

---

**Algorithm CNR**

---

**Input:** A state  $q \in Q_R(q_0)$ ,  $D_1$ ,  $D_2$ ,  $D_3$ , and  $S$ ;

**Output:**  $\Phi$ ; /\*  $\Phi$  stores the number of required raw parts for all part types \*/

```
1: Set  $q(\lambda) = 1$ ;
2: for ( $j = 1 : n$ ) do
3:    $\Phi[j] = 0$ ;
4: end for
5: for ( $i = 1 : m$ ) do /*set the capacity of resources to the infinity*/
6:    $\psi_i = \text{INF}$ ;
7: end for
8: for ( $i = 1 : n$ ) do /*clear all parts not at stages in  $D_1$  to avoid disturbance */
9:   for ( $j = 1 : l_i$ ) do
10:    if ( $P_{ij} \notin D_1$ )
11:       $q(x_{ij}) = 0$ ;
12:       $q(y_{ij}) = 0$ ;
13:    else /*suppose that all parts have finished their current operations*/
14:       $q(x_{ij}) = q(x_{ij}) + q(y_{ij})$ ;
15:       $q(y_{ij}) = 0$ ;
16:    end if-else
17:   end for
18: end for
19: for ( $i = 1 : |D_1|$ ) do /*advance those parts that can be advanced to their proper positions*/
20:   Suppose  $D_1[i] = P_{jk}$ ,  $D_3[i] = P_{vw}$ ;
21:   while ( $D_2[i] > 0$ ) do
22:     if (the part at stage  $D_1[i]$  can be advanced to  $D_3[i]$ )
23:       Let  $\zeta$  be the sequence of stages which are gone through from  $D_1[i]$  to  $D_3[i]$ ;
24:       Advance a part at stage  $D_1[i]$  to  $D_3[i]$ ;
25:       for ( $t = 1 : c$ ) do /*update  $q$ ,  $D_1$  and  $D_2$ */
26:         let  $\Xi = P_{at} \cap \zeta$ ;
27:         for ( $P_{ef} \in \Xi$ ) do
28:           for ( $P_{gh} \in \{\Pi(P_{ef}) \setminus \zeta\}$ ) do
29:              $q(x_{gh}) = q(x_{gh}) - 1$ ;
30:             find  $d \in \{1, 2, \dots, |D_1|\} \ni P_{gh} = D_1[d]$ ;
31:              $D_2[d] = D_2[d] - 1$ ;
32:             if ( $D_2[d] = 0$ )
33:               set  $D_1[d] = P_{vw}$ ;
34:             end if
35:           end for
36:         end for
37:          $q(x_{jk}) = q(x_{jk}) - 1$  and  $q(x_{vw}) = q(x_{vw}) + 1$ ;
38:          $D_2[i] = D_2[i] - 1$ ;
39:         if ( $D_2[i] = 0$ )
40:            $D_1[i] = P_{vw}$ ;
41:         end if
42:       end for
43:     esle
44:       find  $c \in \{1, 2, \dots, l_j - k\}$  such that  $P_{j(k+c)} \in \mathfrak{R}_{jk}$ ,  $P_{j(k+c)} \in \Pi_a$  and  $P_{j(k+w)} \notin \Pi_a$ 
45:        $\forall w \in \{1, 2, \dots, c-1\}$ ;
46:        $D_1[i] = P_{j(k+c)}$ ;
47:        $q(x_{jk}) = q(x_{jk}) - D_2[i]$  and  $q(x_{j(k+c)}) = q(x_{j(k+c)}) + D_2[i]$ ;
48:       break;
49:     end if-else
50:   end while
51: end for
52: for ( $i = 1 : c$ ) do /* starts the computation from the first class assembly operations */
53:   for ( $P_{jk} \in P_{ai}$ ) do
54:     for ( $t = 1 : |D_1|$ ) do
55:       if ( $D_1[t] \in \Pi(P_{jk})$ )
56:         Let  $z = \max\{q(x_{vw}) + q(y_{vw}) : P_{vw} \in \Pi(P_{jk})\}$ ;
57:         for ( $P_{vw} \in \Pi(P_{jk})$ ) do
```

