2. Attention quand on dit path on veut en fait dire walk (c'est un path mais on peut revenir sur une même node).

On peut définir un vecteur $v = (1\ 0\ 0\ 0\ 0)$. Avec $v * A^t$ (avec $t > 0$), on a le même résultat qu'avec un random walk.
Il sort un peu de nul part que v est un eigenvector (coucou le Vietnam) et que son eigenvalue est $\lambda = 1$. Il sort (toujours de nul part) qu'un graph avec un degré d (si le graph contient un seul component) à comme eigenvector $(d, d, \dots, d)$ et comme eigenvalue $\lambda = d$. Bon allez encore une notion inutile mais il a l'air de l'aimer. Le eigengap c'est $\lambda_1 - \lambda_2$ (Avec λ_1 et λ_2 les eigens value les plus grandes). Au plus ce truc tend vers 0, au plus le graph est connecté en 1 component.

Cas où ça marche pas

— *Si pas connecter* : si on a deux components et qu'on commence un random walk sur un des component, on a 0 chances d'arriver sur l'autre.
— *Si biparti* : Les nombres pair de pas on sera sur un coté et le nombre impaire sur l'autre.

Et pour les graphs directed ?
Il faut qu'il soit *strongly connected* et *aperiodic* :

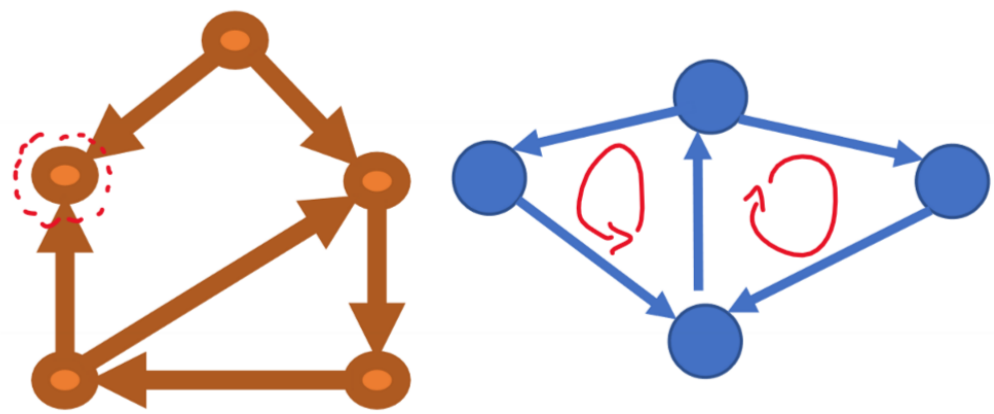
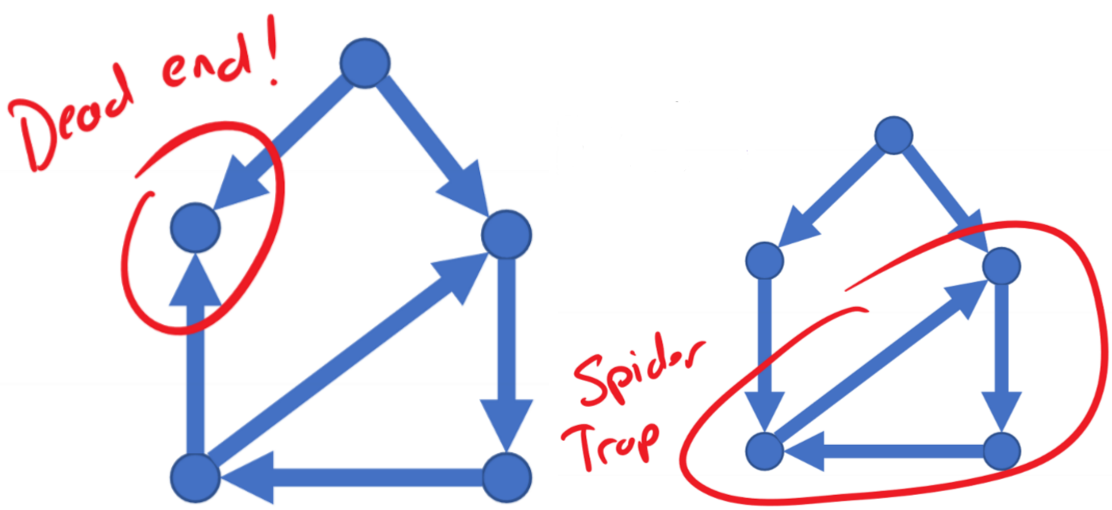


Figure 26 – Exemples de problèmes

8.2.4 Google PageRank

ENFIN !!!
L'idée de l'algo est de donner à chaque node un "budget de vote" qu'ils distribueront à leur out-links. Si la page r_j a n out-links, chaque lien recevra r_j/n votes.
Enemies de PageRank :



