

**“Excuse me, Sir,
we ordered 31
minutes ago!”**

How to address time delays
in food delivery



**We are the
world's leading
local delivery
platform.**



70 + countries



11 Brands



+3,400 Heroes in Berlin DHSE



+1.4 M Riders



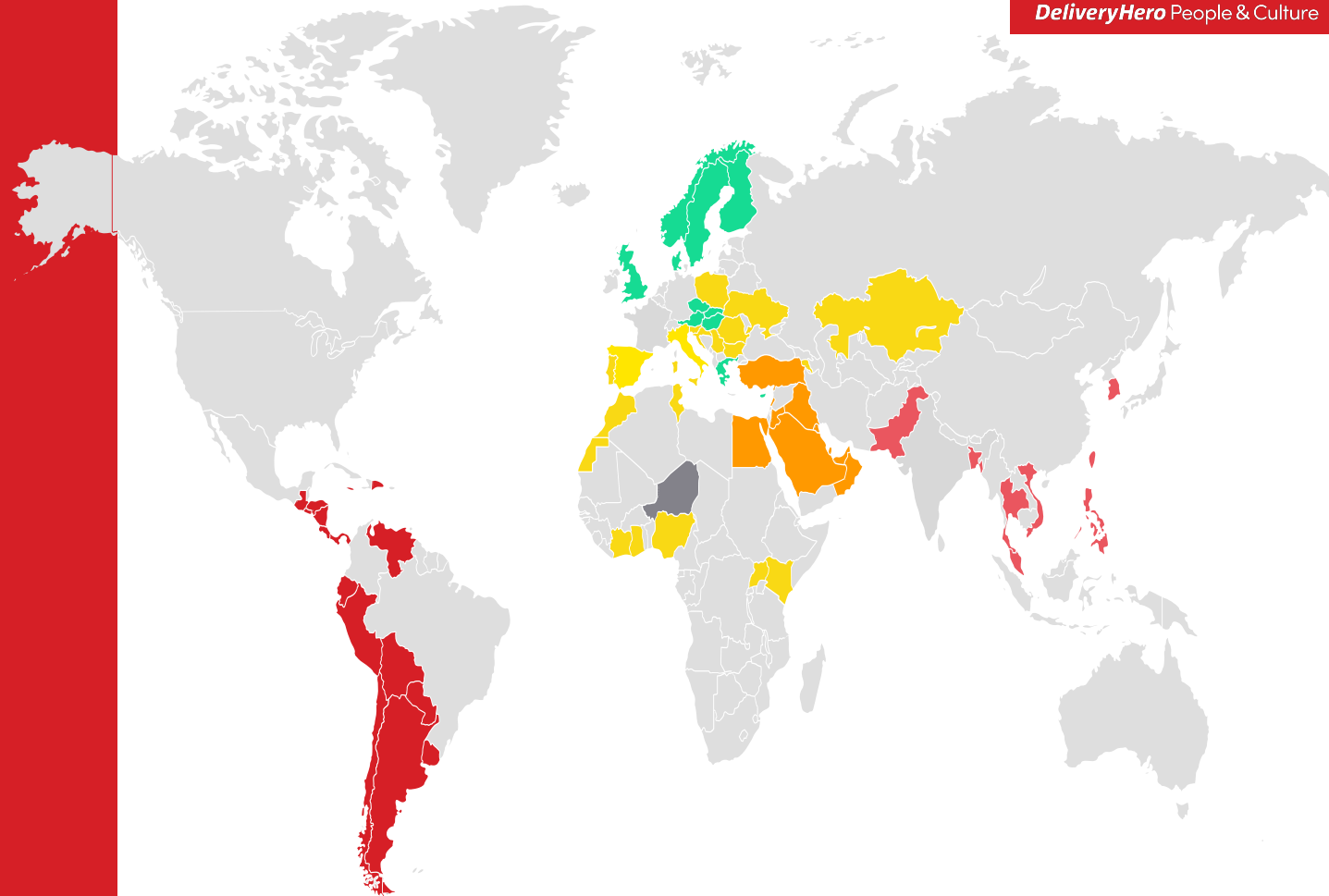
+1.3 M Restaurant partners & vendors



10+ Tech Hubs



42 Quick Commerce countries



Our brands

Glovo!

 InstaShop

 efood

 foodora

 foodpanda

 foody

**HUNGER
STATION**

배달의민족

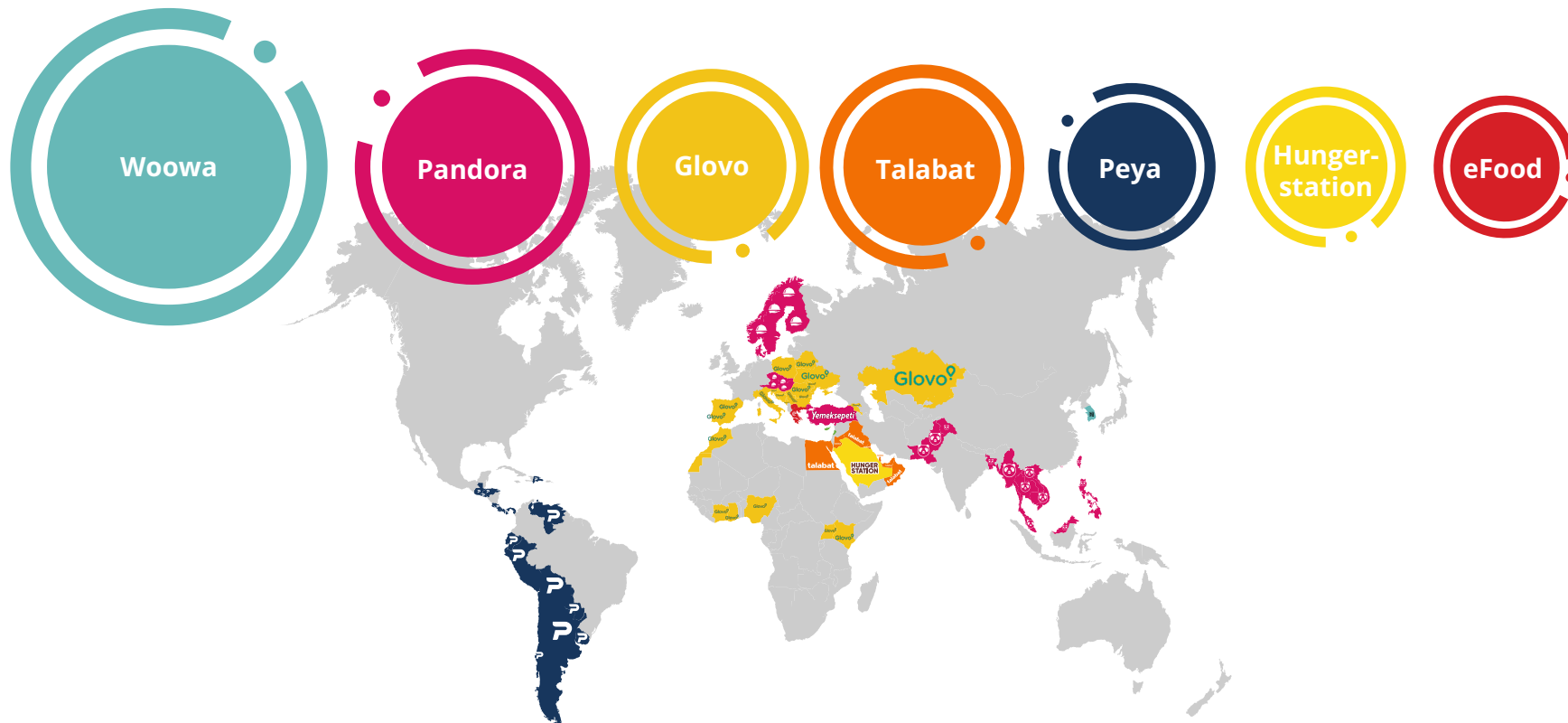
 PedidosYa

talabat

Yemeksepeti



Our Tech Platforms





Onboarding Experience



Our vision:

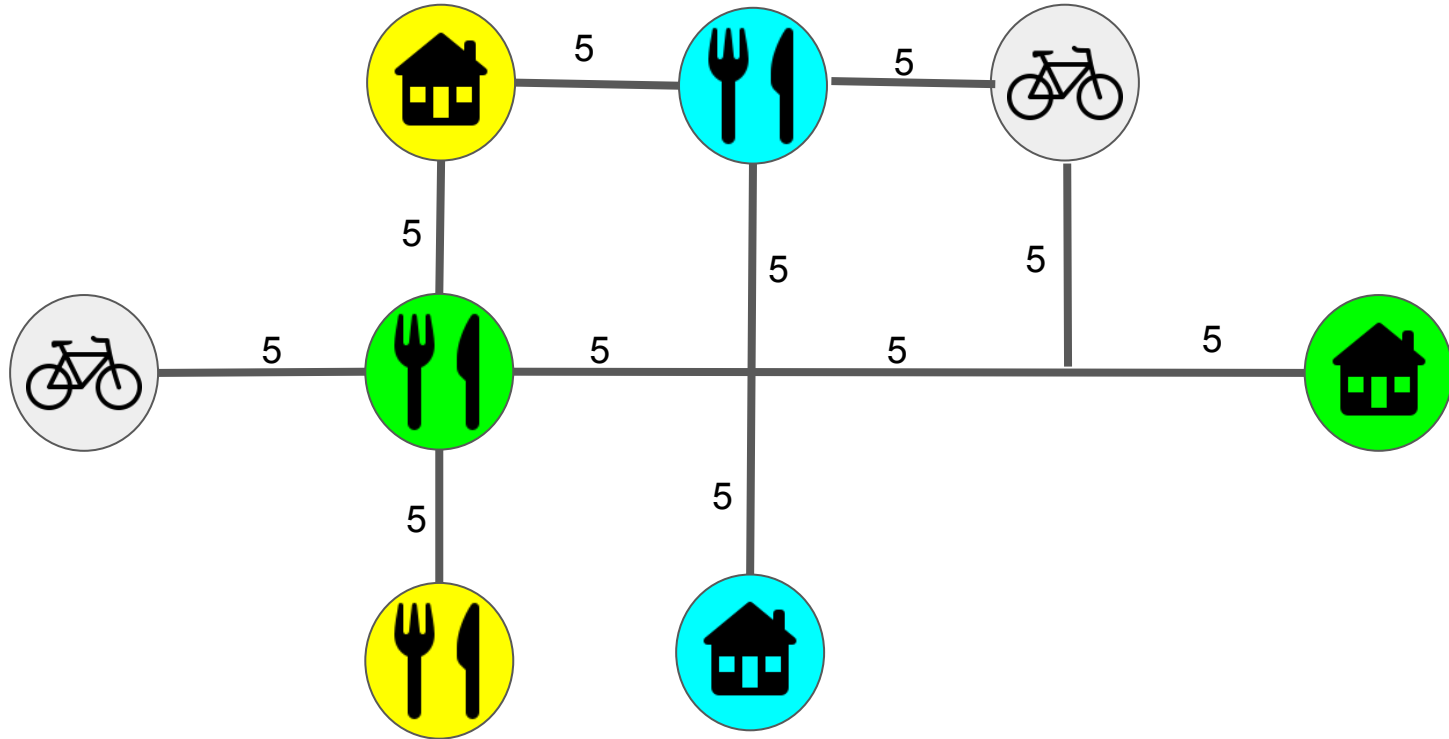
**Always delivering an
amazing experience**

