

ASYNCHRONOUS NETWORKS AND EVENT DRIVEN DYNAMICS

CHRISTIAN BICK AND MICHAEL FIELD

ABSTRACT. Real-world networks in physics, biology and technology often exhibit dynamics that cannot be adequately reproduced using network models given by smooth dynamical systems and a fixed network topology. Asynchronous networks give a theoretical and conceptual framework for the study of network dynamics where nodes can evolve independently of one another, be constrained, stop, and later restart, and where the interaction between different components of the network may depend on time, state, and stochastic effects. This framework is sufficiently general to encompass a wide range of applications ranging from engineering to neuroscience. Typically, dynamics is piecewise smooth and there are relationships with Filippov systems. We make the notion of a functional asynchronous network rigorous, discuss the phenomenon of dynamical locks, and present a theorem about the spatiotemporal factorization of the dynamics for a class of deadlock free functional asynchronous networks of feedforward type. We conclude with some examples and applications related to asynchronous networks with a stochastic connection structure.

CONTENTS

1. Introduction	3
2. Synchronous networks	7
2.1. Generalities on networks	7
2.2. Example: N -body problem of classical mechanics	10
2.3. Kuramoto's model: phase oscillator dynamics	10
2.4. Synchronous networks	11
3. Asynchronous networks: examples	13
4. A Mathematical model for deterministic asynchronous networks	19
4.1. Preliminaries and an overview	19

Date: December 12, 2015.

Research supported in part by NSF Grant DMS-1265253 & Marie Curie IEF Fellowship (Project 626111).

Research supported in part by NSF Grant DMS-1265253 & Marie Curie IIF Fellowship (Project 627590).

4.2.	Constrained nodes and admissible vector fields	21
4.3.	Asynchronous networks	26
4.4.	Dynamics on an asynchronous network	27
4.5.	Asynchronous networks with additive input structure and constraints	34
5.	Products and amalgamations of asynchronous networks	36
5.1.	Products	36
5.2.	Indecomposability	38
5.3.	Sufficient conditions for decomposability	40
5.4.	Amalgamating asynchronous networks	42
6.	A simple example: transport with a passing loop	46
6.1.	The basic example	46
6.2.	Local times	50
6.3.	Adding some dynamics	51
6.4.	Initialization, termination and function	52
7.	Functional asynchronous networks and deadlocks	53
7.1.	Initialization, termination and network function	53
7.2.	Deadlocks	55
7.3.	To stop or not to stop? Consequences for network function	58
8.	Weakly regular and regular FANs	59
8.1.	Geometric FANs	59
8.2.	Weakly regular FANs	61
8.3.	Generalized initialization of FANs.	63
8.4.	Hidden deadlocks	65
8.5.	Weakly regular FANs of simple type	66
8.6.	Concatenating regular networks of simple type	68
8.7.	Forced stopping	70
9.	A factorization theorem for a class of FANs	71
9.1.	Primitive, stably primitive, trivial and basic FANs	72
9.2.	FANs of feedforward type and factorization	76
10.	Stochastic asynchronous networks	80
10.1.	Passing loop revisited: stochastic events and network stalls	81
10.2.	Numerical solution of differential equations	82
10.3.	Dynamics on a stochastic discrete asynchronous network & commingled basins of attraction	83
11.	Concluding comments & outstanding problems	87
	References	89

1. INTRODUCTION

Recent decades have seen an ever increasing interest in *network dynamics* with applications ranging from biology and neuroscience to engineering ([98, 2, 34, 91, 101] are some characteristic works from a vast literature on network dynamics and applications). Since Kuramoto's celebrated work on oscillator synchronization [72, 73], much of the mathematical effort has been directed towards questions related to *synchronization* in networks, especially in coupled phase oscillator systems of the type originally investigated by Kuramoto. More recently, Golubitsky, Stewart *et al.* have developed a formal framework for the study of (exact) synchronization and bifurcation in *coupled cell networks*: typically networks of identical cells that do not have all-to-all coupling [51, 52, 53] and which are modelled by a single system of analytic (or smooth) differential equations. Although it is not yet clear that this approach is useful in the study of large (heterogeneous) and, in particular, distributed networks, there have been significant applications to small networks such as models for bipedal gaits [50, 85], where symmetry plays a major role (some form of symmetry appears to be important in the structure and bifurcation theory of general identical coupled cell networks [92, 93, 83]). In a related direction, there is the concept of a *network motif* developed by Alon and Kastan [67, 10]. Their work identifies certain small network configurations that occur frequently in (large) biological networks (for example, the feed-forward motif [10, Chapter 4, §3]).

The identification and description of relatively simple and small dynamical units, such as non-linear oscillators or motifs, suggests a modular approach to dynamics: identify building blocks, connect together to form networks and then describe dynamical properties of the resulting network in terms of the dynamics of its components:

“Ideally, we would like to understand the dynamics of the entire network based on the dynamics of the individual building blocks.” Alon [10, page 27]

Unfortunately, a reductionist approach of this type has a limited range of application to classical models of network dynamics where networks are modelled by a single system of analytic differential equations (we discuss some of the issues in more detail in the next section).

Furthermore, many networks of interacting dynamical systems that are of interest in contemporary science and engineering do not fit naturally into the classical model. For example, connectivity may vary, there may be switching between dynamical equations, thresholds, hybrid dynamics (a mix of continuous and discrete or spiking dynamics),

and individual nodes may stop and later restart (solutions are strongly non-analytic). At a deeper level, changes in network evolution may be caused by ‘events’, nodes may evolve independently of each other for periods of time (local autonomy), there may be no intrinsic (global) time and, in practice, the network may also have a function – for example, getting from a set of initial states to a set of prescribed final states in *finite* time. Successful running of the network may require adaptation, (distributed) control and the avoidance of dynamical deadlocks (situations where the dynamics is permanently stalled and the final state cannot be reached). As compensation for the loss of analyticity in the network, we believe that it is *only* by dropping the rigid structure of a fixed network of analytic equations that we can effectively employ reductionist methods to networks not consisting of identical nodes. Our main result will be to identify a large class of networks of non-classical type which are modular and to which we can apply a classical reductionist approach to the analysis of network dynamics.

We aim in this work to formalize an abstraction for networks that takes account of issues such as time and state dependent network topology, piecewise smooth dynamics (and failure of analyticity), local times (no global network clock), event driven dynamics, limits on internode communication, and statistical effects. We refer to these types of non-classical networks as *asynchronous* and, when appropriate, *event driven*, and classical networks modelled by a single system of differential equations as *synchronous*¹.

An underlying theme and guide for much of what we say is that of efficiency and cost in large distributed networks. It is worth citing the guidelines for scalability in large distributed networks given by Tannenbaum & van Steen [100, page 11]:

- No machine has complete information about the (overall) system state.
- Machines make decisions based only on local information.
- Failure of one machine does not ruin the algorithm.
- There is no implicit assumption of global time.

It is expensive and often difficult to run a large distributed network synchronously (total central control). By way of illustration, we refer

¹We caution the reader that our usage of ‘asynchronous’ and ‘synchronous’ comes from computer science rather than mathematics. Indeed, the term asynchronous is often used by mathematicians to refer to network *dynamics* that is not, in some sense, synchronized or synchronous [36]. The term asynchronous has also been widely used in the context of random Boolean networks [56, 8, 30], and artificial neural networks [16] and while there are some connections with our work, our perspective is quite different.

to [54, Chapter 1] for a discussion of the issues involved in maintaining a classical power grid network and to [79, 87] for some of the mathematical issues involved in the design of ‘smart grids’ where control is partially distributed. From a modelling perspective, it is unrealistic in very large networks (for example, neural networks) to assume network nodes continuously communicate *complete* information about their states to all other nodes to which they are connected or to a central controller. Such communication would be both inefficient and wasteful of energy. Usually communication channels, when open, will be low dimensional (but not too low), even if the nodes they connect have high dimensional state spaces and complex dynamics, and there will be a mix of local and centralized control. Issues of this type arise, for example, in the use of decentralized search algorithms in large networks [69], and the use of decentralized algorithms for collision avoidance in aircraft control [96] or in computation [68]. As the authors point out in [70, §1], ‘centralized scheduling requires accurate, centralized information about the state of the whole system’.

In general terms, networks dynamics, in either technology or biology, is likely to involve a complex mix of synchronous and asynchronous components. In particular, timing (clocks, whether local or global) may be used to trigger the onset of events or processes rather than as part of a mechanism for centralized control. Evolution is opportunistic – whatever works well will be adopted (and adapted) whether synchronous or asynchronous in character. In specific cases, especially in biology, it may be a matter of debate as to which viewpoint – synchronous or asynchronous – is the most appropriate. The models we present include synchronous modules coupled together asynchronously – a globally asynchronous, locally synchronous architecture (GALS [88]) – but our framework is sufficiently flexible to allow for a wide mix of synchronous and asynchronous structure at the global or local level.

Mathematically speaking, much of what we say has significant overlap with other areas and past work. We cite in particular, the general area of nonsmooth dynamics, Filippov systems and hybrid systems (for example, [44, 12, 81, 22]) and time dependent network structures (for example [19, 77, 55, 63]). While the theory of nonsmooth dynamics focuses on problems in control, impact, and engineering problems, rather than networks, there is significant past work studying bifurcation (for example [74, 21]) which is likely to apply to parts of the theory we describe. At the same time, much of the work on time dependent network topologies has been on discrete time systems.

Other approaches are possible and have been extensively used. In particular, the use of time or ensemble averages to reduce a non-smooth

system to a classical system of differential equations or maps (see [28] for a characteristic averaging approach to Spike-Timing Dependent Plasticity – STDP). The drawback of averaging is that it usually assumes a large complex system and potentially washes out the (often hidden) structural features that are essential for the efficient and effective operation of the system. This is a significant problem with any biological or technological system that is in a highly evolved form. For example, the mechanism of STDP likely originally involved small sets of neurons and later evolved into a much more complex system taking account of multiple factors to fine tune performance. The internal combustion engine of a car in 1950 was usually a relatively simple device which was synchronized through mechanical means. A modern internal combustion engine is structurally complex and employs a mix of synchronous and asynchronous systems controlled by multiple computer processors, sensors and complex computer code. One aspect of the methods we develop is their potential to work at a level where it is possible to detect key structural features that are often hidden by subsequent system evolution and refinement. The approach and methodology is suggestive of ways to unravel the dynamics of networks that were originally built from relatively simple subnetworks and coupled together in an asynchronous fashion before evolutionary specialization and optimization led to blurring of the underlying structures.

Our intention is to provide an *introduction* to the theory of asynchronous networks and consequently our presentation is selective. In particular, we do not give general models for non-autonomous asynchronous networks (these involve local clocks – and were the original motivation for this work) and we limit discussion of the (crucial) stochastic aspects to examples or remarks (for all these matters, and more, we refer to a more extended exposition in [42] that is in preparation). Our aim here is to illustrate some of the main ideas, many of which come from engineering and may not be familiar to all mathematicians and physicists, and make a serious attempt to develop an appropriate framework for the study and description of the dynamics on the type of networks encountered in contemporary science and technology. Finally, we hope to stimulate critical thought and discussion on these matters both within and outside the mathematics community.

In more detail, in section 2 we discuss classical (or synchronous) networks and indicate limitations of this class of networks, and of the reductionist approach, in applications to real networks. In section 3 we give several instructional examples of asynchronous networks. We conclude the section with a description of one general tractable class

of asynchronous networks of feedforward type that we return to in sections 7–9. In section 4, we develop a formal mathematical model for deterministic asynchronous networks and introduce the critical concept of an event map. We develop the theory further in section 5 where we discuss products and amalgamations of asynchronous networks. Section 6 is devoted to the development of a very simple example: two trains on a passing loop. We consider various issues that arise such as stopping, local times, coupled dynamics and initialization and termination events. We also sketch the relation of the basic model to Filippov systems [44, 22]. In section 7, we give a careful definition of a functional asynchronous network and indicate one obstruction to fulfilling network function: dynamical deadlocks. In section 8, we develop foundational mathematical theory for regular functional asynchronous networks. We apply the results of section 8 in section 9 where we prove the main result of the paper: *The Factorization of Dynamics Theorem* (theorem 9.10). This result gives conditions for the computability and analysis of network function in terms of constituent subnetworks and allows the use of reductive methods in the study of a class of functional asynchronous networks of feedforward type. In section 10, we briefly describe examples of asynchronous networks where there is a stochastic connection structure and indicate some of the rich and novel dynamics that can occur in such networks. We conclude with comments about outstanding problems and the potential for bifurcation theory on asynchronous networks.

2. SYNCHRONOUS NETWORKS

2.1. Generalities on networks. We start by establishing a few mostly standard notational conventions used throughout. Let \mathbb{N} denote the natural numbers (the strictly positive integers), \mathbb{Z}_+ denote the set of positive integers (including zero), $\mathbb{R}_+ = \{x \in \mathbb{R} \mid x \geq 0\}$, and $\mathbb{R}(> 0) = \{x \in \mathbb{R}_+ \mid x \neq 0\}$. Given $n \in \mathbb{N}$, define $\mathbf{n} = \{1, \dots, n\}$. Let $\mathbf{n}^\bullet = \{0, 1, \dots, n\}$ and, more generally, for $A \subset \mathbb{N}$ define $A^\bullet = A \cup \{0\}$.

Many dynamical systems from engineering, physics, biology, computer science, etc, can be regarded as *networks* of interacting dynamical systems. In a (large) system, variables can often be naturally grouped together to define *nodes* (or subsystems) of the main system².

We need to establish our conventions on network notation; these we follow throughout this work.

We associate a graph structure with the network that encodes the interactions between nodes: a directed connection from node V to node

²Possibly in more than one way – see remark 3.5.

W , denoted $V \rightarrow W$, signifies that the dynamical evolution of W depends *nontrivially* on the state of V (we explain what we mean by nontrivial dependence below).

Suppose that the network \mathcal{N} has k nodes, N_1, \dots, N_k , and that dynamics on \mathcal{N} is given by a system of ordinary differential equations. We denote the state or phase space for N_i by M_i ³ and set $\mathbf{M} = \prod_{i \in \mathbf{k}} M_i$ – the network phase space. We denote the state of node N_i by $\mathbf{x}_i \in M_i$ and the network state by $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_k) \in \mathbf{M}$.

Dynamics on \mathcal{N} will be given by a system of ordinary differential equations of the form

$$(1) \quad \mathbf{x}'_i = f_i(\mathbf{x}_i; \mathbf{x}_{j_1}, \dots, \mathbf{x}_{j_{e_i}}), \quad i \in \mathbf{k},$$

such that the following conditions are satisfied.

(N1) For all $i \in \mathbf{k}$, $j_1 < \dots < j_{e_i}$ are distinct elements of $\mathbf{k} \setminus \{i\}$ (and so $e_i < k$).

Set $J(i) = \{j_1, \dots, j_{e_i}\} \subset \mathbf{k}$, $i \in \mathbf{k}$ ($J(i)$ may be empty).

(N2) For each $i \in \mathbf{k}$, the evolution of N_i depends nontrivially on the state of N_j , $j \in J(i)$, in the sense that there exists a choice of $\mathbf{x}_i \in M_i$ and $\mathbf{x}_{j_s} \in M_{j_s}$, $j_s \in J(i) \setminus \{j\}$, such that $f_i(\mathbf{x}_i; \mathbf{x}_{j_1}, \dots, \mathbf{x}_{j_{e_i}})$ is not constant as a function of \mathbf{x}_j .

(N3) The notation implies that the evolution of N_i depends nontrivially on the state of N_i (unless f_i does not depend on \mathbf{x}_i when we write $f_i(\mathbf{x}_{j_1}, \dots, \mathbf{x}_{j_{e_i}})$). If $J(i) \neq \emptyset$, then our convention on nontrivial dependence is that of (N2). Else, if $J(i) = \emptyset$, then the dependence of f_i on \mathbf{x}_i is nontrivial if and only if f_i is not identically zero. If the dependence is nontrivial we write $f_i(\mathbf{x}_i)$, else $f_i \equiv 0$.

Remarks 2.1. (1) Our conventions are quite different from general formalisms involving multiple connection (edge) types (for example, see [53, 7] for continuous dynamics and [4] for discrete dynamics). We emphasize that we allow at most one connection between distinct nodes of the network graph: connections encode dependence. We do not use self-loops – there are no connections $N_i \rightarrow N_i$. Later, in section 4, we give a convention to distinguish nodes of the graph whose evolution does not depend on their internal state.