La solution de Google est de mettre des *tiny-links* entre toutes les nodes. C'est à dire une chance de se téléporter où qu'on soit. Avec une probabilité de β on utilisera un vrai lien et avec une probabilité de $1-\beta$ on se téléportera. (Généralement β vaut entre 0.8 et 0.9). Pour les *dead ends*, on va mettre 1β à 1.

$$M_{\text{pageRank}} = \beta \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 1-\beta & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1-\beta & 0 & 0 \\ 0 & 0 & 0 & 1-\beta & 0 \\ 0 & 0 & 0 & 0 & 1-\beta \end{pmatrix} \begin{pmatrix} 0 & 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 0 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 0 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 0 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 & 0 \end{pmatrix}$$

Sur l'image, on voit bien que la node sans out-links à un facteur de téléportation de 1.

Problème ?
C'est super lourd à stocker walla. Du coup au lieu de faire $r^{new} = r^{old} * M_{pagerank}$ (Is this a formule qui sort de nul part ? :pog :). On va faire $r^{new} = \beta * (r^{old} * M) + c$ avec $c = (1 - \beta)/N$

8.2.5 *Personalized PageRank et SimRank*

8.3 Web Spam

C'est l'idée d'essayer de niquer internet pour que les gens viennent vers ta page.

8.3.1 *Term spam*

C'est tout con, tu mets des mots caché sur ta page. Genre si tu vends des t-shirts, tu mets 1000 fois le mots "cinéma" sur ton site pour attirer les gens qui aime le cinéma.

Solution :
Mesurer ce que les gens disent de la page plutôt que ce que la page dit d'elle même. Le h@ck3r pourrait essayer aussi de faire 1000 pages qui pointent vers sa page sauf que bas ces pages n'auront aucun in-links du coup rip.

8.3.2 *Coordinated attack*

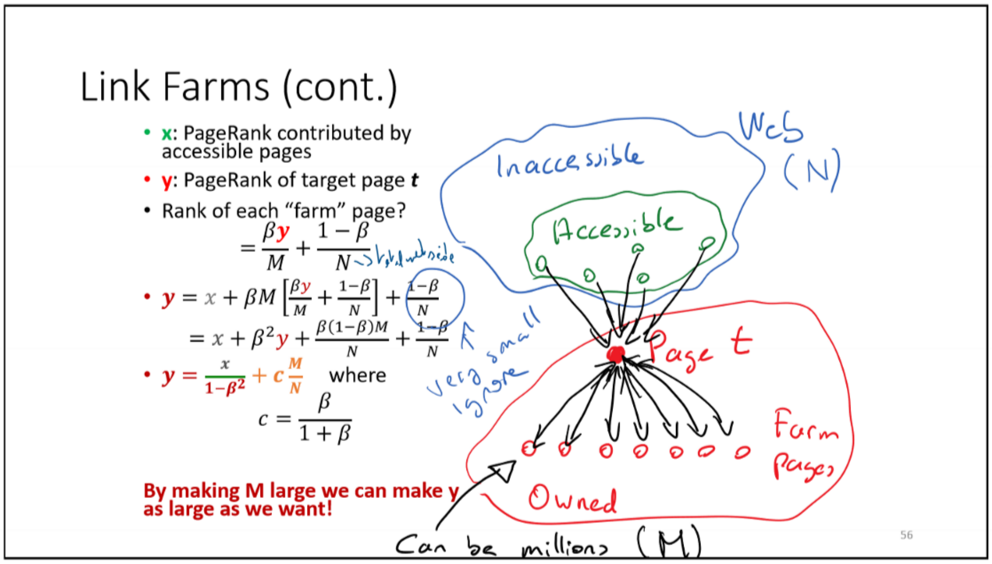


Figure 27 – Architecture

Les pages *accessibles* sont les pages publiques où les h@ck3rs peuvent poster des trucs (blogs, reviews, etc).

Solution :
Faire un TrustRank en utilisant uniquement des pages trusted. Comment choisir ?

1. **PageRank** : Utiliser PageRank pour sélectionner k pages trusted (on compte qu'il n'y aura que des pages trusted qui auront un high rank)
2. **Trusted domains** ex : .edu, .mil, .gov, ...

(C'est fini pour pagerank !)

9 NETWORK DYNAMICS

Comment le network influence les comportements.

9.1 Cascades

Mise en contexte : t'arrive devant deux restos. Y'en a un qui est full, un qui est vide. Tu choisis lequel ? T'as pas d'informations sur les deux mais juste tu vois les gens. Bah sale mouton de merde tu vas aller là où y'a full gens. (Selon la théorie)
Ce chapitre est giga long à suivre mais il dit pas grand chose du coup walla

Baye's Rule

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(B|A)}{P(B)}$$

Bon fin j'espère que t'as bien suivi stats parce que j'ai la masta flemme d'expliquer comme ça marche.

