



Bölüm 7: Yakınsama Algoritmaları

Algoritmalar



Yakınsama Algoritmaları

- Bazı problemlerin çözümlerini bulmak çok zor veya imkansız olabilir.
- Yakınsama algoritmaları,
 - Mükemmel çözümü bulmak yerine,
 - Orijinal çözüme mümkün olduğunca yakın çözümler üretir.
 - Karmaşık problemler makul bir süre içerisinde çözülebilir.



Yakınsama Algoritmaları

- Polinom zaman karmaşıklığına sahiptir.
- Orijinal çözüme belli bir oran veya yüzde ile yakın çözümler üretir.
- Hesaplama kaynakları kısıtlı olduğunda tercih edilir.
- Terazinin bir kefesine bir ağırlık koyulduğunda,
 - diğer kefeye tam olarak aynı ağırlığı koymak zor olabilir.
 - Bu durumda, çeşitli küçük ağırlıklar kullanılarak,
 - büyük ağırlığa olabildiğince yakın bir denge kurulabilir.



Yakınsama Algoritmalarının Uygulamaları

- Seyir satıcısı (Traveling salesman) problemi:
 - Seyyar satıcının en kısa sürede en fazla müşteriye ulaşmasını sağlayan rotaya karar vermek.
- Sırt çantası (Knapsack) problemi:
 - Çantaya sığacak, maksimum fayda sağlayacak eşyaların seçilmesi.
- İşlemci (Job scheduling) planlama:
 - Bilgisayarda birden fazla sürecin en verimli şekilde yürütülmesi.

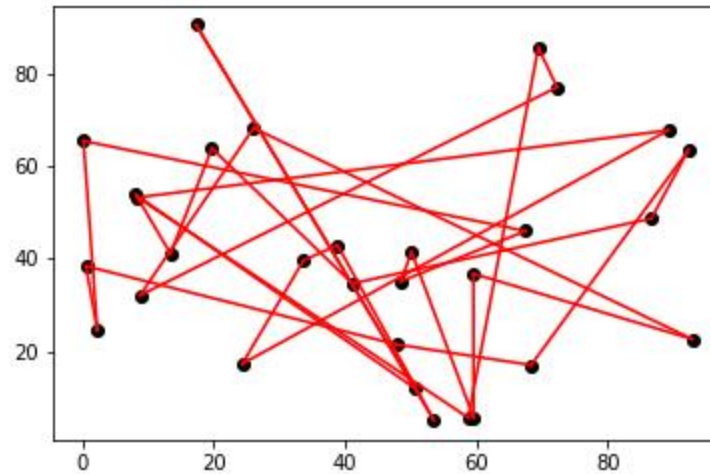


Yakınsama Algoritmaları Türleri

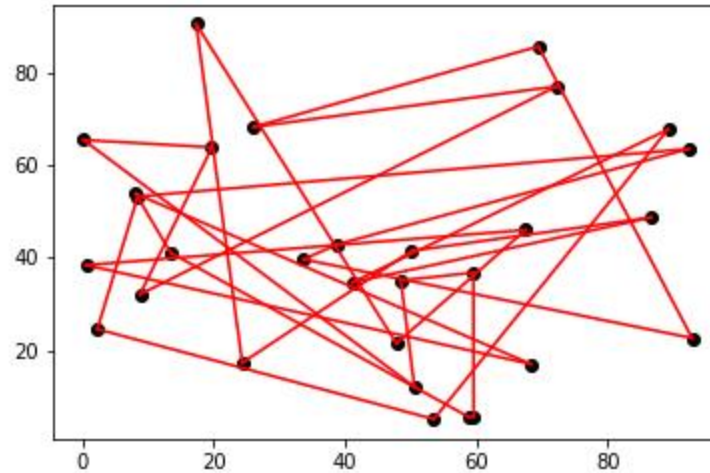
- Yakınsama:
 - Orijinal problemin daha basit bir versiyonunu çözülür.
- Açgözlü (Greedy):
 - Her adımda en iyi görünen seçenek seçilerek ilerlenir.
- Yerel Arama (Local search):
 - Mevcut çözüm küçük değişikliklerle iyileştirilir, daha iyi çözüm bulunur.