³We assume the phase space for each node is a connected differential manifold – usually a domain in \mathbb{R}^n or the n -torus, \mathbb{T}^n .

(2) Given network equations (1) which do not necessarily satisfy (N1–3), we can first redefine the f_i so as to satisfy (N1). Next we remove trivial dependencies so as to satisfy (N2). Finally, we check for the dependence on internal state and modify the f_i as necessary to achieve (N3). If $f_i \equiv 0$, we can remove the node from the network. Consequently, it is no loss of generality to always assume that (N1–3) are satisfied, with $f_i \not\equiv 0$. A consequence is that any *network vector field* $\mathbf{f} = (f_1, \dots, f_k) : \mathbf{M} \rightarrow T\mathbf{M}$ can be uniquely written in the form (1) so as to satisfy (N1–3).

We always assume that network vector fields are at least C^1 (usually C^∞ or real analytic) guaranteeing existence and uniqueness of solutions to (1).

Systems of differential equations like (1) can be used to model general networks of interacting systems modelled by differential equations.

In many cases of interest, we may assume an *additive input structure* [40] and decompose the components f_i of \mathbf{f} as

$$(2) \quad f_i(\mathbf{x}_i; \mathbf{x}_{j_1}, \dots, \mathbf{x}_{j_{e_i}}) = F_i(\mathbf{x}_i) + \sum_{s=1}^{e_i} F_{ij_s}(\mathbf{x}_{j_s}, \mathbf{x}_i), \quad i \in \mathbf{k}.$$

Additive input structure implies that there are no interactions between inputs $N_j, N_k \rightarrow N_i$, as long as $j, k \neq i$, $j \neq k$, and allows us to add and subtract inputs and nodes in a consistent way. We may think of $\mathbf{x}'_i = F_i(\mathbf{x}_i)$ as defining the *intrinsic dynamics* of the node (but note that coupling terms may depend on \mathbf{x}_i).

Remarks 2.2. (1) If we identify a null or zero state \mathbf{z}_j^* for each node N_j , then the decomposition (2) will be unique if we require $F_{ij}(\mathbf{z}_j^*, \mathbf{x}_i) \equiv 0^4$. If a node is in the null state then we assume it has no output to other nodes – effectively, it is invisible to the rest of the network. If we have an additive structure on the phase spaces M_i (for example, each M_i is a domain in \mathbb{R}^n or an n -torus \mathbb{T}^n) it is natural to take $\mathbf{z}_i^* = 0$.

(2) Suppose M is a manifold with additive (Abelian) group structure and $M_i = M$, all $i \in \mathbf{k}$. If $F_{ij_s}(\mathbf{x}_{j_s}, \mathbf{x}_i) = G_{ij_s}(\mathbf{x}_{j_s} - \mathbf{x}_i)$, all $i, s \in \mathbf{k}$, the coupling is *diffusive*. For general phase spaces, we may define diffusive coupling using the scalar signalling model of [4, 2.5].

⁴In the case of identical phase spaces, we suppose inputs are asymmetric – $F_{ij} \neq F_{i\ell}$, if $j \neq \ell$. For symmetric inputs we require that the corresponding coupling functions are equal. See [53, 5] for background on symmetric inputs.

2.2. Example: N -body problem of classical mechanics. We identify N nodes P_1, \dots, P_N . The variable $\mathbf{X}_j = (\mathbf{x}_j, \mathbf{v}_j) \in \mathbb{R}^3 \times \mathbb{R}^3$ associated with P_j will denote the position and velocity of the body corresponding to P_j . The coupling between nodes will be all-to-all and reflects the gravitational interactions given by Newton's laws. Network differential equations are analytic and may be written uniquely in additive form as

$$\begin{aligned} \mathbf{x}'_i &= \mathbf{v}_i \\ \mathbf{v}'_i &= \sum_{j \neq i} G_{ij}(\mathbf{x}_i - \mathbf{x}_j), \quad i \in \mathbf{N}, \end{aligned}$$

where G_{ij} gives the gravitational interaction between P_i and P_j and the coupling terms G_{ij} are defined and analytic except when P_i, P_j have the same position.

Here a network based reductionist approach has little to offer – even if all the bodies have the same mass. Indeed, the 1-body problem has trivial dynamics: motion in a straight line at constant velocity. In the multi-body problem, the dynamics of the N bodies bears little or no relationship to the trivial dynamics of a single body; network and reductionist approaches to dynamics fail.

2.3. Kuramoto's model: phase oscillator dynamics. Huygens initiated the subject of coupled oscillators in 1665 when he observed the phenomenon of synchronization in coupled pendulum clocks (see [20, 86, 99, 3] for history and contemporary work). Since then there has been continuing strong interest in the global synchronization of a set of coupled pendula or coupled oscillators. Arising out of a study on chemical oscillations, Kuramoto [73] showed that the dynamics of k weakly coupled approximately identical non-linear oscillators could be reduced via averaging to that of a system of k phase oscillators with dynamics given by

$$(3) \quad \theta'_i = \omega_i + \frac{1}{N} \sum_{j=1, i \neq j}^k H_{ij}(\theta_j - \theta_i), \quad i \in \mathbf{k}.$$

The phase variable $\theta_i \in \mathbb{T} = \mathbb{R}/\mathbb{Z}$, and the functions H_{ij} are 1-periodic (we refer to Strogatz' review [98] for more detail and historical background). Just as for the N -body problem, we have additive input structure, diffusive coupling, and the equations are analytic.

Following the work of Sakaguchi & Kuramoto [95], and Ashwin & Swift [15], a much studied case has assumed identical coupling $H_{ij} = H$, $i, j \in \mathbf{k}$ with $H(\theta)$ a trigonometric polynomial $\sum b_n \sin(2\pi(n\theta + \alpha))$ and identical frequencies $\omega_i = \omega$, $i \in \mathbf{N}$. In this situation, the phase

oscillator system is equivariant under the natural action of the symmetric group S_k and there are many dynamically invariant subspaces: If X_1, \dots, X_r are proper disjoint subsets of \mathbf{k} , with $|X_i| \geq 2$, $i \in \mathbf{r}$, then the subspace V_{X_1, \dots, X_r} defined by $\theta_i = \theta_j$ if $i, j \in X_\ell$, for some $\ell \in \mathbf{r}$, is an invariant subspace for the flow of (3).

A single uncoupled phase oscillator is modelled by $\theta' = \omega$ and has frequency ω . If the coupled oscillators have common frequency ω and $H_{ij}(0) = 0$, all i, j , then we have a fully synchronized solution $\theta_i = \theta_0 + \omega t$, $i \in \mathbf{k}$, to (3). This approach works particularly well if H is a trigonometric polynomial with $H(0) = 0$ and for all $i, j \in \mathbf{k}$ either $H_{ij} = H$ or $H_{ij} \equiv 0$. In this case, it is often possible to find stability criteria for the fully synchronized solution, criteria for stable groups of synchronized solutions, bifurcations from the fully synchronized solution, etc. Using mean field techniques, much has also been done on the case where frequencies are approximately identical [98]. However, analytic and reductive methods are largely restricted to the synchronized case. Without some hypothesis about existing synchronization (and the coupling), it is difficult to say much about the dynamics of (3). For example, even small networks with nonsinusoidal coupling can give rise to complicated dynamics [25]. This is, in part, a consequence of the analyticity and the strongly connected character of the network. Working against the use of reductive techniques is that the analytic character of the equations means that changes in the state of any individual node affects the states of all other nodes: it is not usually possible to ‘decouple’ the nodes. Moreover, it has been argued that in many neuroscience applications, the case when the network is fully synchronized is not useful for information processing and is more indicative of pathology than function (for example, Izhkevich [64, §5.1]).

2.4. Synchronous networks. Systems of ordinary differential equations (ODEs) such as (1,3) give mathematical models for what we call *synchronous* networks. By synchronous we mean that all nodes are synchronized to a global clock – the terminology comes from computer science. Indeed, if we assume that each node comes with a local clock, then all the clocks can be set to the same time provided that the network is connected (we ignore the issue of delays, but see [75]). For networks where dynamics is governed by a system of ODEs, synchronization of these clocks is essentially forced by the model and the connectivity of the graph defined by nodes and connections; nodes cannot evolve independently of one another unless the network consists of two disjoint components.

The motivating principle underlying much of the study of network dynamics is that of *reductionism*: the idea that one can infer properties about dynamics of the complete network in terms of dynamics of the individual nodes. The reductionist approach can work well for linear systems but is less clear for synchronous networks governed by a system of non-linear analytic equations. As we have already indicated, it is not helpful at all in the study of the N -body problem. For networks of identical, or approximately identical nodes, the reductionist approach is useful for the study of synchronized behaviour.

Part of the problem that is faced in the analysis of large synchronous networks relates to the *curse of analyticity*: if we assume analytic equations, then it is only under special conditions, such as the assumption of (approximately) identical nodes and coupling, that dynamics decouples within the network to reflect dynamics of individual nodes. Yet reductive methods seem to offer our main hope to understand the detailed structure of dynamics in complex systems (but see [11] for what one can lose by a too strict adoption of a reductionism – obviously, in the global dynamics we want to see more than just the dynamics of individual nodes). Abandoning the hypothesis of analyticity means that we are forced to move away from the standard methods of bifurcation theory for smooth dynamical systems. This, of course, is well (and long) understood by those working in control and piecewise smooth systems [44, 81, 74].

Before we describe what we mean by an asynchronous network, we list characteristic features of synchronous networks⁵ modelled by analytic equations.

Global clock: Nodes all run on the same clock (use the same time).

Global evolution: If we perturb the state of any node, we expect (generically) an immediate effect on the evolution of the states of *all* the other nodes: nodes *never* evolve independently of each other.

Stopped nodes: If a node (or subset of node variables) is at equilibrium or “stopped” for a period of time, it will remain stopped for all future time. If a node has a non-zero initialization, it will never stop (in finite time).

Fixed connection structure: The connection structure of a synchronous network is fixed: it does not vary in time and is not dependent on node states. A consequence is that *one* system of ODEs suffices to model the network dynamics.

⁵We assume continuous dynamics; similar comments hold for discrete dynamics.

Reversibility: Solutions are uniquely defined in backward time.

Remarks 2.3. (1) From the Newtonian perspective of 19th century mathematical physics, the properties we list above are quite unremarkable. Indeed, they are exactly what one would expect of an analytic system of differential equations such as the those used to model the N -body problem. However, from the perspective of the 21st century, biology and modern technology, these properties are quite remarkable: they rarely hold.

(2) We caution that the assumption of a global clock means that ‘events’ in the network – which might include changes in the connection structure – are governed by the global clock rather than by local variables or stochastically. Many networks may have both synchronous and asynchronous features and so it is not always possible to designate a network as either synchronous or asynchronous (but not both).

3. ASYNCHRONOUS NETWORKS: EXAMPLES

In this section we identify some of the features that we incorporate into our model of asynchronous networks and look at a number of characteristic examples. We call these networks *asynchronous*, partly to emphasize the analogy to asynchronous computation in computer science [60]. In general terms, the theory and formalism of asynchronous networks are designed to incorporate commonly observed dynamical events such as stopping and restarting of nodes, and connection, disconnection, and synchronization events (and more, see [42]). It is a consequence of the definition that dynamics on asynchronous networks will only be piecewise smooth as loss of smoothness is forced by variation in connection structure as well as the occurrence of stopping and starting events.

Example 3.1 (Threaded and parallel computation). Threaded or parallelized computer programs provide examples of asynchronous processes from computation. Computation based on a single processor proceeds synchronously and sequentially. The speed of the computation is directly dependent on the clock speed of the processor: the processor clock synchronizes the various steps in the computation. In threaded or parallel computation, computation is broken into blocks or ‘threads’ which are then computed *independently* of each other at a rate that is partly dependent on the clock rates of the processors involved in the computation (these need not be identical). At certain points in the computation, threads need to exchange information with other threads. This process involves stopping and synchronizing (updating) the thread states. Thus, a thread may have to stop and wait for other

threads to complete their computations and update data before it can continue with its own computation.

Threaded computation has a strong non-deterministic flavour – in practice, the running times of each thread yield a stochastic process and differ from run to run. Nevertheless, if the code is correct, then each time the program is run, the computed result will be the same.

Each thread will have its own clock (determined by the processor it belongs to) but threads will be unaware of the times other threads are running on – except during the stopping and synchronization events (these can be managed synchronously – central control – or asynchronously – local control).

This example shows many characteristic features of an asynchronous network: nodes (threads) evolving independently of each other, stopping, synchronization and restarting events. As well the network has a function – transforming a set of initial data into a set of final data in finite time. Here there is the possibility of incorrect code that can lead to inconsistent results (errors) or a process that stops before the desired outcome is achieved (a deadlock). \diamond

Example 3.2 (Thresholds, spiking networks and adaptation). Many mathematical models from engineering and biology incorporate thresholds. For networks, when a node attains a threshold, there are often changes (addition, deletion, weights) in connections to another nodes. For networks of neurons, reaching a threshold can result in a neuron firing (spiking) and short term connections to other neurons (for transmission of the spike). For learning mechanisms, such as Spike-Timing Dependent Plasticity (STDP) [46] relative timings (the order of firing) are crucial [47, 29, 82] and so each neuron, or connection between a pair of neurons, comes with an implicit local clock that governs the adaptation in STDP. In general, networks with thresholds, spiking and adaptation provide characteristic examples of asynchronous networks where dynamics is piecewise smooth and hybrid – a mix of continuous and discrete dynamics. Spiking networks also highlight the importance of efficient communication in large networks: spiking induced connections between neurons are brief and low cost; memory is important for local timings; there is no oscillator clock governing all computations along the lines of a single processor computer. We develop asynchronous aspects of this class of examples further in [42]. \diamond

Example 3.3 (Power grids). Power grids are undergoing a major transformation as traditional centralized power production is replaced by smaller, distributed energy sources. In order to understand the dynamics of large scale power networks, new models have been proposed

and studied where nodes are either generators or loads [45, 35, 34, 94]. In the simplest models, loads are motors and the system near equilibrium. The state of node j may be given by a phase $\theta_j \in \mathbb{T}$, associated with a rotor angle, and a parameter P_j which determines whether the node is a generator, $P_j > 0$, or a load, $P_j < 0$. Neglecting power loss along the line, the transmitted power between two nodes depends on the phase difference between rotors (see [71, Chapter 6] for details). The resulting network dynamics are modelled by the ODEs

$$(4) \quad \theta_j'' + \alpha_j \theta_j' = P_j - \sum_{i=1}^n k_{ij} \sin(\theta_i - \theta_j), \quad j \in \mathbf{n},$$

where (k_{ij}) is a symmetric matrix, all entries positive (zero is allowed). If $\sum_j P_j = 0$ the system can reach an equilibrium. This model assumes we are close to power balance and that generators and motors are all tightly frequency synchronized. In current power grids, synchronization of (large) generators is centrally controlled; a matter of considerable complexity.

While the network described by (4) is synchronous (and the main interest is in the stability of the equilibrium solution), the dynamics of real-world power grids are subject to factors that cannot be adequately described by a synchronous model. In practice it is important to find conditions that guarantee the stability of equilibrium solutions of (4) under extreme events. For example, what happens if the power transfer along a connection $i \rightarrow j$ is too high and trips a circuit breaker or if a severe storm breaks one or more of the connections from a generator? More formally, suppose that each connection from a generator i to a load j has a maximum allowable power transfer P_{ij} . We may redefine the coefficients k_{ij} to be state-dependent

$$\tilde{k}_{ij} = \begin{cases} k_{ij}, & \text{if } |k_{ij} \sin(\theta_i - \theta_j)| < P_{ij}, \\ 0, & \text{if } |k_{ij} \sin(\theta_i - \theta_j)| \geq P_{ij}. \end{cases}$$

We may further suppose that once the circuit breaker has tripped, it is not reset for a specified time which makes the network locally non-autonomous.