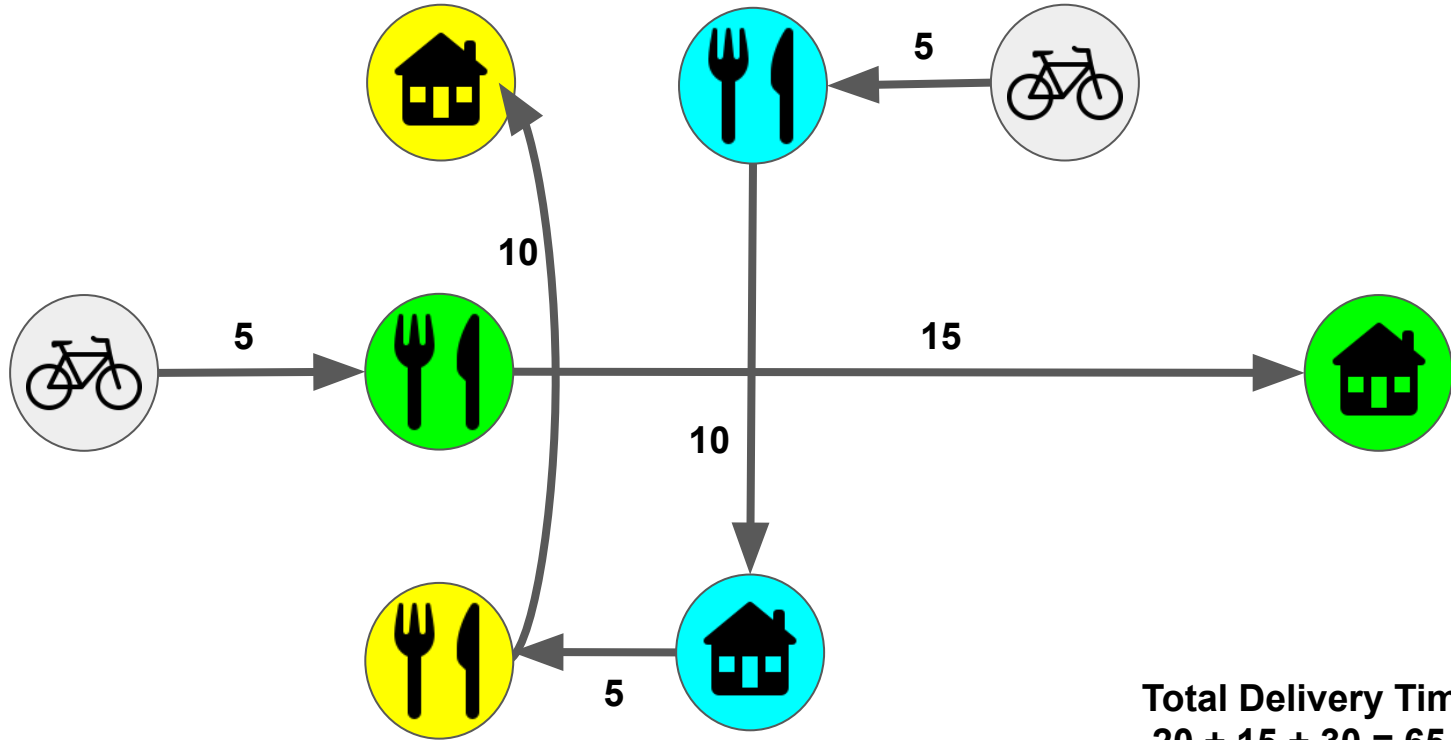
...fast, easy and to your door.

The Vehicle Routing Problem with Pickup and Delivery (VRPPD)