Alors en relisant le cours on se demande vraiment pourquoi Mr Roy n'a pas fait la LSM du coup voilà le truc important :

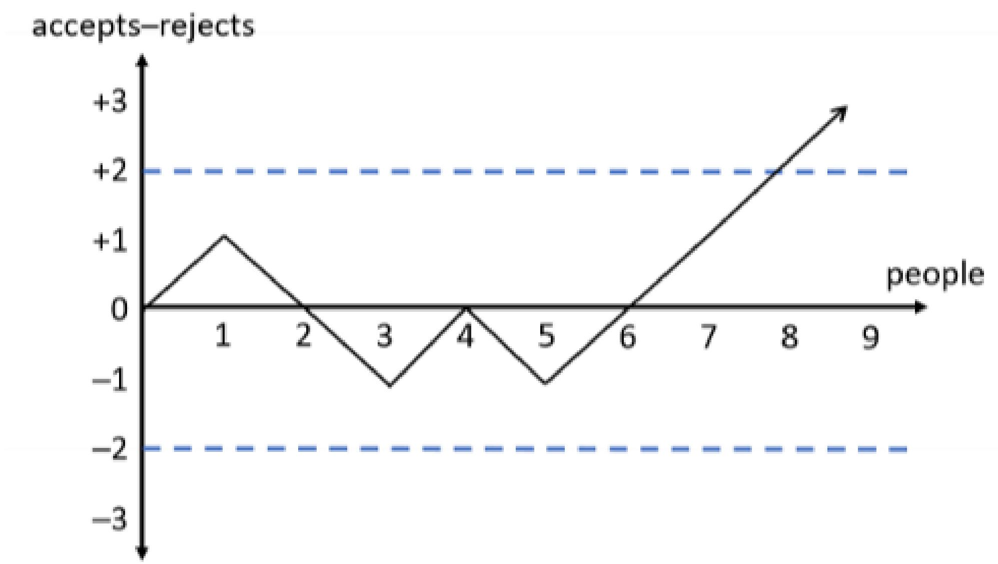


FIGURE 28 – Début d'une cascade

Quand on passe un seuil de 2 réponse les mêmes les gens qui arrivent par après auront tendance à suivre

Fragilité

La cascade est plus fragile que toi en soirée et c'est pour dire. Il suffit qu'a un moment t'ai deux zigotos qui suivent pas la masse pour inverser la tendance.

Attention

Ici on parle de choix successifs quand on observe ce que fait l'autre devant soit. Y'a des cas néfastes comme des cas bénéfiques.

9.2 Direct-benefit effects

Rappel d'éco-pol (Khatibi, celle là est pour toi)

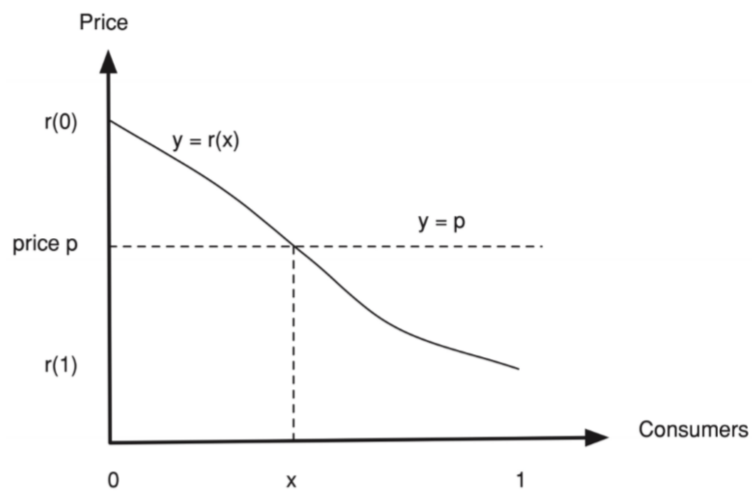


FIGURE 29 – Relation prix et demande

Sur l'axe des x on a une proportion des consommateurs. Par exemple, à 0.6 on aura 60% des consommateurs. La fonction r(x) donne le prix que sont prêt à mettre x pourcent des consommateurs. Cette fonction est décroissante, au plus c'est cher, au moins y'aura de consomateurs (logique).

L'optimum sera en p*, c'est à dire que r(x*)=p* avec p* le coût de production.

Et le network effect ?