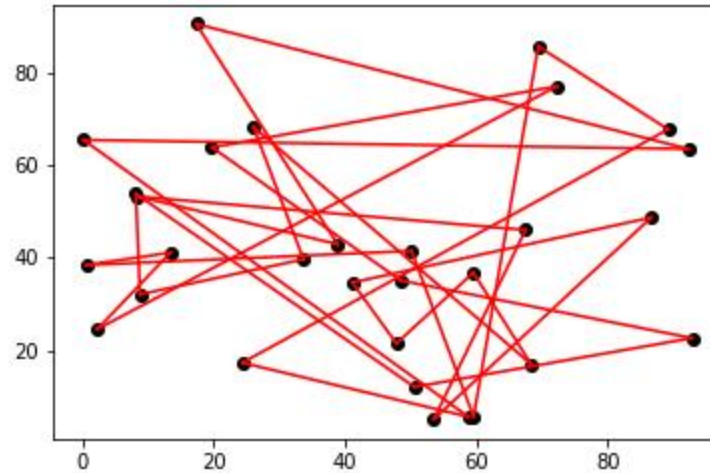
Traveling Salesman



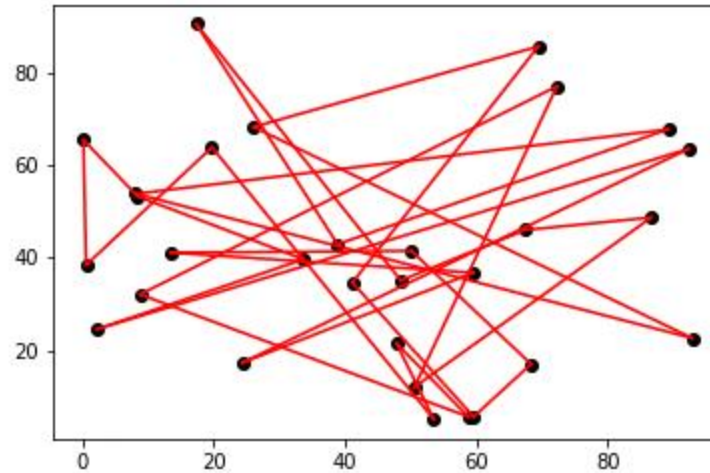
Traveling Salesman



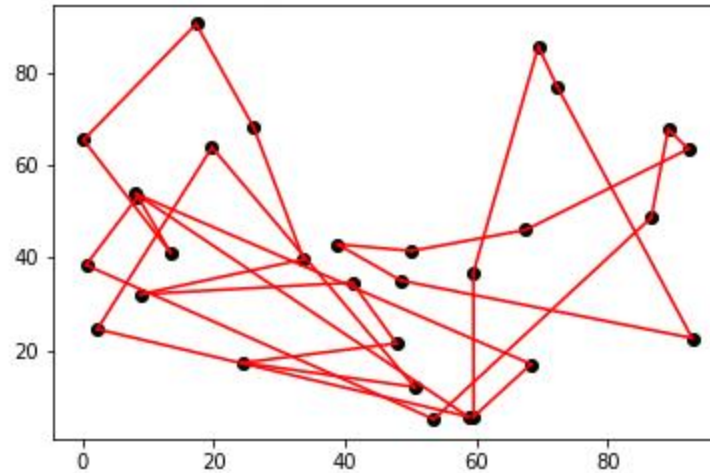
Traveling Salesman



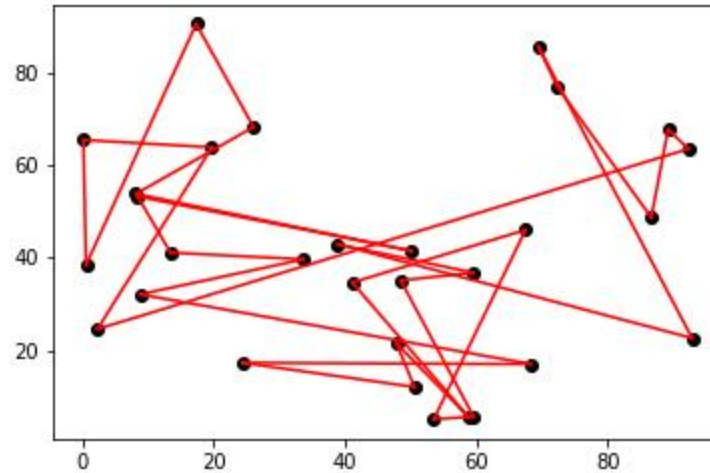
Traveling Salesman



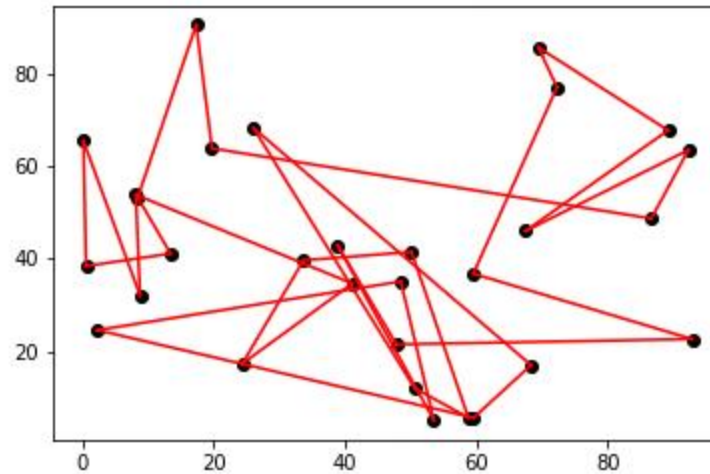
Traveling Salesman



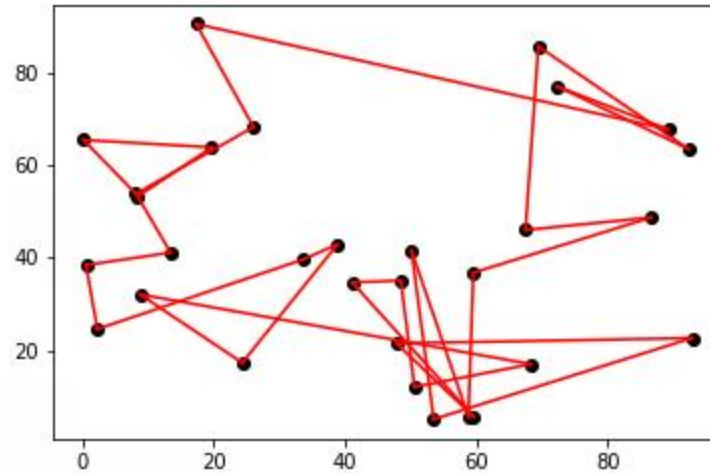
Traveling Salesman



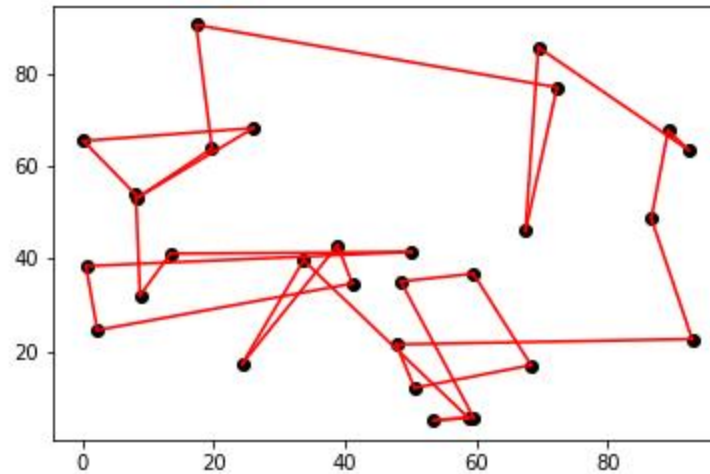