The resulting dynamics is that of an asynchronous network. If the power distribution network is correctly set up, circuit breakers will never be tripped because of an initial unbalanced power distribution. So we assume a (slow) Poisson process that randomly removes connections from generators to loads and then reconnects after a time given by a renewal statistic. In this situation, the network may sometimes become disconnected and so a further protocol needs to be implemented

to achieve power balancing (“selective blackouts” on nodes originally fed by the disconnected generator). In this way the network is both asynchronous and adaptive. We caution, however, that even when considering (4) as a state dependent network it is likely to miss important transient effects that occur in a real power network when there are disconnections and/or overloads. Moreover, removing a connection can sometimes *improve* the capacity of the network (Braesse’s paradox [103]). Similar issues arise in connection with synchronization in diffusively coupled networks [84].

Asynchronous features of the dynamics become even more important in power grids that use many (small) generators (for example, wind and solar) whose power output may be time dependent. In this situation, generators are likely to be frequently switched in and out of the network and the control of the power grid is likely to involve a mix of synchronous and asynchronous logic and the appearance of local self-organised structure (see the discussion in [94]). \diamond

Example 3.4 (Transport & production networks). Our final example relates to transport and production networks (including gene transcription networks). Here we look at the simple case of two trains on a single track line with a passing loop. We develop this example more formally in section 6.

Consider two trains travelling in opposite directions along a single track railway line of infinite length. To pass each other, the trains have to go through a passing loop. We model using an asynchronous network for which each train is a node with position determining its state. We describe the dynamics of the asynchronous network. While both trains are approaching the passing loop, they move at nonzero speed and are essentially decoupled (we do not assume the existence of a centralized control). When one train enters the passing loop, it stops (a discontinuous event) and waits for the second train to enter. Entering the passing loop can be seen as a connection event since now the position of the second train becomes relevant for the dynamics of the first train. Once both trains are in the passing loop (a synchronization event), the trains may proceed out of the loop (a second discontinuous event). If one train never arrives, there is a deadlock – the train in the passing loop cannot proceed. Additional features can be added to this example. For example, the trains can be required to stay in the passing loop for a minimum amount of time (the passing loop has a station), the arrival times and/or wait times in the loop can be governed by a stochastic process, the trains might have some internal dynamics

running that has to be synchronized to some level of approximation before they can depart (relevant to production networks). Finally, if we add stations at the end of the line we can view the network as having a *function*: getting from one station to the other in finite time. Networks dynamics will (successfully) complete when both trains have reached their destination stations. If we have a functional network, there is the possibility that networks will not complete for some initial conditions. This leads to the concepts of a *deadlock* where nodes are stopped and cannot restart, and a *livelock* where nodes becomes periodic after a finite time and the network cannot complete. \diamond

We summarize some of the key features of asynchronous networks revealed by the previous examples.

- (1) Variable connection structure and dependencies between nodes. Changes in connection structure may depend on the state of the system or be given by a stochastic process. (All the examples.)
- (2) Synchronization events associated with stopping or waiting states of nodes. (Examples 3.1, 3.4.)
- (3) Order of events may depend on the initialization of the system or stochastic effects. (All the examples.)
- (4) Dynamics is only piecewise smooth and there may be a mix of continuous and discrete dynamics. (All the examples.)
- (5) Aspects involving function, adaptation and control. (Examples 3.2, 3.3, 3.4.)
- (6) Evolution only defined for forward time – systems are non-reversible. (All the examples.)

We conclude this section with a heuristic description of one general class of asynchronous network for which it is often possible, in principle, to quantify the dynamics (we give details in sections 7–9).

Consider a network \mathcal{N} of k nodes N_1, \dots, N_k . Assume that each node has nontrivial intrinsic dynamics – for example, the state of uncoupled node N_i might be governed by an ODE $\mathbf{x}'_i = F_i(\mathbf{x}_i)$. We assume that generally nodes evolve independently. A node N_i will ‘stop’ or enter a ‘waiting state’ when it attains certain thresholds – typically given by a hypersurface in the state space of N_i . Stopping or waiting may involve all or a proper subset of node variables being held fixed. The node N_i will restart when synchronization conditions have been met that involve a subset E of stopped nodes. We illustrate the general process in figure 1 where we model a system with 9 nodes. The nodes evolve independently until they reach certain states. For example, referring to the figure, suppose N_1 has stopped but not N_2 . The node N_2 will continue to evolve until it reaches its stopping state. When both nodes

We remark that there is the possibility that the network function will not be completed (deadlock) or that extreme events (low, non-zero probability) will cause the network to go into a pathological state (a familiar problem in transport and distributed manufacturing and production networks).

Remark 3.5. The setup described above extends to allow for different numbers of inputs and outputs at each stop-synchronize-restart event (in particular, the number of network nodes may vary). One possibility is to define a different partition of variables over a subset of nodes: over time, new ways of grouping network variables may emerge. This approach amounts to a dynamical version of a *Petri net* [31] and is applicable to a variety of transport and production networks.

4. A MATHEMATICAL MODEL FOR DETERMINISTIC ASYNCHRONOUS NETWORKS

In this section we formalize the notion of an asynchronous network. We restrict to the case of deterministic and continuous time asynchronous networks (for the discrete and stochastic cases, see section 10 as well as [42]).

4.1. Preliminaries and an overview. We start by extending the network formalism established in section 2.1 and then give a quick overview of the definition of an asynchronous network, together with an illustrative example.

We consider a network with k nodes, N_1, \dots, N_k . Each node N_i has phase space M_i , $i \in \mathbf{k}$. Set $\mathbf{M} = \prod_{i=1}^k M_i$ – the network phase space. A vector field \mathbf{f} on \mathbf{M} is a *network vector field*.

Stopping, waiting, or synchronization are characteristic features of asynchronous networks. If nodes of a network are stopped or partially stopped, then node dynamics will be constrained to subsets of node phase space. We codify this situation by introducing a *constraining node* N_0 that, when connected to N_i , implies that dynamics on N_i is constrained. We only allow connections $N_0 \rightarrow N_i$, $i \in \mathbf{k}$, and do not consider connections $N_i \rightarrow N_0$, $i \in \mathbf{k}^\bullet$. Henceforth we always assume there is a constraining node and let $\mathcal{N} = \{N_0, N_1, \dots, N_k\}$ denote the set of nodes. Abusing terminology, we often refer to the network \mathcal{N} .

The constraining node N_0 has no dynamics and no associated phase space. In a network with no constraints (that is, there are no connections $N_0 \rightarrow N_i$), the constraining node N_0 plays no role and can be omitted. If we allow constraints, it is possible that there may be more

than one type of constraint on a node N_i . We allow for this possibility in our general setup, but for the present we assume at most one constraint on each node.

We represent interactions between distinct nodes in the network by the network graph. Connections $N_j \rightarrow N_i$ encode dependencies, if $i, j \in \mathbf{k}$, and constraints if $j = 0, i \in \mathbf{k}$. A *connection structure* α is a directed network graph on the nodes \mathcal{N} such that for all $i \in \mathbf{k}, j \in \mathbf{k}^\bullet, i \neq j$, there is at most one directed connection $N_j \rightarrow N_i$. We do not consider connections $N_i \rightarrow N_i, i \in \mathbf{k}$ (self-loops) or connections to the constraining node. If $i \in \mathbf{k}$, then the node N_i is a vertex of α if and only if N_i is the origin or target of a connection (including from N_0). It is convenient to regard the empty connection structure \emptyset as a connection structure (with no connected nodes).

If α is a connection structure on \mathcal{N} , then an α -*admissible vector field* $\mathbf{f} = (f_1, \dots, f_k)$ is a network vector field such that for $i, j \in \mathbf{k}, i \neq j, f_i$ depends on the state \mathbf{x}_j of N_j only if α contains a connection $N_j \rightarrow N_i$. Here we follow the conventions (N1–3) of section 2.1. In particular, if there is a connection $N_j \rightarrow N_i, i, j \in \mathbf{k}$, there is a nontrivial dependence of f_i on the state \mathbf{x}_j of N_j (see also remarks 4.22(4), 5.7(2)). If there is a connection $N_0 \rightarrow N_i$, there is a nontrivial constraint on N_i . An α -admissible vector field has constrained dynamics if there are connections from the constraining node.

A *generalized connection structure* \mathcal{A} is a (nonempty) set of connection structures on \mathcal{N} . An \mathcal{A} -*structure* \mathcal{F} is a set $\mathcal{F} = \{\mathbf{f}^\alpha \mid \alpha \in \mathcal{A}\}$ of network vector fields such that each $\mathbf{f}^\alpha \in \mathcal{F}$ is α -admissible.

Interactions between nodes in asynchronous networks are dynamic rather than static; they may be state or time dependent or both. Here we focus on state dependence; that is, interactions and constraints are determined by the state of the network through an *event map* $\mathcal{E} : \mathbf{M} \rightarrow \mathcal{A}$.

Given a network \mathcal{N} , generalized connection structure \mathcal{A} , \mathcal{A} -structure \mathcal{F} and event map \mathcal{E} , the quadruple $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ defines an *asynchronous network*. Dynamics on \mathcal{N} is given by the state dependent network vector field \mathbf{F} defined by

$$\mathbf{F}(\mathbf{X}) = \mathbf{f}^{\mathcal{E}(\mathbf{X})}(\mathbf{X}), \quad \mathbf{X} \in \mathbf{M}.$$

Subject to regularity conditions, which we give later, the dynamical system will have a uniquely defined semiflow.

In summary, an asynchronous network is a dynamical system involving varying interactions and constraints. Before filling in the details, we give a simple illustration of the general setup that illustrates constrained dynamics and approximate synchronization.

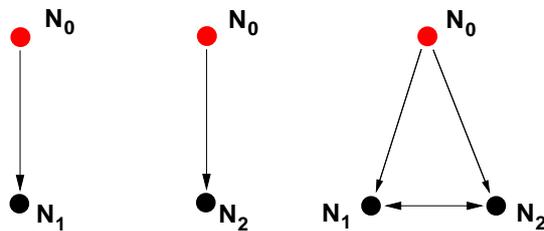


FIGURE 2. Constrained connection structures.

Example 4.1. Suppose $k = 2$ and $M_1 = M_2 = \mathbb{R} \times \mathbb{T}$. Assume that the dynamics of either (uncoupled) node is given by the skew product system

$$\begin{aligned} x' &= f(x) \\ \theta' &= g(\theta, x). \end{aligned}$$

For $x = 0$, assume dynamics is constrained to lie on the line $\{0\} \times \mathbb{T}$ by

$$\begin{aligned} x' &= 0 \\ \theta' &= g(\theta, 0). \end{aligned}$$

When both nodes are constrained, assume coupling of the form

$$\begin{aligned} \theta'_1 &= g(\theta_1, 0) + h(\theta_2 - \theta_1) \\ \theta'_2 &= g(\theta_2, 0) + h(\theta_1 - \theta_2). \end{aligned}$$

Revert to standard (unconstrained) dynamics when $|\theta_1 - \theta_2| < \varepsilon$. Other than the empty structure \emptyset (no coupling), the connection structures for this system are shown in figure 2. Reading from the left, the first connection structure is obtained when $x_1 = 0, x_2 \neq 0$, the second when $x_1 \neq 0, x_2 = 0$, and the final when $x_1 = x_2 = 0$ and $|\theta_1 - \theta_2| \geq \varepsilon$ (this defines the event map $\mathcal{E} : (\mathbb{R} \times \mathbb{T})^2 \rightarrow \mathcal{A}$). Note that the cross-coupling when both nodes are partially stopped is compatible with the constraints $x_1 = 0, x_2 = 0$. \diamond

4.2. Constrained nodes and admissible vector fields.

4.2.1. *Foliations.* Conditions for a constrained node N_i will be given in terms of local *foliations* of the node phase space M_i . We start by recalling basic definitions and refer the reader to the review [76] for general background on foliations.

Recall that a p -dimensional smooth (always C^∞ here) foliation \mathcal{L} of the m -dimensional manifold W consists of a partition $\{L_\alpha \mid \alpha \in \Lambda\}$

of W into connected sets, called *leaves*, such that for every $x \in W$, we can choose an open neighbourhood U of x and smooth embedding $\psi : U \rightarrow \mathbb{R}^m$ such that for each leaf L_α , the components of $\psi(L_\alpha \cap U)$ are given by equations $x^{p+1} = \text{constant}, \dots, x^m = \text{constant}$. Each leaf of a foliation will be an immersed p -dimensional submanifold of W . In the simplest cases, the leaves will be properly embedded (closed) submanifolds of W (it is possible (and easy) to find examples where each leaf is dense in W). We assume in what follows that foliations are proper ($p < m$) and that the manifold W has *finitely* many connected components.

More generally, a smooth foliation of the manifold W will consist of a smooth foliation of each connected component of W . In this case, we require the dimension of leaves to be constant on each connected component of W but the dimension of leaves may vary between components.

Examples 4.2. (1) Every smooth nonsingular vector field on W defines a 1-dimensional smooth foliation of W (“flow-box” theorem of dynamical systems). The leaves are trajectories of the vector field.

(2) If $W = A \times B$, where A and B are manifolds, we have the product foliations $\mathcal{L}(A)$ and $\mathcal{L}(B)$ of W defined by $\mathcal{L}(A) = \{A \times \{b\} \mid b \in B\}$ and $\mathcal{L}(B) = \{\{a\} \times B \mid a \in A\}$. Each leaf $\mathcal{L}(A)$ is transverse to every leaf of $\mathcal{L}(B)$. More generally, foliations $\mathcal{L}, \mathcal{L}'$ are transverse if leaves are transverse. A foliation of W , even by compact 1-dimensional leaves, need not have a transverse foliation. The best-known example is given by the foliation of S^3 into circles given by the Hopf fibration. \diamond

Suppose that \mathcal{L} is a p -dimensional smooth foliation of W with leaves $\{L_\alpha \mid \alpha \in \Lambda\}$. The *tangent bundle along the foliation* $\tau : \mathbb{L} \rightarrow W$ is the smooth vector sub-bundle of the tangent bundle TW of W defined by

$$\mathbb{L} = \bigcup_{x \in L_\alpha, \alpha \in \Lambda} T_x L_\alpha \subset TW.$$

Let $\pi_{\mathbb{L}}$ be a choice of smooth vector bundle projection $\pi_{\mathbb{L}} : TW \rightarrow \mathbb{L}$. The projection map $\pi_{\mathbb{L}}$ is never unique (unless $p = 0$ or m). However, there are often natural choices if, for example, there is a given transverse foliation (as in the case of product foliations).

Definition 4.3. (Notation and assumptions as above.) The pair $(\mathcal{L}, \pi_{\mathbb{L}})$ on W is an \mathcal{L} -*projection structure* on W .

4.2.2. *Constrained nodes.* Following section 4.1, we assume a network $\mathcal{N} = \{N_0, N_1, \dots, N_k\}$, where the nodes N_i have phase space M_i , $i \in \mathbf{k}$. Fix a k -tuple $\mathbf{P} = (p_1, \dots, p_k) \in \mathbb{Z}_+^k$ and set $P = |\mathbf{P}| = \max_{i \in \mathbf{k}} p_i$. In what follows, we assume $P > 0$.

Definition 4.4. (Notation and assumptions as above.) A *constraint structure* $\mathbf{C} = \{(\mathbf{W}_i, \mathcal{L}_i) \mid i \in \mathbf{k}\}$ on \mathcal{N} consists of

- (1) For each $i \in \mathbf{k}$, such that $p_i > 0$, a family $\mathbf{W}_i = \{W_i^\ell \mid \ell \in \mathbf{p}_i\}$ of nonempty open subsets of M_i .
- (2) For each $i \in \mathbf{k}$, such that $p_i > 0$, a family $\mathcal{L}_i = \{\mathcal{L}_i^\ell \mid \ell \in \mathbf{p}_i\}$ where each \mathcal{L}_i^ℓ is a smooth foliation of W_i^ℓ .

The *indexing set* $\iota(\mathbf{C})$ for \mathbf{C} is defined to be $\prod_{p_i \neq 0} \mathbf{p}_i$.

Remarks 4.5. (1) If $p_i = 0$, there are no constraints on N_i .