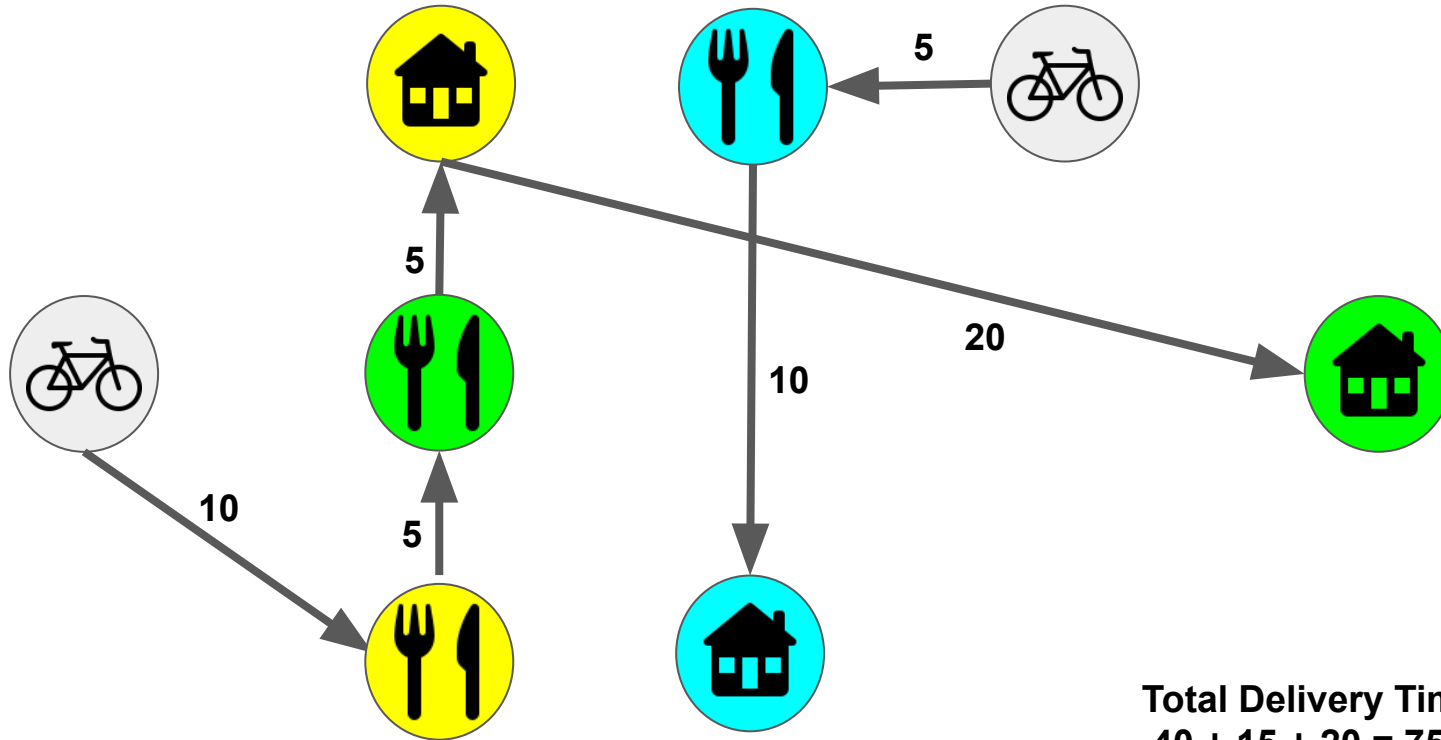







Total Delivery Time
 $20 + 15 + 30 = 65$

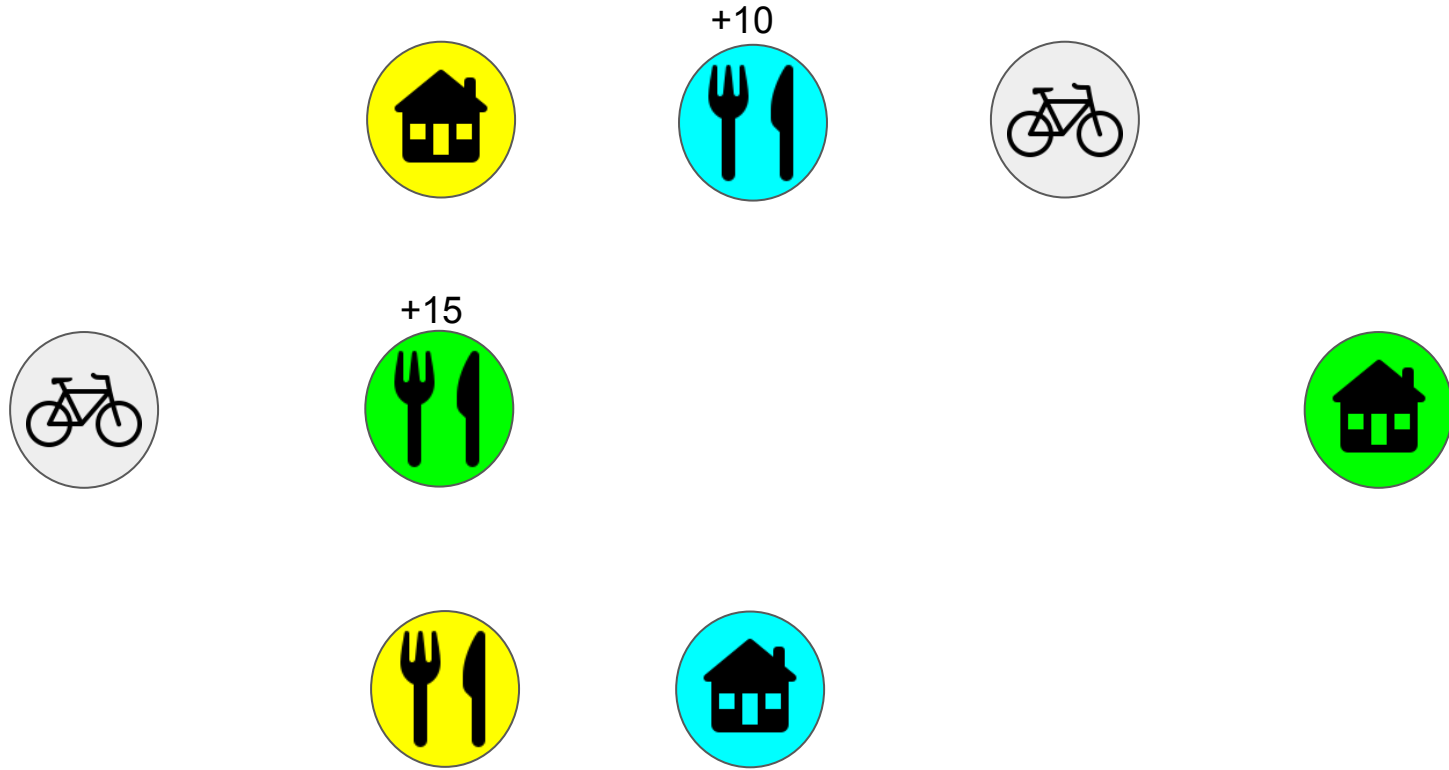


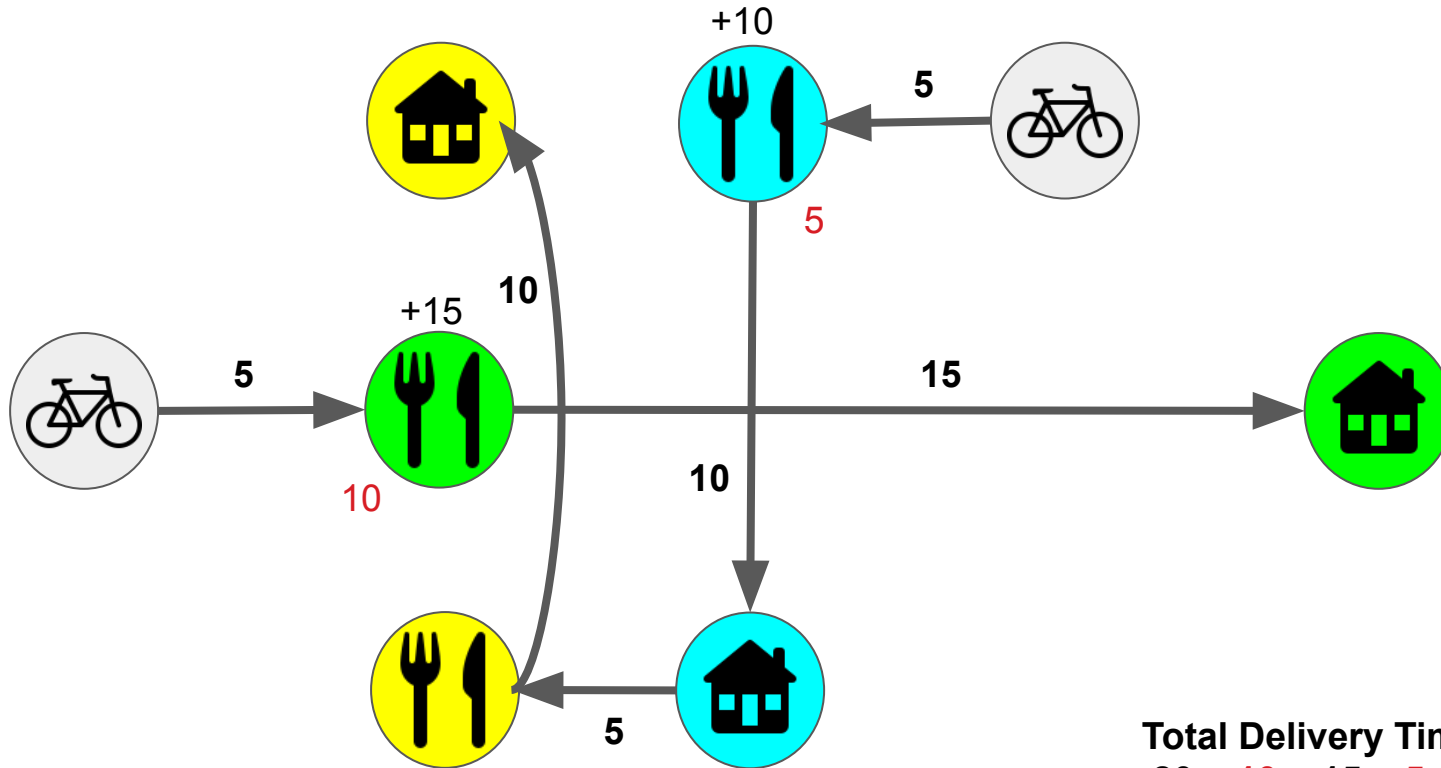


Total Delivery Time
 $40 + 15 + 20 = 75$

The Vehicle Routing Problem with Pickup and Delivery and Time Windows (VRPPDTW)

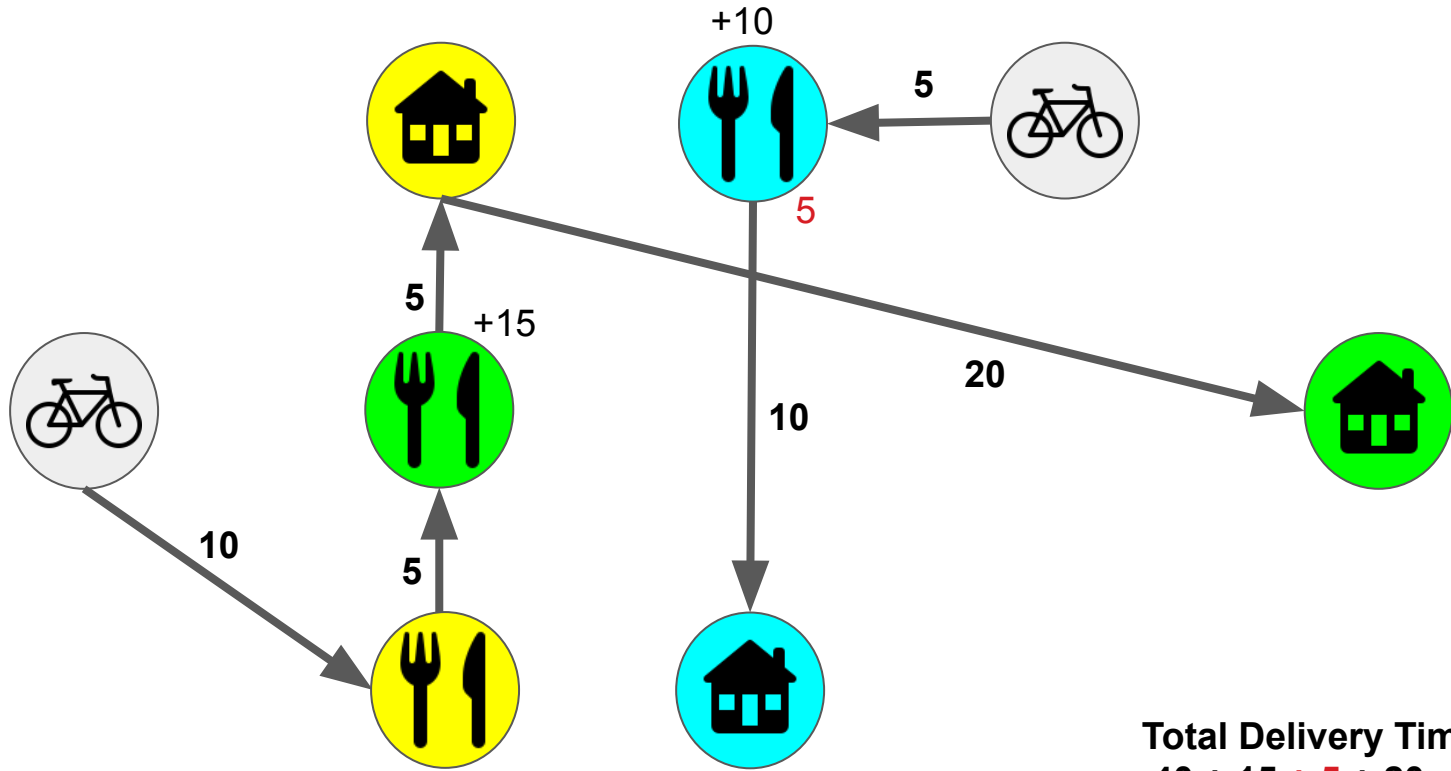






Total Delivery Time
 $20 + 10 + 15 + 5 + 30 + 5 = 85$





Total Delivery Time
40 + 15 + 5 + 20 = 80





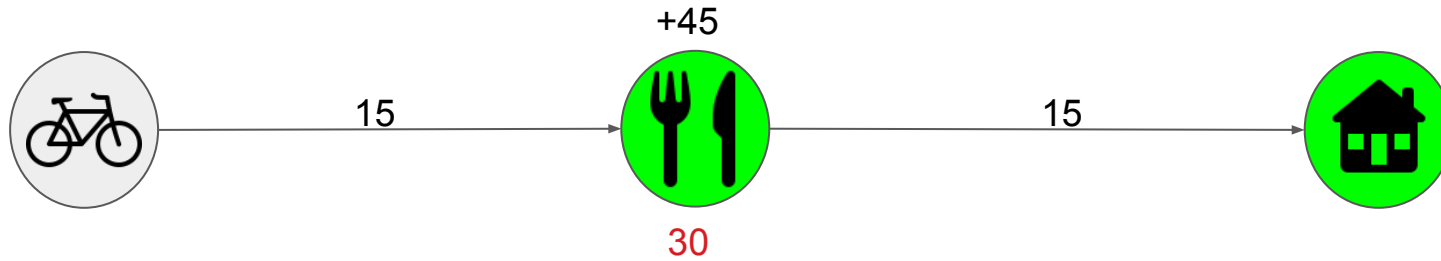
**Once upon a time in
South America...**



... some random dude ordered a Bolognese at 10AM | Mamma mia!