Traveling Salesman



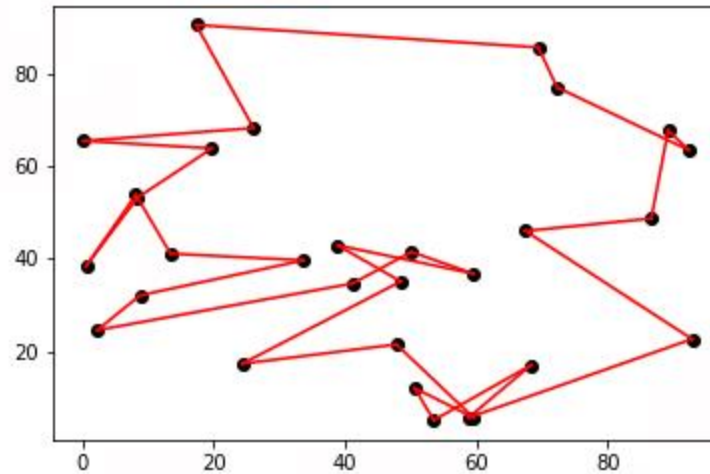
Traveling Salesman



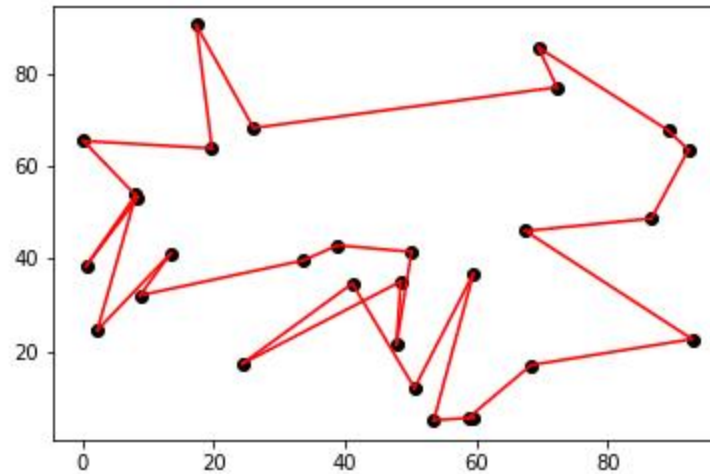
Traveling Salesman



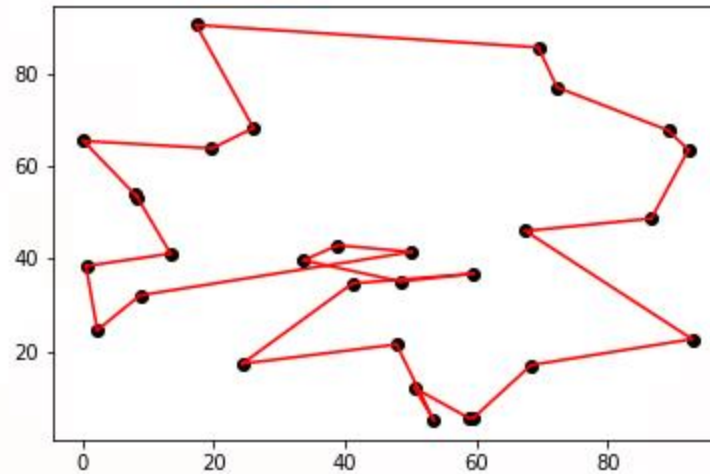
Traveling Salesman



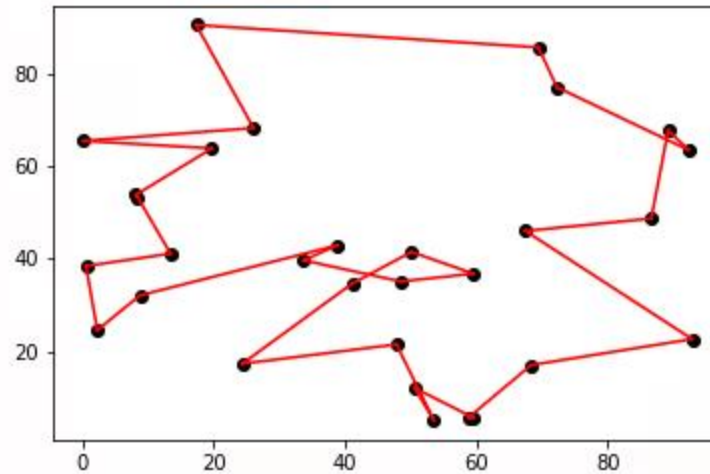
Traveling Salesman



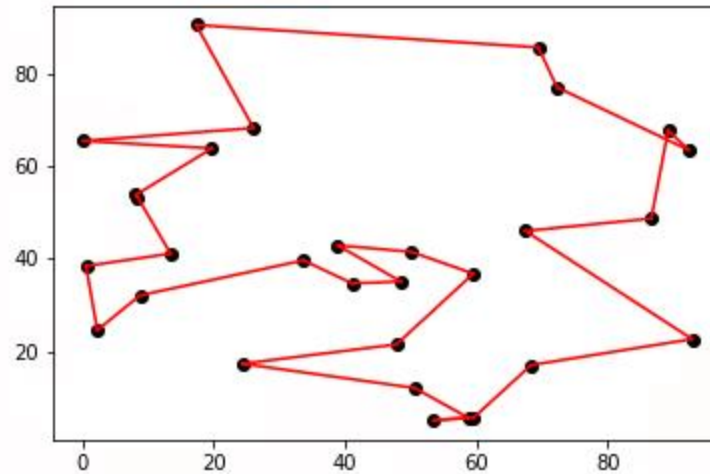
Traveling Salesman



Traveling Salesman

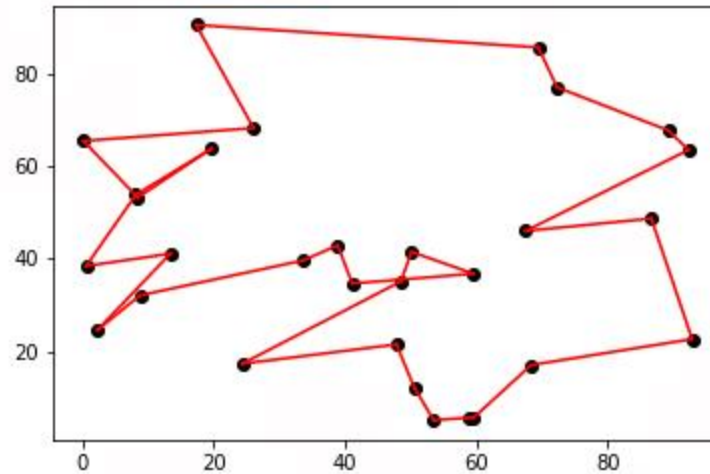


Traveling Salesman

