

Optimization in short local supply chains

Sharing of deliveries between producers in local food logistic

Adrien CALLICO, Pierre DESPORT, Caroline PRODHON, Jean-Charles BILLAUT

ROADEF 2024



Summary

1. Context
2. Problem modelisation
3. Results
4. Perspectives



Context

Short food supply chains

- Geographic proximity and at most one intermediary
 - Growing demands for sustainable and high-quality products
 - Supported by government initiatives
 - Diversity of SFSC : AMAP, farmer's markets, baskets ...
-
- Objective: to promote a sustainable and resilient food system
 - Challenges: coordination, quality assurance, greenhouse gas emissions...

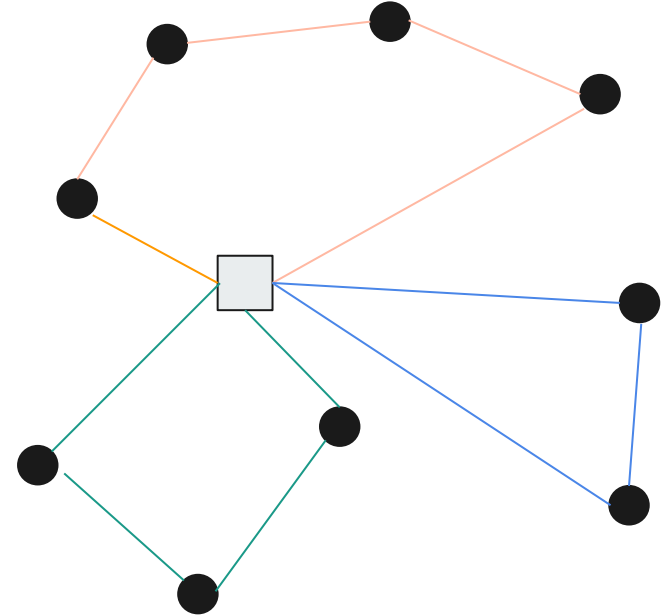




OR & SFSC

- Location of platforms/warehouses
- Inventory management, demand forecasting
- Delivery planning and organization

Vehicle Routing Problem (VRP)





Existing works on delivery planning and SFSC

PhDs :

- M. Ogier (2013) *“Contributions to digital supply chain : design of short and local supply chains and decentralized planning”*
- W. Gu (2019) *“Multiple commodities routing problems with applications in the local fresh food supply chain”*

ROADEF

- Bayir et al. (2022) *“Multi-Depot Vehicle Routing Problem with Branch-and-Cut: An application in Short Food Supply Chains”*



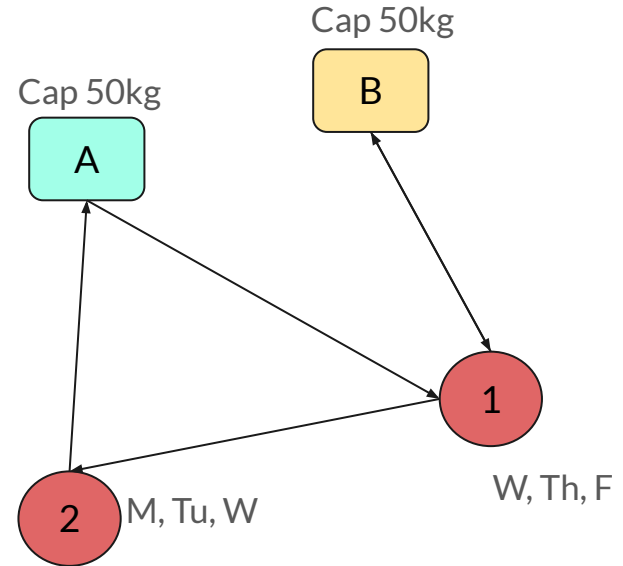
Sharing deliveries VRP ≠ Collaborative VRP

- VRP with delivery sharing focuses on optimizing routes by sharing the goods to be delivered
- Collaborative VRP focuses on optimizing routes by exchanging the requests to be fulfilled.