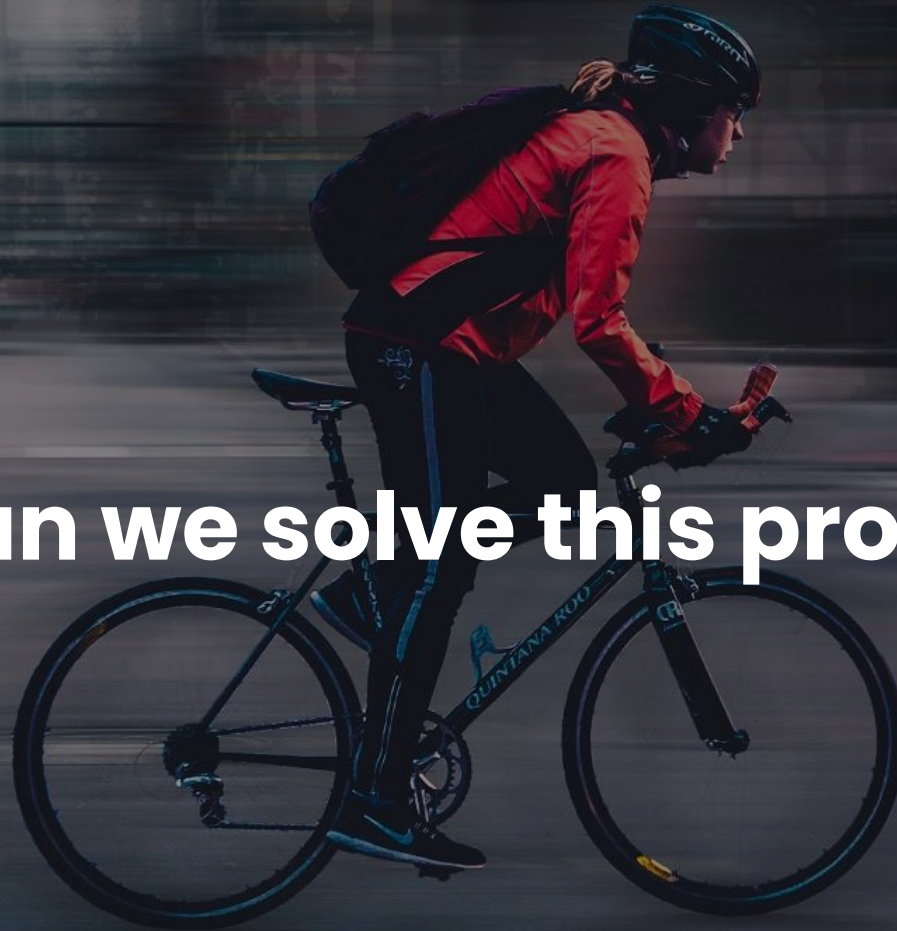


Problem | Excessive Waiting Time at Vendor



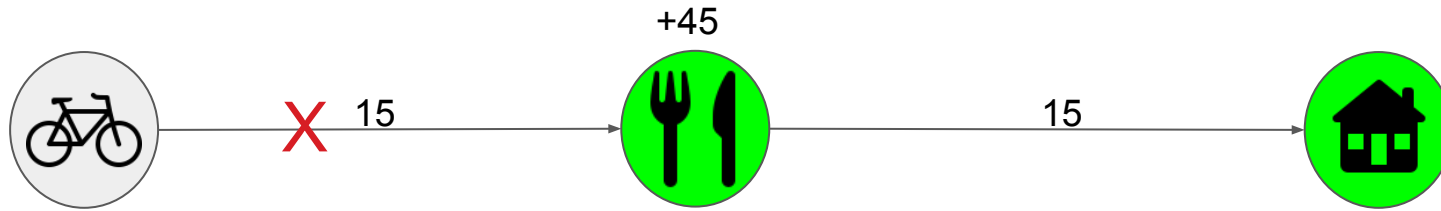
- Riders (have the feeling that) they are wasting their time within which they could earn more money
- Vendors do not like riders waiting at their premises as they believe it negatively impacts their customers experience
- I don't like it: We do not have the possibility to redispach as rider and delivery are blocked for further optimization

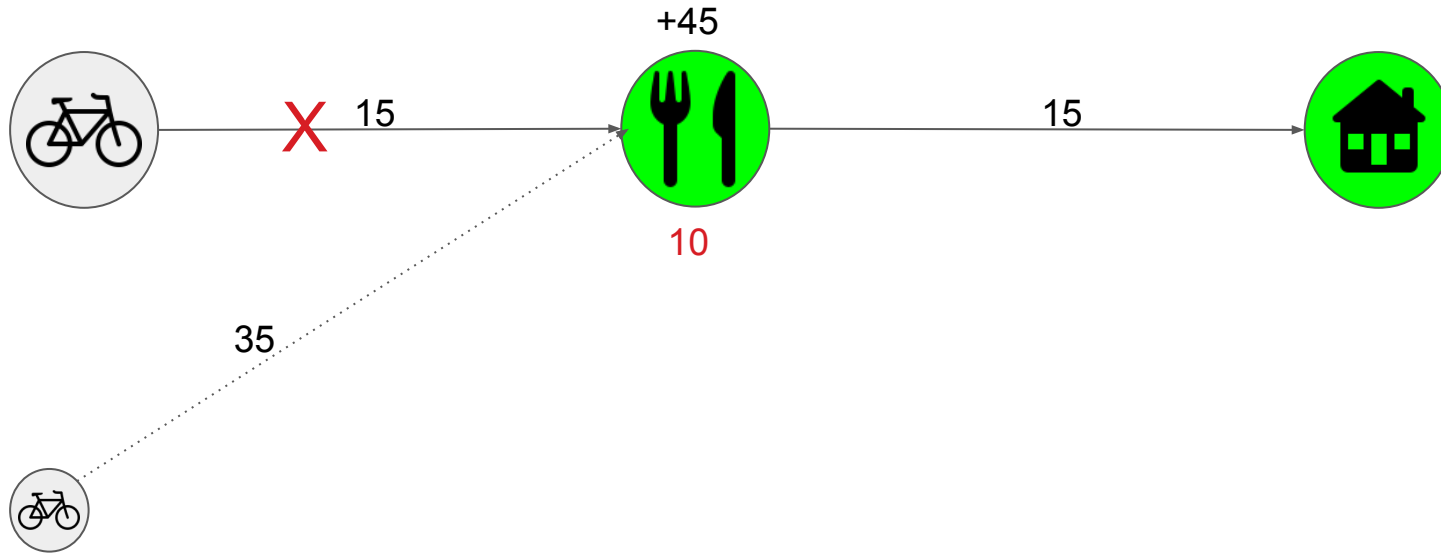
How can we solve this problem?



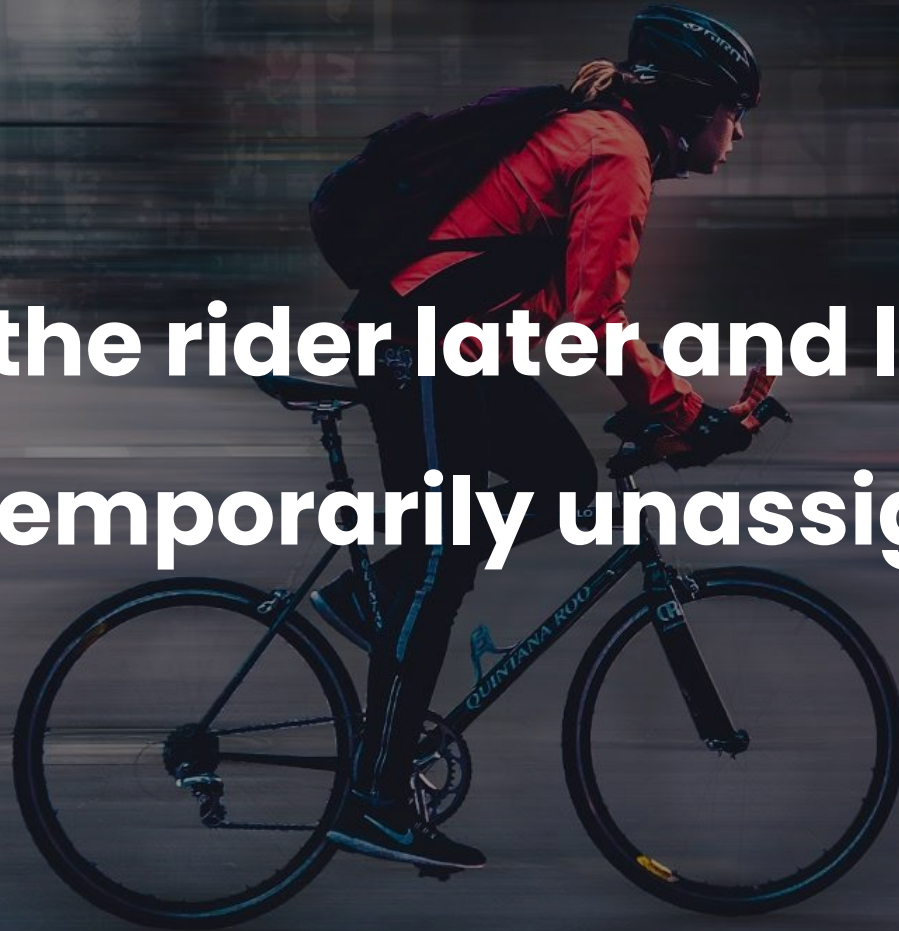
Introduce a constraint?!



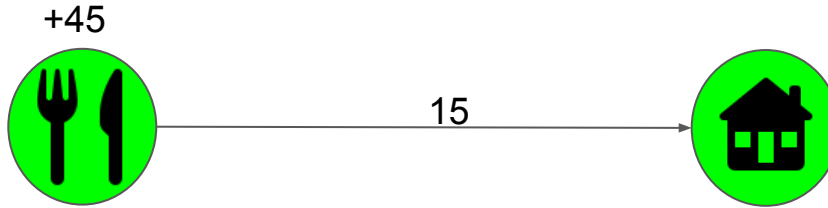




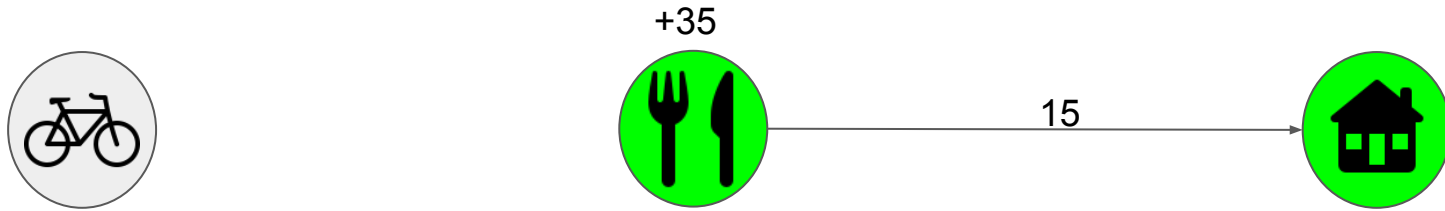
**Notify the rider later and leave
order temporarily unassigned?!**



