CPN = [P, T, F, C, V, m0]

P - непустое конечное множество позиций

T - непустое конечное множество переходов, непересекающееся с P

F - подмножество (P x T) v (T x P), множество дуг

C - это отображение, связывающее непустое конечное множество цветов C(x) с каждым узлом x ∈ P v T

V - это отображение, определенное для множества всех дуг F так, что f = [p, t] ∈ F (соотв. f = [t, p] ∈ F), а значение V(f) это отображение C(t) в BAG(C(p))

m0 - это начальная маркировка CPN, соотносящая каждое мультимножество m0(p) из BAG(C(p)) с каждым p ∈ P

Обозначения:

BAG(P) - это множество всех мультимножеств для P

Пусть p ∈ P, c ∈ C(p), t ∈ T, d ∈ C(t). Тогда m0(p)[c] это количество токенов цвета c, которые изначально содержит позиция p, и V(p,t)[d][c] это количество токенов цвета c, которые будут захвачены firing’ом перехода t с цветом d из позиции p

Введем множества:

Pf = {[p,c] | p ∈ P ∩ c ∈ C(p)} - множество определяющее возможные сочетания позиций с их цветами;

Tf = {[t,d] | t ∈ T ∩ d ∈ C(t)} - множество, определяющее возможные сочетания переходов с их цветами;

CPN = [P, T, X, Y, Z, Wx, Wy, Wz, Q, A, G, m0]

P - конечное множество позиций, T - конечное множество переходов, *X*⊆*P*x*T* – множество входных дуг переходов с порогом-минимумом (дуги с проверкой на “больше”); *Y*⊆*T*x*P* – множество выходных дуг переходов; *Z*⊆*P*x*T* – множество входных дуг переходов с порогом-максимумом (дуги с проверкой на “меньше”); *WX*: *X→N*+x*N*0 – функция весов дуг типа *X*; *WY*:*Y*→*N+* – функция весов дуг типа *Y*; *WZ*: *Z*→*N+* – функция весов дуг типа *Z*; *Q*: *T*→*N*0 – функция приоритетов переходов (чем меньше значение, тем выше приоритет); *m*0 – начальная маркировка. Первый компонент веса дуги типа *X* указывает минимальный порог для срабатывания перехода, а второй – число удаляемых меток из инцидентной дуге позиции.

============================

CPN = [P, T, ]

P - конечное множество позиций, T - конечное множество переходов,

*CPN* = (*P,T,A,**,V,C,G,E, I*), where:

1. *P* is a finite set of **places**.

2. *T* is a finite set of **transitions** *T* such that *P∩T* = / 0.

3. *A ⊆ P×T ∪T ×P* is a set of directed **arcs**.

4. is a finite set of non-empty **colour sets**.

5. *V* is a finite set of **typed variables** such that *Type*[*v*] *∈* for all variables *v ∈ V*.

6. *C* : *P→*is a **colour set function** that assigns a colour set to each place.

7. *G* : *T →EXPRV* is a **guard function** that assigns a guard to each transition *t* such

that *Type*[*G*(*t*)] = *Bool*.

8. *E* : *A→EXPRV* is an **arc expression function** that assigns an arc expression to

each arc *a* such that *Type*[*E*(*a*)] =*C*(*p*)*MS*, where *p* is the place connected to the

arc *a*.

9. *I* : *P→EXPR*/0 is an **initialisation function** that assign