Problem definition

Hypotheses (VRP)

- Each customer has a demand at all or some of the producers
- Delivery slots in "half-day" intervals: a customer can have one or several available slots and can be visited multiple times
- One vehicle per producer with limited capacity
- Maximum one tour per producer per "half-day"



- A delivers 10 kg to 1 and 15 kg to 2 on Tuesday morning.
- B delivers 15 kg to 1 on Wednesday afternoon

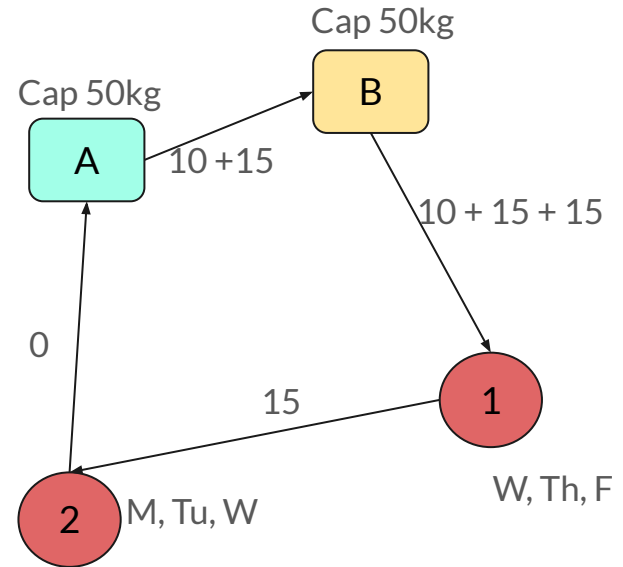
Hypotheses (sharing deliveries)

A producer p can go to another producer q to:

- Drop off goods for a customer c that producer q will visit.
- Pick up goods from producer q for the customer c that producer p will visit.

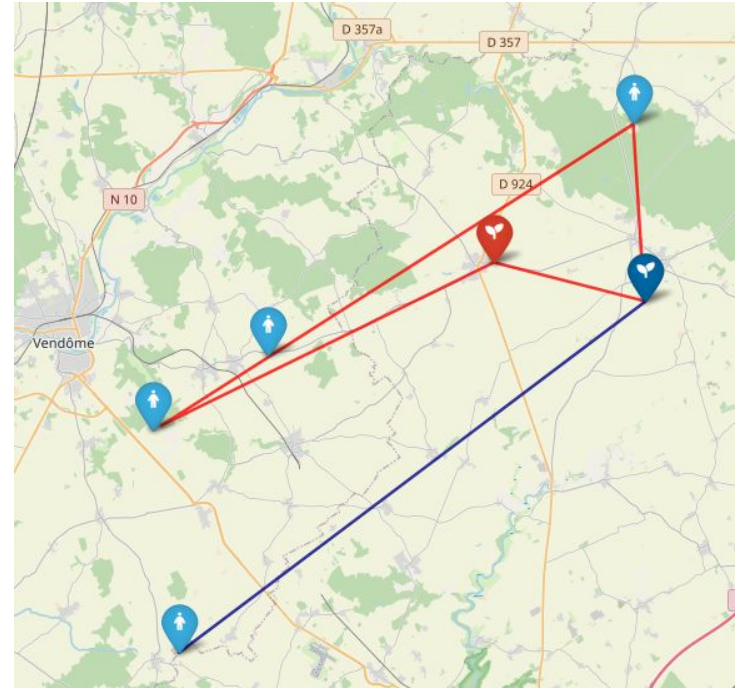
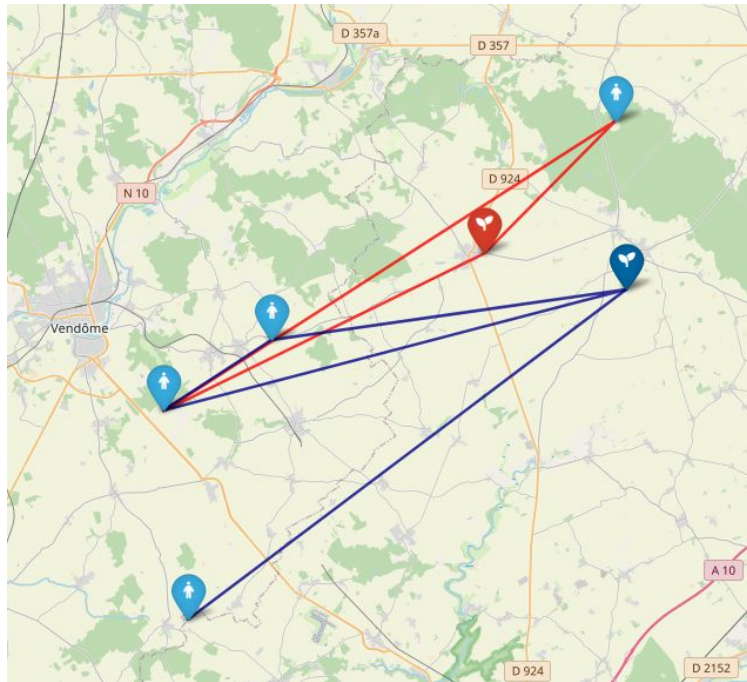
If a producer q receives goods from a producer p , they must deliver it themselves.