---

---

**CNR (continued)**

```
57:      Let  $\zeta(P_{vw}) = \Theta(P_{vw}) \cap P_b$ ;  
58:      for ( $P_{ob} \in \zeta(P_{vw})$ ) do  
59:          Let  $\Phi[o] = \Phi[o] + z - q(x_{vw}) - q(y_{vw})$ ;  
60:      end for  
61:       $q(x_{vw}) = 0$ ;  
62:      for ( $f = 1 : |D_1|$ ) do  
63:          if ( $f \neq t$  and  $D_1[f] \in \Pi(P_{jk})$ )  
64:               $D_2[f] = 0$ ;  
65:               $D_1[f] = D_3[f]$ ;  
66:          end if  
67:      end for  
68:      end for  
69:       $q(x_{jk}) = q(x_{jk}) + z$ ; /* assemble the parts at stages in  $\Pi(P_{jk})$   $z$  times */  
70:       $D_1[t] = P_{jk}$ ;  
71:       $D_2[t] = z$ ;  
72:      while ( $D_2[t] > 0$ ) do  
73:          Suppose  $D_3[t] = P_{vw}$ ;  
74:          if (a part at stage  $D_1[t]$  can be advanced to  $D_3[t]$ )  
75:              /*advance the obtained parts one by one to  $D_3[t]$  */  
76:              Let  $\zeta$  be the sequence of stages which are gone through from  $D_1[t]$   
77:              to  $D_3[t]$ ;  
78:              Advance a part at stage  $D_1[t]$  to  $D_3[t]$ ;  
79:              for ( $d = 1 : c$ ) do /*update  $q$ ,  $D_1$  and  $D_2$ */  
80:                  let  $\Xi = \{P_{ad} \cap \zeta\}$ ;  
81:                  for ( $P_{ef} \in \Xi$ ) do  
82:                      for ( $P_{gh} \in \{\Pi(P_{ef}) \setminus \zeta\}$ ) do  
83:                           $q(x_{gh}) = q(x_{gh}) - 1$ ;  
84:                          find  $o \in \{1, 2, \dots, |D_1|\} \ni P_{gh} = D_1[o]$ ;  
85:                           $D_2[o] = D_2[o] - 1$ ;  
86:                          if ( $D_2[o] = 0$ )  
87:                               $D_1[o] = D_3[t]$ ;  
88:                          end if  
89:                      end for  
90:                  end for  
91:                   $q(x_{jk}) = q(x_{jk}) - 1$  and  $q(x_{vw}) = q(x_{vw}) + 1$ ;  
92:                   $D_2[t] = D_2[t] - 1$ ;  
93:                  if ( $D_2[t] = 0$ )  
94:                       $D_1[t] = D_3[t]$ ;  
95:                  end if  
96:              else  
97:                  find  $c \in \{1, 2, \dots, l_j - k\}$  such that  $P_{j(k+c)} \in \mathfrak{R}_{jk}$ ,  $P_{j(k+c)} \in \Pi_a$  and  
98:                   $P_{j(k+w)} \notin \Pi_a \forall w \in \{1, 2, \dots, c - 1\}$ ;  
99:                   $D_1[t] = P_{j(k+c)}$ ;  
100:                   $q(x_{jk}) = q(x_{jk}) - D_2[t]$ ;  
101:                   $q(x_{j(k+c)}) = q(x_{j(k+c)}) + D_2[t]$ ;  
102:                  break;  
103:              end if-else  
104:          end while  
105:      end if  
106:  end for  
107:  end for  
108:  return  $\Phi$ ;  
109: End
```

---

**Theorem 1:** The complexity of Algorithm CNR is polynomial.

*Proof:* The complexity of Lines 2-4 is  $O(n)$ , that of Lines 5-7 is  $O(m)$ , that of Lines 8-18 is  $O(L \times |D_1|)$ , where  $L = l_1 + l_2 + \dots + l_n$ . The for-loop from Line 19 to Line 50 can execute no more than  $|D_1|$ . Since a part occupies a buffer slot and all part at the same stage occupy the buffer slots of the same resource,  $\forall i \in \{1, 2, \dots, |D_1|\}$ ,  $D_2[i] \leq \psi_{\max} = \max\{\psi_j \mid j \in \{1, 2, \dots, m\}\}$ . The while-loop from Line 21 to Line 49 can execute no more than  $\psi_{\max}$  times. Since the number of stages between stage  $D_1[i]$  to  $D_3[i]$  is no more than  $l_{\max} = \max\{l_j \mid j \in \{1, 2, \dots, n\}\}$ , the complexity of advancing a part at stage  $D_1[i]$  to  $D_3[i]$  is bound by  $l_{\max}$ . Since there are at most  $L$  stages needing to be updated, the complexity of the for-loop from Line 25 to Line 42 is bounded by  $L$ . The complexity of Lines 44 to Line 47 is dominated by that of Line 44, which is bound by  $O(l_{\max})$ . Thus, the complexity of Lines 19-50 is  $O(|D_1| \times \psi_{\max} \times (l_{\max} + L + l_{\max})) = O(|D_1| \times \psi_{\max} \times L)$ .

The for-loop from Line 53 to Line 104 the can execute  $|P_{a1}| \times |D_1| + |P_{a2}| \times |D_1| + \dots + |P_{ac}| \times |D_1| = |P_a| \times |D_1|$  times. Since  $|\Pi(P_{jk})| \leq L$ , the complexity of Line 55 is  $O(L)$  and the for-loop from 56-68 can execute no more than  $L$  times. Since  $|\zeta(P_{vw})| \leq n$ , The complexity of Lines 58-60 is  $O(n)$ . The complexity of Lines 62-67 is  $O(|D_1|)$ . Thus, the complexity of the for-loop from 56-68 is  $O(L \times (n + |D_1|))$ . Since a part occupies a buffer slot and all part at the same stage occupy the buffer of the same resource,  $\forall i \in \{1, 2, \dots, |D_1|\}$ ,  $D_2[i] \leq \psi_{\max}$ . The while-loop from Line 72 to Line 102 can execute no more than  $\psi_{\max}$  times. Since the number of stages between stage  $D_1[t]$  to  $D_3[t]$  is no more than  $l_{\max}$ , the complexity of advancing a part at stage  $D_1[t]$  to  $D_3[t]$  is bound by  $l_{\max}$ . The for-loop from Lines 77 to Line 89 updates  $q$ ,  $D_1$ , and  $D_2$ . As stated before, its complexity is bounded by  $L$ . The complexity of Lines 96 to Line 99 is dominated by that of Line 96, which is bound by  $O(l_{\max})$ . Thus, the complexity of the for-loop from Line 51 to Line 106 is  $O(|P_a| \times |D_1| \times (L \times (n + |D_1|) + \psi_{\max} \times (l_{\max} + L))) = O(|P_a| \times |D_1| \times L \times (n + |D_1| + \psi_{\max}))$ .