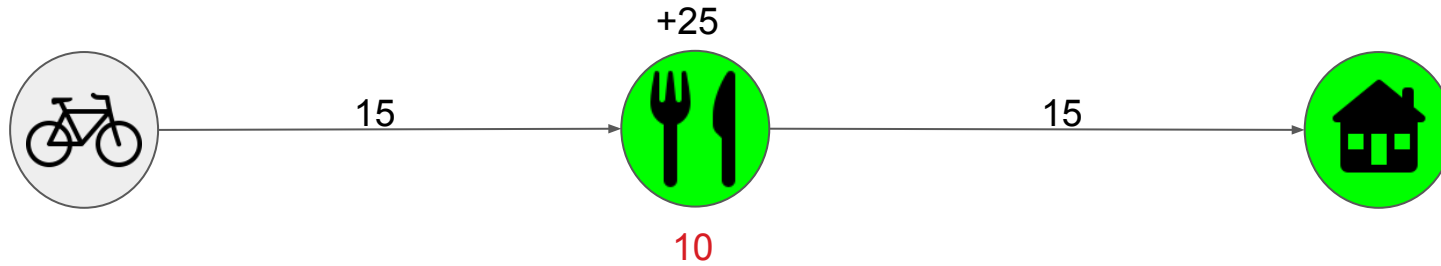
10:00 AM



10:10 AM



10:20 AM

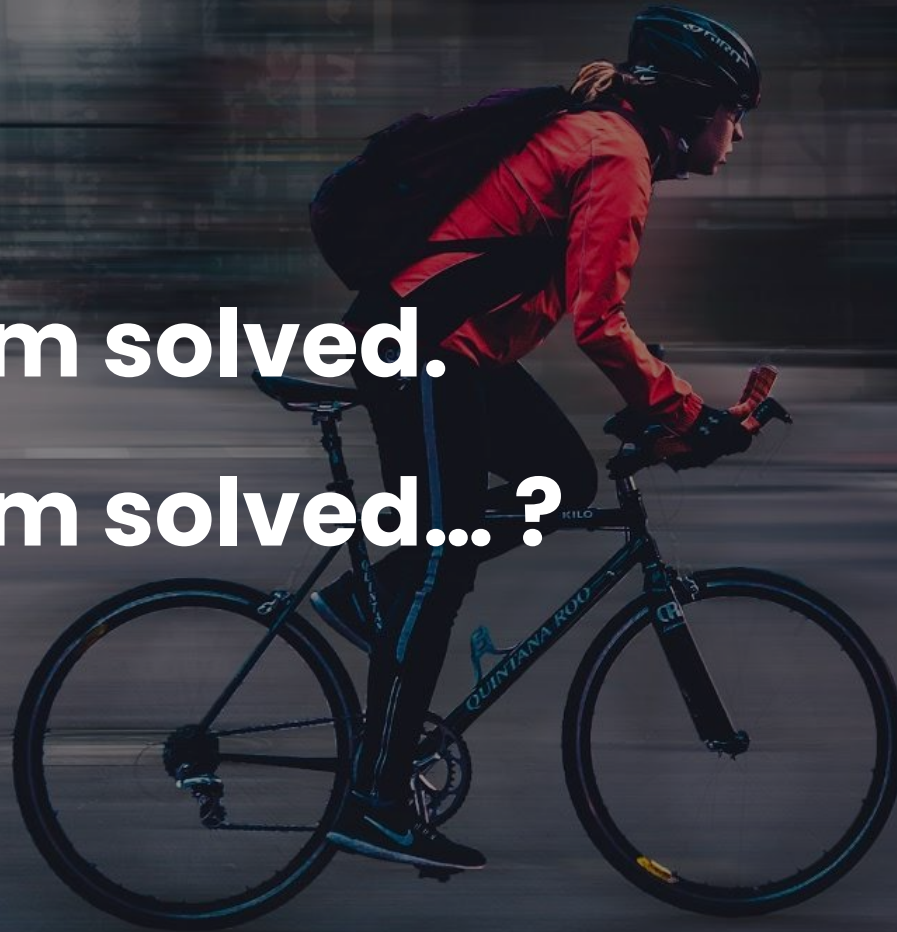


Delivery Hero
Logistics

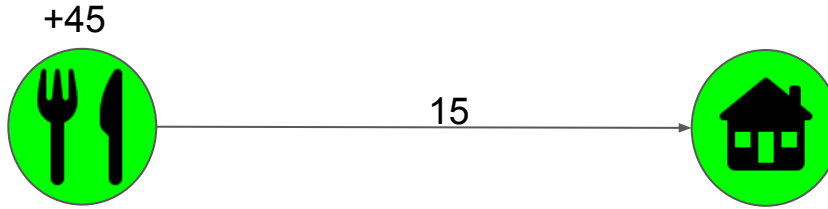
Problem solved.



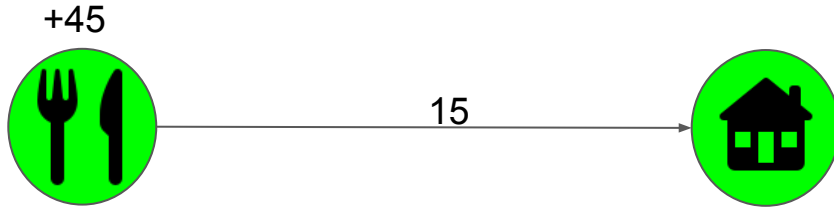
Problem solved.
Problem solved...?



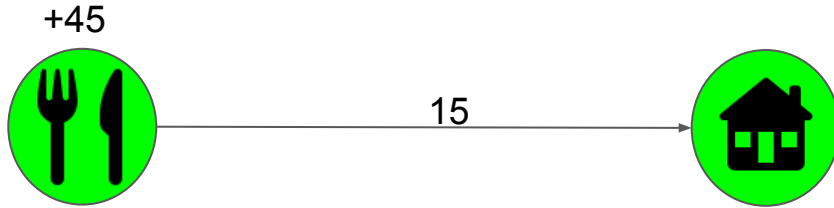
10:00 AM



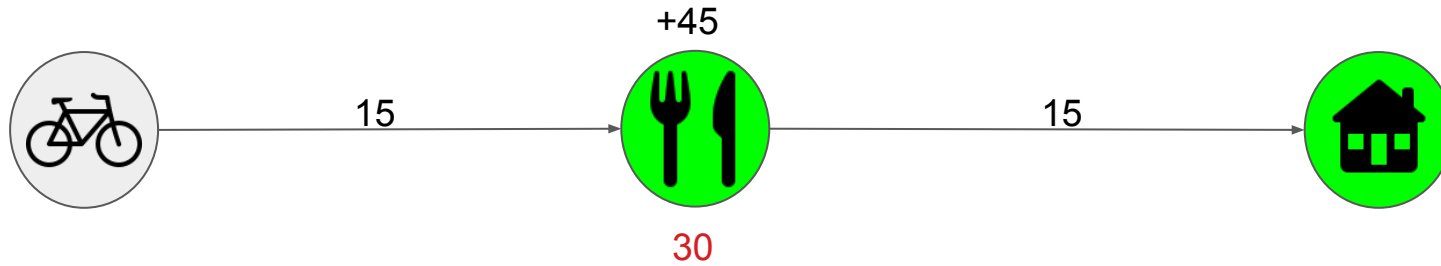
10:10 AM



10:20 AM

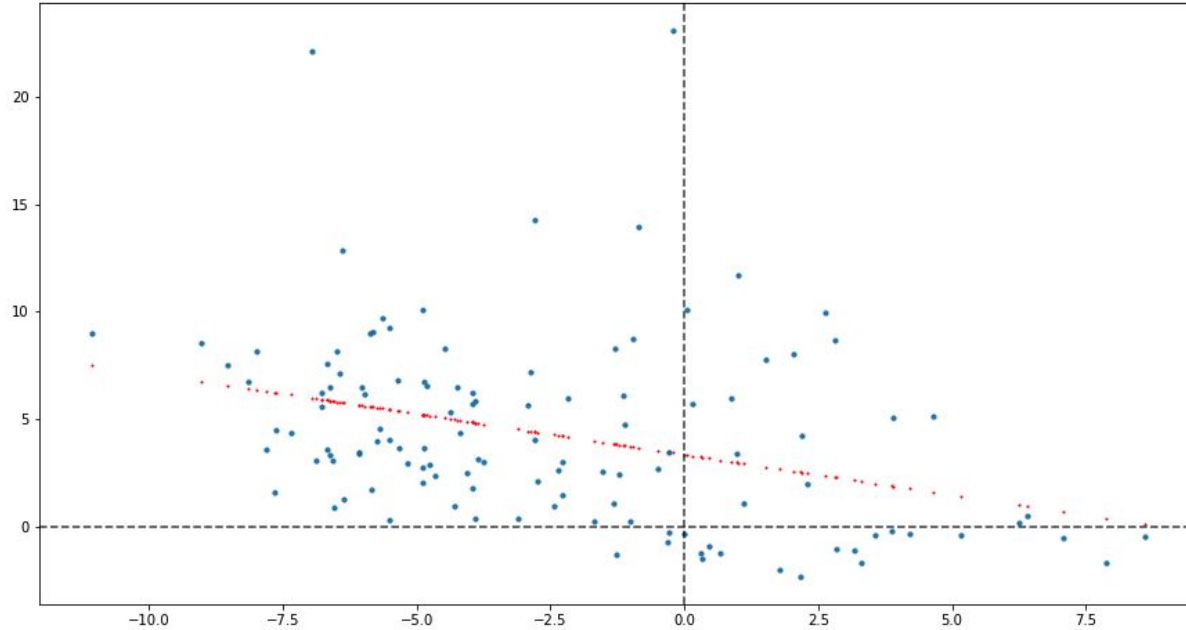


10:20 AM

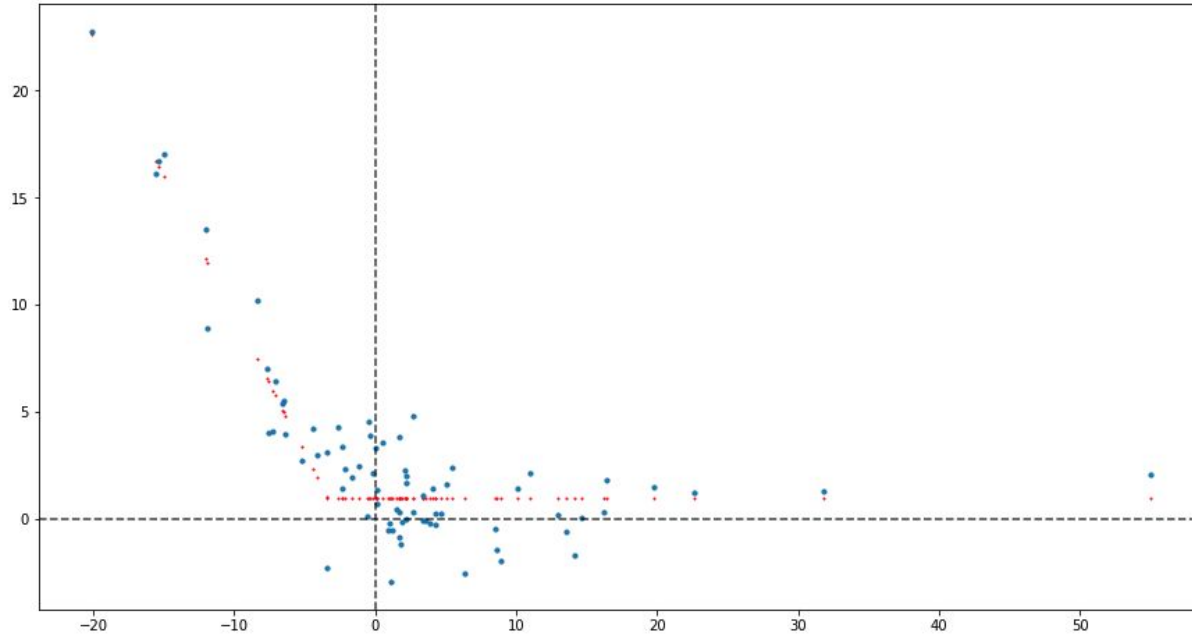


Introduction of a state-of-the-art, cutting-edge machine learning approach to determine the reliability of a vendor with respect to food preparation times.