A producer has a maximum number of partners.



- A picks up goods of B for client 1 and delivers 1 (10kg + 15kg) and 2 (15kg) on Wednesday morning

Before / after sharing deliveries : example



Problem modelisation



Objective function

- Sum of distances travelled by producers (Min)

Alternately :

- Customer satisfaction (Max)
- Number of working days then total distance traveled (Min)
- ...



Classic decision variables of the VRP

$x_{pij}^d = 1$ If node j is visited directly after i by producer p during the half-day d , 0 otherwise

$y_{pc}^d = 1$ If producer p visits client c during the half-day d , 0 otherwise

$t_{pi}^d \geq 0$ Arrival time of producer p at node i during half-day d

$q_{pi}^d \in [0, Q]$ Load level of producer p just before node i during half-day d



Decision variables specific to delivery sharing

$z_{pq}^d = 1$ If producer p visits producer q during the half-day d , 0 otherwise

$Z_{pq} = 1$ If producers p and q are partners, 0 otherwise

$w_{pqc}^{dd'} = 1$ If p visits q during half-day d , and q visits c during half-day d' ($d \leq d'$)

$w'_{pqc}{}^{dd'} = 1$ If q visits p during half-day d , and q visits c during half-day d' ($d \leq d'$)



Classic constraints of the VRP

- Route continuity
- Demand satisfaction
- Time windows : a customer can only be visited during the periods when they are available
- The travel time of a vehicle per half-day cannot exceed the duration of an “half-day”
- Capacities: we check the loading rate at the departure of the vehicle and update it at each node visit (loading/unloading)



Specific constraints related to delivery sharing

- The goods from each producer are either transported directly, deposited at another producer, or entrusted to a producer on a visit.

$$\sum_{d \in D_c} y_{pc}^d + \sum_{\substack{d \in D, d' \in D_c \\ d \leq d'}} \sum_{\substack{q \in P \\ q \neq p}} w_{pqc}^{dd'} + \sum_{\substack{d \in D, d' \in D_c \\ d \leq d'}} \sum_{\substack{q \in P \\ q \neq p}} w'_{pqc}{}^{dd'} \geq 1 \quad \forall c \in C, \forall p \in P_c$$

- Maximum number of partners

$$Z_{pq} \geq \frac{1}{2 \times |D|} \sum_{d \in D} (z_{pq}^d + z_{qp}^d) \quad \forall p, q \in P, p \neq q$$

$$\sum_{\substack{q \in P \\ p \neq q}} Z_{pq} \leq \text{MaxPartners}_p \quad \forall p \in P$$



Model size

Binary variables : $\mathcal{O}(|P|^2 * |C| * |D|^2)$

Continuous variables : $\mathcal{O}(|P| * (|P| + |C|) * |D|)$

Constraints : $\mathcal{O}(|P|^2 * |C| * |D|^2)$

Results



Instances generation & resolution

Instances generated according to several parameters :