(2) If $p_i = 1$, we set $\mathbf{W}_i = (W_i, \mathcal{L}_i)$ and \mathcal{L}_i is a smooth foliation of the nonempty open subset W_i of M_i . This is quite general. If we allow the dimension of leaves to vary between different connected components, and the families \mathbf{W}_i to consist of disjoint open subsets of M_i , $i \in \mathbf{k}$, then we can reduce to the case $p_i \leq 1$ by taking $W_i = \bigcup_{\ell} W_i^\ell$ and \mathcal{L}_i to be the foliation determined on W_i by $\mathcal{L}_i|_{W_i^\ell} = \mathcal{L}_i^\ell$, $\ell \in \mathbf{p}_i$.

(3) We have not said anything yet about conditions required of the domains W_i^ℓ except for the requirement of finitely many connected components. We impose additional conditions after we have defined the event map.

(4) The constraints we have defined are always framed in terms of the phase space of a single node. It is straightforward to generalise this concept to consider, for example, foliations of the joint phase space $M_i \times M_j$ for nodes $i \neq j$; this can be interpreted as nodes i and j having a fixed phase relationship. However, we do not consider these possibilities further here.

4.2.3. Connection matrices and admissible vector fields. Interaction between (distinct) nodes in a network can be codified using adjacency matrices. Let $M(k)$ denote the space of $k \times k$ matrices $\beta = (\beta_{ij})_{i,j \in \mathbf{k}}$ with coefficients in $\{0, 1\}$ and $\beta_{ii} = 0$, all $i \in \mathbf{k}$. We refer to elements of $M(k)$ as 0-1 matrices. Each $\beta \in M(k)$ determines a directed graph Γ_β with vertices N_1, \dots, N_k and a directed connection $N_j \rightarrow N_i$ if and only if $\beta_{ij} = 1$. The matrix β is the adjacency matrix of Γ_β .

Fix $\mathbf{P} = (p_1, \dots, p_k) \in \mathbb{Z}_+^k$. Let $M_\bullet(k; \mathbf{P})$ denote the space of $k \times (k+1)$ matrices $\alpha = (\alpha_{ij})_{i \in \mathbf{k}, j \in \mathbf{k}^\bullet}$ such that

- (1) $(\alpha_{ij})_{i,j \in \mathbf{k}} \in M(k)$.
- (2) $\alpha_{i0} \in \mathbf{p}_i^\bullet$, $i \in \mathbf{k}$.

We write $N_j \rightarrow N_i \in \alpha$ if and only if $\alpha_{ij} = 1$. If $\alpha_{i0} \neq 0$, this will indicate a connection $N_0 \rightarrow N_i$, $i \in \mathbf{k}$. We usually abbreviate $M_\bullet(k; \mathbf{P})$ to $M_\bullet(k)$ in what follows.

Remark 4.6. In our applications it will usually be the case that $P = |\mathbf{P}| \leq 1$ and $M_\bullet(k)$ will consist of 0-1 matrices.

Definition 4.7. An element $\alpha \in M_{\bullet}(k)$ defines a *connection structure* on \mathcal{N} .

If $\alpha \in M_{\bullet}(k)$, let α^0 denote the first column $(\alpha_{i0})_{i \in \mathbf{k}}$ of α . We have a natural projection $\pi : M_{\bullet}(k) \rightarrow M(k)$; $\alpha \mapsto \alpha^b$, defined by omitting the column α^0 . We write $\alpha \in M_{\bullet}(k)$ uniquely as

$$\alpha = (\alpha^0 \mid \alpha^b).$$

The column vector α^0 codifies the connections from the constraining node and α^b encodes the connections between the nodes $\{N_1, \dots, N_k\}$.

Example 4.8. Consider the connection structure

$$\alpha = \left(\begin{array}{c|ccc} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{array} \right) \in M_{\bullet}(3).$$

with the associated directed graph depicted in Figure 3. The connec-

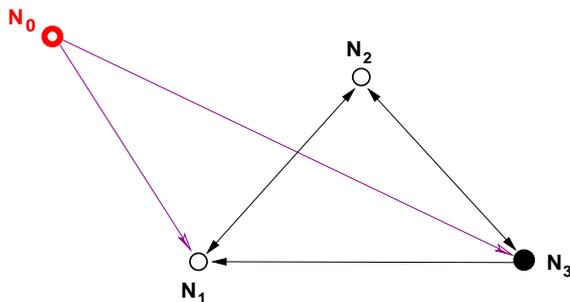


FIGURE 3. The graph defined by the connection structure α . Rather than drawing self-loops, we use a ‘filled node’ if the dynamics of a node depend on its internal state.

tions $N_0 \rightarrow N_1, N_3$ signify that the nodes N_1, N_3 are constrained (for example, they are stopped or constrained to lie on the leaves of a foliation). The evolution of N_1 and N_2 does not depend on the internal state of the nodes (if there is additive input structure, the nodes have no intrinsic dynamics). \diamond

Definition 4.9. Fix a constraint structure $\mathbf{C} = \{(\mathbf{W}_i, \mathcal{L}_i) \mid i \in \mathbf{k}\}$ on \mathcal{N} with indexing set $\iota(\mathbf{C})$ and let $\alpha \in M_{\bullet}(k)$. A smooth⁷ vector field $\mathbf{f} = (f_1, \dots, f_k)$ on \mathbf{M} is an α -admissible vector field if

⁷ C^1 suffices.

- (1) For $i, j \in \mathbf{k}$, $i \neq j$, f_i does not depend on \mathbf{x}_j if $\alpha_{ij} = 0$.
- (2) If $\alpha_{i0} = \ell > 0$, then f_i is tangent to the smooth foliation \mathcal{L}_i^ℓ at all points of $W_i^\ell \subset M_i$. Equivalently, $f_i|_{W_i^\ell}$ defines a section of \mathbb{L}_i^ℓ , the tangent bundle along the foliation \mathcal{L}_i^ℓ .

We extend earlier notation from section 2.1, to define the *dependency set* of node N_i , $i \in \mathbf{k}$, by

$$J(i, \alpha) = \{j \in \mathbf{k} \mid \alpha_{ij} = 1\} \cup \{(0, \alpha_{i0})\}.$$

Remarks 4.10. (1) If $\alpha^0 = \mathbf{0}$, the dependency set encodes functional dependencies for node N_i : f_i will depend on $\mathbf{x}_j \in M_j$ only if $j \in J(i, \alpha)$. If $\alpha_{i0} \neq 0$, then N_i will be subject to the constraint $(W_i^\ell, \mathcal{L}_i^\ell)$, where $\ell = \alpha_{i0}$.

(2) If $\alpha^0 = \mathbf{0}$, then dependencies are governed by (N1–3) of section 2.1. These conditions are already quite weak: we have $\alpha_{ij}^b = 1$ if there exists a dependence of N_i on N_j only on a very small subset of \mathbf{M} . If vector fields are real analytic, then the connection matrix gives an optimal encoding of dependencies: if $\alpha_{ij}^b = 1$, we do not have nonempty open subsets of \mathbf{M} where there is no dependency of N_i on N_j . When there are constraints, (N1–3) still apply but we can give a more satisfactory formulation after we have defined the event map and associated dynamics.

Example 4.11. Suppose that $p_i = 1$ and $\alpha_{i0} = 1$ so that there is a constraining connection $N_0 \rightarrow N_i$. Assume that $\mathbf{f} = (f_1, \dots, f_k)$ is α -admissible, $M_i = \mathbb{R}^\ell$, and \mathcal{L}_i is an $(\ell - p)$ -dimensional foliation of M_i with leaves given by $x_{r_1} = c_1, \dots, x_{r_p} = c_p$. The components $f_i^{r_1}, \dots, f_i^{r_p}$ of $f_i = (f_i^1, \dots, f_i^\ell)$ will be identically zero and the node is partially stopped on each leaf. This is the situation described in example 4.1 of our overview where the 1-dimensional foliation of $\mathbb{R} \times \mathbb{T}$ is $\{\{x\} \times \mathbb{T} \mid x \in \mathbb{R}\}$. \diamond

Remark 4.12. Fix $\alpha \in M_\bullet(k)$. Suppose that $p_i = 1$, $\alpha_{i0} = 1$, and (W_i, \mathcal{L}_i) is the associated constraint structure on M_i . Let $(\mathcal{L}_i, \pi_{\mathcal{L}_i})$ be an \mathcal{L}_i -projection structure on W_i (see definition 4.3). Suppose there are no other nodes connected to N_i (other than the constraining node N_0). Dynamics on $W_i \subset M_i$ will typically be given by the vector field $\pi_{\mathcal{L}_i} F_i$, where F_i gives the unconstrained intrinsic dynamics on N_i . If $W_i = A \times B$, where A, B are submanifolds of W_i , $\mathcal{L}_i = \{A \times \{b\} \mid b \in B\}$, and $\pi_{\mathcal{L}_i}$ is the tangent map of the projection $A \times B \rightarrow A$, then we may write $F_i = (G_i, H_i)$, where $G_i = \pi_{\mathcal{L}_i} F_i$, and the constrained dynamics on W_i is given by the vector field G_i . In particular, the B -variables will be stopped. We handle coupling similarly. If $N_0 \rightarrow N_i \leftarrow N_j$,

then the coupling vector field associated with N_j will be tangent to the foliation \mathcal{L}_i on W_i and so will not break the constrained dynamics.

4.2.4. *Generalized connection structures and families of admissible vector fields.*

Definition 4.13. A nonempty subset \mathcal{A} of $M_\bullet(k)$ defines a *generalized connection structure* on \mathcal{N} .

Assume given a fixed constraint structure $\mathbf{C} = \{(\mathbf{W}_i, \mathcal{L}_i) \mid i \in \mathbf{k}\}$ on \mathcal{N} with indexing set $\iota(\mathbf{C})$.

Definition 4.14. Suppose that \mathcal{A} is a generalized connection structure on \mathcal{N} . If $\mathcal{F} = \{\mathbf{f}^\alpha \mid \alpha \in \mathcal{A}\}$ is a set of vector fields on \mathbf{M} such that each \mathbf{f}^α is a α -admissible vector field, we refer to the family \mathcal{F} as an *\mathcal{A} -structure* on \mathcal{N} .

Remark 4.15. We do not insist that the admissible vector fields defined in definition 4.14 for different connection matrices have to be related. In applications, there will usually be strong relations between the components of \mathbf{f}^α . For example, if the input structure is additive, not only is addition or deletion of connections natural but the additive structure also gives explicit relationships between the \mathbf{f}^α (see sections 2.1, 4.5).

4.3. **Asynchronous networks.** We now have most of the basic infrastructure in place for the definition of an asynchronous network.

Definition 4.16. An *asynchronous network structure* $(\mathcal{N}, \mathcal{A}, \mathcal{F})$ consists of

- (1) A finite set $\mathcal{N} = \{N_0, N_1, \dots, N_k\}$ of nodes with associated phase spaces M_1, \dots, M_k (N_0 denotes the constraining node and has no phase space defined).
- (2) A generalized connection structure \mathcal{A} on \mathcal{N} .
- (3) An \mathcal{A} -structure \mathcal{F} on \mathcal{N} .

4.3.1. *Event maps and event sets.*

Definition 4.17. An *event map* for an asynchronous network structure $(\mathcal{N}, \mathcal{A}, \mathcal{F})$ is a surjective map $\mathcal{E} : \mathbf{M} \rightarrow \mathcal{A}$.

Remark 4.18. More generally we allow for event maps that are time and space dependent. Typically, the event map will be stochastic in time and/or involve local clocks. We restrict attention here to autonomous deterministic systems but indicate some of the interesting dynamics that can occur when we allow a stochastic component in section 10.

Let $\mathcal{E} : \mathbf{M} \rightarrow \mathcal{A}$ be an event map. For each $\alpha \in \mathcal{A}$, define the *event set* $E^\alpha \subset \mathbf{M}$ by

$$E^\alpha = \{\mathbf{X} \in \mathbf{M} \mid \mathcal{E}(\mathbf{X}) = \alpha\}.$$

The event sets partition the network phase space \mathbf{M} . We require additional conditions on the event map when there are constraints. These conditions relate the event sets to the constraint structure $\mathbf{C} = \{(\mathbf{W}_i, \mathcal{L}_i) \mid i \in \mathbf{k}\}$ on \mathcal{N} .

Let $\pi_i : \mathbf{M} \rightarrow M_i$ denote the projection map onto the phase space of N_i , $i \in \mathbf{k}$. Given $i \in \mathbf{k}$, $\ell \in \mathbf{p}_i$, define

$$E_i^\ell = \bigcup_{\{\alpha \mid \alpha_{i0} = \ell\}} \pi_i(E^\alpha) \subset M_i.$$

Definition 4.19. The event map $\mathcal{E} : \mathbf{M} \rightarrow \mathcal{A}$ is *constraint regular* if for all $i \in \mathbf{k}$, $\ell \in \mathbf{p}_i$, we have

$$\overline{E_i^\ell} \subset W_i^\ell$$

In future, we always assume the event map is constraint regular (and omit the prefix constraint regular).

Definition 4.20. An (autonomous deterministic) asynchronous network $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ consists of an asynchronous network structure $(\mathcal{N}, \mathcal{A}, \mathcal{F})$ together with an event map \mathcal{E} .

Remark 4.21. If \mathcal{A} consists of a single connection structure α (with or without constraints), then \mathcal{F} consists of one vector field $\mathbf{f} = \mathbf{f}^\alpha$, with dependencies given by α . We recover a synchronous network with dynamics defined by \mathbf{f} and a fixed connection structure.

4.4. Dynamics on an asynchronous network. An asynchronous network $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ uniquely determines the *network vector field* \mathbf{F} by

$$(5) \quad \mathbf{F}(\mathbf{X}) = \mathbf{f}^{\mathcal{E}(\mathbf{X})}(\mathbf{X}), \quad \mathbf{X} \in \mathbf{M}.$$

Remarks 4.22. (1) It is easy to give a discrete dynamics version of definition 4.20. In this case dynamics is given by the map $\mathbf{F} : \mathbf{M} \rightarrow \mathbf{M}$ defined by

$$(6) \quad \mathbf{F}(\mathbf{X}) = \mathbf{f}^{\mathcal{E}(\mathbf{X})}(\mathbf{X}).$$

There is no issue about the iteration (time evolution) being well defined though, of course, without further conditions the map \mathbf{F} may be everywhere discontinuous.

(2) Equations (5,6) define *state dependent* dynamical systems and similar structures have been used previously in engineering applications (see, for example, [57]). As we shall see in section 6.1.1, there is a

relationship with Filippov systems (which is explored further in [23]). However, the notion of an integral curve for an asynchronous network is generally different from that of a Filippov system, see section 4.4.1. (3) The network vector field (5) does not uniquely determine \mathcal{A} , \mathcal{E} or \mathcal{F} even though the choice of \mathcal{A} , \mathcal{E} and \mathcal{F} is usually naturally determined by the problem. Sometimes it is convenient to view the vector field (6) as the basic object and regard asynchronous networks $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$, $(\mathcal{N}, \mathcal{A}', \mathcal{F}', \mathcal{E}')$ on \mathcal{N} as being *equivalent* if $\mathbf{f}^{\mathcal{E}(\mathbf{X})}(\mathbf{X}) = \mathbf{f}'^{\mathcal{E}'(\mathbf{X})}(\mathbf{X})$ for all $\mathbf{X} \in \mathbf{M}$.

(4) Since the event sets $\{E^\alpha \mid \alpha \in \mathcal{A}\}$ partition \mathbf{M} , the network vector field \mathbf{F} only depends on the values of \mathbf{f}^α on E^α . Nevertheless, we assume that \mathbf{f}^α is smooth on \mathbf{M} . We could instead have required that each \mathbf{f}^α was defined as smooth map in the sense of Whitney [102] on $\overline{E^\alpha}$ (and so admits a smooth extension to \mathbf{M}). Similar remarks hold for the constrained case, though now we cannot expect the foliation to extend to the phase space M_i (for example, there are no 1-dimensional foliations of the two-dimensional sphere) and so we require local existence of the foliation and global extension of the vector field. We opt for the definition of admissible vector fields on the entire phase space and smooth foliations on open neighbourhoods of event related sets, because in practice these vector fields and foliations are usually naturally defined this way.

4.4.1. *Regularity of event maps and existence of integral curves.* Without further conditions on the event map, the vector field \mathbf{F} determined by an asynchronous network $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ may not have integral curves through every (or any) point of the phase space. Before we give the formal definition of an integral curve, we give an instructive example.

