1. #net = modello
2. arcs = net.arcs # lista di archi di M
3. places = net.places # lista di place di M
4. transitions = net.transitions # lista di transition di M
5. a = alignemnt(log,trace) # log composto dalle trace in cui occorre la sub
6. **for** pos=pos\_start to pos\_end # pos\_start=posizione in "a" di start,
7. # pos\_end=posizione in "a" di end
8. **if** move[pos] == [L]
9. **if** move[pos-1] == [L/M]
10. **if** p1!=p2 per ogni p1 **in** uscita da h1, per ogni p2 **in** uscita da h2
11. places.add(pl1)
12. arcs.add(t[pos-1], pl1)
13. transitions.add(h1)
14. arcs.add(pl1,h1)
15. transitions.add(h2)
16. arcs.add(pl1,h2)
17. arcs.add(h2,p2) per ogni p2 **in** uscita da t[pos-1]
18. arcs.add(h1,p1) per ogni p1 **in** ingresso a t[pos]
19. arcs.delete(t[pos-1], p2) per ogni p2 **in** uscita da t[pos-1]
20. **if** move[pos+1] == [L/M]
21. transitions.add(h3)
22. arcs.add(p3,h3)
23. **if** deleted\_arcs > 1
24. arcs.add(h3, pl1)
25. **else**
26. arcs.add(h3,p2)
27. **if** move[pos+1] == [L]
28. **if** p4 != p4 per ogni p4 **in** uscita da h3,h4
29. places.add(pl2)
30. arcs.add(t[pos], pl2)
31. transitions.add(h3)
32. arcs.add(pl2,h3)
33. transition.add(h4)
34. arcs.add(pl2,h4)
35. arcs.add(h3,p4) per ogni p4 **in** uscita da t[pos]
36. arcs.add(h4,p4) per ogni p4 **in** ingresso a t[pos+1]
37. arcs.delete(t[pos], p4) per ogni p4 **in** uscita da t[pos]

PS: Nell’immagine HRRAN è indicato come t[pos+1], ma è da considerare come t[pos+1] nel secondo giro di for in cui t[pos] diventa AHRRPC e non quando t[pos] è SHRRPC. Nella condizione in riga 28 è indicato solo p4 perché nell’immagine di riferimento c’è solo p4 ma sarebbero da considerare come place diversi, poi nel caso di quella immagine sono uguali e quindi la condizione sarebbe falsa.