Abstract— Petri nets provide a compact and graphical way to model large and complex discrete event systems (DES). For such systems, the state-space explosion is problematic. Fluid stochastic event graphs are decision free Petri nets, which can represent systems with failures. This paper presents an estimation algorithm for state space estimation and optimization of failure-prone DES.

Keywords— Discrete event systems, fluid Petri nets, space estimation.

I. INTRODUCTION

According to the complexity of modern technological processes, many model analysis and control algorithms are based on the model state-space sizes. In this paper we propose, based on the works in [1] and [2], an algorithm for analyzing and optimizing manufacturing systems subject to failures. It is based on the fluid approximation of a class of Petri nets and called fluid-event graphs. In a fluid/event graph, places hold fluids instead of discrete tokens. Transitions fire continuously, drawing fluids out of its input places and injecting fluids into its output places. Firing speed of transition is limited by a maximal speed. For failures systems, transitions can be either in operating state or in failure-state. The discussed systems are hybrid, that means they have discrete-event components characterized by failures and repair of transitions, and continuous components characterized by markings and transition firing speeds. We assume that the discrete-event component does not depend on the continuous component [2]. This assumption is related to the time dependent failures of the failure – Prone manufacturing systems. The following users for the state–space size estimation can be mentioned:

- Evaluating the trade-off between model detail and solution complexity. This is necessary because, at some point, dependent on the specific model, additional computation time is not worth the improvement in accuracy.
- Determining the appropriateness of a particular analysis technique. Algorithms optimized for “small” or “large” problems can be applied appropriately.

The problem of state–space size estimation of Petri nets (PN) is being pursued in two manners: top-down and bottom–up [3], [4]. At the expense of complete generality, the bottom–up approach offers better accuracy.

II. FLUID PETRI NETS

Fluid Petri nets are an extension of classical Petri nets [5]. In fluid PN (FIPN), marking a place is a real number called the token content. Also, because transitions fire continuously according to some firing speed, we shall say firing speed instead of firing sequence. In fluid Petri nets, we associate to each transition a maximal firing speed, and flowing notations are used:

\[ A_i = \text{maximal firing speed of transition } i; \]
\[ a_{it} = \text{firing speed of transition } i \text{ at time } t; \]
\[ m_{it} = \text{marking of place } p_i \text{ at time } t; \]
\[ m_{oi} = \text{initial marking of place } p_i (m_{oi} = m_{oi}); \]
\[ q_{it} = \text{cumulative firing quantity of transition } i \text{ up to time } t. \]

A control policy \( a_{it} \) is feasible if \( m_{it} \geq 0 \)
A transition can fire at its maximal firing speed if each of its input places has positive markings. A generally firing policy for fluid Petri nets it is [2]:

\[ a_{it} = \min A_j , \forall i \in \{T\} \]
\[ T_j/m_{it} = 0 \quad (1) \]

Where \( T_j \) is the transition which follow the place \( p_i \) with marking \( m_{it} \). \( \{T\} \) is the set of transitions in the considered fluid Petri net.

For example let us consider the fluid Petri net in Fig.1.
In Fig.1 consider the place \( p_1 \) and assume that \( A_1 \geq A_2 \). Then transition \( T_2 \) can fire at its maximal speed and the fluid level of place \( p_1 \) evolves as shown in Fig.2.a.
If $A_1 \geq A_2$, then $T_3$ fires at its maximal speed until place $p_1$ becomes empty, and then it fires at reduced speed $A_1$.

Fig.2.b illustrates the evolution of the fluid level. In general, the fluid level of each place evolves piecewise linearly according to the firing speed of its input/output transitions.

The $r$ tokens possible distribution among the places corresponding to the subnets $SN_1$, and $SN_2$ is due to the fluid repartition among the respective places, separated by transitions $T_i$, $i=1,2,3$, with $a_i$ firing speed.