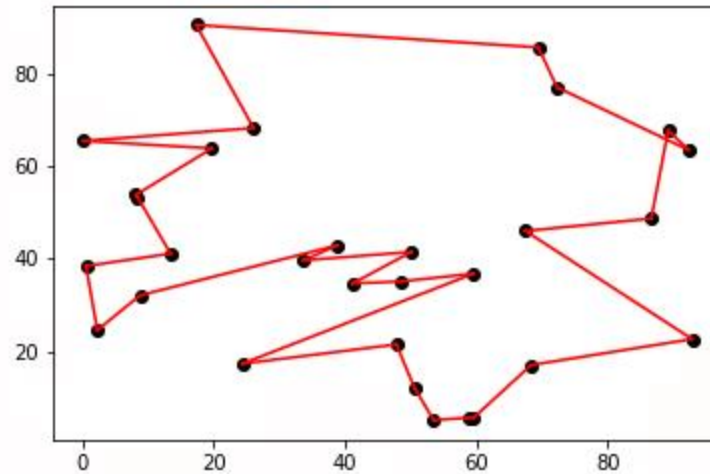




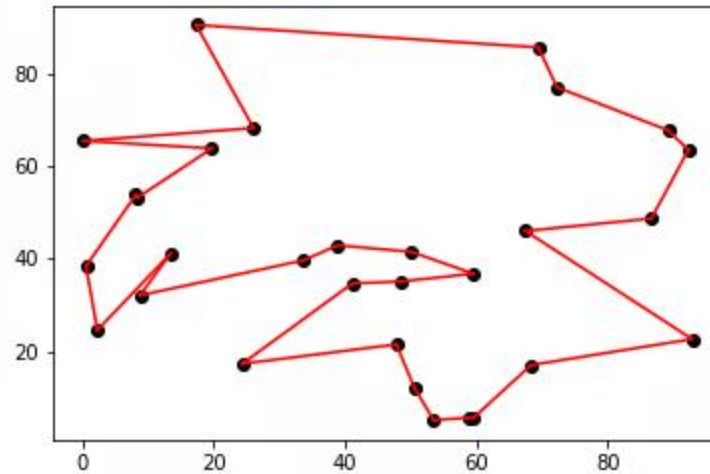
Traveling Salesman



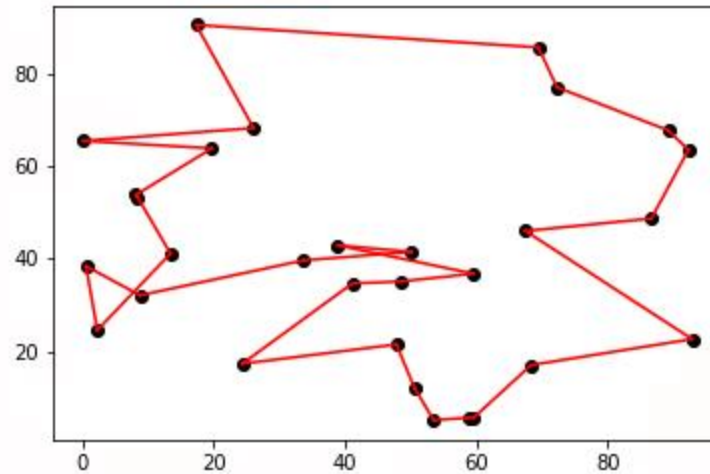
Traveling Salesman



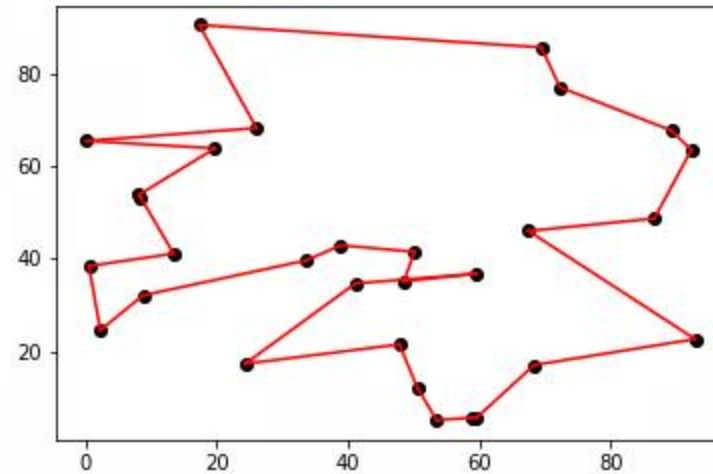
Traveling Salesman



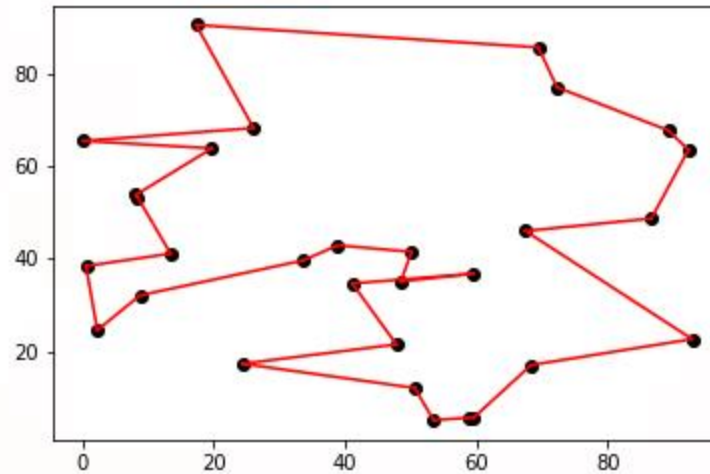
Traveling Salesman



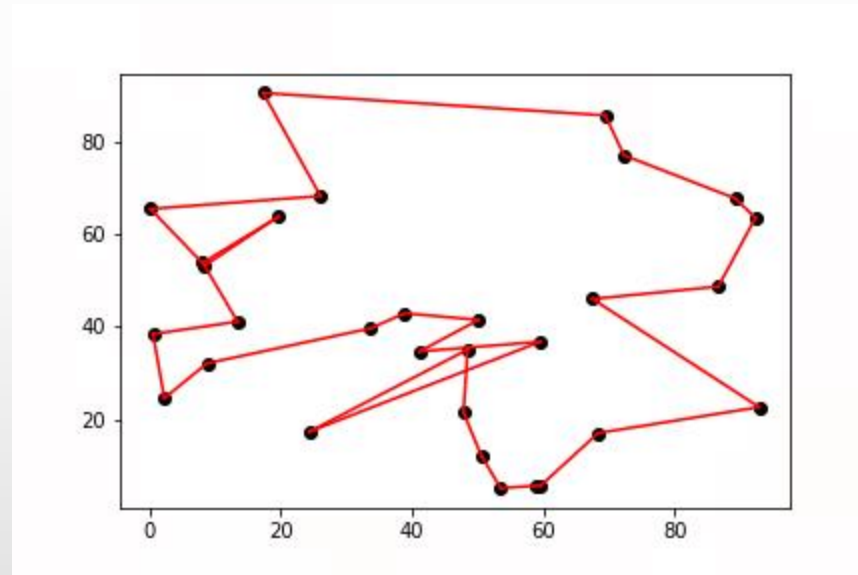
Traveling Salesman



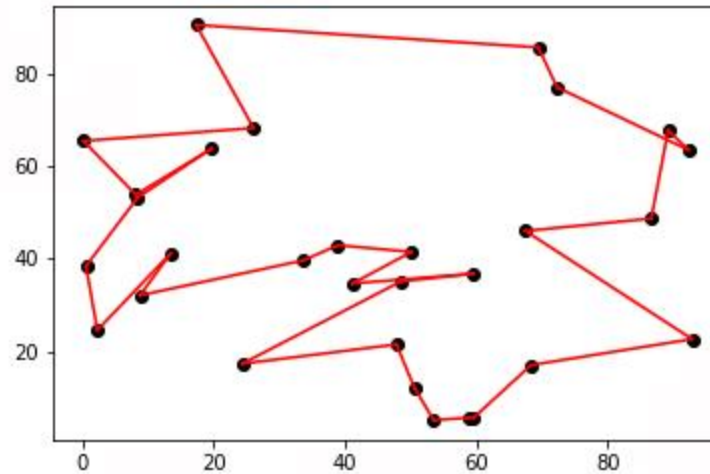
Traveling Salesman