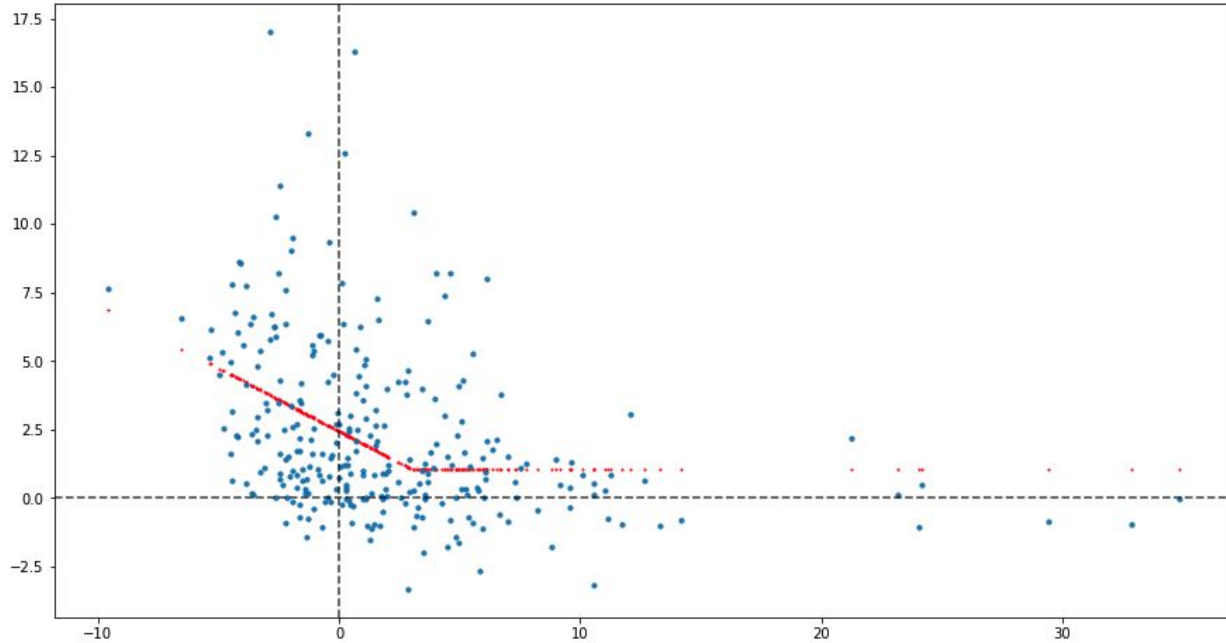


Always send the rider to the vendor right away!

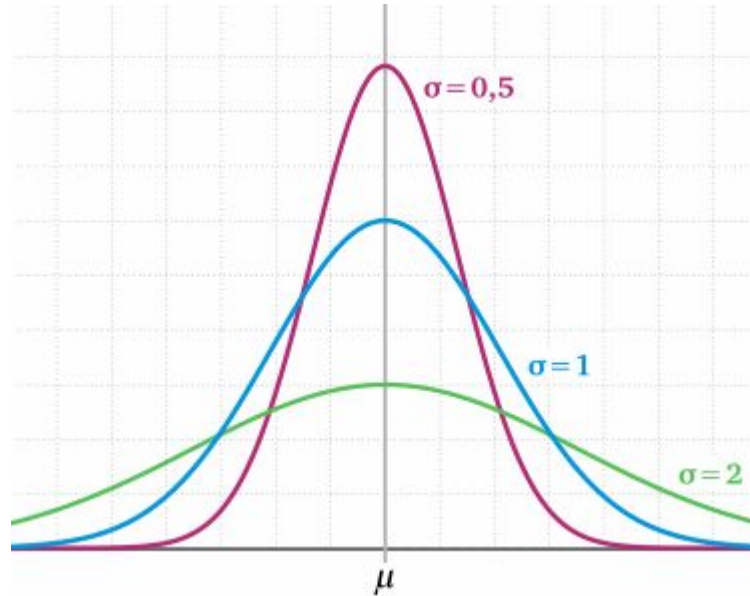


Could increase prep time estimate in the future.
Send the rider just-in-time with respect to prep time!



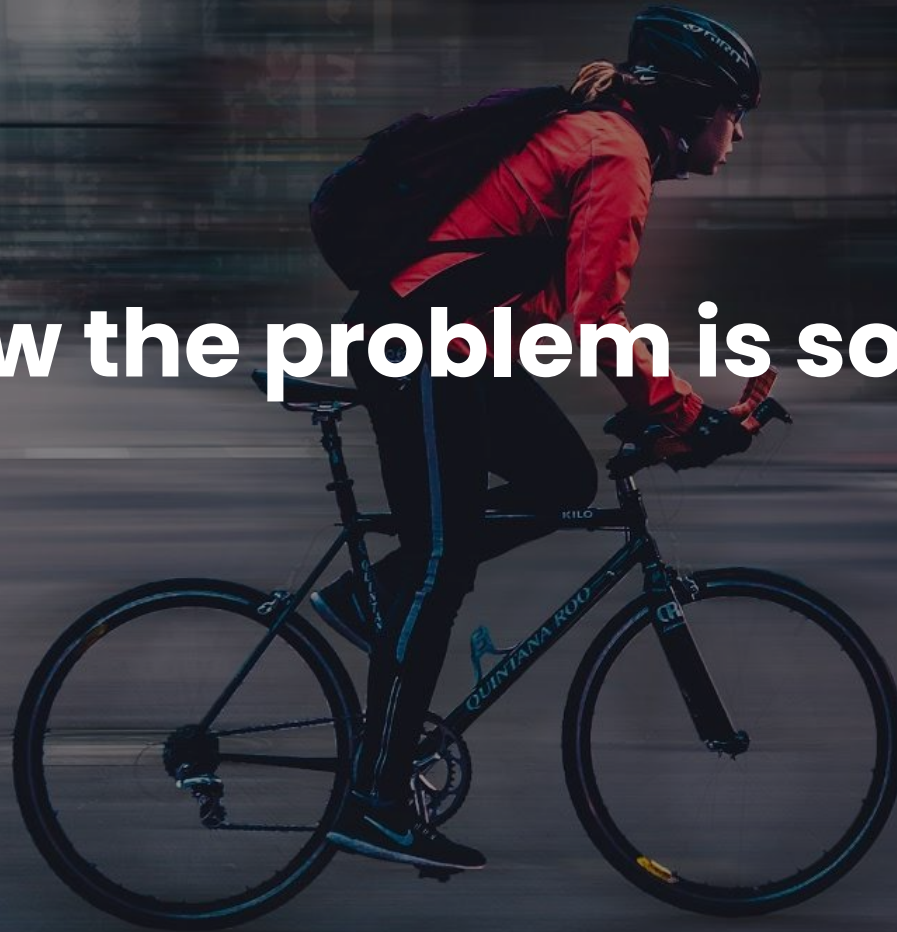


Could increase prep time estimate in the future.
However, when do we send the rider?

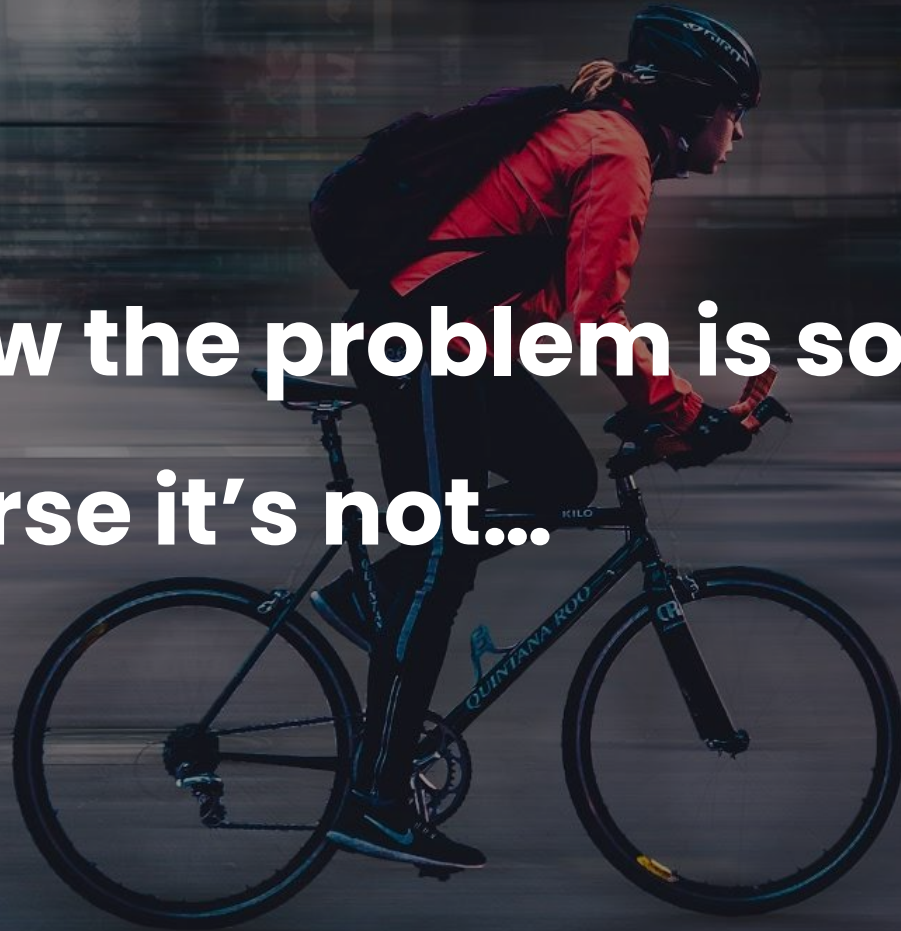


Assume the prep time follows a normal distribution.
Decide to when to send the rider based on the variance

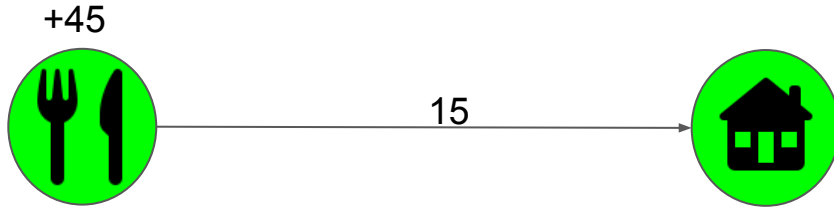
But now the problem is solved.