WAW TU TE CALME DIRECT ! J'allais y venir.
Dans la vie réelle, on va tenir aussi compte de l'avis des autres. On va ajouter une fonction f(z) qui donne le payoff de savoir que d'autres gens consomme le même bien. Cette fonction est strictement croissante. (Au plus t'as de gens qui consomme au plus que t'auras envie de consommer) La formule va devenir : $p* = r(z) * f(z)$

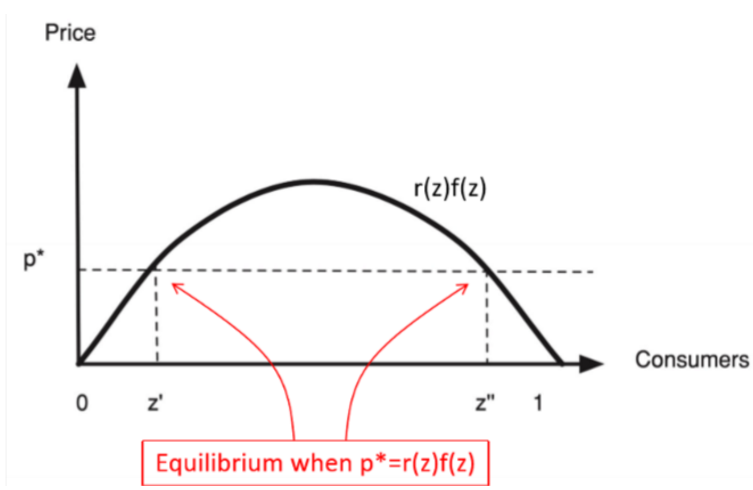


FIGURE 30 – Relation prix et demande et nombre de personne qui consomment

Equilibre

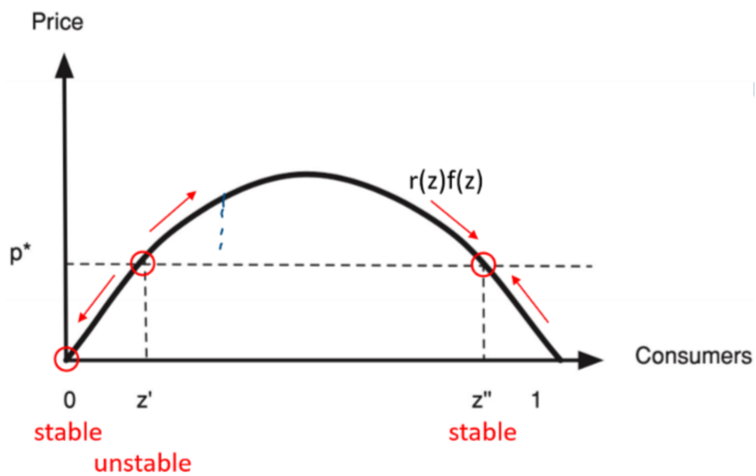


FIGURE 31 – Notion d'équilibre

— *En dessous du point z'* : les gens pensent que le produit est useless vu que y'a pas beaucoup de gens qui l'utilisent et que c'est cher
— *Entre z' et z"* : les gens sont hypé par le produit et auront tendance à plus en consommer
— *Au dessus de z"* : Il aime du coup downward pressure

Changer de p*

Si p* est baissé, z' bouge à gauche tandis que z" bouge vers la droite. On attendra donc le facilement le point où on va monter stonks.

9.2.1 Dynamic view of the market

Les consomateurs sont débiles du coup ils n'ont jamais la bonne vision de combien de gens consomment. On va noter $\hat{z}$ le nombre réel de consommateurs et z le nombre que les consommateurs pensent qui est bon. On a donc que $r(\hat{z})f(z) = p*$. En bougeant des trucs on a que $\hat{z} = r^{-1}(p * /f(z))$. Par facilité on va noté que $g(z) = \hat{z}$.

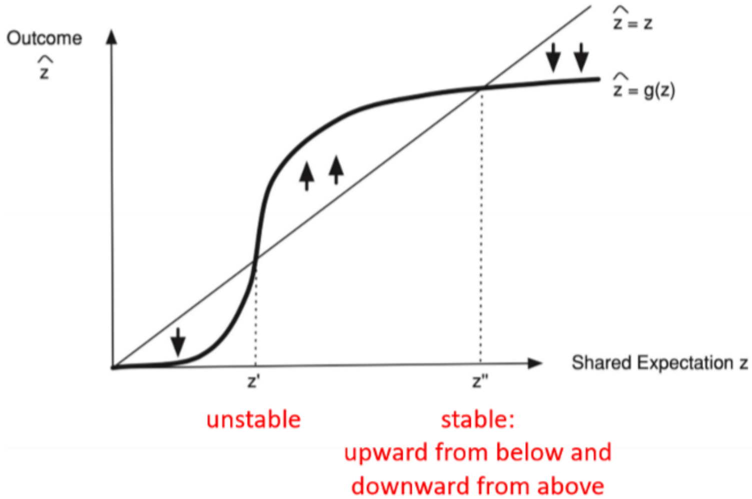


FIGURE 32 – fonction g(z)