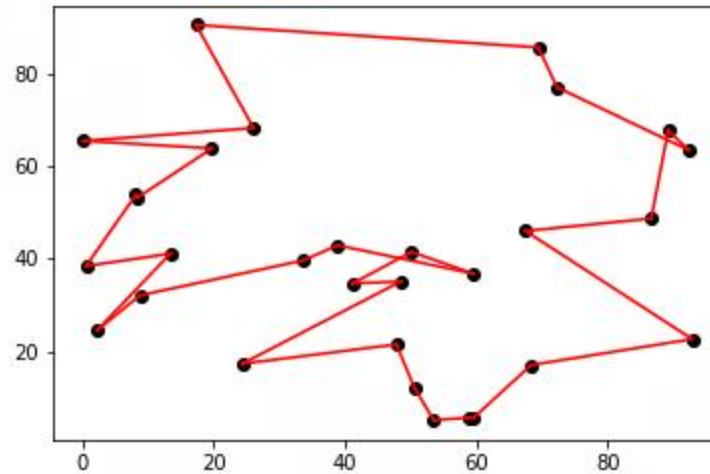
Traveling Salesman



Traveling Salesman



Traveling Salesman

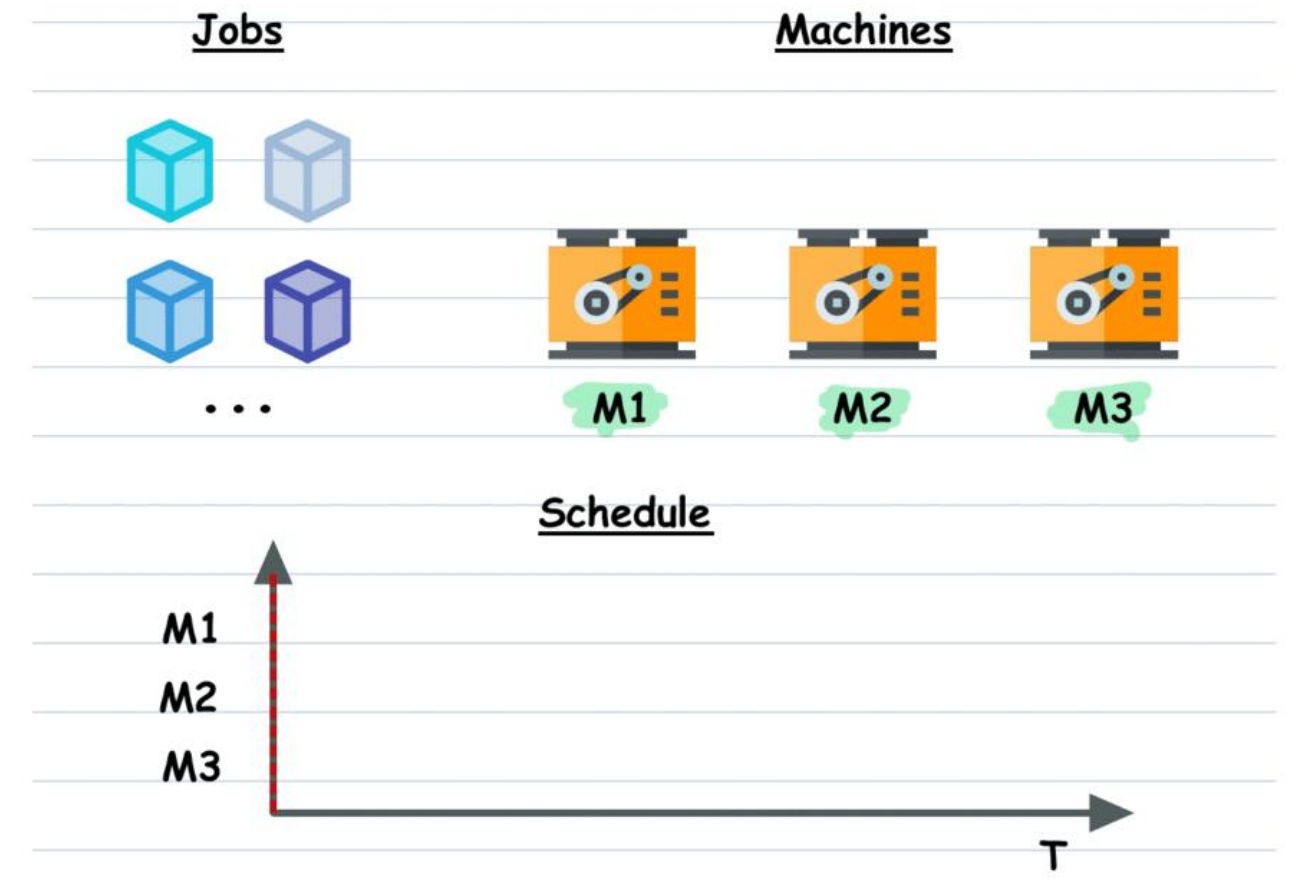






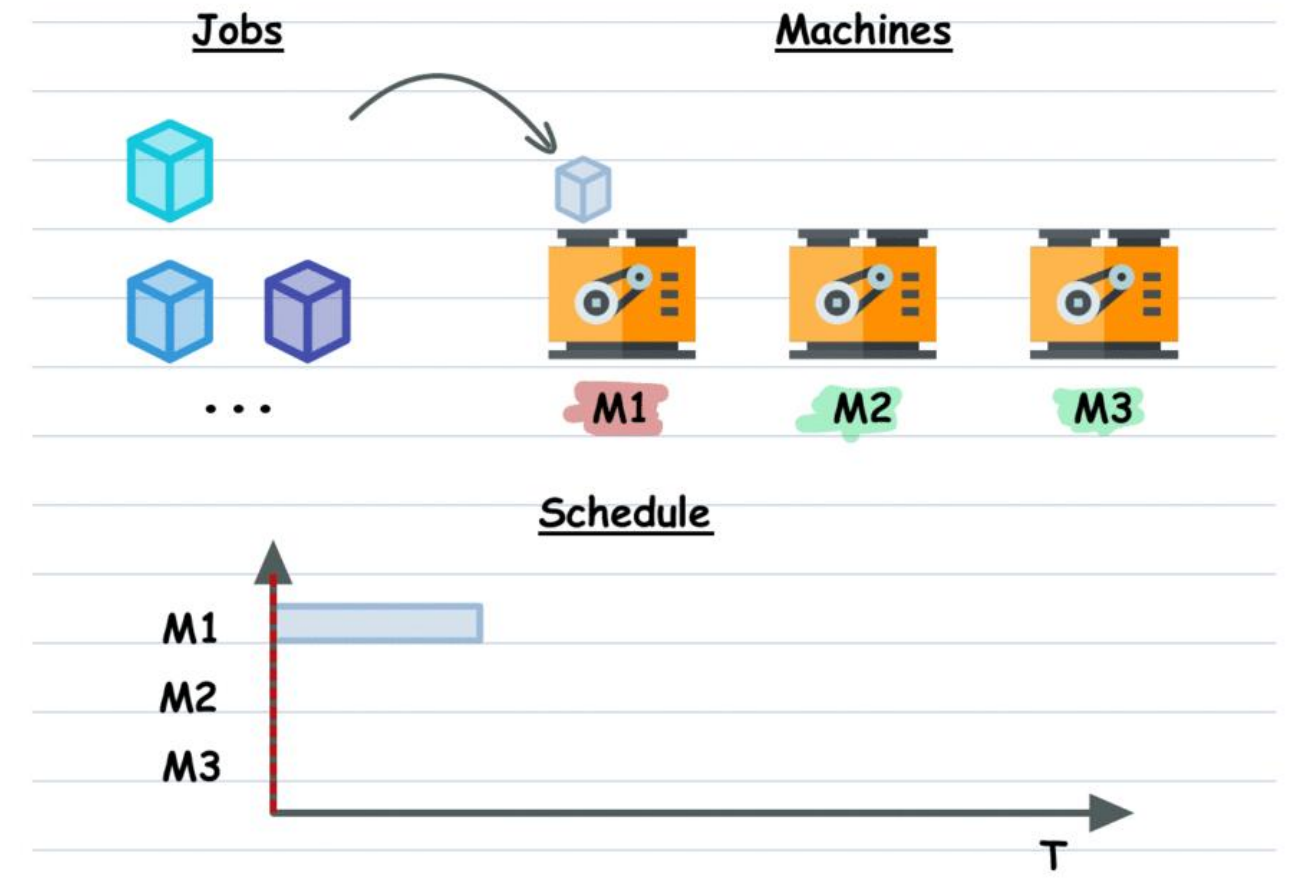


Job Scheduling



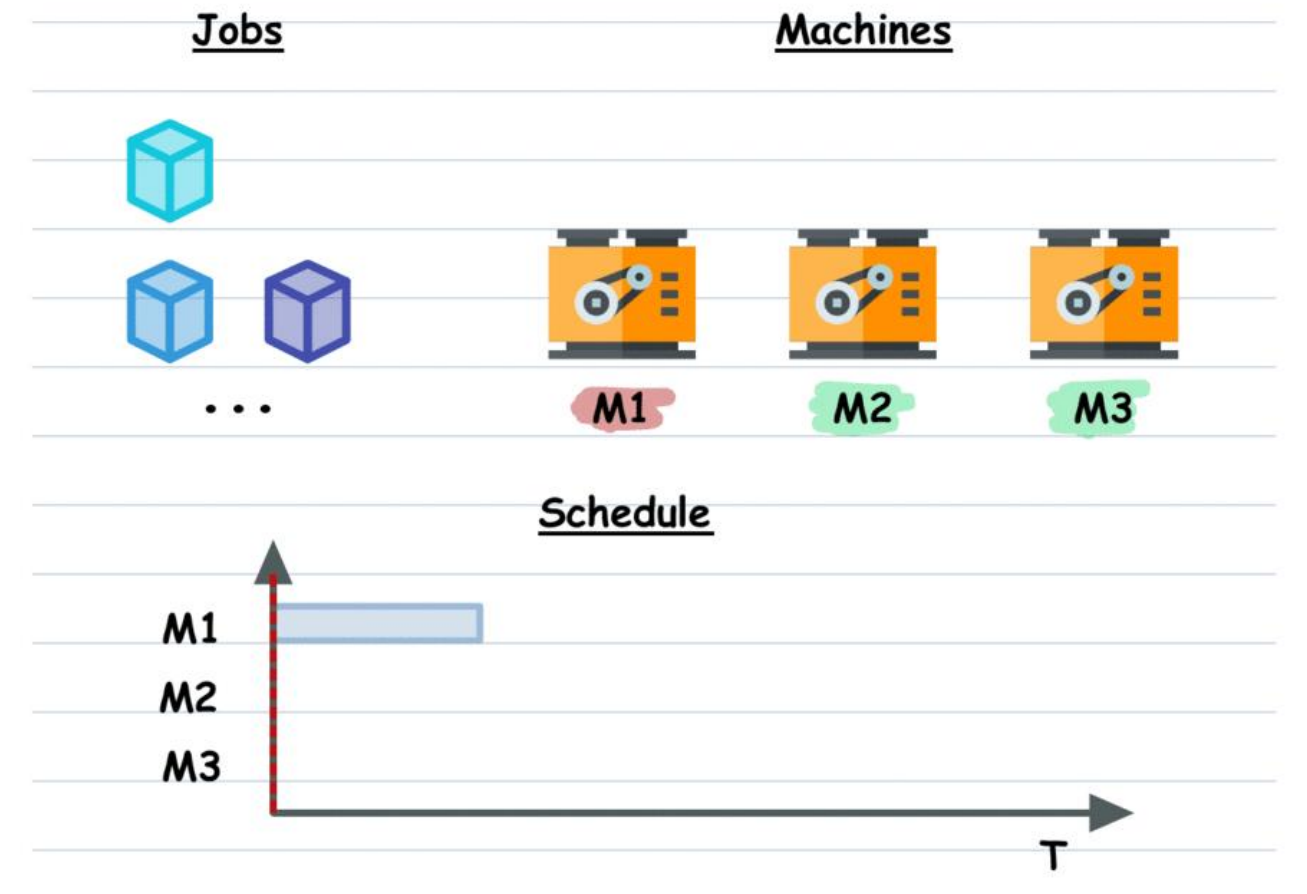


Job Scheduling



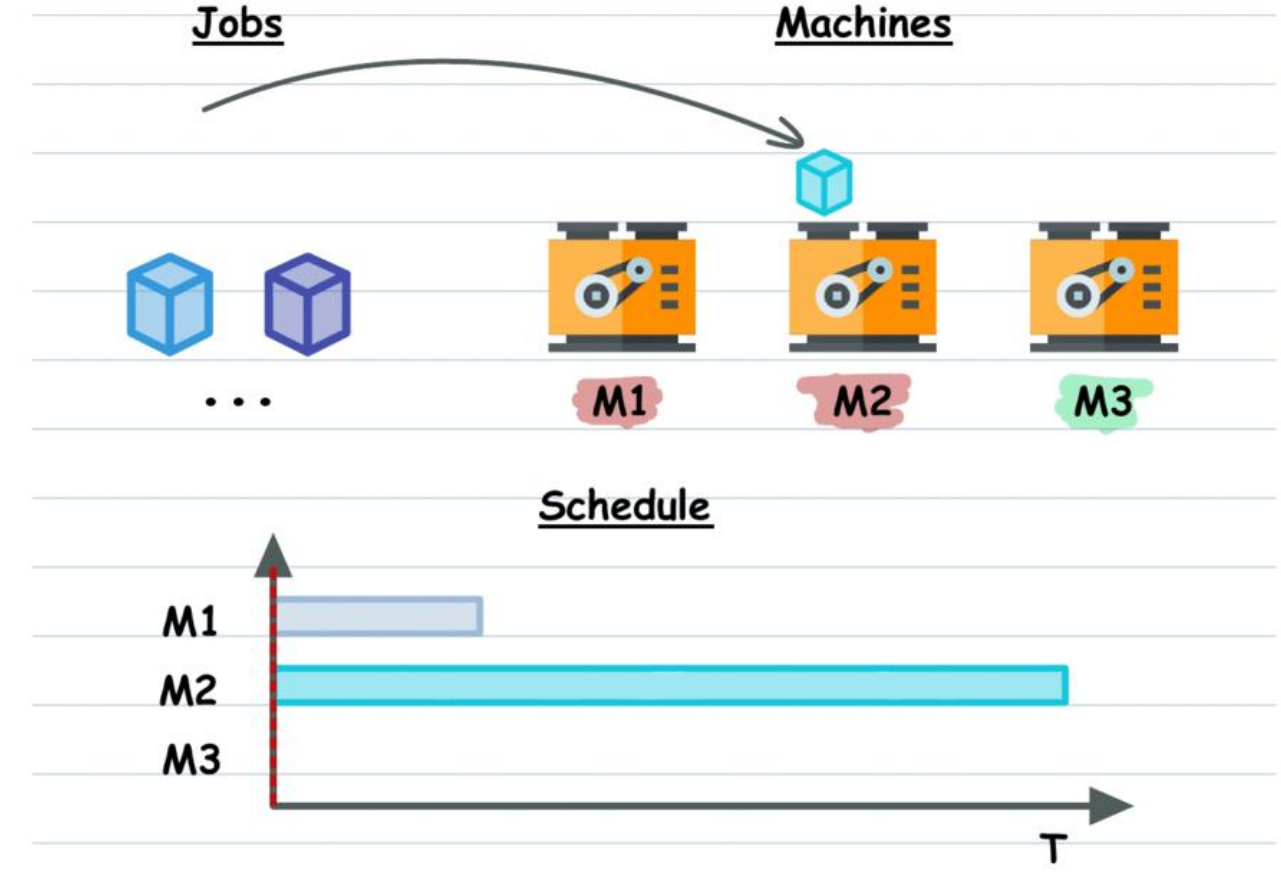


Job Scheduling



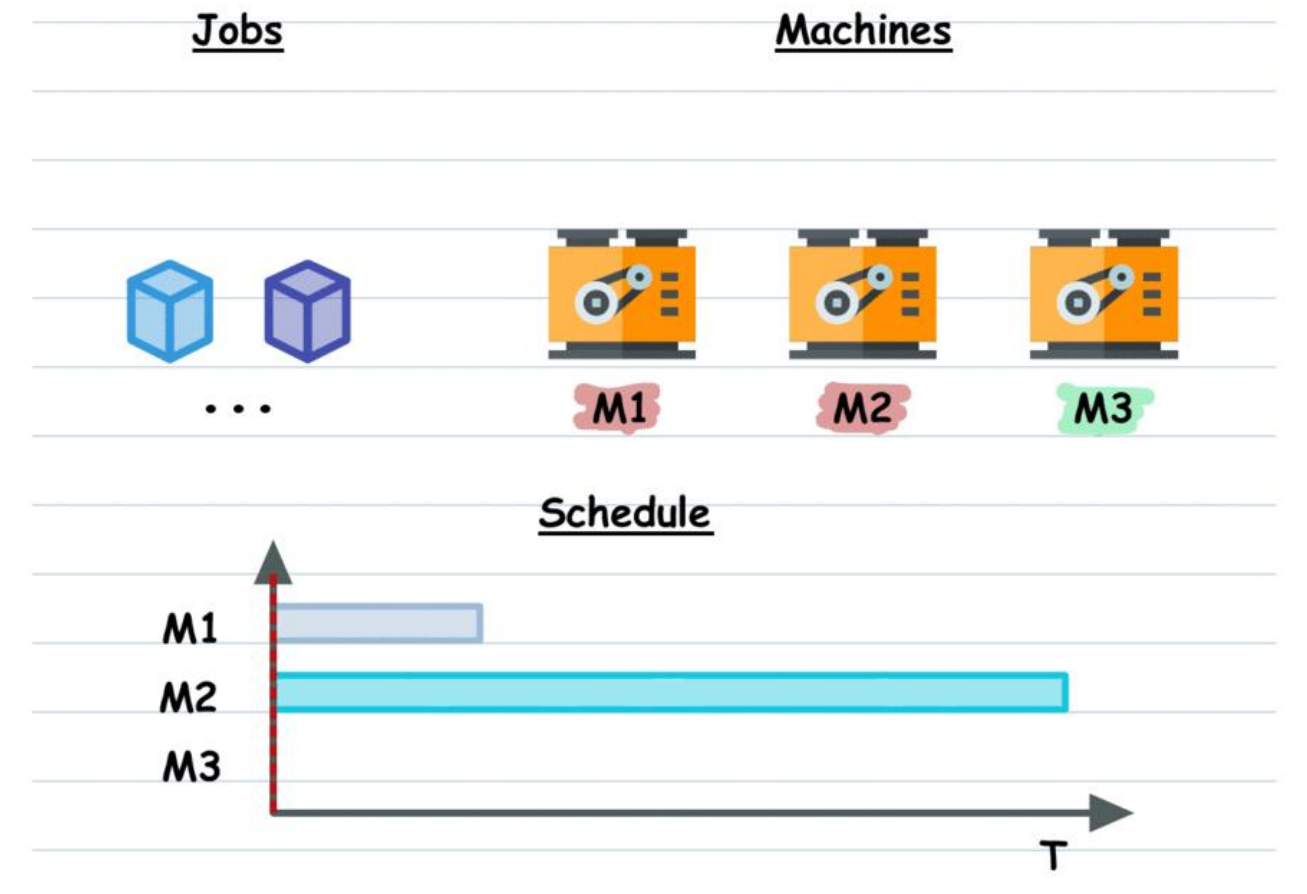


Job Scheduling

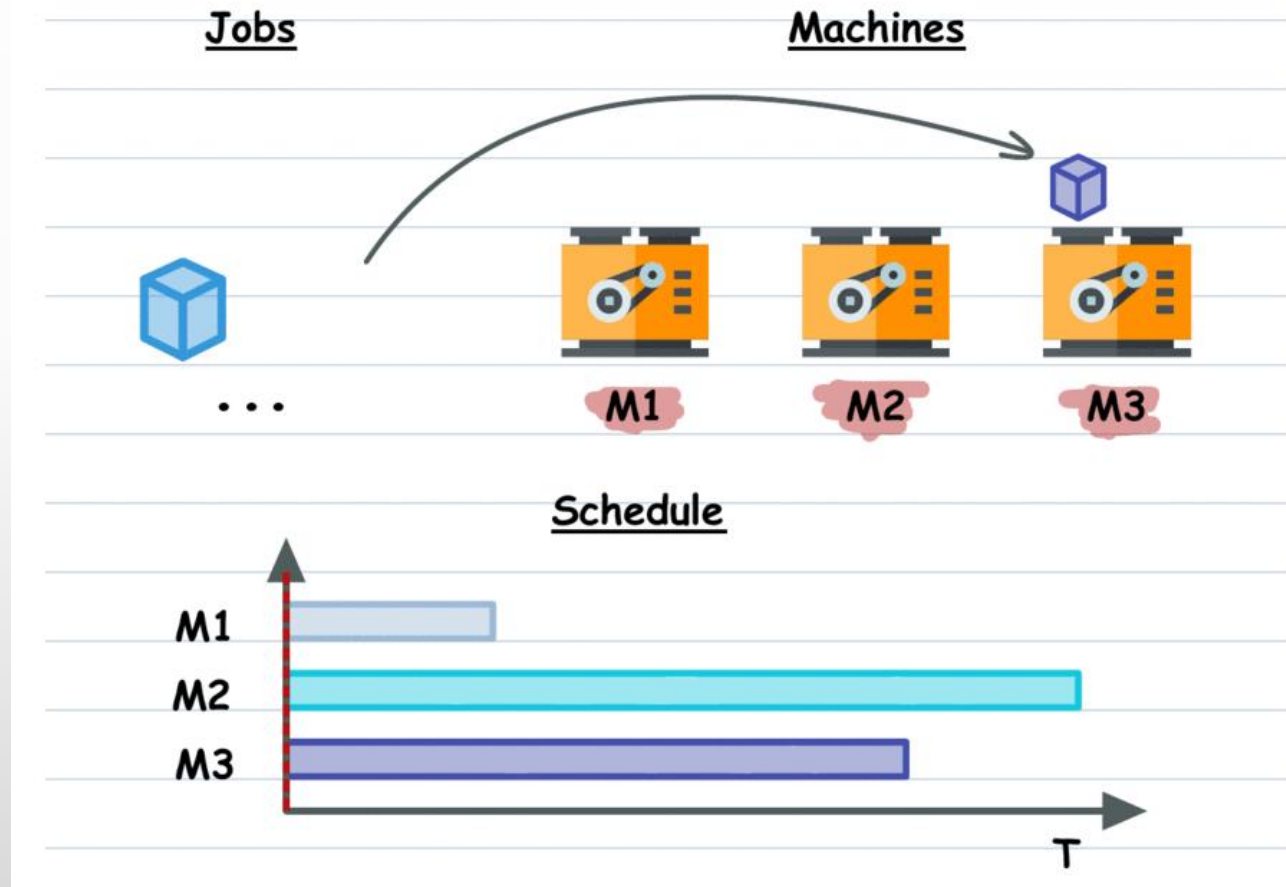