Example 4.23. Consider an asynchronous network with two nodes N_1, N_2 , a constraining node N_0 and connection structure

$$\mathcal{A} = \{\alpha_1 = \{N_0 \rightarrow N_1\}, \alpha_2 = \{N_0 \rightarrow N_2\}\}.$$

Define associated vector fields $\mathbf{F}^{\alpha_1} = (0, -1)$, $\mathbf{F}^{\alpha_2} = (1, 0)$. Let $B = \{x \in \mathbb{R}^2 \mid \|x\| \leq 1\}$ denote the closed unit ball. We take as event map

$$(7) \quad \mathcal{E}(\mathbf{X}) = \begin{cases} \alpha_1 & \text{if } \mathbf{X} \notin B \\ \alpha_2 & \text{if } \mathbf{X} \in B \end{cases}$$

and obtain a differential equation with discontinuous right hand side given by $\mathbf{X}' = \mathbf{F}^{\mathcal{E}(\mathbf{X})}(\mathbf{X})$. We indicate dynamics in Figure 4. Observe that it is not possible to define the trajectory $\phi_{\mathbf{X}} : \mathbb{R}_+ \rightarrow \mathbb{R}^2$ through any point $\mathbf{X} = (x, y) \in B$ with $1 \geq y > 0$. Indeed, such a trajectory will meet the arc $(a, b] \subset \partial B$ where it can no longer be continued. All

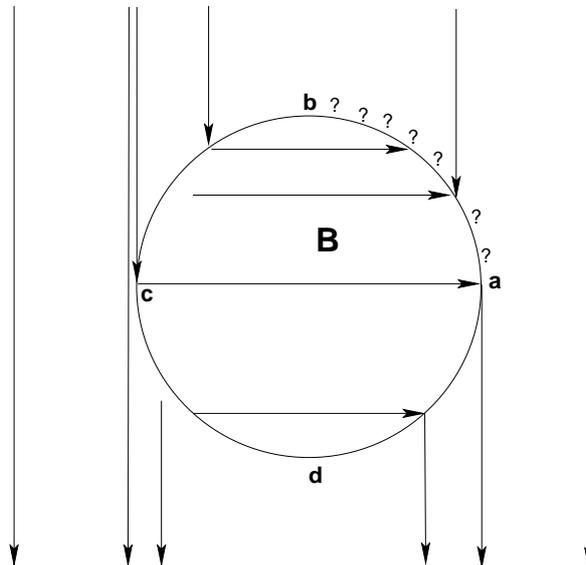


FIGURE 4. Dynamics on an asynchronous networks is not necessarily well defined according to definition 5: trajectories with initial condition $(x, y) \in B$, $y \in (0, 1)$ cannot be continued across ∂B .

other trajectories are well defined. One way of resolving the problem of continuation is Filippov’s approach [44, Chapter 2, page 50] to get a vector field on the arc $[a, b]$: at each point $\mathbf{X} \in [a, b] \subset \partial B$ define a vector field by taking the unique convex combination $\lambda \mathbf{F}^{\alpha_1}(\mathbf{X}) + (1 - \lambda) \mathbf{F}^{\alpha_2}(\mathbf{X})$ which is tangent to ∂B . In this case, the trajectory “slides” along the arc $[a, b]$. We can also apply the Filippov construction along the boundary arc $[c, d]$ – this leads to different dynamics. Alternatively, and more appropriately, for this problem, we can modify the event set B by writing $B = B_2 \cup B_3$, where $B_3 = (a, b] \subset \partial B$. Take say $\alpha_3 = \{N_1 \leftrightarrow N_2\}$ and define $\mathcal{E}(\mathbf{X}) = \alpha_j$ if $\mathbf{X} \in B_j$, $j = 2, 3$. Let \mathbf{F}^{α_3} be any smooth vector field on \mathbb{R}^2 such that

$$\mathbf{F}^{\alpha_3}(x, y) = (y, -x), \quad (x, y) \in [a, b].$$

With these modifications, trajectories $\phi_{\mathbf{X}}$ are defined for all forward time. Moreover, we move along trajectories with constant speed 1,

$$\|\mathbf{F}^{\mathcal{E}(\mathbf{X})}(\mathbf{X})\| = 1, \quad \text{for all } \mathbf{X} \in \mathbb{R}^2.$$

This condition does not hold for the Filippov vector field discussed above (we can modify the constraints to obtain the second solution in

the Filippov setting). We discuss the relationship between asynchronous networks and Filippov systems further in section 6.1.1. \diamond

Definition 4.24. Let $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ be an asynchronous network with associated vector field \mathbf{F} given by $\mathbf{F}(\mathbf{X}) = \mathbf{f}^{\mathcal{E}(\mathbf{X})}(\mathbf{X})$, $\mathbf{X} \in \mathbf{M}$. An *integral curve* or *trajectory* for \mathbf{F} with initial condition $\mathbf{X}_0 \in \mathbf{M}$ is a map $\phi : [0, T) \rightarrow \mathbf{M}$, $T \in (0, \infty]$, satisfying

- (1) $\phi(0) = \mathbf{X}_0$.
- (2) ϕ is continuous.
- (3) There exists a closed countable subset D of $[0, T)$ such that for every $u \in D$, there exists $v \in D \cup \{T\}$, $v > u$, such that
 - (a) $(u, v) \cap D = \emptyset$.
 - (b) ϕ is C^1 on (u, v) and $\phi'(t) = \mathbf{F}(\phi(t))$, $t \in (u, v)$.
 - (c) $\lim_{t \rightarrow u^+} \phi'(t) = \mathbf{F}(\phi(t))$.

Remarks 4.25. (1) It is routine to verify (see the proof of proposition 4.31) that if $\psi : [0, S) \rightarrow \mathbf{M}$ is another integral curve with initial condition \mathbf{X}_0 , then $\psi = \phi$ on $[0, \min\{S, T\})$ (uniqueness). As a consequence we can define the *maximal* integral curve $\phi : [0, T_{\max}) \rightarrow \mathbf{M}$ with initial condition \mathbf{X}_0 . In the sequel, we assume that integral curves are maximal unless otherwise indicated.

(2) If $T = \infty$ in the definition, the trajectory $\phi : \mathbb{R}_+ \rightarrow \mathbf{M}$ is *complete*.

(3) The set D may have accumulation points in D – accumulation is always from the left on account of condition (3a). In the examples we consider D , will always be a finite set.

(4) Typically, it will be the case that there exists $\alpha \in \mathcal{A}$ such that $\mathcal{E}(\mathbf{X}) = \alpha$ for $\mathbf{X} \in \phi((u, v))$ and so $\phi((u, v)) \subset E^\alpha$. Condition (3c) implies that if $\mathcal{E}(\phi(u)) = \beta \neq \alpha$, we must have $\mathbf{f}^\alpha(\phi(u)) = \mathbf{f}^\beta(\phi(u))$.

Definition 4.26. The asynchronous network $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ is *proper* if

- (1) for every $\mathbf{X} \in \mathbf{M}$ there is a trajectory $\phi_{\mathbf{X}}$ with initial condition \mathbf{X} and
- (2) every maximal trajectory of \mathbf{F} is complete (in the case of $\partial\mathbf{M} \neq \emptyset$, non-compact \mathbf{M} , or open regions in \mathbb{R}^n , trajectories may eventually exit \mathbf{M} or escape to infinity in finite time).

Remarks 4.27. (1) If $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ is proper and \mathbf{M} is compact, network dynamics is given by a semiflow $\Phi : \mathbf{M} \times \mathbb{R}_+ \rightarrow \mathbf{M}$. Although $\Phi(\mathbf{X}, t)$ will be continuous as a function of $t \in \mathbb{R}_+$, it need not be continuous as a function of $\mathbf{X} \in \mathbf{M}$ (fixed $t > 0$).

(2) Suppose $\partial M_i \neq \emptyset$, $i \in \mathbf{k}$, and let $\phi = (\phi_1, \dots, \phi_k)$ be a trajectory such that for some smallest $s \in \mathbb{R}_+$, there exists $i \in \mathbf{k}$ such that $\phi_i(s) \in \partial M_i$ with $\mathbf{F}_i(\mathbf{X})$ outward pointing. The domain of ϕ is necessarily $[0, s]$. Under additional hypotheses, it may be possible to extend ϕ

beyond $[0, s]$ (for example, by requiring $\phi_i(t) = \phi_i(s)$, $t \geq s$ – see also sections 8.7, 7.3).

Let $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ be an asynchronous network. As we shall see, the event sets E^α need be neither open nor closed subsets of \mathbf{M} . It is helpful to assume some regularity that implies that each event set can be partitioned into a (locally) finite set of smooth submanifolds of \mathbf{M} . For example, if \mathbf{M} is a domain in \mathbb{R}^n (or \mathbb{T}^n), we might assume the E^α are all *semialgebraic* sets⁸. This condition implies that E^α has finitely many connected components. More generally we might suppose event sets are *semianalytic*⁹. We say the event structure is *regular* if the event sets E^α are all semianalytic. For the examples we consider in this paper, event sets will usually be quite simple (for example, half-lines or half-planes) and semialgebraic (the semialgebraic structure will reflect logical conditions on the underlying dynamics).

Definition 4.28. An asynchronous network $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ is *amenable* if

- (1) The event structure is regular.
- (2) If $\mathbf{X} \in E^\alpha$, $\alpha \in \mathcal{A}$, then there exists $t(\mathbf{X}) \in (0, \infty) \cup \{\infty\}$ such that the trajectory $\phi_{\mathbf{X}}$ of \mathbf{f}^α through \mathbf{X} lies in E^α for $t \in [0, t(\mathbf{X}))$

$$\phi_{\mathbf{X}}(t) \in E^\alpha, \quad t \in [0, t(\mathbf{X})).$$

- (3) Either \mathbf{M} is compact or, in the case when \mathbf{M} is a domain in \mathbb{R}^n , the vector fields have at most linear growth on \mathbf{M} : $\exists a, b > 0$ such that

$$\|\mathbf{f}^\alpha(\mathbf{X})\| \leq a + b\|\mathbf{X}\|, \quad \mathbf{X} \in \mathbf{M}, \quad \alpha \in \mathcal{A}.$$

Remarks 4.29. (1) Condition (2) of definition 4.28 suggests that the vector field \mathbf{f}^α should in some sense be tangent to E^α . The issue of tangency can be made more precise using the regularity assumption which implies that E^α has a locally finite stratification into submanifolds without boundary. For example, we may choose the canonical Whitney regular stratification of each event set [48, 78]. This allows us to unambiguously define tangency at points of E^α which do not lie in the boundary of strata. Matters are not so straightforward at points lying in the boundary of strata and in the second example below we indicate how the geometric structure of the event set can impose strong constraints on associated vector fields.

⁸Defined by algebraic equalities and inequalities. See [48, 27] for precise definitions and properties.

⁹Locally defined by real analytic equalities and inequalities.

(2) If an event set is a connected closed submanifold without boundary, then it follows from definition 4.28(2) that any trajectory that meets the event set will never leave the event set.

(3) Condition (3) of definition 4.28 implies that trajectories are either defined for all positive time or trajectories eventually exit the domain (for example, if $\mathbf{M} \subsetneq \mathbb{R}^n$).

(4) We can assume the time $t(\mathbf{X})$ of definition 4.28 is maximal. That is, $\phi_{\mathbf{X}}(t(\mathbf{X})) \notin E^\alpha$.

(5) Noting remark 4.22(3), we may extend our definition of amenability to include all asynchronous networks which are equivalent to an amenable network in the sense of definition 4.28.

Examples 4.30. In figure 5 we show two examples of event sets in \mathbb{R}^2 (we ignore the explicit dependence of events on connection structure).

(1) In figure 5(1), we have four semialgebraic event sets defined by

$$E^1 = \{(x, 0) \mid x < 0\}, \quad E^2 = \{(0, y) \mid y > 0\}, \quad E^3 = \{(x, x) \mid x \neq 0\},$$

$$E^0 = \mathbb{R}^2 \setminus \bigcup_{i=1}^3 E^i.$$

The event sets are neither open nor closed. We define associated vector

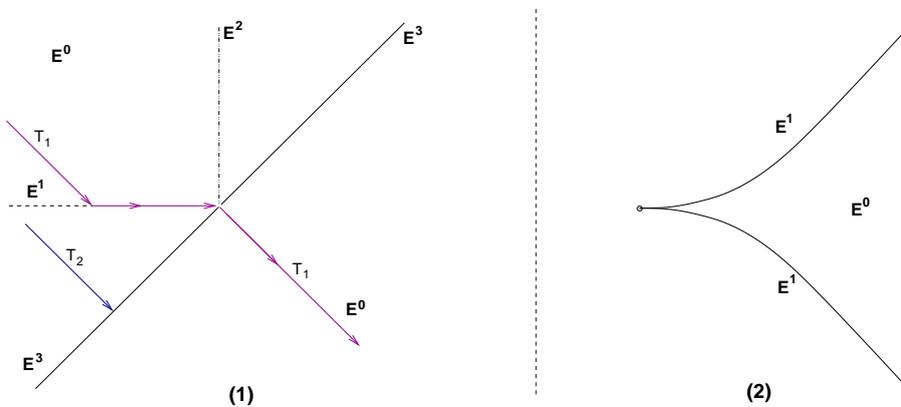


FIGURE 5. Event sets in \mathbb{R}^2

fields f^j on \mathbb{R}^2 by

$$f^1(x, y) = (1, 0), \quad f^2(x, y) = (0, -1), \quad f^3 \equiv 0, \quad f^0 = f^1 + f^2.$$

We show two forward trajectories in figure 5(1). The trajectory T_1 passes through the origin and enters the bottom right quadrant (as do all trajectories starting in the top left quadrant). On the other hand the trajectory T_2 stops on the diagonal (the event set E^3). As

we show later, we can interpret this setup as a model of dynamics on a passing loop (see example 3.4) with the addition of a collision event $-E^3$. The network is amenable: the dynamics determines a unique semiflow $\Phi : \mathbb{R}^2 \times \mathbb{R}_+ \rightarrow \mathbb{R}^2$ which does *not* extend to a flow. Observe also that the semiflow is not continuous on $\overline{E^1 \cup E^2 \cup E^3}$. However, the semiflow is continuous if we restrict to $\{(x, y) \mid xy \leq 0\}$. In the next section we discuss a continuous extension of this restricted semiflow to \mathbb{R}^2 motivated by Filippov systems. However, notice that dynamics near E^3 are not those given by the standard Filippov construction – the line E^3 would be regarded as a removable singularity.

(2) In the second example, the event set E^1 is the cusp defined by $y^2 = x^3$, $x \neq 0$. In this case any smooth (at least C^1) vector field on \mathbb{R}^2 which is tangent to E^1 must vanish at $\{(0, 0)\}$ (an example of such a vector field is $(2ax, 3ay)$, $a \in \mathbb{R}$). If we require amenability, then all trajectories which meet E^1 will never leave E^1 . \diamond

Proposition 4.31. *An amenable asynchronous network is proper.*

Proof. We give details for the case when \mathbf{M} is compact. Fix $\mathbf{X} \in \mathbf{M}$. Suppose that $\phi_i : [0, s_i) \rightarrow \mathbf{M}$ are forward trajectories for \mathbf{F} through \mathbf{X} , $i \in \mathbf{2}$. Using uniqueness of solutions of differential equations and definition 4.28(2), it is easy to see that $\phi_1 = \phi_2$ on $[0, s_1) \cap [0, s_2)$. It follows that if we define

$$T = \sup \{t \mid \text{there is a trajectory } \psi : [0, t) \rightarrow \mathbf{M} \text{ through } \mathbf{X}\}$$

then we have a unique trajectory $\phi : [0, T) \rightarrow \mathbf{M}$ through \mathbf{X} . If $T = \infty$, we are done. But if $T < \infty$, then we can extend ϕ to $[0, T]$ by $\phi(T) = \lim_{t \rightarrow T^-} \phi(t)$ (remarks 4.29(3)). If $\phi(T) \in E^\alpha$ then by definition 4.28(2), ϕ extends to $[0, T + t(\phi(T))]$, where $t(\phi(T)) > 0$. This contradicts the maximality of T and so $T = \infty$. \square

Remarks 4.32. (1) The proof of proposition 4.31 says nothing about the number of changes in the event map that occur along a trajectory. Without further conditions on the vector fields \mathbf{f}^α there may be a countable infinity of changes with countably many accumulation points (this is catered for in definition 4.24).

