Syntax and semantics of SN

You now:

- have an idea on Coloured Nets in general . . .
- . . . and Symmetric Nets in particular

You now:

- have an idea on Coloured Nets in general . . .
- . . . and Symmetric Nets in particular

Let's go for a more precise semantics

Recall: multisets (Bags)

- Let A be a non empty finite set.
- A bag a on A is a function:

$$\begin{aligned} \mathtt{a} : \mathtt{A} &\rightarrow \mathbb{N} \\ \mathtt{x} &\rightarrow \mathtt{a(x)} \end{aligned}$$

  a(x) denotes the number of occurrences of x in a.
- We note: $a = \sum_{x \in A} a(x).x$
- Bag(A) denotes the set of multisets on A.
- Consider the following functions:

$$\begin{aligned} f &: Bag(C_1) \rightarrow Bag(C_2) \\ g &: Bag(C'_1) \rightarrow Bag(C'_2) \\ h &: Bag(C) \rightarrow Bag(C_1) \end{aligned}$$

  then

$$\begin{aligned} \langle f, g \rangle : Bag(C_1) \times Bag(C'_1) &\rightarrow Bag(C_2) \times Bag(C'_2) \\ (x, y) &\rightarrow \langle f(x), g(y) \rangle \end{aligned}$$

  and

$$\begin{aligned} f \circ h : Bag(C) &\rightarrow Bag(C_2) \\ x &\rightarrow f(h(x)) \end{aligned}$$

SN definition: syntax

- **A Symmetric Net** is a tuple $\langle P, T, C, W^-, W^+, M_0 \rangle$ where:
  - P is the set of places, T is the set transitions ($P \cap T = \emptyset$, $P \cup T \neq \emptyset$).
  - C defines for each place and transition a colour domain.
  - W^- (= Pre) (resp. W^+ = Post), indexed on $P \times T$, is backward (resp. forward) incidence matrix of the net.
  - $W^-(p, t)$ and $W^+(p, t)$ are linear colour functions defined from $Bag(C(t))$ to $Bag(C(p))$.
  - M_0 is the initial marking of the net ($M_0(p) \in Bag(C(p))$).
  - Transitions may be guarded by functions: $Bag(C(t)) \rightarrow \{0, 1\}$.
  - Colour domains are generally cartesian products.

SN definition: semantics

- Let $N = \langle P, T, C, W^-, W^+, M_0 \rangle$ be an SN:
  - A **marking** M of N is a vector: $M(p) \in Bag(C(p))$.
  - A transition t is **enabled** for an instance $c_t \in C(t)$ and a marking M *iff*:
    - either t is not guarded, or the guard evaluates to true (for c_t), and
    - $\forall p \in P, M(p) \geq W^-(p,t)(c_t)$
  - M', the marking reached after the firing of t for an instance c_t, from the marking M is defined by:

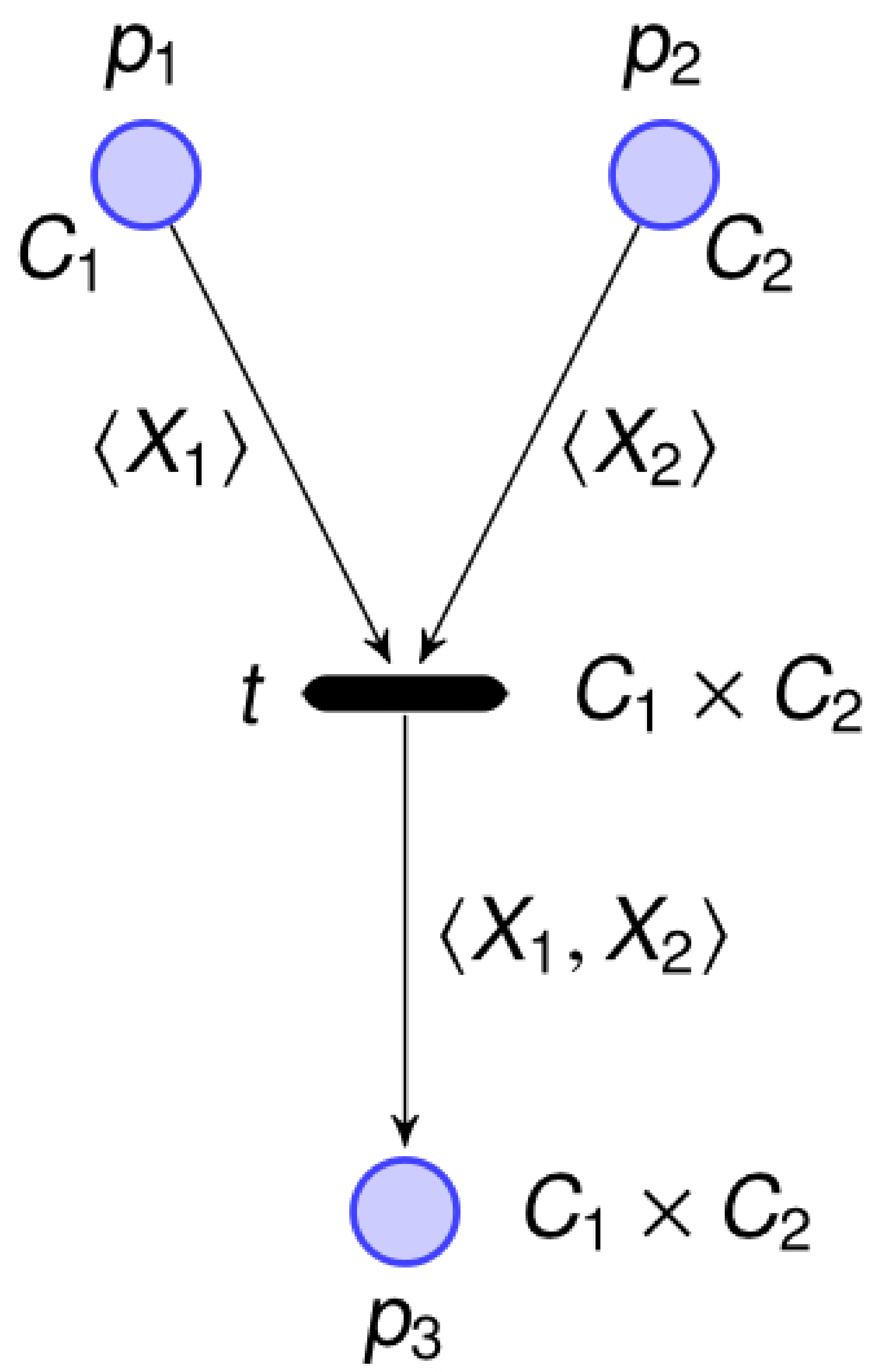
$$\forall p \in P, M'(p) = M(p) - W^-(p,t)(c_t) + W^+(p,t)(c_t)$$

We note:

$$M[(t,c_t)\rangle M' \text{ or } M \xrightarrow{(t,c_t)} M'$$

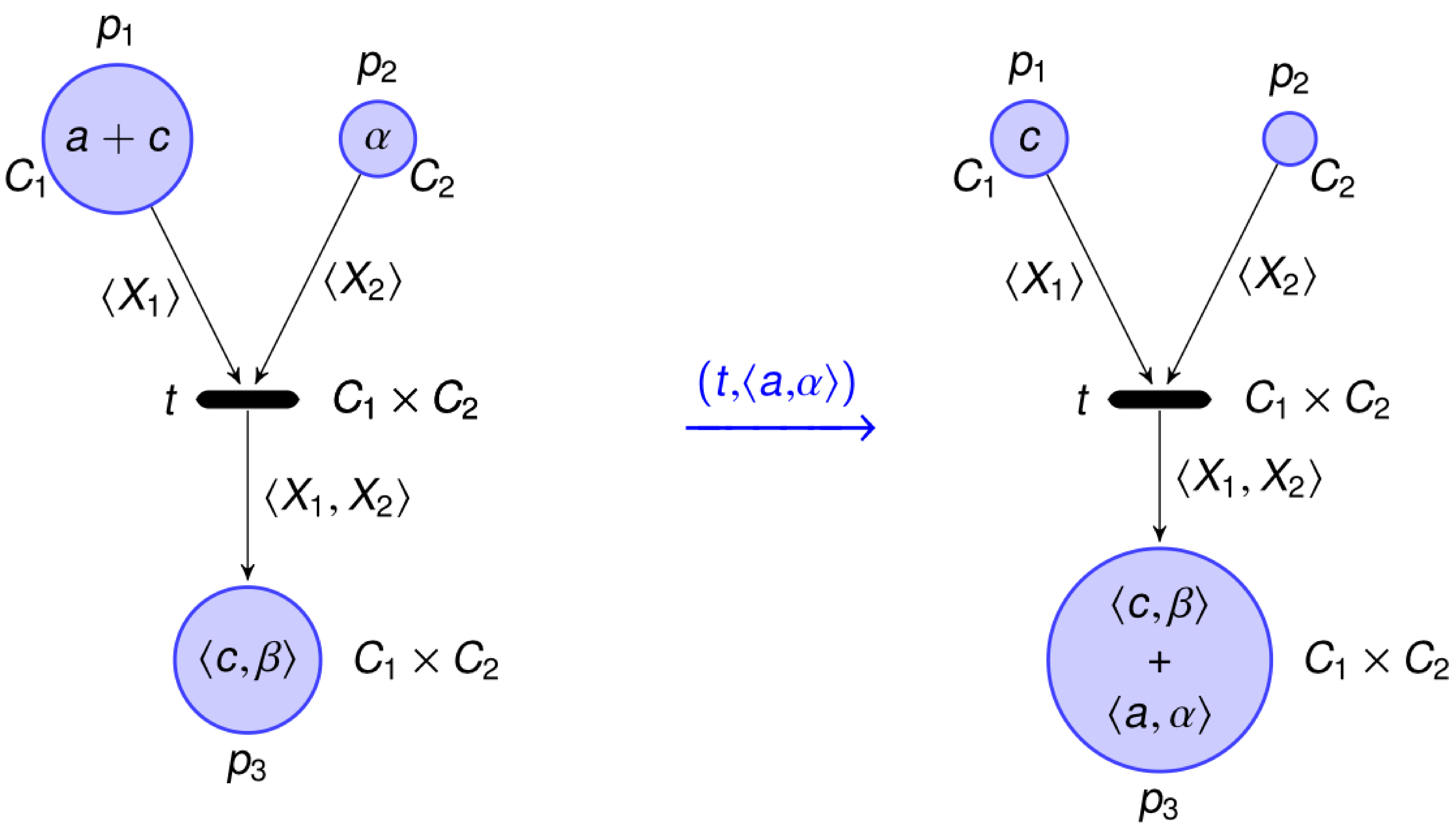
Example of firing in SN (1/2)

- Let $c_1 \in C1, c_2 \in C_2$.
- t is **enabled** for (c_1, c_2) *iff*:
  1. p_1 is marked by a token of colour $\langle X_1 \rangle$
  2. p_2 is marked by a token of colour $\langle X_2 \rangle$
- If t is **fired** for (c_1, c_2) then:
  1. A token of colour $\langle X_1 \rangle$ is removed from p_1
  2. A token of colour $\langle X_2 \rangle$ is removed from p_2
  3. A token of colour $\langle c_1, c_2 \rangle$ is produced in $p_3 : \langle X_1, X_2 \rangle (c_1, c_2) = \langle c_1, c_2 \rangle$



$$X_i(c_1, c_2) = c_i$$

$C_1 = \{a, b, c\} \quad C_2 = \{\alpha, \beta\}$



Basic colour functions

- Let:
  - $C = \prod_{i=1}^{n} \prod_{j=1}^{e_i} C_i$, and
  - $c = \langle c_1^1, c_1^2, ..., c_1^{e_1}, ..., c_n^1, c_n^2, ..., c_n^{e_n} \rangle \in C$

Basic colour functions

A colour domain constructed on top of a Cartesian product of colour classes, in which C_i appears e_i times.