- Number of clients (5, 10, 15), number of producers (5, 10) and number of half-days (10)
 - Producers/clients disposition (clients in the middle, producers in the middle, random)
 - Demands (high / low)
 - Clients disponibilities (restrained to 2-3 days or not)
 - Demands at all producers or only at certain ones
-
- 10 instances by configuration
 - Resolutions for 0,1, ... maximum partners
 - CPLEX (32 threads), time limit : 30 minutes
 - CaSciModOT (computing center)

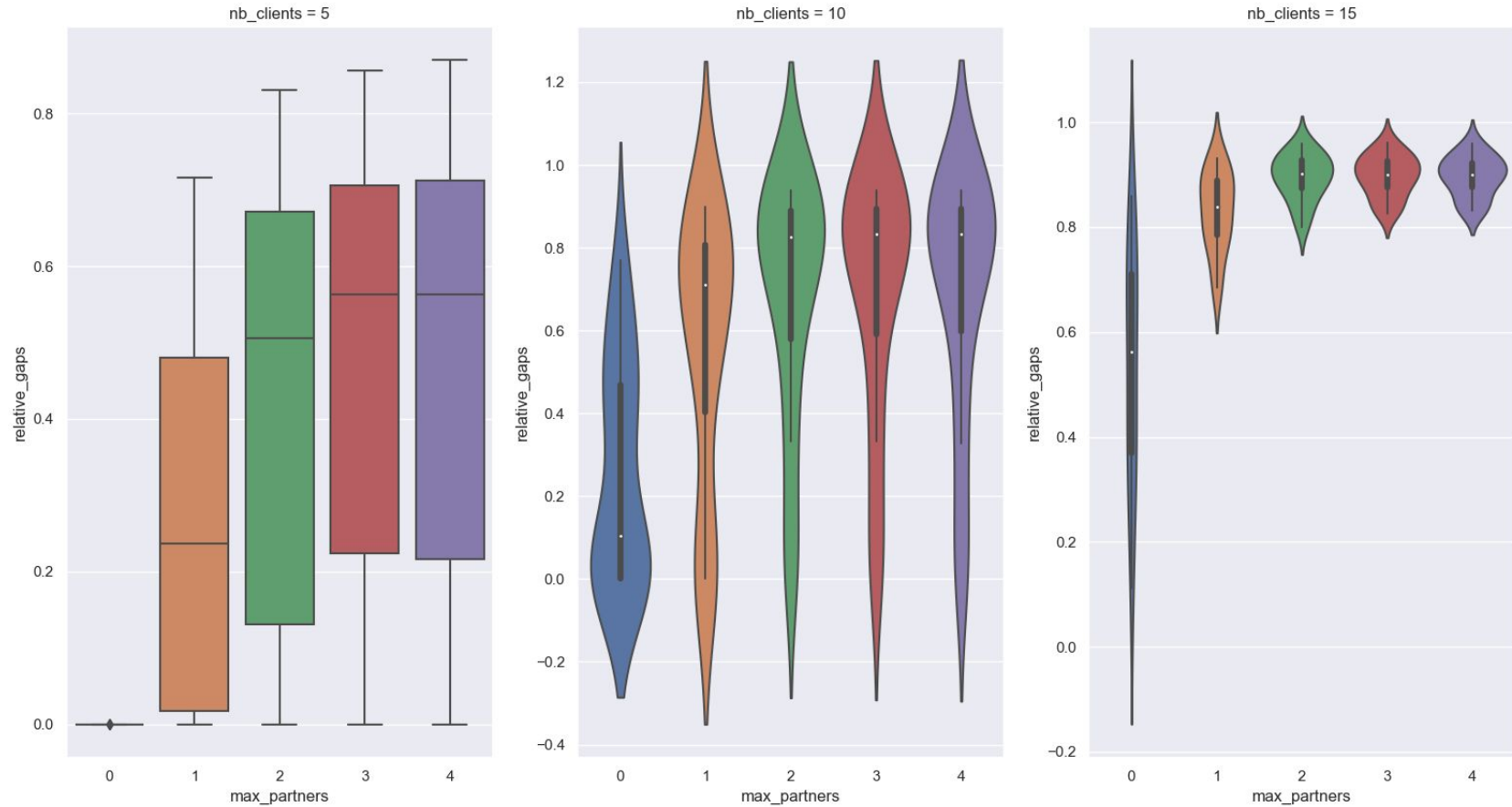