Pour bouger ça va se faire en steps qui vont ressembler à ça

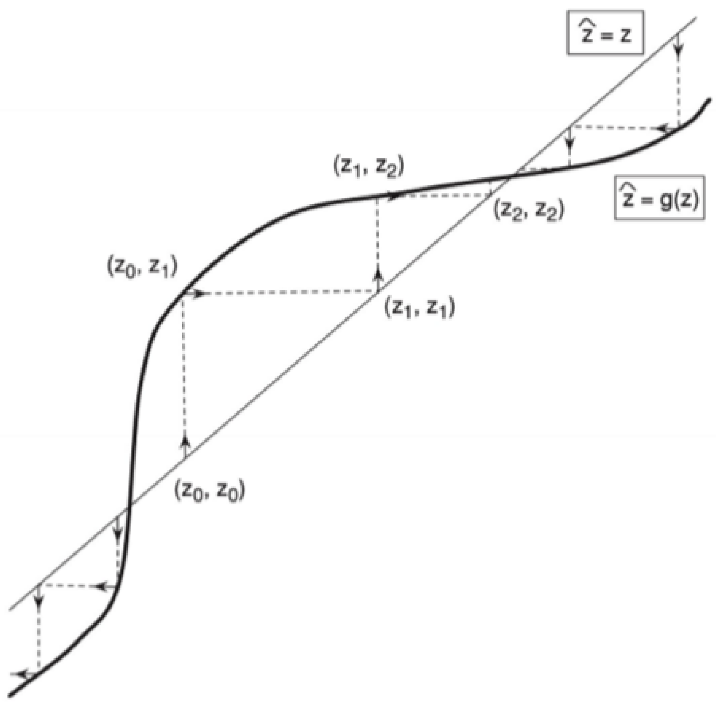


FIGURE 33 – Faut vraiment que je décrive ?

On voit bien qu'en dessous de z', on va aller vers 0, qu'entre z' et z″ on va aller vers z″ et qu'au dessus de z″ on va aller vers z″.

Et dans le monde réel ?

Les gens ont pas tous la même importance. Ainsi les firms auront intérêt à toucher des personnes clefs. J'en profite d'ailleurs pour remercier *Raid Shadow Legend* de sponsoriser cette synthèse.

Attention parfois le modèle peut faire de la merde en fonction de comment tu fais tes tarifs tout ça et ça peut donner :

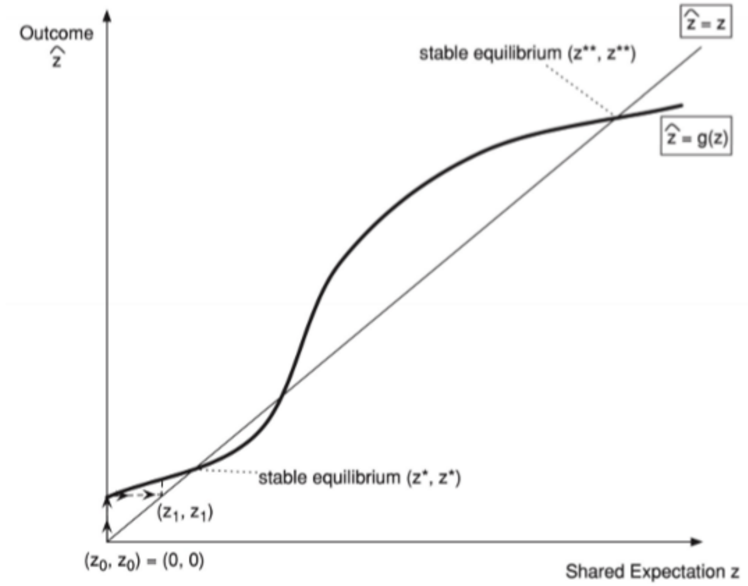


FIGURE 34 – Ici y'a un équilibre stable qui est bas du coup si t'y es c'est chiant

Solution :

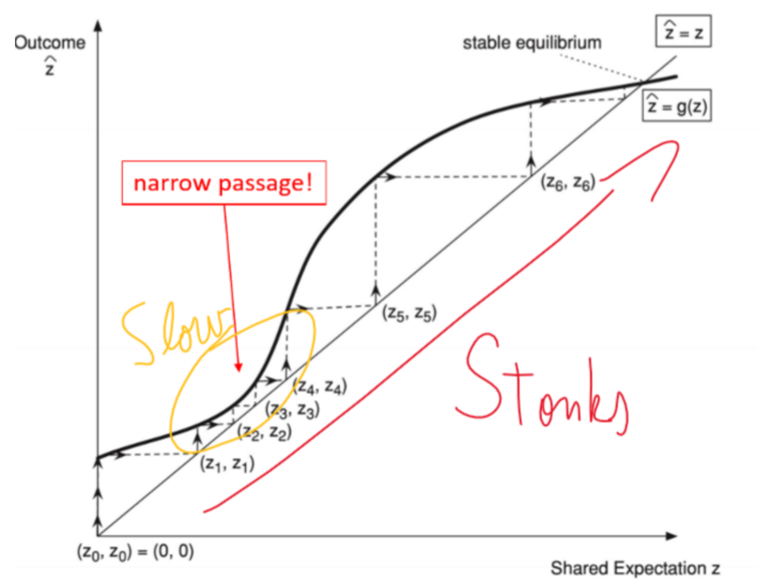


FIGURE 35 – La firm réduit son prix pour supprimer l'équilibre stable d'en bas

9.3 Power law

J'ai aucune idée de pourquoi il parle de ça ni à quoi ça sert.