(2) As shown in examples 4.30(1), the semiflow given by proposition 4.31 need not be continuous (as a function of (\mathbf{X}, t)). In practice, it is usually not hard to give conditions that imply continuity of the semiflow on amenable networks. For example, suppose that $E^1, E^0 = \mathbf{M} \setminus E^1$ are the event sets of an amenable asynchronous network and $\overline{E^1}$ is a smooth hypersurface with nonempty boundary $Y = \overline{E^1} \setminus E^1$. If \mathbf{f}^1 is outward pointing along Y , in particular, non-zero, then the semiflow given by proposition 4.31 is continuous. Note

that amenability implies that \mathbf{f}^0 cannot be nonzero and inward pointing to E^1 anywhere along Y . Also it is not enough for E^1 to be smooth as is shown by examples 4.30(2).

(3) Amenability is sufficient but not necessary for properness.

4.4.2. *Dynamics as semiflows for an amenable asynchronous network.* We assume $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ is an amenable asynchronous network. For simplicity we assume that for each $\alpha \in \mathcal{A}$, \mathbf{f}^α has flow $\Phi^\alpha : \mathbf{M} \times \mathbb{R} \rightarrow \mathbf{M}$ (for example, if \mathbf{M} is compact or $\mathbf{M} = \mathbb{R}^n$ and \mathbf{f}^α has linear growth). We have a well defined semiflow

$$\Phi : \mathbf{M} \times \mathbb{R}_+ \rightarrow \mathbf{M}.$$

that may be computed in terms of the flows Φ^α . In order to construct Φ , it suffices to define the trajectory $\Phi_{\mathbf{X}}$ through $\mathbf{X} \in \mathbf{M}$. We proceed just as in the proof of proposition 4.31. If $\Phi_{\mathbf{X}}$ is defined on $[0, T]$ and $\Phi_{\mathbf{X}}(T) \in E^\alpha$, then there exists $\delta > 0$ such that $\Phi_{\mathbf{X}}(t) \in E^\alpha$ for $t \in [T, T + \delta]$ and so

$$\Phi_{\mathbf{X}}(t) = \Phi^\alpha(\Phi_{\mathbf{X}}(T), t - T), \quad t \in [T, T + \delta].$$

In the case of most interest there there will only be finitely many events, occurring at times $0 < t_1 < t_2 < \dots < t_\ell < \infty$, and we have

$$\Phi_{\mathbf{X}}(t) = \Phi^{\alpha(k)}(\dots \Phi^{\alpha(2)}(\Phi^{\alpha(1)}(\mathbf{X}, t_1), t_2 - t_1) \dots, t - t_p), \quad t \in [t_p, t_{p+1}),$$

where $\alpha(j) = \mathcal{E}(\Phi_{\mathbf{X}}(t_j)) \in \mathcal{A}$, $j \in \ell$, and $t_{\ell+1} = +\infty$ (note that t_j , $\alpha(j)$ and ℓ all depend implicitly on \mathbf{X}).

Assuming the case of finitely many events and following the previous notation, choose $i \in \mathbf{k}$, $\mathbf{X} \in \mathbf{M}$ and let $\mathbf{x}_i(t) = \pi_i(\mathbf{X}(t)) \in M_i$ be the component of the trajectory through \mathbf{X} lying in M_i . The evolution of $\mathbf{x}_i(t)$ for $t \in [t_s, t_{s+1}]$, $s \leq \ell$, depends only on the nodes connected to N_i as determined by $\alpha(s)$. That is,

$$\mathbf{x}_i(t) = \pi_i(\Phi_{\mathbf{X}}(t)) = \Phi^{\alpha(s)}((\mathbf{x}_{j_1}(t), \dots, \mathbf{x}_{j_s}(t)), t), \quad t \in [t_s, t_{s+1}],$$

where $j \in \{j_1, \dots, j_p\}$ if and only if $N_j \rightarrow N_i \in \alpha(s)$.

4.5. **Asynchronous networks with additive input structure and constraints.** Synchronous networks with additive input structure give rise to asynchronous networks and we outline this ‘‘asynchronization’’ of a synchronous networks below. We do not discuss conditions on the event sets for an asynchronifed network to be proper.

Recall from section 2.1 that a k node (synchronous) network \mathcal{N} with additive input structure has dynamics defined by

$$(8) \quad f_i(\mathbf{x}_i; \mathbf{x}_{j_1}, \dots, \mathbf{x}_{j_{e_i}}) = F_i(\mathbf{x}_i) + \sum_{s=1}^{e_i} F_{ij_s}(\mathbf{x}_{j_s}, \mathbf{x}_i), \quad i \in \mathbf{k}.$$

We view F_i as defining the intrinsic dynamics of the node N_i and F_{ij} as giving the (nontrivial) coupling from N_j to N_i . Here it is natural to remove and later reinsert connections between nodes.

Let Γ be the network graph determined by (8) with associated 0-1 matrix $\gamma \in M(k)$. Let $\mathcal{A} \subset M_\bullet(k)$ be a generalized connection structure such that

- (1) $(0 | \gamma) \in \mathcal{A}$,
- (2) for all $\alpha = (\alpha^0 | \alpha^b)$ the matrix α^b defines a subgraph of Γ , and
- (3) $\alpha_{i0} \in \{0, 1\}$ for all $i \in \mathbf{k}$, $\alpha \in \mathcal{A}$.

For each $\alpha \in \mathcal{A}$, define the α -admissible vector field \mathbf{f}^α by

$$(9) \quad f_i^\alpha(\mathbf{x}_i; \mathbf{x}_{j_1}, \dots, \mathbf{x}_{j_{e_i}}) = (1 - \alpha_{i0}) \left(F_i(\mathbf{x}_i) + \sum_{s=1}^{e_i} \alpha_{ij_s} F_{ij_s}(\mathbf{x}_{j_s}, \mathbf{x}_i) \right), \quad i \in \mathbf{k}.$$

In this case there is one global constraint structure with M_i foliated by single points. If we choose an event map $\mathcal{E} : \mathbf{M} \rightarrow \mathcal{A}$ and take $\mathcal{F} = \{\mathbf{f}^\alpha | \alpha \in \mathcal{A}\}$, then $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ will be an asynchronous network. In this way, every synchronous network with additive input structure leads naturally to many asynchronous networks.

Definition 4.33. An asynchronous network such that the admissible vector fields \mathbf{f}^α are defined as in (9) is an *asynchronous network with additive input structure*.

For an asynchronous network with additive input structure there is close relationship admissible vector fields: all are derived from the network vector field of a synchronous network. In particular, the dynamics of any node is fully determined by the nodes that are connected to it.

Definition 4.34. An asynchronous network $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ is *input consistent* if for any node N_i and $\alpha, \beta \in \mathcal{A}$ with dependency sets satisfying $J(i, \alpha) = J(i, \beta)$ we have $f_i^\alpha = f_i^\beta$.

Lemma 4.35. *Asynchronous networks with additive input structure are input consistent.*

Proof. Obvious from the explicit dependencies given in (9). □

We conclude this section by generalizing the previous construction to allow for arbitrary constraint structures. We start again with the system (8) but now assume there is a constraint structure $\mathbf{C} = \{(\mathbf{W}_i, \mathcal{L}_i) | i \in \mathbf{k}\}$ on \mathcal{N} . We follow the notational conventions and assumptions of section 4.2. In particular, $p_i \in \mathbb{Z}_+$ will be the number of constraints on node N_i , $\mathbf{P} = (p_1, \dots, p_k) \in \mathbb{Z}_+^k$ and $P = |\mathbf{P}| = \max_{i \in \mathbf{k}} p_i$. For each constraint $(W_i^\ell, \mathcal{L}_i^\ell)$ on N_i , choose a projection structure $(\mathcal{L}_i^\ell, \pi_{\mathbb{L}_i^\ell})$.

Just as above, let $\gamma \in M(k)$ be the 0-1 matrix determining the graph Γ defined by (8) and let $\mathcal{A} \subset M_\bullet(k) = M_\bullet(k; P)$ be a generalized connection structure such that

- (1) $(0 | \gamma) \in \mathcal{A}$,
- (2) for all $\alpha \in \mathcal{A}$ the matrix α^b defines a subgraph of Γ ,
- (3) $\alpha_{i0} \in \mathbf{p}_i^\bullet$ for all $i \in \mathbf{k}$, $\alpha \in \mathcal{A}$.

Let $\mathcal{E} : \mathbf{M} \rightarrow \mathcal{A}$ be a (regular) event map. For each $\alpha \in \mathcal{A}$, define the α -admissible vector field \mathbf{f}^α by

$$(10) \quad f_i^\alpha(\mathbf{x}_i; \mathbf{x}_{j_1}, \dots, \mathbf{x}_{j_{e_i}}) = F_i^\alpha(\mathbf{x}_i) + \sum_{s=1}^{e_i} F_{is}^\alpha(\mathbf{x}_{j_s}, \mathbf{x}_i), \quad i \in \mathbf{k},$$

where

- (1) $F_i^\alpha = \pi_{\mathbb{L}_i^\ell} F_i$ if $\alpha_{i0} = \ell \in \mathbf{p}_i$, and $F_i^\alpha = F_i$, if $\alpha_{i0} = 0$.
- (2) $F_{is}^\alpha = \pi_{\mathbb{L}_i^\ell} F_{is}$, if $\alpha_{i0} = \ell \in \mathbf{p}_i$, and $F_{is}^\alpha = F_{is}$, if $\alpha_{i0} = 0$.

If we define $\mathcal{F} = \{\mathbf{f}^\alpha | \alpha \in \mathcal{A}\}$, then $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ will be an asynchronous network with constraint structure \mathbf{C} .

5. PRODUCTS AND AMALGAMATIONS OF ASYNCHRONOUS NETWORKS

In this section we define the product of asynchronous networks and find conditions that imply the network cannot be written as a product. We also define the amalgamation of a set of asynchronous networks with a common node set.

5.1. Products. Given $\alpha, \beta \in M(k)$, define $\alpha \vee \beta \in M(k)$ (the join of α and β) by

$$(\alpha \vee \beta)_{ij} = \max\{\alpha_{ij}, \beta_{ij}\}, \quad i, j \in \mathbf{k}$$

(the max-plus addition of tropical algebra [61]). Let \emptyset denote the empty connection structure and note that $\alpha \vee \emptyset = \alpha$ for all $\alpha \in M(k)$. If $\mathcal{A}, \mathcal{B} \subset M(k)$ are generalized connection structures, define the generalized connection structure $\mathcal{A} \vee \mathcal{B}$ by

$$\mathcal{A} \vee \mathcal{B} = \{\alpha \vee \beta | \alpha \in \mathcal{A}, \beta \in \mathcal{B}\}.$$

Note that $\emptyset \in \mathcal{A} \vee \mathcal{B}$ if and only if $\emptyset \in \mathcal{A} \cap \mathcal{B}$. Consequently, if $\emptyset \in \mathcal{A} \vee \mathcal{B}$, then $\mathcal{A}, \mathcal{B} \subset \mathcal{A} \vee \mathcal{B}$.

Suppose that A is a nonempty subset of \mathbf{k} containing k_A elements. There is a natural embedding of $M(k_A)$ in $M(k)$ defined by relabelling

the matrices in $M(k_A)$ according to A . Specifically, map the matrix $(\alpha_{ij})_{i,j \in A} \in M(k_A)$ to the matrix $\hat{\alpha} \in M(k)$ defined by

$$\hat{\alpha}_{ij} = \begin{cases} \alpha_{ij} & \text{for } i, j \in A, \\ 0 & \text{otherwise.} \end{cases}$$

This embedding extends naturally to an embedding of $M_\bullet(k_A)$ in $M_\bullet(k)$ if we set

$$\hat{\alpha}_{i0} = \begin{cases} \alpha_{i0} & \text{for } i \in A, \\ 0 & \text{otherwise.} \end{cases}$$

Given disjoint nonempty subsets A, B of k , regard $M_\bullet(k_A), M_\bullet(k_B)$ as embedded in $M_\bullet(k)$. Given $\alpha \in M_\bullet(k_A), \beta \in M_\bullet(k_B)$, we define

$$\alpha \vee \beta = \hat{\alpha} \vee \hat{\beta} \in M_\bullet(k).$$

This extends to the join $\mathcal{A} \vee \mathcal{B}$ of generalized connection structures on disjoint sets of nontrivial nodes.

Let $\mathcal{N} = \{N_0, \dots, N_k\}$ and let A be a proper subset of \mathbf{k} . Recall that $A^\bullet = A \cup \{0\}$. Define $\mathcal{N}_A = \{N_j \mid j \in A^\bullet\}$ and $\mathbf{M}_A = \prod_{i \in A} M_i$. Denote points in \mathbf{M}_A by \mathbf{X}_A . Suppose $B = \mathbf{k} \setminus A$. We have $\mathcal{N}_A \cap \mathcal{N}_B = \{N_0\}$ and $\mathbf{M}_A \times \mathbf{M}_B \approx \mathbf{M}$. If $\mathbf{C}_A, \mathbf{C}_B$ are constraint structures on $\mathcal{N}_A, \mathcal{N}_B$ respectively, let $\mathbf{C} = \mathbf{C}_A \vee \mathbf{C}_B$ denote the induced constraint structure on \mathcal{N} .

More generally, given finite disjoint node sets $\mathcal{N}_A = \{N_j \mid j \in A^\bullet\}$, $\mathcal{N}_B = \{N_j \mid j \in B^\bullet\}$, we can identify A, B with complementary subsets of \mathbf{k} , where k is the total number of elements in $A \cup B$, and then follow the conventions described above..

Definition 5.1. (Notation and assumptions as above.) Given asynchronous networks $\mathfrak{N}_X = (\mathcal{N}_X, \mathcal{A}_X, \mathcal{F}_X, \mathcal{E}_X)$, $X \in \{A, B\}$, define the product $\mathfrak{N}_A \times \mathfrak{N}_B$ be the asynchronous network $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ with

- (1) node set $\mathcal{N} = \mathcal{N}_A \cup \mathcal{N}_B$ (we identify A, B with complementary subsets of \mathbf{k}),
- (2) constraint structure $\mathbf{C} = \mathbf{C}_A \vee \mathbf{C}_B$,
- (3) generalized connection structure $\mathcal{A} = \mathcal{A}_A \vee \mathcal{A}_B$,
- (4) \mathcal{A} -structure $\mathcal{F} = \mathcal{F}_A \times \mathcal{F}_B$, and
- (5) event map defined by $\mathcal{E}(\mathbf{X}_A, \mathbf{X}_B) = \mathcal{E}_A(\mathbf{X}_A) \vee \mathcal{E}_B(\mathbf{X}_B)$, for $(\mathbf{X}_A, \mathbf{X}_B) \in \mathbf{M}_A \times \mathbf{M}_B$.

Remark 5.2. If $\mathfrak{N}_A, \mathfrak{N}_B$ are proper (or amenable), then so is the product network.

Lemma 5.3. (Notation of definition 5.1.) *If the asynchronous network $\mathfrak{N} = \mathfrak{N}_A \times \mathfrak{N}_B$, then dynamics on the network phase space $\mathbf{M}_A \times \mathbf{M}_B$*

is given by

(11)

$$\mathbf{F}(\mathbf{X}_A, \mathbf{X}_B) = (\mathbf{f}^{\mathcal{E}_A(\mathbf{X}_A)}(\mathbf{X}_A), \mathbf{f}^{\mathcal{E}_B(\mathbf{X}_B)}(\mathbf{X}_B)), (\mathbf{X}_A, \mathbf{X}_B) \in \mathbf{M}_A \times \mathbf{M}_B.$$

Proof. Immediate from the definitions. \square

5.2. Indecomposability.

Definition 5.4. An asynchronous network $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ is *decomposable* if it can be written as a product of asynchronous networks. If the network is not decomposable, it is *indecomposable*.

Before giving our next definition, recall that the projection $\pi : M_\bullet(k) \rightarrow M(k); \alpha \mapsto \alpha^\flat$, is defined by omitting the zero column of $\alpha \in M_\bullet(k)$.

Definition 5.5. The *connection graph* of the asynchronous network $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ is the graph defined by the 0-1 matrix $\Gamma_{\mathfrak{N}} = \bigvee_{\alpha \in \mathcal{A}} \alpha^\flat$.

