"Semi-controllability"

Sometimes you have events that are not totally controllable but not totally uncontrollable either.

1. You can prevent the event, but you cannot start it by will.

2. You can start it, but it can also start spontaneously.

This can be modeled by using two different events, one controllable and one uncontrollable.
Consider the uncontrollable event $uc$, which can neither be started nor prevented.

If we can also start it at will we simply add an controllable event $c$ with the same effect:
If we can prevent the starting up to some point we can use the following construction:

One could also add a timeout or similar event which takes us from q1 back to initial state, and thus disables the uncontrollable event.

Timeouts are also a bit semicontrollable, they cannot be executed at will, so they have to be uncontrollable, but they are typically not that critical.

Timeouts are also internal, and not external as non-controllable events usually are.

This means that the timeout is only present in the specifications, and not in the plants, so the event cannot cause a forbidding of states in the compare-operation.

Such uncontrollable events are thus not as restrictive.
If we want at each time decide if we can actually allow the uncontrollable event we can use the following construction:

Here the desired action takes places in the event ok. The main point with this construction is that we can postpone the treatment of the event until later.