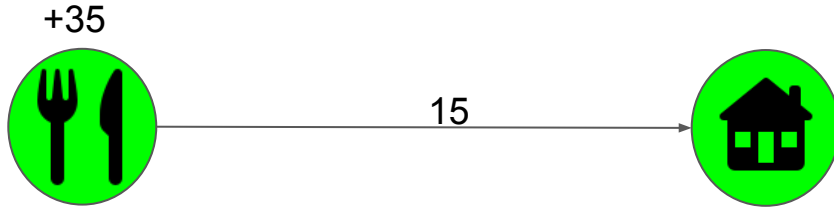
**But now the problem is solved.
Of course it's not...**



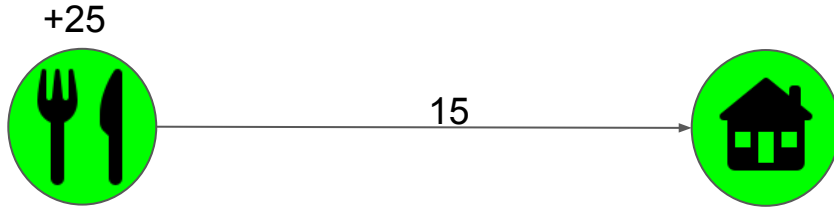
10:00 AM



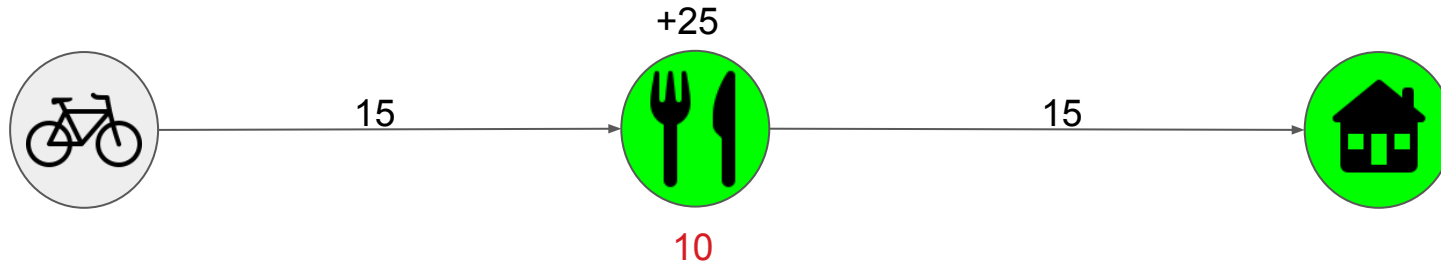
10:10 AM



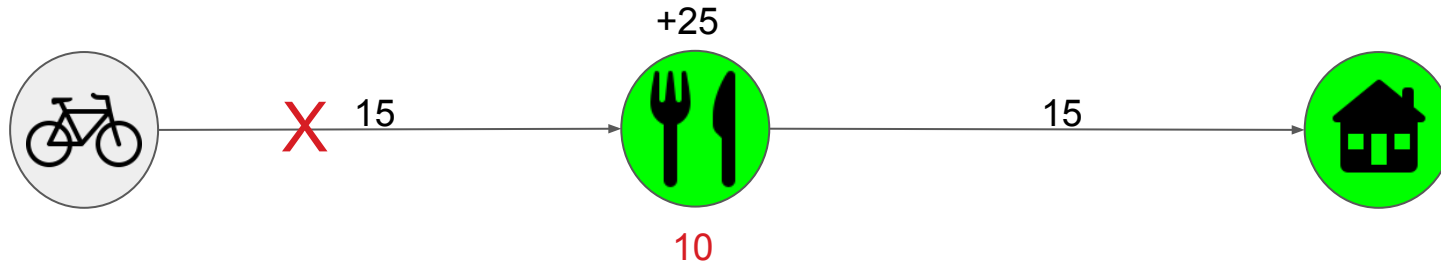
10:20 AM



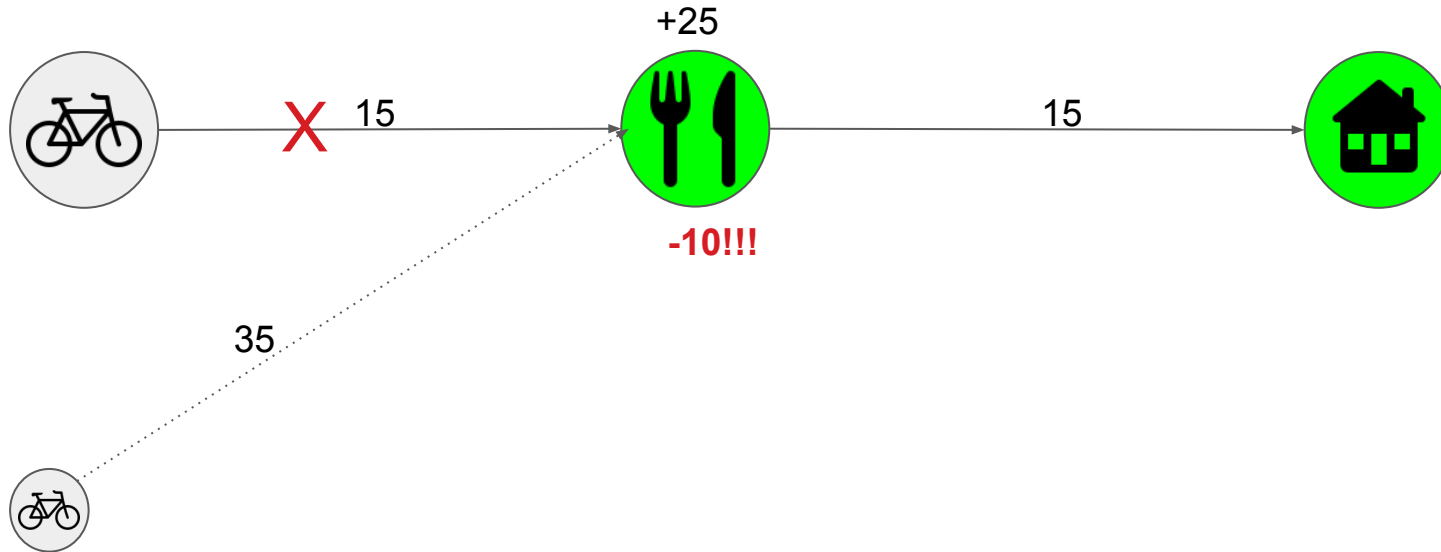
10:20 AM



10:20 AM



10:20 AM

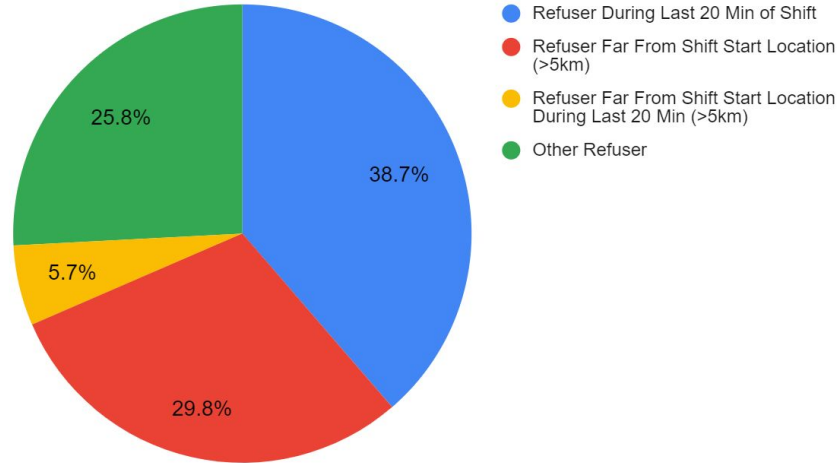




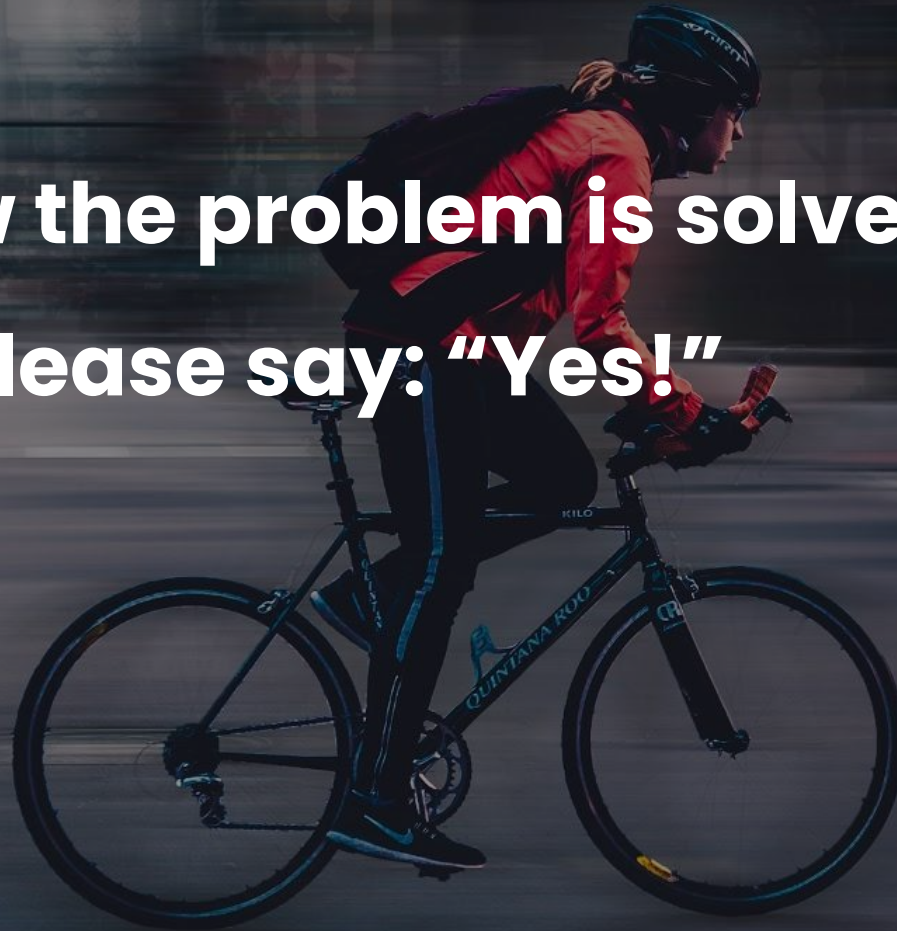


Determine the reasons why riders decline orders and try to avoid corresponding situations using optimization techniques.