The interconnection $Se$–function is determined from the fact that tokens can be distributed between the two subnets according to the following relation [2], [6]:

$$Se_{series} = \sum_{i=0}^{r} Se_{a(j)} (i) Se_{a(2j)} (r-i)$$  
(2)

Where $Se_{a(i)} (j)$, the $k^{th}$ partition of $r$, is an $n$-vector of non-negative integers that sum to $r$.

$A_i$ is the firing speed of transition $i$, $i=1,2$, at time $t$.

For $n$ subnets, the $Se$-function is:

$$Se_{series} = \sum_{i=0}^{r} Se_{a(j)} (i) Se_{a(2j)} (i-1) Se_{a(j)} (r-i) + Se_{a(j)} (r-i)$$  
(3)

Parallel execution of operation is depicted in Fig.4. For every token entering through $T_1$, there is one in each subnet. The $Se$-function is given by relation (4):

$$Se_{parallel} = \prod_{i=1}^{n} Se_{a(i)} (r)$$  
(4)

The choice among sub-nets is depicted in Fig.5. There are three constructs to consider: $SN_1$, $SN_2$, and, as a group the places $P_1$ and $P_2$. Having $r$ tokens among $P_1$ and $P_2$ generates $C_r^2 = r + 1$ states.

For $n$ sub-nets, we have:

$$Se_{choice} = \sum_{i=0}^{r} Se_{a(j)} (i) Se_{a(2j)} (i-1) Se_{a(j)} (r-i) + Se_{a(j)} (r-i)$$  
(5)

FIPN of systems containing unreliable components, may include models for the failure and repair of these components. Fig.6 depicts a common model for these operations. Under normal functioning, $T_2$ fires instead of $T_{fail}$, leaving $SN_2$ out of consideration.
The choice between \( T_2 \) and \( T_{\text{fail}} \) allows \( SN_2 \) to be used by all \( r \) tokens. Thus, \( SN_1 \) and \( SN_2 \) are, in series, producing:

\[
(Se)_i (Se) = \sum_{i=0}^{r} (Se)_{i} \cdot (Se)_{i-1} (r-i) \quad (7)
\]

Where, \( r \) is the content of place \( P_6 \).

Using (7) and the \( Se \)-functions given in (6) for \( SN_i \), respectively in (2) for \( SN_1 \) and \( SN_2 \), we obtain, as a function of \( r \), the state estimating function for the net given in Fig.7. For example, we suppose (in order to simplify the calculus) that \( A_j = \text{const} = 1 \), where \( j=1,\ldots,5 \) is the number of transitions in Fig.7 (see relation (1)). Thus, we obtain the following \( Se_{\text{global}} \) for the net given in Fig.7:

**IV. AN EXAMPLE OF FLPN WITH FAILURE AND REPAIR**

In Fig.7 a number of modeling assumptions were made for the convenience of the presentation:

a) The tasks performed by \( M_2 \) and \( R_2 \) were aggregated into a single transition representing loading and unloading at \( M_2 \).

b) The time associated with the unloading tasks performed by \( R_1 \) is incorporated into transition \( T_2 \).

c) The failure-repair loop is added to \( M_1 \).

d) The time delays are associated with transitions only.

e) A Firing speed of a transition equals to the reciprocal of the average firing delay time of the corresponding event or operation.
CONCLUSION
A bottom-up state-space size estimation technique for fluid Petri nets (FIPN) has been described. The estimation relies on the computation of state estimating functions (Se functions). Several researchers have documented the state-space explosion behavior but there is no direct correlation between the specific behavior of FPN and the used mechanisms such as top-down and bottom-up. A lot of model analysis and control algorithms are based on the model state space, and there it is why they are affected by large state space sizes. Benefits of this approach include simple representation of Se-functions that facilitate automation, and the possibility to interject hand-computed results into the estimation. Errors in the estimation may result from changes in the FIPN model to permit analysis in the Se functions. Further research will include the colored Petri nets into the FIPN, and the calculus of theirs Se-functions.

REFERENCES