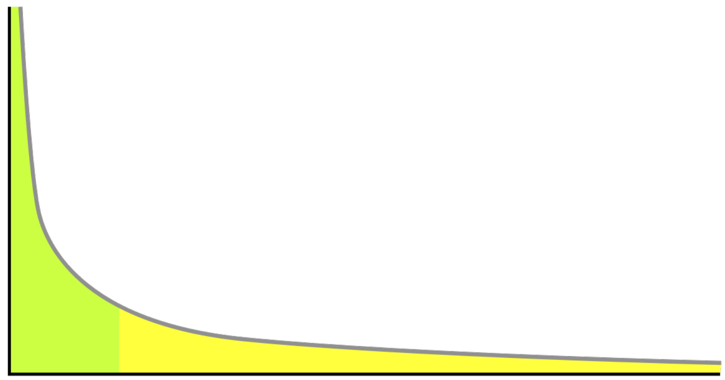


FIGURE 36 – Le vert et le jaune valent respectivement 50% de la population

On peut voir ce graphe comme le nombre de truc en fonction de sa popularité. Y'a full gens sur terre et pourtant y'a peut de star célèbres.

9.4 Long tail

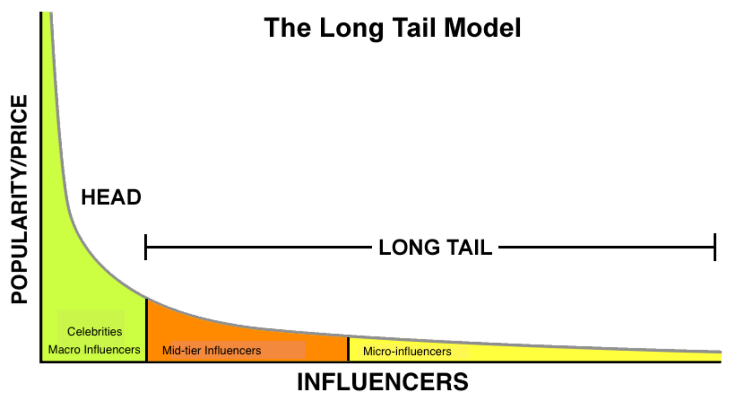


FIGURE 37 – Comme une impression de déjà vu

Pour moi c'est exactement la même que le power law mais comme le dit Mr Roy, c'est encore un sujet de recherche.

10 CASCADING BAHAVIOR IN NETWORKS

10.1 Diffusion of innovations

On va voir comment le choix des gens dépendent de leur voisins (ce cours t'apprend juste que t'es un mouton j'ai l'impression). Le process se fait en 2 étapes :

1. T'apprends le truc toi
2. Tu le montres à tes voisins

Exemple :

1. J'ai installé Linux et je trouve ça trop bien
2. Je vais répandre la bonne parole et dire au gens comment c'est bien comme truc

On passe sur des games theories. Imaginons t'as v et l'autre c'est w :

Attention

Un équilibre pareto optimal n'est pas nécessairement un équilibre de Nash (et l'inverse)

		w: A	w: B
v	A	a, a	0, 0
	B	0, 0	b, b

Table 8 – Dans le cas où vous êtes deux

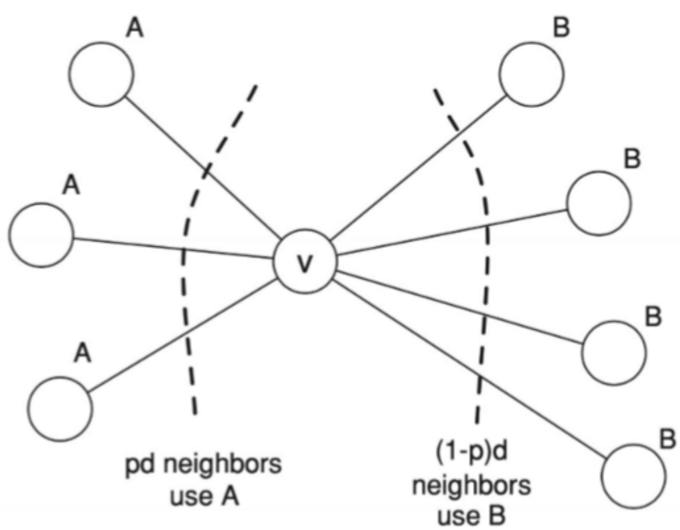


Figure 38 – Dans le cas où y'a plus de gens

— V a d voisins
— V a une fraction p de voisins qui choisissent A
— V a une fraction $(1-p)$ de voisins qui choisissent B
— Si V choisit A son payoff est de pda
— Si V choisit B son payoff est de $(1-p)db$