- Let:
  - $C = \prod_{i=1}^{n} \prod_{j=1}^{e_i} C_i$, and
  - $c = \langle c_1^1, c_1^2, ..., c_1^{e_1}, ..., c_n^1, c_n^2, ..., c_n^{e_n} \rangle \in C$

Basic colour functions

- Let:
  - $C = \prod_{i=1}^{n} \prod_{j=1}^{e_i} C_i$, and
  - $c = \langle c_1^1, c_1^2, ..., c_1^{e_1}, ..., c_n^1, c_n^2, ..., c_n^{e_n} \rangle \in C$
- **Identity/Projection**:
  - noted by a variable $\langle X \rangle$, Y, or $\langle X_1 \rangle$, or X_1^1, or p, q,...
  - $X_i^j(c) = c_i^j$ (e.g. $q(\langle p, q, r \rangle) = q$).

Basic colour functions

- Let:
  - $C = \prod_{i=1}^{n} \prod_{j=1}^{e_i} C_i$, and
  - $c = \langle c_1^1, c_1^2, ..., c_1^{e_1}, ..., c_n^1, c_n^2, ..., c_n^{e_n} \rangle \in C$
- **Identity/Projection**:
  - noted by a variable $\langle X \rangle$, Y, or $\langle X_1 \rangle$, or X_1^1, or p, q,...
  - $X_i^j(c) = c_i^j$ (e.g. $q(\langle p, q, r \rangle) = q$).
- **Successor (on a circularly ordered C_i)**:
  - noted X_i++ or ($X_i \oplus 1$) or X_i!
  - $X_i^j{++}(c) = successor(c_i^j)$.

Basic colour functions

- Let:
  - $C = \prod_{i=1}^{n} \prod_{j=1}^{e_i} C_i$, and
  - $c = \langle c_1^1, c_1^2, ..., c_1^{e_1}, ..., c_n^1, c_n^2, ..., c_n^{e_n} \rangle \in C$

- **Identity/Projection**:
  - noted by a variable $\langle X \rangle$, Y or $\langle X_i \rangle$ or X_i^1 or p, q
  - $X_i^j(c) = c_i^j$ (e.g. $q(\langle p, q, r \rangle) = q$).

The successor relation is defined by the enumeration order of elements in class C_i.

- **Successor (on a circularly ordered C_i)**:
  - noted X_i++ or $(X_i \oplus 1)$ or X_i!
  - $X_i^j{+}{+}(c) = successor(c_i^j)$.

Basic colour functions

- Let:
  - $C = \prod_{i=1}^{n} \prod_{j=1}^{e_i} C_i$, and
  - $c = \langle c_1^1, c_1^2, ..., c_1^{e_1}, ..., c_n^1, c_n^2, ..., c_n^{e_n} \rangle \in C$

- **Identity/Projection**:
  - noted by a variable $\langle X \rangle$, Y, or $\langle X_1 \rangle$, or X_1^1, or p, q,...
  - $X_i^j(c) = c_i^j$ (e.g. $q(\langle p, q, r \rangle) = q$).

- **Successor (on a circularly ordered C_i)**:
  - noted X_i++ or $(X_i \oplus 1)$ or X_i!
  - $X_i^j{+}{+}(c) = successor(c_i^j)$.

- **Diffusion / Synchronisation (on C_i)**
  - noted $C_i.All$ or S_{C_i}
  - $C_i.All(c) = \sum_{x \in C_i} x$

Conclusion

At this stage:

- you know the formal syntax of Symmetric Nets
- the enabling conditions
- their firing rule

Conclusion

At this stage:

- you know the formal syntax of Symmetric Nets
- the enabling conditions
- their firing rule

Let's go for a detailed example (next sequence)