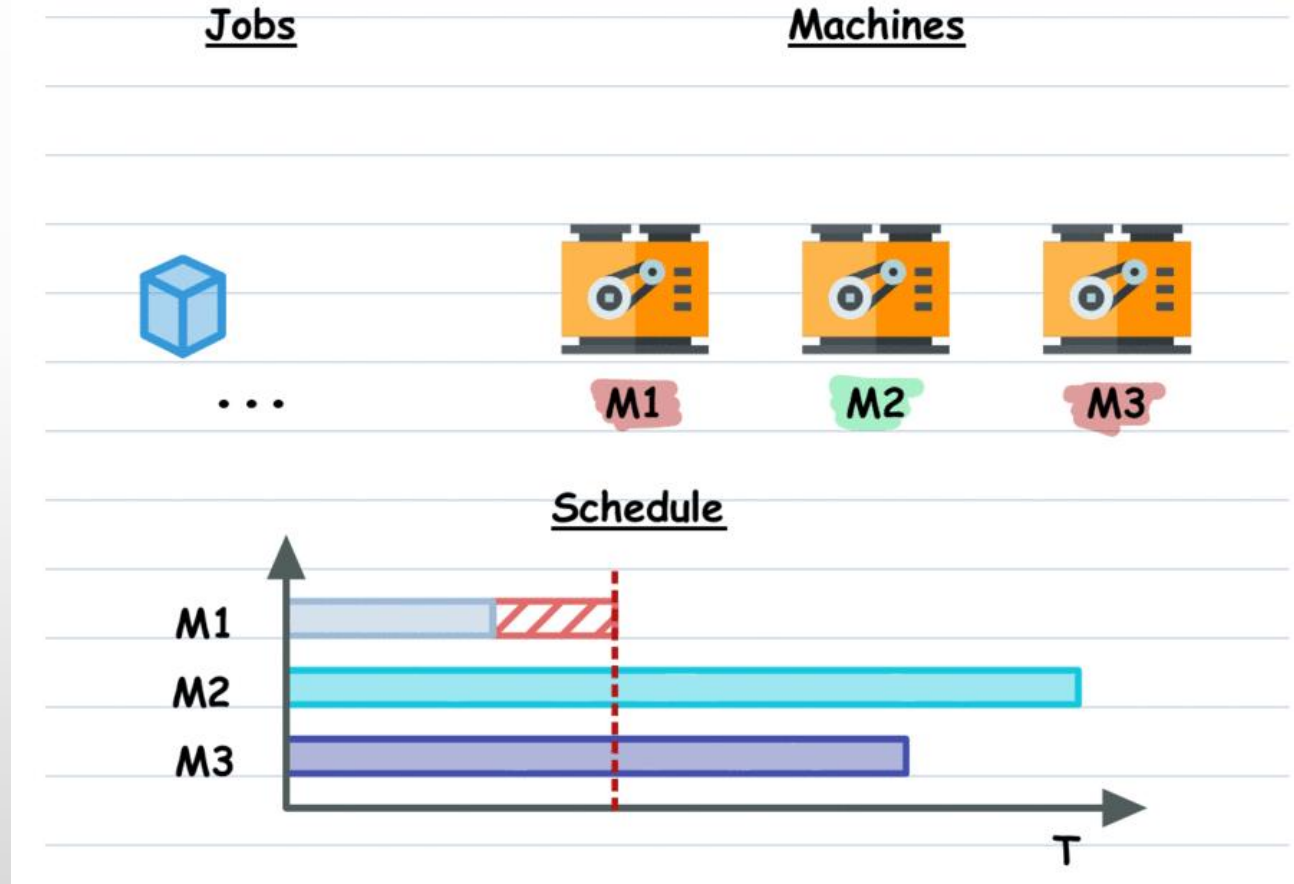
Job Scheduling



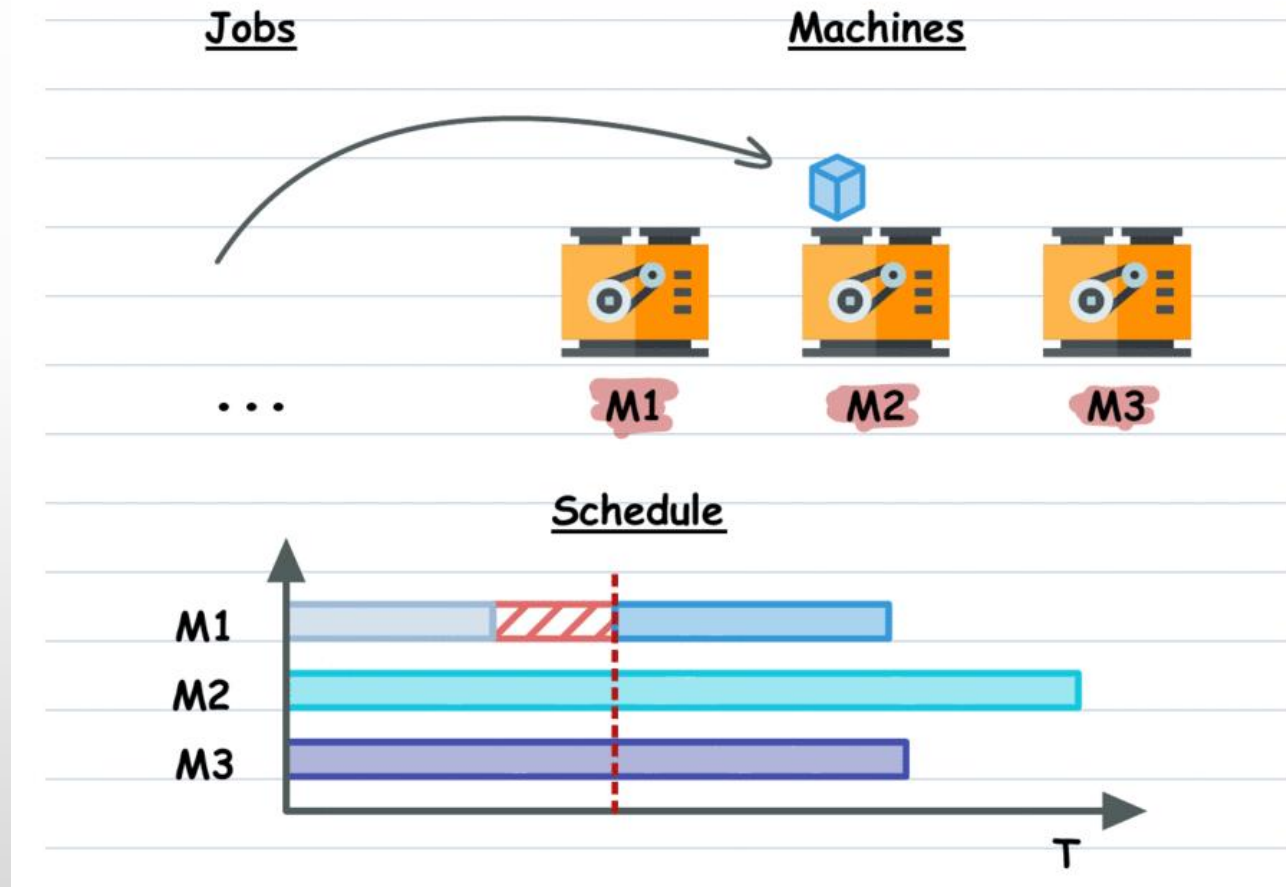
Job Scheduling



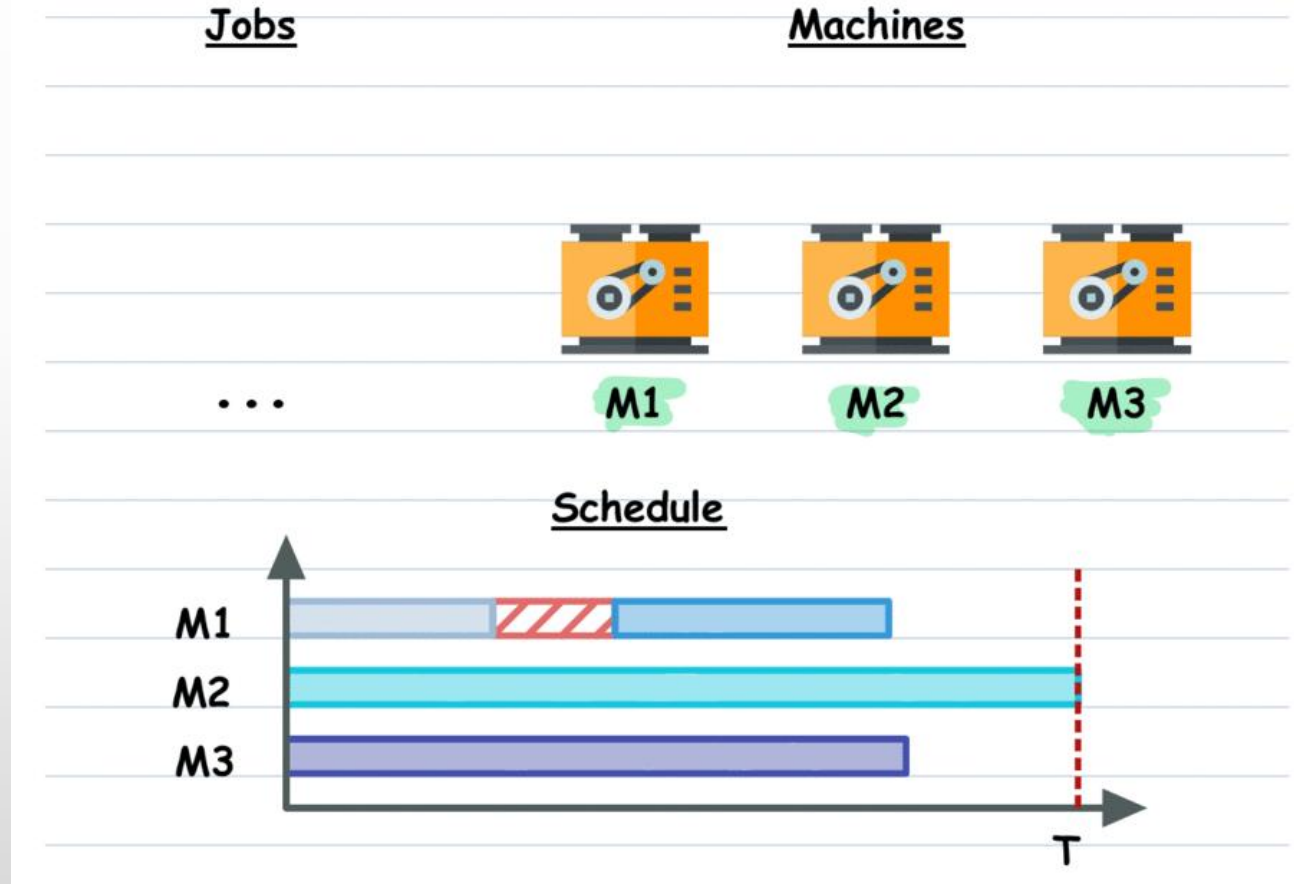
Job Scheduling



Job Scheduling



Job Scheduling







Job Scheduling

Job ID	Profit	Deadline
3	40	2
4	30	2
1	20	4
2	10	1

-1	-1	-1	-1
----	----	----	----



Job Scheduling

Job ID	Profit	Deadline
3	40	2
4	30	2
1	20	4
2	10	1

-1	3	-1	-1
----	---	----	----