A est le meilleur choix si $p \geqslant \frac{b}{a+b}$ (=q)
Exemple

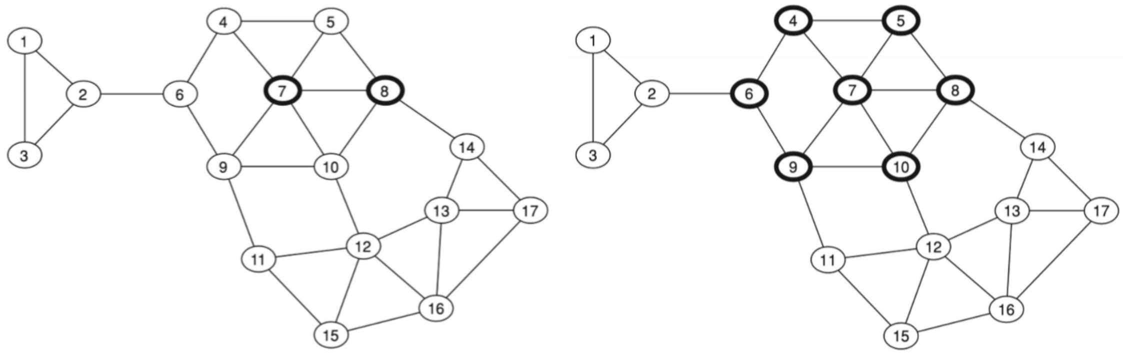


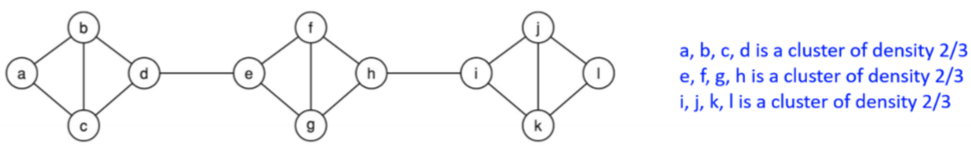
FIGURE 39 – Avec a=3 et b=2 (le threshold est donc de q=2/5)

Pour continuer à propager le choix A, on peut :

— Augmenter la qualité du produit (passer a=3 à a=4 et donc q=1/3)

— Convaincre des personnes cléfs (par exemple 12 ou 13)

10.2 Clusters



La densité d'un cluster de densité p est un set de nodes connectés pour lesquelles chacune à au moins une fraction p de ses voisins dans le cluster.

Pour l'exemple de la figure 39, les zones non "contaminées" sont des clusters de densité 2/3. Il y a donc au maximum 1/3 de pression pour passer à A. Or il faut que cette fraction soit supérieur au treshold (p).

10.3 Cascade ++

Bah c'est gentil ton truc mais dans la vrai vie chacun a sa propre opinion. Y'a des gens qui vont être influençable facilement (quand tu convaincs ton petit frère de faire une connerie a ta place) et d'autre tu auras giga dur (genre convaincre ton papy raciste que tout ses problèmes ne viennent pas des étrangers). On aura donc :

v \ w	*A*	*B*
A	a_v, a_w	0, 0
B	0, 0	b_v, b_w

TABLE 9 – Table à deux avec chacun ses préférence

A est un meilleur choix si $p \geqslant \frac{b_v}{a_v+b_v} = q_v$ (threshold personnel)