producers = 5

Average gain vs no_sharing

Parameter	# instances	# Instances solved optimally
Demands = low	150	49 (33%)
Demands = high	170	16 (10%)
# clients = 5	120	45 (37,5%)
# clients = 10	120	20 (16,6%)
# clients = 15	80	0
Maximum partners = 0	64	34 (53,1%)
Maximum partners = 1	64	13 (20,3%)
Maximum partners = 2	64	7 (10,9%)
All configurations	320	65 (20,3%)

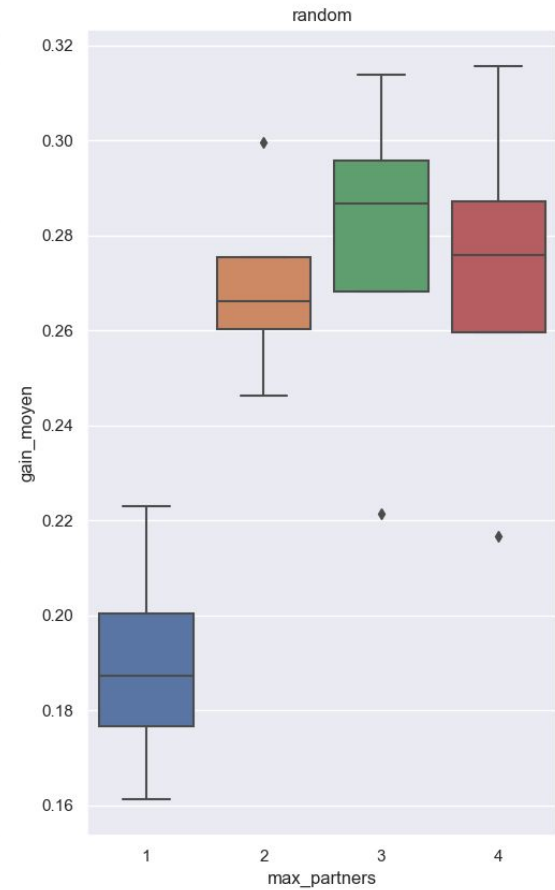
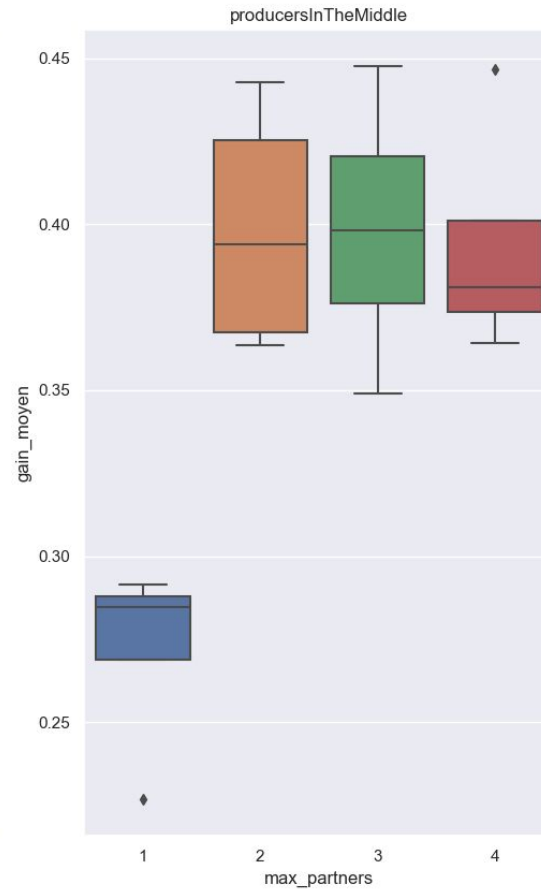
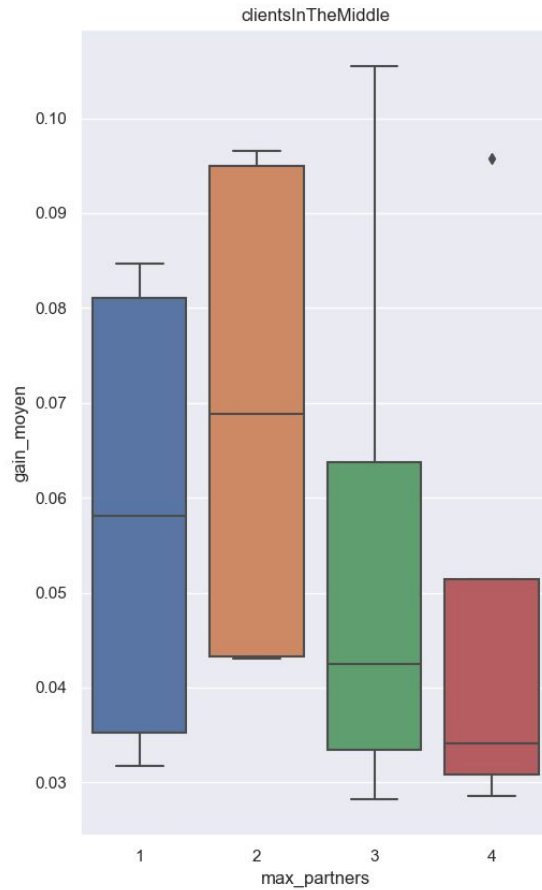
Parameter	MaxPartners = 1	MaxPartners = 2	MaxPartners = 3	MaxPartners = 4
Demands = low	8.65%	11.07%	10.51%	10.28%
Demands = high	0.21%	0.08%	0.13%	0.08%
# clients = 5	8.87%	12.38%	12.29%	11.91%
# clients = 10	2.91%	2.978%	2.47%	2.33%
# clients = 15	1.09%	0.642%	0.42%	0.58%
Disponibilities = restrained	5.48%	6.79%	6.58%	6.36%
Disponibilities = all	3.86%	4.99%	4.64%	4.58%
Producers_per_client = random	5.43%	6.22%	5.91%	5.72%
Producers_per_client = all	3.73%	5.53%	5.29%	5.22%
All configurations	4.69%	5.92%	5.64%	5.50%