Profit = 40



Job Scheduling

Job ID	Profit	Deadline
3	40	2
4	30	2
1	20	4
2	10	1

4	3	-1	-1
---	---	----	----

Profit = 40 + 30



Job Scheduling

Job ID	Profit	Deadline
3	40	2
4	30	2
1	20	4
2	10	1

4	3	-1	1
---	---	----	---

Profit = 40 + 30 + 20



Job Scheduling

Job ID	Profit	Deadline
3	40	2
4	30	2
1	20	4
2	10	1

4	3	-1	1
---	---	----	---

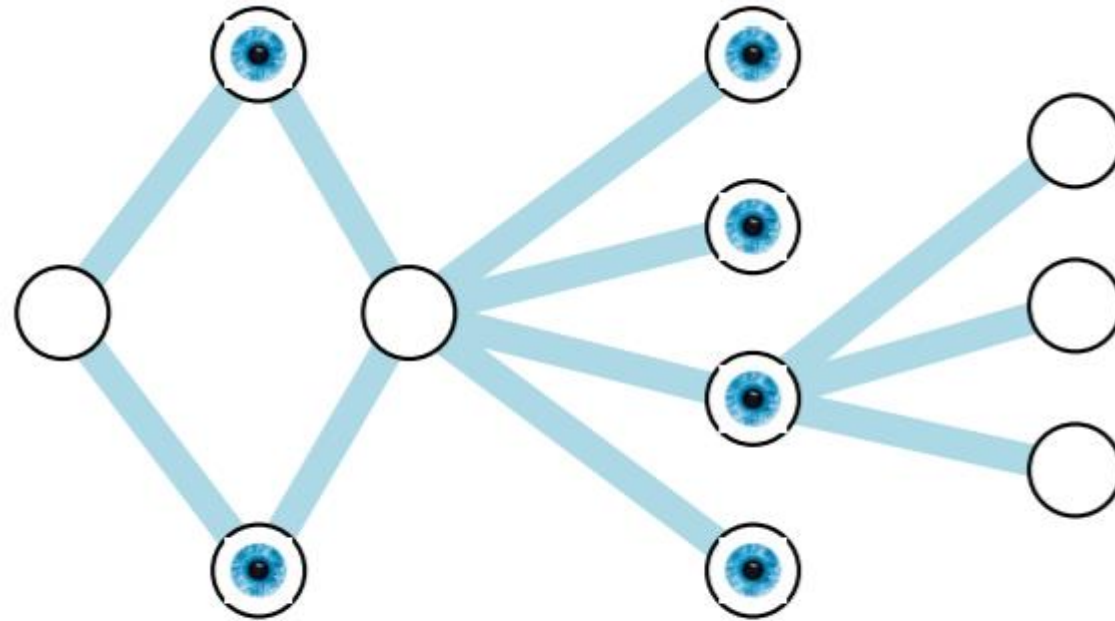
$$\text{Profit} = 40 + 30 + 20 + 0^*$$