Lemma 5.6. *If an asynchronous network $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ is decomposable, then the connection graph $\Gamma_{\mathfrak{N}}$ of \mathfrak{N} has at least two connected components.*

Proof. If \mathfrak{N} is decomposable, we can find proper complementary subsets A, B of \mathbf{k} such that $\mathfrak{N} = \mathfrak{N}_A \times \mathfrak{N}_B$, where the node sets for $\mathfrak{N}_A, \mathfrak{N}_B$ are $\mathcal{N}_A, \mathcal{N}_B$ respectively. Since there are no connections between nodes in \mathcal{N}_A and \mathcal{N}_B , $\Gamma_{\mathfrak{N}}$ has at least two connected components. \square

Remarks 5.7. (1) Lemma 5.6 gives a simple necessary condition for decomposability which is clearly not sufficient. There are two obstructions to inferring decomposability from the disconnectedness of the connection graph. First, the event map encodes information about spatial dependence of node interactions that cannot be deduced from the connection graph. Second, the admissible vector fields may encode implicit dependencies that are incompatible with decomposability (see example 5.8 below).

(2) While \mathfrak{N} may be indecomposable, it may be equivalent to a decomposable network (for equivalence, see remarks 4.22(3)). Indeed, the dynamics of an asynchronous network only depends on the values of the α -admissible vector fields on the corresponding event sets E^α . In particular, there may be no functional dependence of $f_i^\alpha|E^\alpha$ on $\mathbf{x}_j \in M_j$, even though f_i^α does depend on $\mathbf{x}_j \in M_j$. In this case, if we fix $\mathbf{x}_j^\alpha \in E^\alpha$, we can replace f_i^α by the smooth vector field \bar{f}_i defined by taking $\mathbf{x}_j = \mathbf{x}_j^\alpha$ and define $\bar{\alpha}_{i\ell} = \alpha_{i\ell}$, $\ell \neq j$, $\bar{\alpha}_{ij} = 0$. After removing all such non-essential dependencies in this way (see also section 2.1 and note that constrained nodes present no new problems), we can then

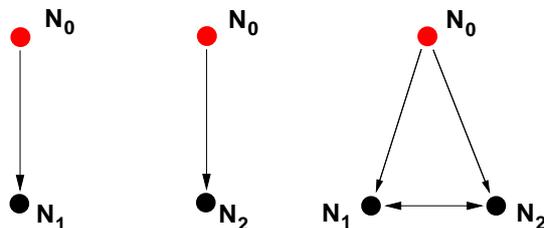


FIGURE 6. Nonempty connection structures for a two node asynchronous network.

ask if the modified network decomposable and so \mathfrak{N} is equivalent to a decomposable network.

Example 5.8. Let $k = 2$, $M_1 = M_2 = \mathbb{R}$ and suppose that, apart from the empty structure, there are three connection structures α, β, γ shown in figure 6.

Suppose the event map is given by

$$\mathcal{E}(x, y) = \begin{cases} \alpha, & \text{if } x < 0, y = 0 \\ \beta, & \text{if } x = 0, y > 0 \\ \gamma, & \text{if } x = -y \\ \emptyset, & \text{otherwise} \end{cases}$$

In this case, \mathcal{A} cannot be written as $\mathcal{A}_1 \vee \mathcal{A}_2$ since the only choice for \mathcal{A}_i is $\{\emptyset, \{N_0 \rightarrow N_i\}\}$, $i \in \mathbf{2}$. The existence of a connection $N_1 \leftrightarrow N_2$ violates the necessary condition given by lemma 5.6. If the connection γ encodes a nontrivial functional dependency between N_1 and N_2 , then the network cannot be decomposable or even equivalent to a decomposable network.

If there is no such nontrivial dependency, then we can remove the redundant link by replacing γ with $\alpha \vee \beta = N_1 \leftarrow N_0 \rightarrow N_2$ which means that the connection graph now has two disjoint connected components. However, there is no way to write $\mathcal{E}(x, y)$ as $\mathcal{E}_1(x) \vee \mathcal{E}_2(y)$ since the event sets involving $x \in M_1$ (respectively, $y \in M_2$) depend nontrivially on y (respectively, x). Thus, the asynchronous network cannot be decomposable or even equivalent to a decomposable network.

The decomposability of a network may also depend on the choice of admissible vector fields. In contrast to the event map above, the event

map

$$\tilde{\mathcal{E}}(x, y) = \begin{cases} \alpha, & \text{if } x = 0, y \neq 0 \\ \beta, & \text{if } y = 0, x \neq 0 \\ \alpha \vee \beta, & \text{if } x = y = 0 \\ \emptyset, & \text{otherwise} \end{cases}$$

factorises. Let $v, w > 0$. Define $\mathbf{f}^\emptyset(x, y) = (v, v)$, $\mathbf{f}^\alpha(x, y) = (0, w)$, $\mathbf{f}^\beta(x, y) = (v, 0)$, and $\mathbf{f}^{\alpha \vee \beta}(x, y) = (0, 0)$. For $v \neq w$ the network cannot be decomposable since the dynamics of N_2 depends implicitly on the state $x \in M_1$ of N_1 despite there being no explicit dependency. However, if $v = w$ the network is decomposable. \diamond

5.3. Sufficient conditions for decomposability. Suppose that \mathfrak{N} is an asynchronous network with k nodes and that C is a proper connected component of the connection graph $\Gamma = \Gamma_{\mathfrak{N}}$. The component C can be identified with a nonempty subset of \mathbf{k} that corresponds to the labels of the nodes in the component. Let $\bar{C} = \mathbf{k} \setminus C$. Since C is a connected component of $\Gamma_{\mathfrak{N}}$, we can write each $\alpha \in \mathcal{A}$ uniquely as $\alpha = \alpha_C \vee \alpha_{\bar{C}}$, where $\alpha_C, \alpha_{\bar{C}}$ are connection structures on \mathcal{N}_C and $\mathcal{N}_{\bar{C}}$ respectively. Set $\mathcal{A}_C = \{\alpha_C \mid \alpha \in \mathcal{A}\}$. We have a well defined projection $\pi_C : \mathcal{A} \rightarrow \mathcal{A}_C$ defined by $\pi_C(\alpha) = \alpha_C$.

Define the event map $\mathcal{E}_C : \mathbf{M}_C \times \mathbf{M}_{\bar{C}} \rightarrow \mathcal{A}_C$ by

$$\mathcal{E}_C(\mathbf{X}_C, \mathbf{X}_{\bar{C}}) = \pi_C(\mathcal{E}(\mathbf{X}_C, \mathbf{X}_{\bar{C}})).$$

Definition 5.9. Let $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ be an asynchronous network. The asynchronous network \mathfrak{N} is *structurally decomposable* if for any connected component C of the connection graph $\Gamma_{\mathfrak{N}}$, the map \mathcal{E}_C is independent of $\mathbf{X}_{\bar{C}} \in \mathbf{M}_{\bar{C}}$ (that is, $\mathcal{E}_C(\mathbf{X}_C, \mathbf{X}_{\bar{C}}) = \mathcal{E}_1(\mathbf{X}_C)$ where $\mathcal{E}_1 : \mathbf{M}_C \rightarrow \mathcal{A}_C$).

Remark 5.10. Structural decomposability implies the existence of certain links. For example, if a node is constrained and there is a condition that the node will restart when the state of another node attains a certain value, then there has to be a link between the nodes. In other words, connection matrices of networks that are structurally decomposable not only encode dynamical interaction (the vector field determining the dynamics depends on the state of the other node) but also “structural” interaction; that is, the event map \mathcal{E} determines a change in connection structure if a certain condition is met (see example 5.8).

We continue to assume that C is a proper connected component of the connection graph of an asynchronous network. Admissibility ensures that each element $\mathbf{f}^\alpha \in \mathcal{F}$ can be written as $\mathbf{f}^\alpha = \mathbf{f}_C^\alpha \times \mathbf{f}_{\bar{C}}^\alpha$ where

$\mathbf{f}_C^\alpha : \mathbf{M}_C \rightarrow T\mathbf{M}_C$ and $\mathbf{f}_{\overline{C}}^\alpha : \mathbf{M}_{\overline{C}} \rightarrow T\mathbf{M}_{\overline{C}}$. For an asynchronous network to be decomposable, this decomposition has to be compatible with the projections $\pi_C : \mathcal{A} \rightarrow \mathcal{A}_C$, $\pi_{\overline{C}} : \mathcal{A} \rightarrow \mathcal{A}_{\overline{C}}$. That is, if a connection in the set of nodes that are in C is added or deleted, the dynamics of the remaining nodes in \overline{C} are not affected. This condition translates into relationships between the network vector fields \mathbf{f}^α .

Definition 5.11. Let $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ be an asynchronous network. The asynchronous network \mathfrak{N} is *dynamically decomposable* if for any connected component C of the connection graph $\Gamma_{\mathfrak{N}}$ and any \mathbf{f}^α , $\alpha \in \mathcal{A}$, the vector field component in $T\mathbf{M}_C$ is independent of $\mathcal{A}_{\overline{C}}$. That is, we have the decomposition

$$\mathbf{f}^\alpha = \mathbf{f}_C^{\pi_C(\alpha)} \times \mathbf{f}_{\overline{C}}^{\pi_{\overline{C}}(\alpha)}, \quad \alpha \in \mathcal{A}$$

of the network vector fields on $T\mathbf{M}$.

Lemma 5.12. (*Notation as above.*) *Input consistent asynchronous networks are dynamically decomposable. Moreover, any asynchronous network with additive input structure is dynamically decomposable.*

Proof. Given $i \in \mathbf{k}$, $\alpha \in \mathcal{A}$, let $J(i, \alpha)$ be the associated dependency set for node N_i . If $\alpha, \beta \in \mathcal{A}$ and $J(i, \alpha) = J(i, \beta)$, then $f_i^\alpha = f_i^\beta$ by input consistency. If $i \in C$, where C is a connected component of the network graph $\Gamma_{\mathfrak{N}}$, then $J(i, \alpha) \cap \mathbf{k} \subset C$ for all $\alpha \in \mathcal{A}$. Hence $J(i, \alpha) = J(i, \alpha_C \vee \alpha_{\overline{C}})$ is independent of $\alpha_{\overline{C}}$. Input consistency implies that $f_i^{\alpha_C \vee \beta} = f_i^{\alpha_C \vee \gamma}$ for all $\beta, \gamma \in \mathcal{A}_{\overline{C}}$ which yields dynamical decomposability.

Any asynchronous network with additive input structure is input consistent and is therefore dynamically decomposable. \square

We can now state the main result of this section.

Theorem 5.13. *Let $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ be a structurally and dynamically decomposable asynchronous network with connection graph Γ . If Γ has connected components C_1, \dots, C_q then there exist indecomposable asynchronous networks $\mathfrak{N}_1, \dots, \mathfrak{N}_q$ such that*

$$\mathfrak{N} = \mathfrak{N}_1 \times \dots \times \mathfrak{N}_q.$$

Proof. We prove in case $q = 2$ (the general case is similar). Suppose Γ has connected components $C_1 = A$ and $C_2 = B$. Let $\mathcal{A}_A = \{\pi_A(\alpha) \mid \alpha \in \mathcal{A}\}$ and $\mathcal{A}_B = \{\pi_B(\alpha) \mid \alpha \in \mathcal{A}\}$. Dynamical decomposability ensures that $\mathbf{f}^\alpha = \mathbf{f}^{\pi_A(\alpha)} \times \mathbf{f}^{\pi_B(\alpha)}$. Moreover, constraint structures are defined for individual nodes and so they factorise naturally.

Define event maps $\mathcal{E}_A, \mathcal{E}_B$ by

$$\mathcal{E}_A(\mathbf{X}_A) = \pi_A(\mathcal{E}(\mathbf{X}_A, \mathbf{X}_B)), \quad \mathcal{E}_B(\mathbf{X}_B) = \pi_B(\mathcal{E}(\mathbf{X}_A, \mathbf{X}_B)).$$

These maps are well defined since the network is structurally decomposable. Hence we have constructed asynchronous networks $\mathfrak{N}_A, \mathfrak{N}_B$ such that $\mathfrak{N} = \mathfrak{N}_A \times \mathfrak{N}_B$. \square

Our concluding result on decomposability is an immediate consequence of lemma 5.6 and theorem 5.13.

Corollary 5.14. *A structurally and dynamically decomposable asynchronous network \mathfrak{N} is decomposable if and only if its connection graph has more than one nontrivial connected component.*

5.4. Amalgamating asynchronous networks. Products are defined for asynchronous networks with disjoint sets of nodes. Asynchronous networks which share a common node set can be *amalgamated* or *merged* to form a new asynchronous network. In terms of binary operations from graph theory, amalgamations roughly correspond to unions of graphs. Amalgamations will become important when we consider the structure of functional asynchronous networks in sections 7–9. While products of amenable (proper) asynchronous networks are automatically amenable (proper), this is not necessarily true for the amalgamation of asynchronous networks. We partially address the question of amenability here and return to the issue in section 9.

Definition 5.15. A node N_i , $i \neq 0$, is *linked* if either there exists $j \in \mathbf{k}^\bullet$, such that $N_j \rightarrow N_i$ or there exists $j \in \mathbf{k}$ such that $N_i \rightarrow N_j$.

Remark 5.16. The constraining node is never a linked node. However, if there is a connection $N_0 \rightarrow N_i$, $i \in \mathbf{k}$, then N_i will be linked. A linked node is either constrained or connected to another node. Every node in a connection structure will be linked, except for the constraining node.

If α is a connection structure, let $v(\alpha)$ denote the set of linked nodes of α . More generally, define

$$V(\mathcal{A}) = \bigcup_{\alpha \in \mathcal{A}} v(\alpha) \subset \mathbf{k}.$$

for a generalized connection structure \mathcal{A} .

Definition 5.17. An asynchronous network $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ with $\emptyset \in \mathcal{A}$ is *weakly input consistent* if $i \notin v(\alpha)$ implies that $f_i^\alpha = f_i^\emptyset$.

This property is crucial in what follows. For the remainder of this section we assume that all asynchronous networks include the empty connection structure $\emptyset \in \mathcal{A}$ and are weakly input consistent.

5.4.1. *Amalgamating independent asynchronous networks.* We start by defining the amalgamation of independent asynchronous networks. In this case, the operation is similar to that of product. Recall that for $A \subset \mathbf{k}$, we defined $\mathbf{M}_A = \prod_{i \in A} M_i$.

Definition 5.18. An asynchronous network $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ is of *elementary type* if the event map \mathcal{E} depends only on variables \mathbf{x}_i , $i \in V(\mathcal{A})$.

Remarks 5.19. (1) Definition 5.18 implies that for all $\alpha \in \mathcal{A}$, there exists $\hat{E}^\alpha \subset \mathbf{M}_{V(\mathcal{A})}$ such that $E^\alpha = \hat{E}^\alpha \times \mathbf{M}_{\mathbf{k} \setminus V(\mathcal{A})}$. Typically, it is too restrictive to require that if $\alpha \in \mathcal{A}$, then $E^\alpha = \tilde{E}^\alpha \times \mathbf{M}_{\mathbf{k} \setminus v(\alpha)}$.

(2) The concept of elementary type is related to the concepts of structural and dynamical decomposability defined in the previous section. In particular, if \mathfrak{N} is of elementary type, then $\mathfrak{N} = \mathfrak{N}_v \times \prod_{i \notin V(\mathcal{A})} \mathfrak{S}_i$ where \mathfrak{N}_v has nodes $\{N_0\} \cup \{N_i \mid i \in V(\mathcal{A})\}$ and each \mathfrak{S}_i gives uncoupled dynamics on N_i .

(3) We can generalise the definition of elementary type by taking $\tilde{V}(\mathcal{A})$ to be the minimal subset of \mathbf{k} such that $\tilde{V}(\mathcal{A}) \supset V(\mathcal{A})$ and definition 5.18 holds with $\tilde{V}(\mathcal{A})$ replacing $V(\mathcal{A})$. The preceding remark continues to hold with this change.