Relative gap vs. maximum number of partners



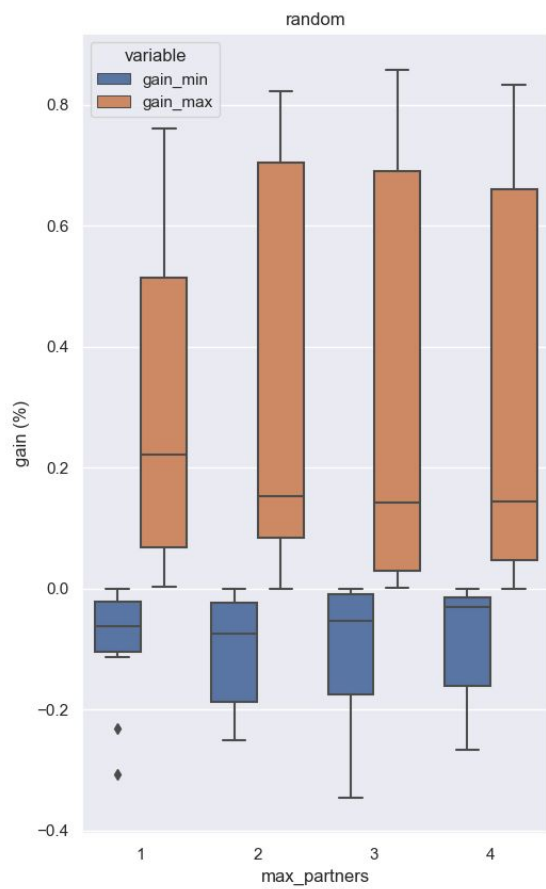
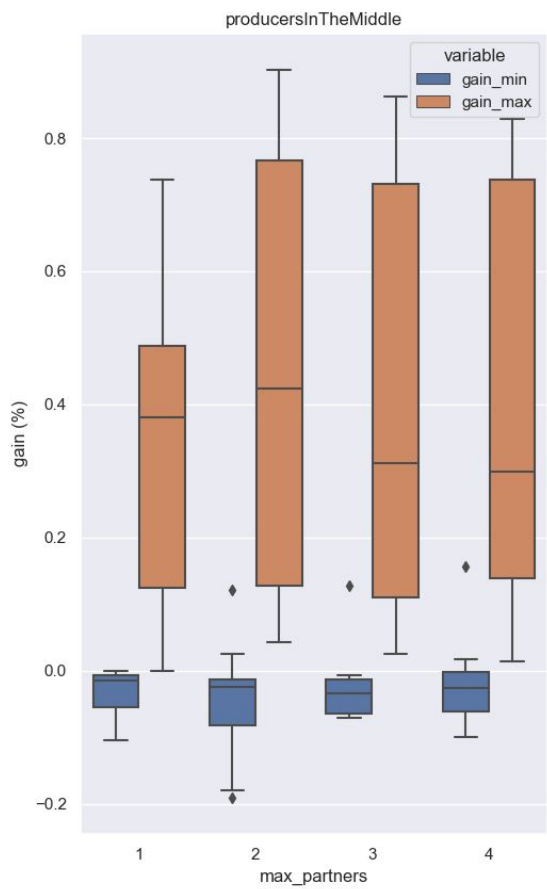
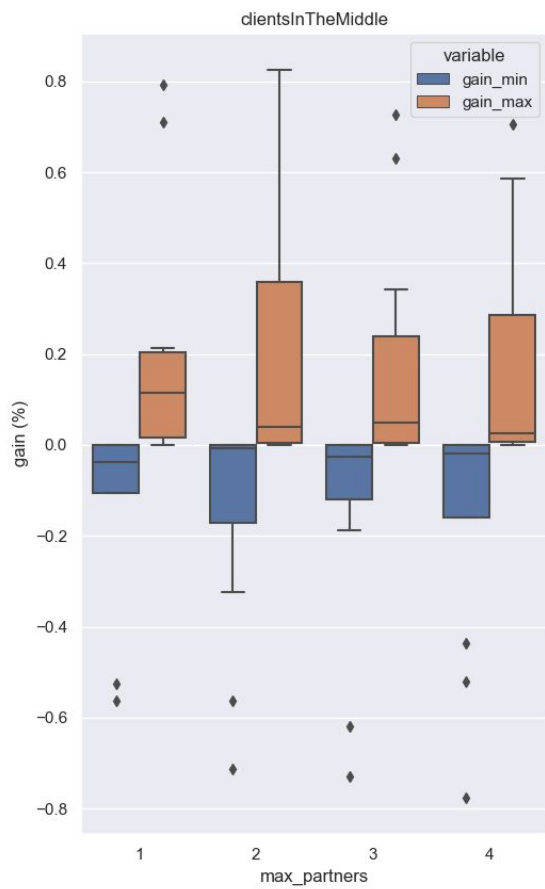
Average gain vs. maximum number of partners

demands = low, nb_clients = 5



Gain vs. Maximum Number of Partners

demands = low, nb_clients = 5





Perspectives

- Equity : prevent a producer from working more than if they were working alone (WIP)
- Develop a metaheuristic (LNS ?)
- Refine the modeling of this problem by incorporating feedback from stakeholders in SFSC



Thank you for your attention

{adrien.callico, pierre.desport, jean-charles.billaut}@univ-tours.fr

caroline.prodhon@utt.fr