Thus,  $O(\text{CNR}) = O(n) + O(m) + O(L \times |D_1|) + O(|D_1| \times \psi_{\max} \times L) + O(|P_a| \times |D_1| \times L \times (n + |D_1| + \psi_{\max}))$ . Since  $L \geq n$ ,  $L \geq |D_1|$ ,  $L \geq |P_a|$ ,  $O(\text{CNR}) = O(m) + O(L^2 \times \psi_{\max}) + O(L^3 \times (L + \psi_{\max})) = O(m + L^3 \times (L + \psi_{\max}))$ . ■

**Theorem 2:** Given a state  $q \in \mathcal{Q}_R(q_0)$ ,  $D_1$ ,  $D_2$ ,  $D_3$  and  $S$ , let  $\Phi_{\min}$  be the minimum number of raw parts required to advance all parts at stages in  $D_1$  to their proper stages in  $D_3$ . Then,  $\Phi = \text{CNR}(q, D_1, D_2, D_3, S) = \Phi_{\min}$ .

*Proof:* We prove that  $\forall c \in \mathbb{Z}$ ,  $\Phi = \Phi_{\min}$  by mathematical induction on the value of  $c$  as follows.

If  $c = 0$ , then no assemble operation exist in  $S$ . All parts at stages in  $D_1$  can be advanced to their proper stages in  $D_3$  without needing any raw parts. Thus,  $\forall i \in \{1, 2, \dots, n\}$ ,  $\Phi[i] = \Phi_{\min}[i] = 0$ .

If  $c = 1$ , after the execution of Lines 1-50, all part remaining in the system stay in  $\Pi(P_{jk})$ , where  $P_{jk} \in P_{a1}$ . In order to advance all parts at stages in  $P_{vw} \in \Pi(P_{jk})$  to their proper stages, stage  $P_{jk}$  must be gone through. Thus, for stage  $P_{vw} \in \Pi(P_{jk})$ , at least  $z = \max\{q(x_{vw}) + q(y_{vw}) : P_{vw} \in \Pi(P_{jk})\}$  parts are needed. Since  $\forall P_{vw} \in \Pi(P_{jk})$ , no part stays in  $\{\theta(P_{vw}) \setminus P_{vb}\}$ , the lacking parts can only be obtained from the fictitious beginning places. Thus,  $\forall i \in \{1, 2, \dots, n\}$ ,  $\Phi[i] = \Phi_{\min}[i]$ .

Suppose that  $\forall i \in \{1, 2, \dots, n\}$ ,  $\Phi[i] = \Phi_{\min}[i]$  when  $c = h \in \mathbb{Z}^+$ . Now, we prove  $\forall i \in \{1, 2, \dots, n\}$ ,  $\Phi[i] = \Phi_{\min}[i]$  when  $c = h + 1$ . Note that CNR computes the number of raw parts from smaller assembly class to larger assembly class and the part obtained after the assembly operation is advanced either to their proper stages (Lines 74-94) or to the next stages in  $\Pi_a$  (Lines 95-101). In other words, before CNR computing the number of raw parts for  $P_{jk} \in P_{a(h+1)}$ , all parts remaining in the system stay in  $\Pi(P_{jk})$ . In order to advance all parts at stages in  $P_{vw} \in \Pi(P_{jk})$  to their proper stages, stage  $P_{jk}$  must be gone through. Thus, for stage  $P_{vw} \in \Pi(P_{jk})$ , at least  $z = \max\{q(x_{vw}) + q(y_{vw}) : P_{vw} \in \Pi(P_{jk})\}$  parts are needed. Since  $\forall P_{vw} \in \Pi(P_{jk})$ , no part stays in  $\{\theta(P_{vw}) \setminus P_{vb}\}$ , the

lacking parts can only be obtained from the fictitious beginning places. Thus, the number of raw parts which is worked out by CNR for  $P_{a(h+1)}$  is minimal. By the assumption, the number of raw parts which is worked out by CNR for  $P_{ai} \forall i \in \{1, 2, \dots, h\}$  is minimal. Thus,  $\forall i \in \{1, 2, \dots, n\}$ ,  $\Phi[i] = \Phi_{\min}[i]$  holds for  $c = h + 1$ . ■