Refuser Rider

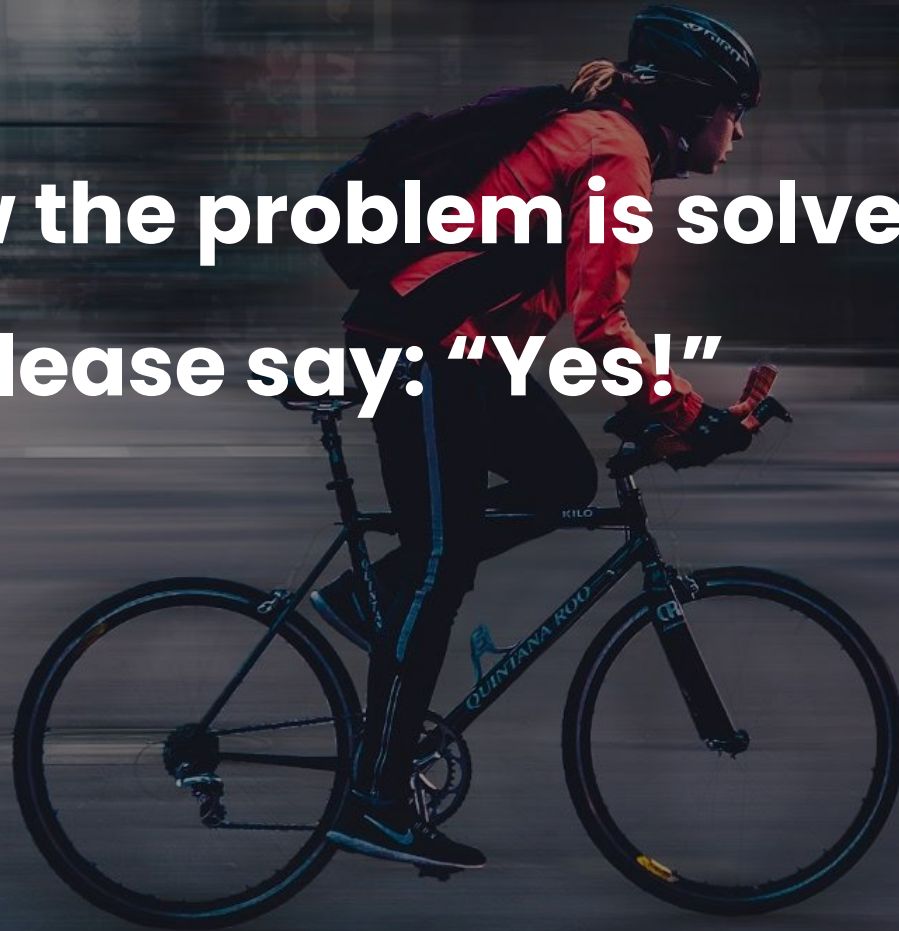


**But now the problem is solved,
right? Please say: "Yes!"**

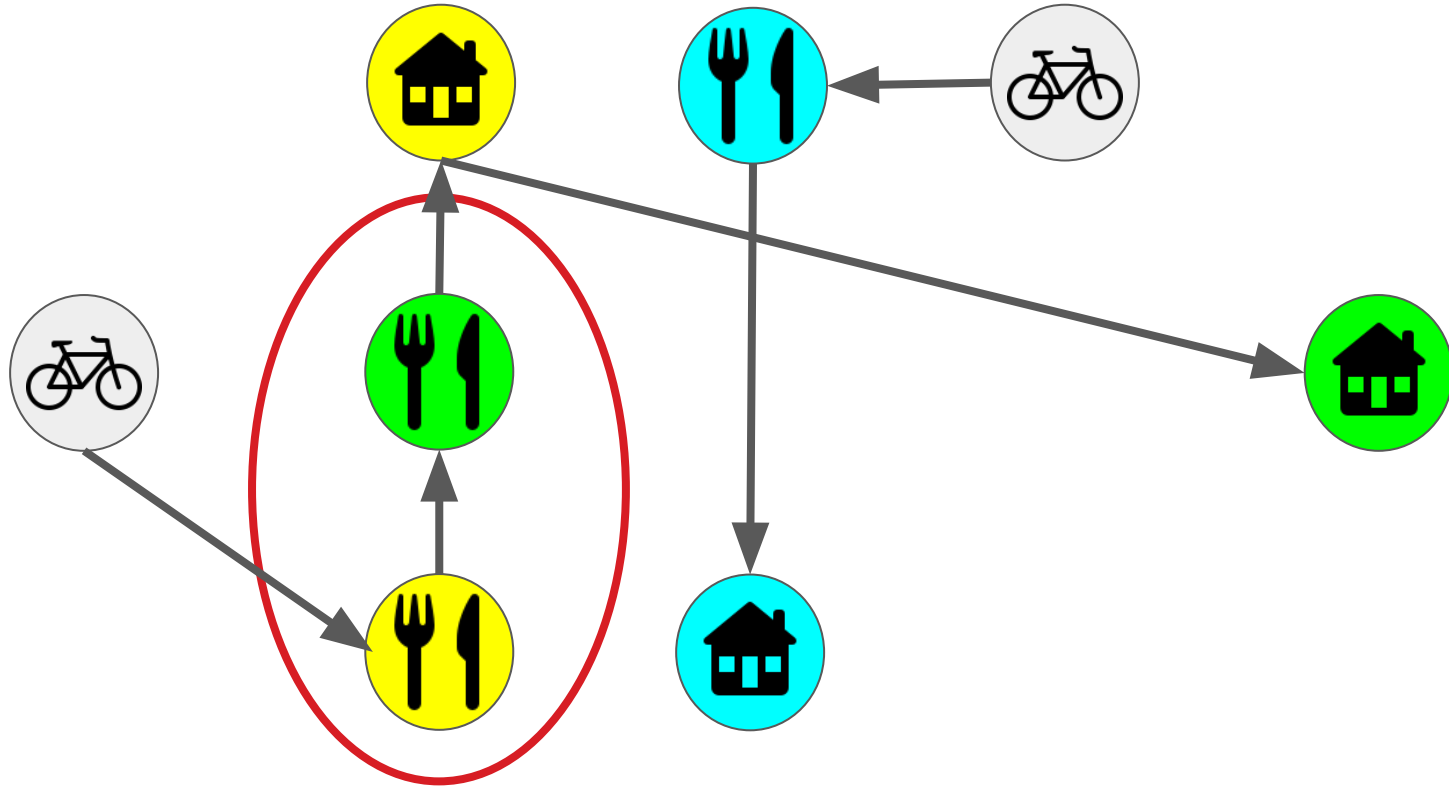


**But now the problem is solved,
right? Please say: "Yes!"**

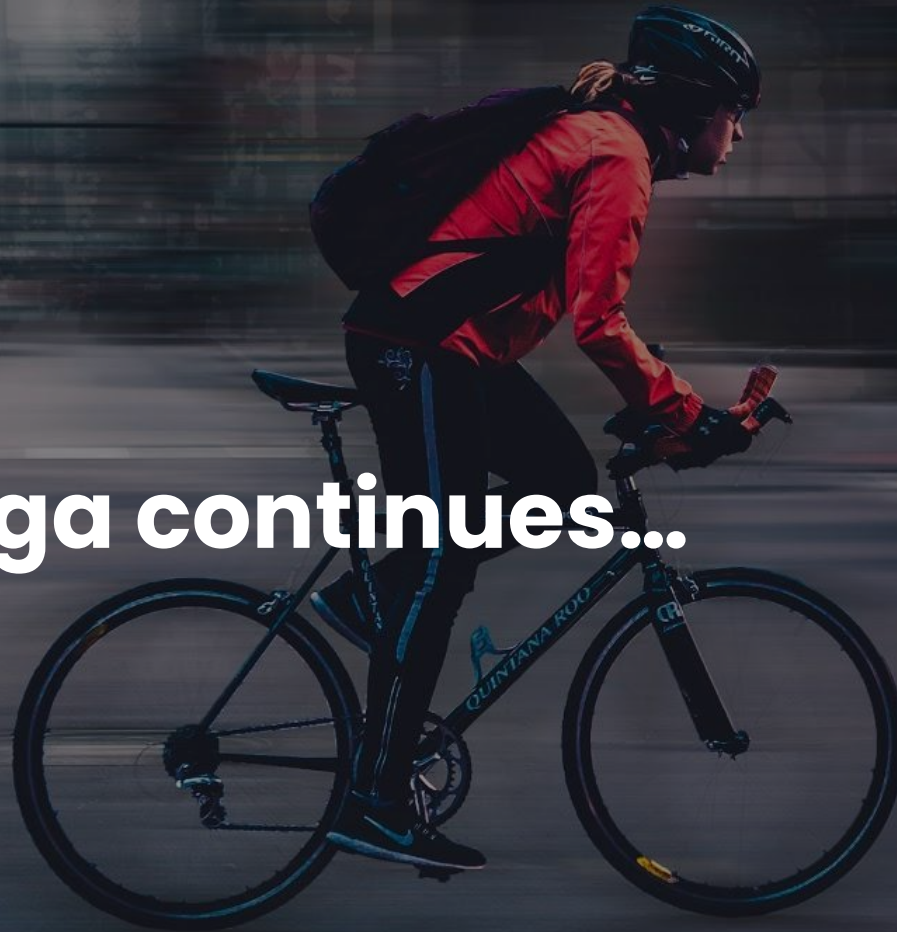
Nope.



Next Problem: How to treat stacked orders?



The saga continues...



What is the moral of the story?



Sometimes, you are forced to conduct state-of-the-art, cutting-edge research using optimization and machine learning techniques to solve real-life problems just because...

... some random dude ordered a Bolognese at 10AM | Mamma mia!





Thank you for your attention!