*cannot be performed



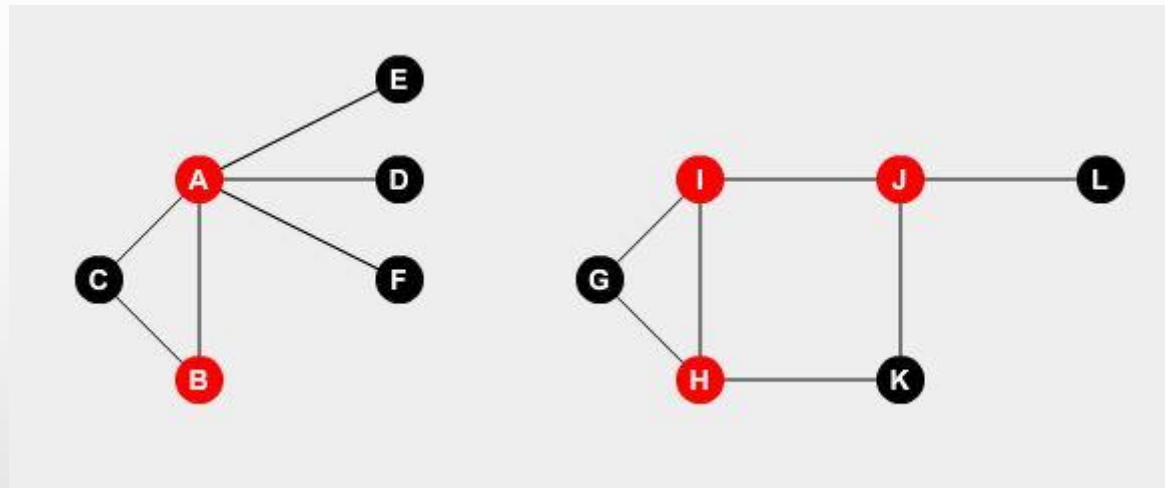


Vertex Cover

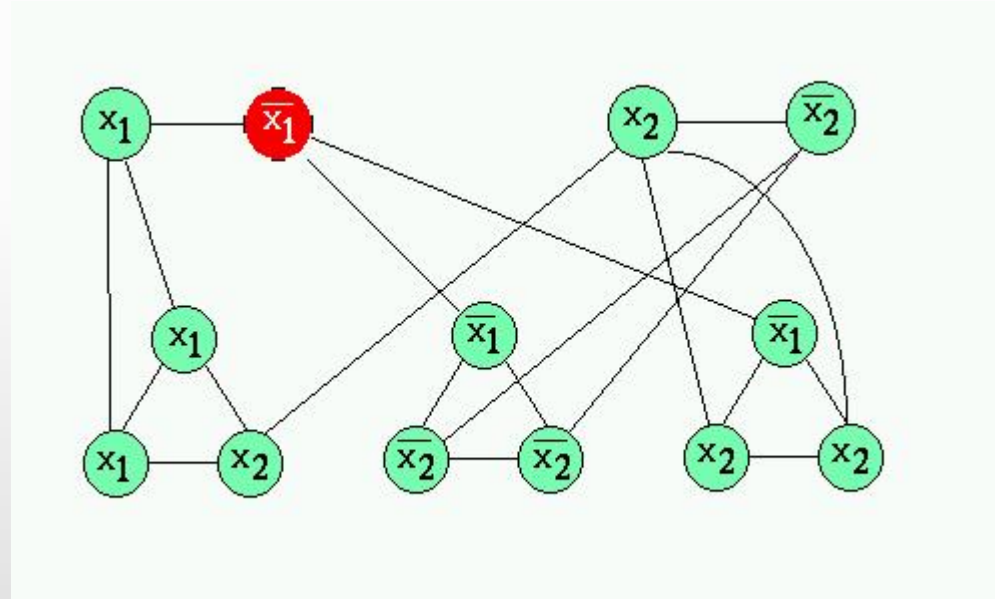


A vertex cover of size 6

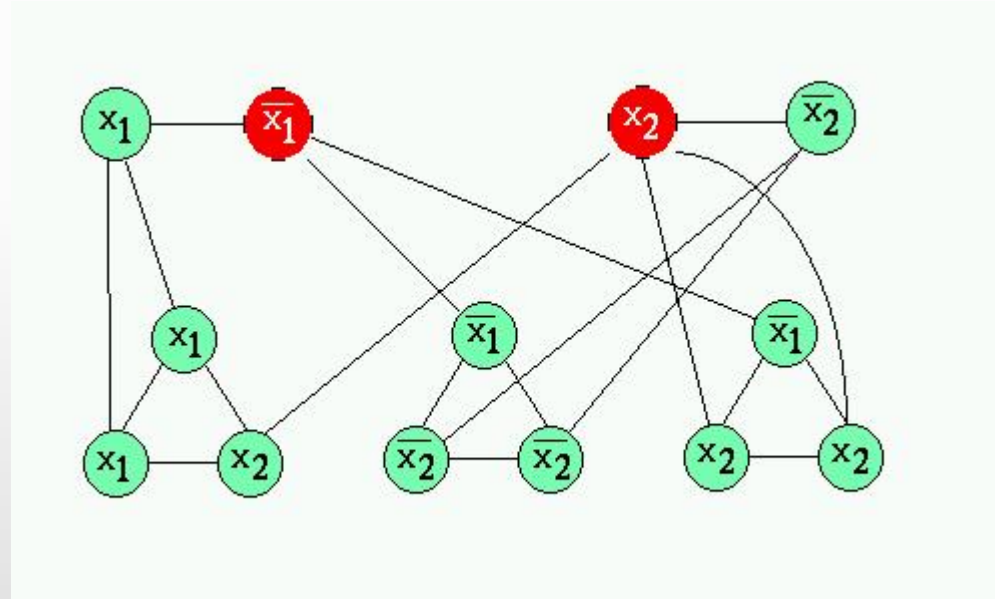
Vertex Cover



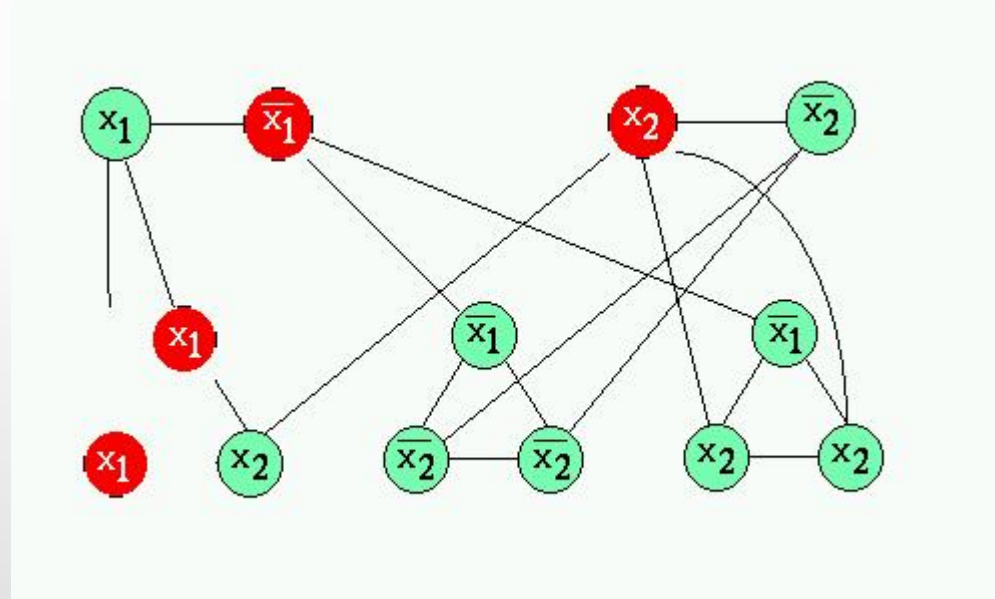
Vertex Cover



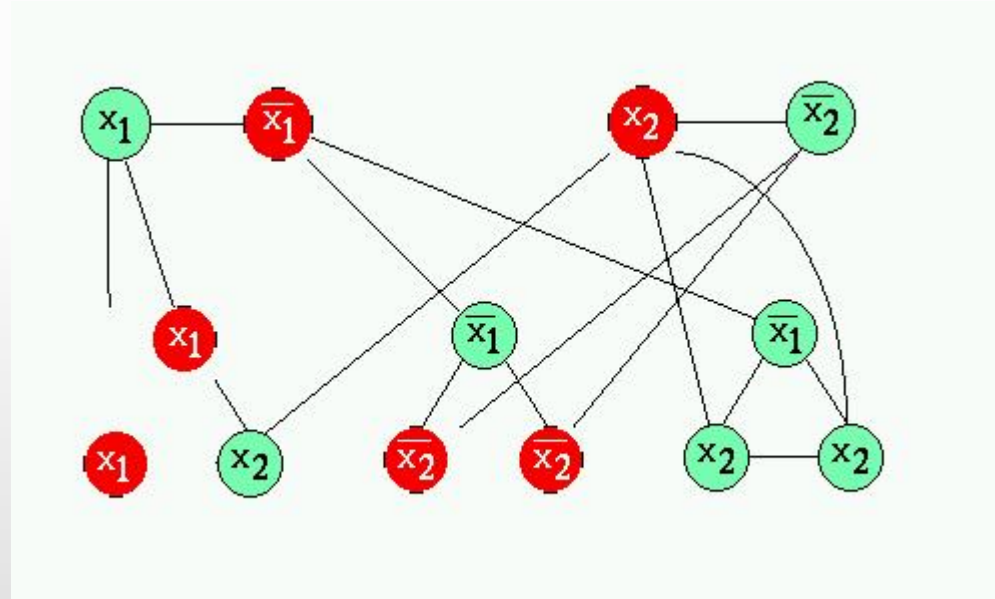
Vertex Cover



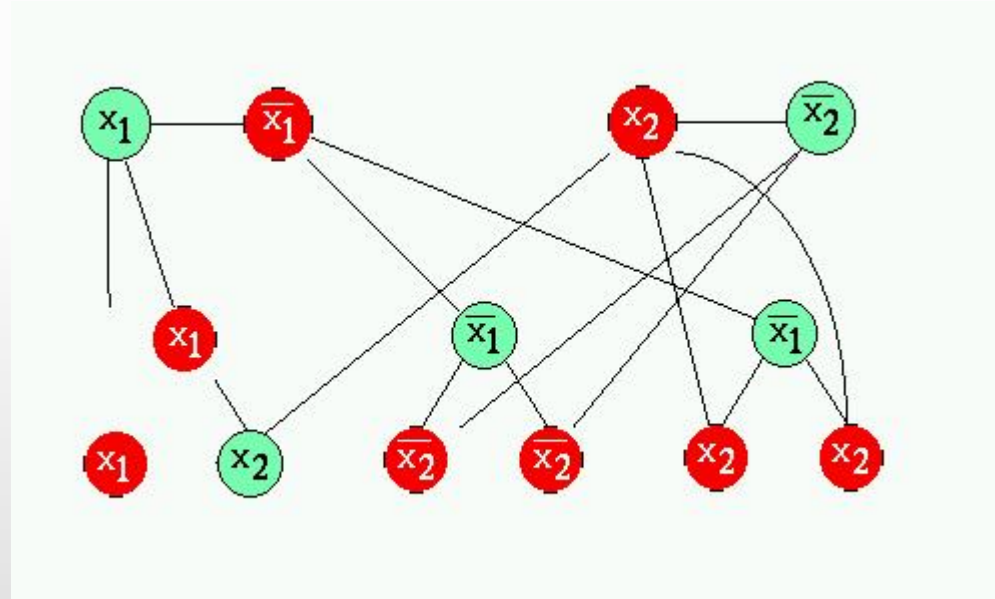
Vertex Cover



Vertex Cover



Vertex Cover





SON