minservice = 0
If the minimum service requirement is set to zero, the model basically allows for no real services to be scheduled at all. From what I observed, many solutions end up assigning almost all trucks to the dummy truck (DD). In these cases, no maintenance tasks are scheduled, and all start times are simply zero. Naturally, this causes both the maintenance load and service time to drop to zero, which means the objective function hits its lowest possible value. The reason behind this is quite straightforward: without any minimum service to enforce, the optimizer prefers to avoid using real trucks to cut down on cost and maintenance. because of the dummy truck doesn't require maintenance or service time, all the associated constraints become irrelevant or are just ignored. So, this means that this solution reflects the scenario where resources are used as little as possible, without any effort to balance workload or optimize the schedule.

minservice = 100
On the other hand, when the minimum =100, the model basically has no choice but to assign all services to real trucks and none go to the dummy truck. This means every truck is running at full capacity. You can see that both the maintenance load and service time increase a lot compared to when the minimum service is zero. The main reason for this is that the model prioritizes meeting the service quota above everything else, even if it means trucks have to go through more frequent or longer maintenance. To squeeze all the required work in, the optimizer often schedules the maximum number of tasks per truck and tends to pick shorter maintenance types so everything fits within the schedule.

minwork = 0
Setting minimum work per truck to zero grants the optimizer flexibility in workload distribution. Most trucks still perform multiple services, but some may be lightly loaded or idle. The dummy truck is seldom used, and maintenance tasks are generally assigned with the shortest possible durations unless other constraints necessitate longer ones. Maintenance load and service times vary widely depending on how the optimizer balances competing constraints. On the other hand, a few trucks can't handle a lot of workload, resulting in high maintenance loads.

minwork = 100
In this scenario, the model outcome indicates it is infeasible (UNSAT), as it fails to identify any valid schedule. The stipulation that each truck must manage a minimum of 100 work units often surpasses the available resources or clashes with additional constraints such as maintenance windows or the total number of services. This results in the optimizer being unable to allocate sufficient work to each truck without violating established rules, leading the solver to conclude that no valid solution is available

maxsty = [1,1,1,0]
Limiting each actual truck type to a single service type drastically reduces scheduling flexibility. Given the diverse array and number of services required, this limitation obstructs the distribution of all tasks without breaching the maximum service type restriction. Services that cannot be assigned to real trucks are redirected to a dummy truck, which does not contribute to the completion of services. If a minimum percentage of services is mandated, the model becomes unfeasible due to unmet service coverage obligations.

maintdur[SH] > maintdur[LG]
When the maintenance duration for SH (Short) exceeds that of LG (Long), the model remains solvable. I have observed that the scheduling process can still accommodate all services and maintenance as long as the constraints are met. However, the maintenance strategy changes as SH maintenances now take longer. The optimizer tends to minimize the use of SH maintenances, preferring LG or MJ types whenever possible. As a result, trucks may spend less overall time on maintenance, or the availability of service time slots may slightly decline due to the lengthened maintenance durations.

maxsep[SH] > maxsep[LG]
If the permitted interval between SH maintenances (maxsep[SH]) is set longer than for LG maintenances (maxsep[LG]), the model demands LG or MJ maintenances to occur more regularly than SH maintenances. This configuration contradicts standard maintenance procedures but remains feasible. In practice, the optimizer schedules LG maintenances more frequently, while SH maintenances become less common or optional, leading to maintenance schedules that feature more frequent larger maintenances and fewer smaller ones.

maxsep[MJ] > maxsep[LG]
As LG maintenances must take place more often than MJ maintenances, the LG maintenance interval serves as the more stringent constraint. This effectively dictates the maintenance timetable, rendering the less stringent MJ interval inconsequential. The model consistently adheres to the stricter LG maintenance frequency requirements.

maxsep[SH] = maxsep[LG] = endtime
Establishing both SH and LG maximum separation intervals equal to the entire scheduling horizon (endtime) considerably relaxes maintenance frequency constraints. In this scenario, the model only needs to schedule one SH and one LG maintenance for each truck throughout the designated period, regardless of schedule duration. This results in a lower number of maintenances overall, typically just one of each type per truck, thereby easing the maintenance burden. Although mathematically valid, such solutions may be less practical in real-world situations.

prework = [0 | t in TRUCK]
When the prework array is initialized to zero for all trucks, it indicates that each truck has just finished a significant maintenance activity at the scheduling start. As a result, there is no accumulated workload influencing the timing of initial maintenance. This "fresh start" simplifies the constraints related to maintenance scheduling, concentrating only on future maintenance requirements based on scheduled services. Generally, this leads to reduced maintenance loads and more straightforward schedules compared to situations where trucks have prework accumulated.

majorwork > endtime
If the majorwork parameter exceeds the total scheduling horizon, it implies that trucks are past due for major maintenance. Consequently, the model usually schedules major or long maintenances early on, sometimes right at the beginning of the planning period. This action decreases scheduling flexibility and frequently raises maintenance load and service time. In contrast to lower majorwork values, this leads to a maintenance schedule loaded at the front with less balance and fewer opportunities for optimization.

maxsty[ty] = 0 for some ty other than DUMMY
When the maximum permissible services for a truck type (apart from DUMMY) is established as zero, trucks of that type are essentially barred from being assigned to services. As a result, all service tasks must be managed by the remaining truck types. Output results indicate that dummy trucks remain unassigned, while the eligible trucks often bear higher maintenance loads. Although the total maintenance burden and service time may experience a slight increase due to this exclusion, the model retains its feasibility as long as the other trucks possess adequate capacity. If they do not, the model becomes infeasible. This evaluation is significant for testing the robustness of the model under limited resources.

start[s] = 0 and end[s] = endtime for some service s.
Allocating a service to occupy the complete scheduling timeline from start to finish effectively prevents the resource (e.g., truck) from being available for that period. This rule stops any other service or maintenance from being scheduled on the same truck. If several services are necessary, this situation leads to conflicts that cannot be resolved, creating an unsatisfiable model (UNSAT).

start includes k services with the same start and end time where k = card(T RUCK).
Scheduling k services with identical start and end times — one for each truck—typically results in direct unsatisfiability. This indicates that all trucks would have to perform their services at the same time, potentially instantaneously, which is impractical. Realistic scheduling models need to ensure that services on the same truck do not overlap and that total service

time meets minimum requirements. Overlapping services that are identical do not contribute to an increase in cumulative service time, thus breaching these constraints and making the model infeasible.

card(STYPE) = 1

The flexibility of the concept is severely limited when all services are restricted to a single category (such as gas). To balance workload and satisfy maintenance requirements, a number of limitations rely on allocating jobs among different service types. Fulfilling these limitations can become impossible with only one service type available, making the business unfeasible.

summary

An analysis of these scenarios reveals that the two constraints most significantly impacting the model's viability are:

Maximum Allocation of Service Type (maxsty).

This fundamental limitation restricts the number of services each vehicle can perform. When demand surpasses the allowed types of service, an excessively low limit constrains flexibility and often leads to infeasibility.

Minimum Quantity of Work and Service (minwork, minservice).

These criteria establish the minimum amount of work and services that must be provided. If set too high relative to the system's capacity and maintenance capabilities, they can create unfeasibility by demanding more than what can be managed effectively.

These constraints effectively define the upper and lower limits of supply and demand in the scheduling problem. Modifying parameters alone rarely resolves conflicts between these thresholds. Experimental findings indicate that increasing minservice or tightening maxsty usually results in significant shifts in scheduling results, often leading to a lack of viable options.