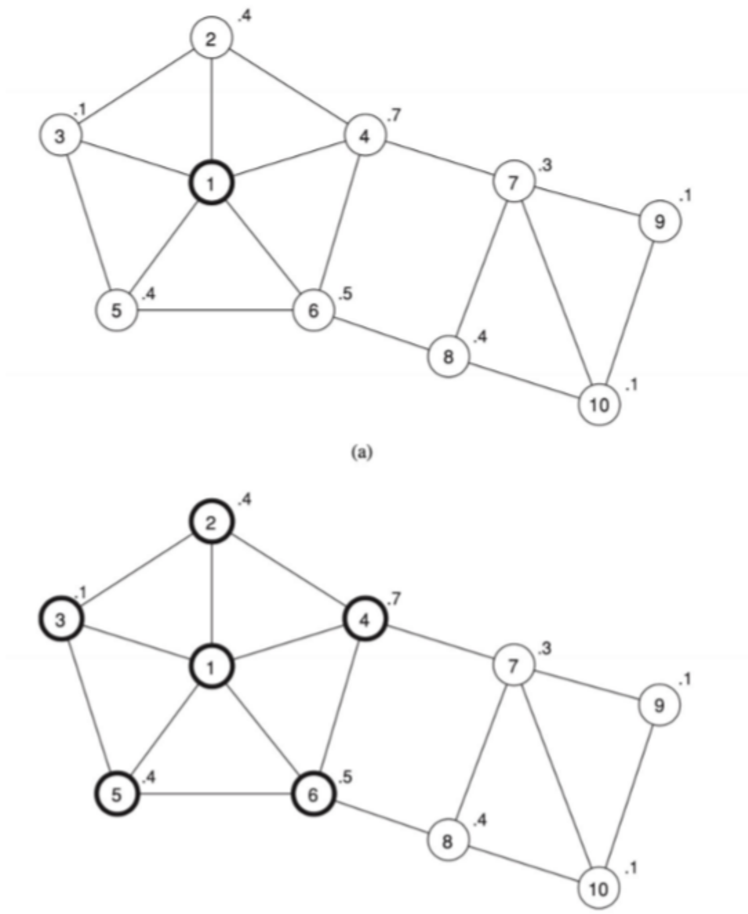


Figure 40 – Exemple avec plus de gens

Ici le cluster à droit va s'appeller un *blocking cluster* chaque node v a plus que $1 - q_v$ fraction de ses voisins dans le set

10.4 Renverser l'oppression

Dans un état autoritaire (coucou la Russie), le but des résistants est de s'allier ensemble pour tenté de renverser le gouvernement. Sauf que s'ils sont solo, ils vont se faire latter à coup de chars ou des trucs plus vénères.

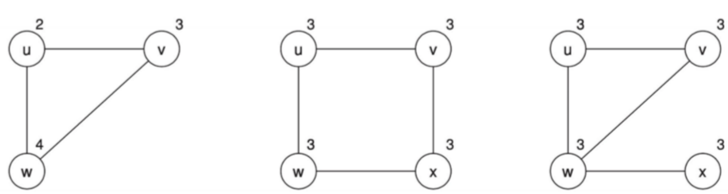


Figure 41 – Exemples

Chaque node a un numéro à partir du quel (avec lui comprit) il va manifesté.

1. Dans ce cas, w sait qu'il y a pas assez de gens du coup il se retire. Bah v il est pas con, y'aura pas assez de gens il vient pas. Et u bah il est tout seul il va pas risquer.

2. u sait que v et w ont le même threshold que lui mais sait pas s'ils se connaissent. Il va donc pas risquer que les deux soient pas coordonné. (ça va faire un effet cascade et donc personne va venir).
3. u sait que v et w on le même threshold. Sauf qu'il sait aussi qu'ils se connaissent. Il sera donc chaud balancer des pavés sur la police, v et w aussi.

On appelle les liens entre v, w, u (dans le 3^{e} cas) un commom knowledge.

11 TUYEAU

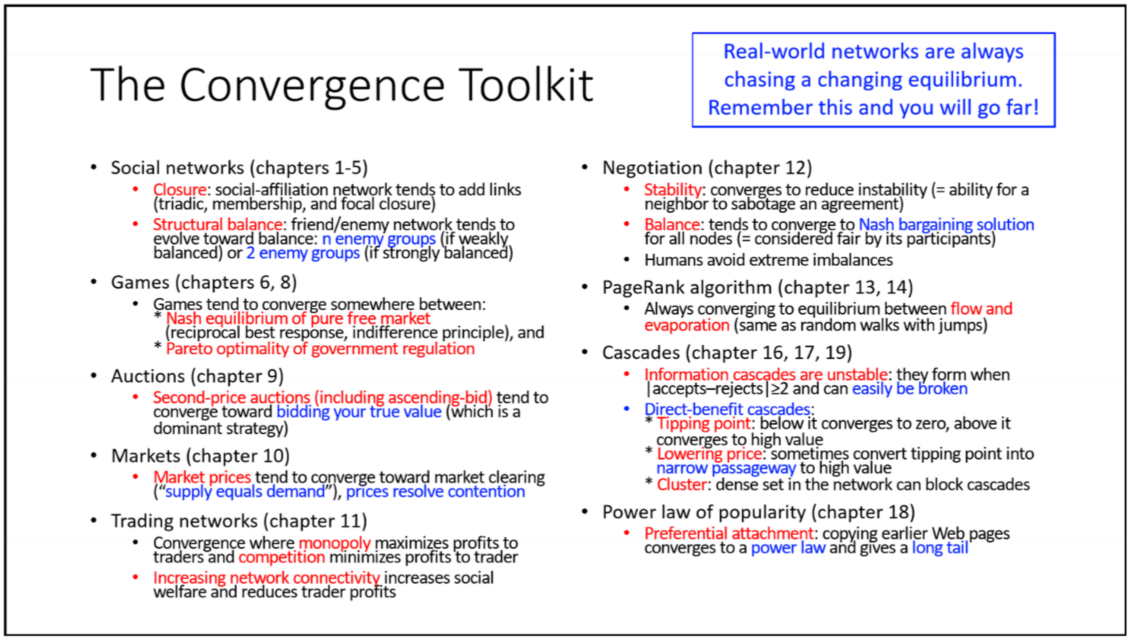


FIGURE 42 – Gros résumé signé Mr. Roy