Let $q \in \mathbb{N}$, $q > 1$. Suppose that $\Lambda = \{\mathfrak{N}^a = (\mathcal{N}, \mathcal{A}^a, \mathcal{F}^a, \mathcal{E}^a) \mid a \in \mathbf{q}\}$ is a family of asynchronous networks of elementary type with common node set \mathcal{N} and network phase space \mathbf{M} . We write elements of \mathcal{F}^a as $\mathbf{f}^{a,\alpha} = (f_1^{a,\alpha}, \dots, f_k^{a,\alpha})$, $\alpha \in \mathcal{A}^a$. We assume that $\mathbf{f}^{a,\emptyset}$ is independent of a and set $\mathbf{f}^{a,\emptyset} = \mathbf{f}^\emptyset$, $a \in \mathbf{q}$.

Definition 5.20. (Notation and assumptions as above.) The family Λ is *independent* if for all $a, b \in \mathbf{q}$, $a \neq b$, we have

$$V(\mathcal{A}^a) \cap V(\mathcal{A}^b) = \emptyset.$$

If Λ is independent, we define $\mathcal{A} = \bigvee_{a \in \mathbf{q}} \mathcal{A}^a$ and $\mathcal{E} : \mathbf{M} \rightarrow \mathcal{A}$ by $\mathcal{E}(\mathbf{X}) = \bigvee_{a \in \mathbf{q}} \mathcal{E}^a(\mathbf{X})$, $\mathbf{X} \in \mathbf{M}$. Obviously, $\mathcal{E}(\mathbf{M}) = \mathcal{A}$ (by independence). Each $\alpha \in \mathcal{A}$ may be written *uniquely* as $\alpha = \bigvee_{a \in \mathbf{q}} \alpha_a$, where $\alpha_a \in \mathcal{A}^a$. We define the \mathcal{A} -admissible vector field \mathbf{f}^α on \mathbf{M} by

$$f_i^\alpha = \begin{cases} f_i^{a,\alpha_a}, & \text{if } i \in v(\alpha_a) \\ f_i^\emptyset, & \text{if } i \notin v(\alpha). \end{cases}$$

Since the family is independent, each \mathfrak{N}^a is of elementary type, and since we assume weak input consistency, \mathbf{f}^α is uniquely defined. Set $\mathcal{F} = \{\mathbf{f}^\alpha \mid \alpha \in \mathcal{A}\}$.

Lemma 5.21. (Notation and assumptions as above.) *If the family $\Lambda = \{\mathfrak{N}^a \mid a \in \mathbf{q}\}$ is independent, then we define the amalgamation $\mathfrak{N} = \bigsqcup_{a \in \mathbf{q}} \mathfrak{N}^a$ to be the asynchronous network $(\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ where \mathcal{A}, \mathcal{F} are as defined above and $\mathcal{E}(\mathbf{X}) = \bigvee_{a \in \mathbf{q}} \mathcal{E}^a(\mathbf{X})$, $\mathbf{X} \in \mathbf{M}$.*

The network \mathfrak{N} is of elementary type and will be amenable (respectively, proper) if each of the networks \mathfrak{N}^a is amenable (respectively, proper).

Proof. Let $a \in \mathbf{q}$. It follows by remarks 5.19(2) that we may write $\mathfrak{N}^a = \mathfrak{N}_v^a \times \mathfrak{S}^a$, where $\mathfrak{S}^a = \prod_{i \notin V(\mathcal{A}^a)} \mathfrak{S}_i$. If we set $V = \cup_{a \in \mathbf{q}} V(\mathcal{A}^a)$, then $\mathfrak{N} = \prod_{a \in \mathbf{q}} \mathfrak{N}_v^a \times \prod_{i \notin V} \mathfrak{S}_i$. Hence \mathfrak{N} is of elementary type with $V(\mathcal{A}) = \cup_{a \in \mathbf{q}} V(\mathcal{A}^a)$. It follows from the representation of \mathfrak{N} as a product that if each \mathfrak{N}^a is amenable (respectively, proper), then the same is true for \mathfrak{N} . \square

5.4.2. *Amalgamating locally independent asynchronous networks.* The assumption that asynchronous networks are of elementary type and independent is strong and allows us to write the amalgamation as a product. For the remainder of the section we assume $\Lambda = \{\mathfrak{N}^a \mid a \in \mathbf{q}\}$ is a family of asynchronous networks but do not require independence or that the component networks \mathfrak{N}^a are of elementary type. We discuss how amalgamations can be defined for this more general class of asynchronous networks. Note that we continue to assume that $\mathbf{f}^{a, \emptyset} = \mathbf{f}^\emptyset$, all $a \in \mathbf{q}$, and that weak input consistency holds.

For $a \in \mathbf{q}$, $\alpha \in \mathcal{A}^a$, and $i \in V(\mathcal{A}^a)$, define

$$\begin{aligned} E^{a, \alpha} &= \{\mathbf{X} \in \mathbf{M} \mid \mathcal{E}^a(\mathbf{X}) = \alpha\} \\ E_i^a &= \{\mathbf{X} \in \mathbf{M} \mid i \in v(\mathcal{E}^a(\mathbf{X}))\}. \end{aligned}$$

Definition 5.22. (Notation and assumptions as above.) The family Λ is *compatible* if for $a, b \in \mathbf{q}$, $a \neq b$,

- (1) $\overline{E_i^a} \cap \overline{E_i^b} = \emptyset$, if $i \in V(\mathcal{A}^a) \cap V(\mathcal{A}^b)$. (Local independence.)
- (2) If $\alpha \in \mathcal{A}^a \cap \mathcal{A}^b$, then $\mathbf{f}^{a, \alpha} = \mathbf{f}^{b, \alpha}$. (Consistency.)

Remarks 5.23. (1) Since we always assume consistency holds for $\emptyset \in \bigcap_{a \in \mathbf{q}} \mathcal{A}^a$, it is easy to show that condition (1) of definition 5.22 implies that we can choose the $\mathbf{f}^{a, \alpha}$, without changing $\mathbf{f}^{a, \alpha} | \overline{E^{a, \alpha}}$, so that the family is compatible. For our purposes, the essential requirement is that consistency holds for \emptyset .

(2) If Λ is independent, then Λ is a compatible family.

Lemma 5.24. *Suppose that Λ is compatible. If we define $\mathcal{E} : \mathbf{M} \rightarrow M_\bullet(k)$, by $\mathcal{E}(\mathbf{X}) = \bigvee_{a \in \mathbf{q}} \mathcal{E}^a(\mathbf{X})$, then for each $\mathbf{X} \in \mathbf{M}$, either $\mathcal{E}(\mathbf{X}) = \emptyset$ or there exist unique $a_1 < \dots < a_s \in \mathbf{q}$ such that*

- (1) $\mathcal{E}^a(\mathbf{X}) = \emptyset$, if and only if $a \notin \{a_1, \dots, a_s\}$.
 (2) $v(\mathcal{E}^{a_i}(\mathbf{X})) \cap v(\mathcal{E}^{a_j}(\mathbf{X})) = \emptyset$, for all $i, j \in \mathbf{s}$, $i \neq j$.

Proof. Definition 5.22(1) implies that for given $\mathbf{X} \in \mathbf{M}$ and $i \in \mathbf{k}$, there can be at most one $a \in \mathbf{q}$ for which $i \in v(\mathcal{E}^a(\mathbf{X}))$. The result follows. \square

Let $\mathbf{X} \in \mathbf{M}$ and $\mathcal{E}(\mathbf{X}) \neq \emptyset$. By lemma 5.24, there are unique $a_1 < \dots < a_s \in \mathbf{q}$ such that $\mathcal{E}(\mathbf{X}) = \bigvee_{i \in \mathbf{s}} \mathcal{E}^{a_i}(\mathbf{X})$. Set $\alpha_{a_i} = \mathcal{E}^{a_i}(\mathbf{X})$, $i \in \mathbf{s}$, so that $\mathcal{E}(\mathbf{X}) = \alpha_{a_1} \vee \dots \vee \alpha_{a_s}$. We extend the concept of generalized connection structure and let define \mathcal{A} to consist of \emptyset together with the set of all $\alpha_{a_1} \vee \dots \vee \alpha_{a_s}$ obtained as values of \mathcal{E} . Note that even if $\alpha = \bigvee_{i \in \mathbf{s}} \alpha_{a_i}$, $\beta = \bigvee_{j \in \mathbf{t}} \beta_{b_j}$ define the same connection structure on \mathcal{N} , we regard α, β as *distinct* elements of \mathcal{A} unless $s = t$ and $\alpha_{a_i} = \beta_{b_i}$ for all $i \in \mathbf{s}$.

For each $\alpha = \alpha_{a_1} \vee \dots \vee \alpha_{a_s} \in \mathcal{A}$, define \mathbf{f}^α by

$$f_i^\alpha(\mathbf{X}) = \begin{cases} f_i^{\alpha_\ell, \alpha_\ell}(\mathbf{X}), & \text{if } i \in v(\alpha_{a_\ell}), \\ f_i^\emptyset(\mathbf{X}), & \text{if } i \notin v(\alpha_{a_1} \vee \dots \vee \alpha_{a_s}). \end{cases}$$

This defines $\mathbf{f}^\alpha = \mathbf{f}^{\alpha_{a_1} \vee \dots \vee \alpha_{a_s}}$ as a smooth vector field on \mathbf{M} . In particular, \mathbf{f}^α is defined as a smooth vector field on the event set E^α (with respect to the joint event map \mathcal{E}). This yields our required \mathcal{A} -admissible family $\mathcal{F} = \{\mathbf{f}^\alpha \mid \alpha \in \mathcal{A}\}$.

Remark 5.25. If $\alpha, \beta \in \mathcal{A}$ define the same connection structure on \mathcal{N} , it will not generally be possible to choose a smooth network vector field \mathbf{f} which coincides with \mathbf{f}^α on E^α and \mathbf{f}^β on E^β . Rather than use the extended concept of generalized connection structure described above, we could instead introduce different edge types for elements of \mathcal{A}^a , $a \in \mathbf{q}$. Whatever conventions are used yield the same dynamics since the admissible vector fields are always uniquely defined on their associated event sets.

Definition 5.26. Let $\Lambda = \{\mathfrak{N}^a \mid a \in \mathbf{q}\}$ be a compatible family of asynchronous networks and $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ be defined as above. Then the asynchronous network

$$\mathfrak{N} = \bigsqcup_{a \in \mathbf{q}} \mathfrak{N}^a$$

is the *amalgamation* of the family Λ .

In general, \mathfrak{N} will not be amenable or proper even if all the constituent networks \mathfrak{N}^a are amenable or proper. The difficulty lies with the structure of \mathcal{E} . Even though for all $\mathbf{X} \in \mathbf{M}$, $N_i \rightarrow N_j \in \mathcal{E}(\mathbf{X})$ if

and only if there exists a unique $a \in \mathbf{q}$ such that $N_i \rightarrow N_j \in \mathcal{E}^a(\mathbf{X})$, the dependence of $E_i^a \cap E_j^a$ on \mathbf{X} cannot be given purely in terms of \mathfrak{N}^a without further conditions on the event maps. We return to this point in section 8.

6. A SIMPLE EXAMPLE: TRANSPORT WITH A PASSING LOOP

In this section we describe and develop a simple example: two trains on single track railway line with a passing loop. The asynchronous network describing the dynamics of the two trains will not be associated to a classical synchronous network. As we develop the example, we include elaborations that illustrate important features of more general and realistic asynchronous networks – local time, dynamics, initialization, and network function. Although our example is that of a transport network, it is not hard to modify the setup to model a production network.

6.1. The basic example. Consider two trains $\mathfrak{T}_1, \mathfrak{T}_2$ travelling in opposite directions along a single track railway line of infinite length modelled by \mathbb{R} ; see figure 7.



FIGURE 7. Two trains on a single track railway line with a passing loop.

To pass each other, the trains must enter a passing loop, located at $0 \in \mathbb{R}$. We model as an asynchronous network with 1-dimensional node dynamics. Assume nodes N_1, N_2 , both with 1-dimensional phase space \mathbb{R} . The position of train \mathfrak{T}_i is given by $x_i \in \mathbb{R}$, $i \in \mathbf{2}$. We assume that, outside of the passing loop, the velocity of the trains is given by smooth vector fields $V_1, V_2 : \mathbb{R} \rightarrow \mathbb{R}$ such that

$$V_1(x) > 0 > V_2(x), \quad x \in \mathbb{R}.$$

That is, \mathfrak{T}_1 is moving to the right and \mathfrak{T}_2 to the left. At this point we will be vague about the precise initialization of the trains except to say that at $t = 0$, we require $x_1(0) < 0 < x_2(0)$. The dynamics of the asynchronous network can now be described as follows. When the trains are approaching the passing loop, they move at nonzero speed and are decoupled (we do not assume the existence of a centralized control or communication between the drivers). Once one train enters

the passing loop, it stops and waits for the second train to enter. Entering the passing loop can be seen as a connection event since now the position of the one train becomes relevant for the dynamics of the other train. Once both trains are in the passing loop, they may proceed. Formalizing, we define the generalized connection structure \mathcal{A} by

$$\mathcal{A} = \{\alpha = \{N_0 \rightarrow N_1\}, \beta = \{N_0 \rightarrow N_2\}, \gamma = \{N_1 \leftrightarrow N_2\}, \emptyset\},$$

where N_0 denote the constraining node and \emptyset is the empty connection structure. We define the event map $\mathcal{E} : \mathbb{R}^2 \rightarrow \mathcal{A}$ by

$$\mathcal{E}(x_1, x_2) = \begin{cases} \alpha, & \text{if } x_1 = 0, x_2 > 0 \\ \beta, & \text{if } x_1 < 0, x_2 = 0 \\ \gamma, & \text{if } x_1 = 0, x_2 = 0 \\ \emptyset, & \text{otherwise.} \end{cases}$$

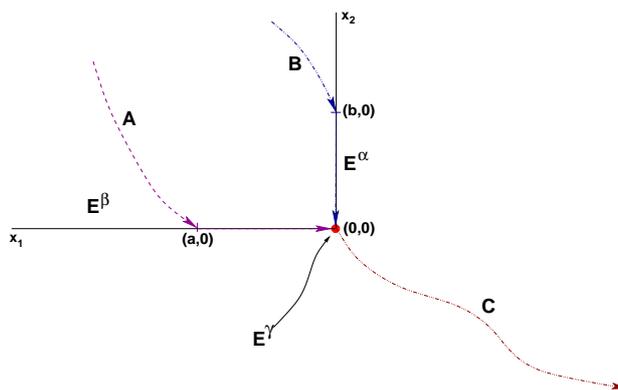


FIGURE 8. Dynamics on a one track line with passing loop.

We can allow for initialization of the trains at general points of \mathbb{R}^2 by adding a collision event along $x_1 = x_2$, $x_1 \neq 0$ – see examples 4.30(1) – we keep matters as simple as possible here by focusing on initializations in the region $\mathbb{R}_\circ^2 = \{(x_1, x_2) \mid x_1 x_2 \leq 0\}$. The event dependent vector fields \mathbf{V}^C , $C \in \mathcal{A}$, are given by

$$\mathbf{V}^\emptyset = \mathbf{V}^\gamma = (V_1, V_2), \quad \mathbf{V}^\alpha = (0, V_2), \quad \mathbf{V}^\beta = (V_1, 0).$$

We show dynamics in figure 8 and note that the associated semiflow $\Phi : \mathbb{R}^2 \times \mathbb{R}_+ \rightarrow \mathbb{R}^2$ will not be continuous on $\overline{E^\alpha \cup E^\beta}$ (see examples 4.30(1)). However, $\Phi : \mathbb{R}_\circ^2 \times \mathbb{R}_+ \rightarrow \mathbb{R}_\circ^2$ is continuous. Referring to figure 8, trajectory **A** corresponds to train \mathfrak{T}_2 reaching the passing loop first and restarting only when \mathfrak{T}_1 reaches the passing loop.

Train \mathfrak{T}_1 reaches the passing loop first for trajectory \mathbf{B} . Note that whichever train reaches the passing loop first, the ‘exit trajectory’ \mathbf{C} is always the same and so the structure imposes an eventual reduction to 1-dimensional dynamics.

In the above, we defined $\mathcal{E}(0,0) = \gamma$. We might just as well have not used γ and defined $\mathcal{E}(0,0) = \emptyset$ (the network is then amenable). The origin is singular in this example – in part, this is because it is possible to initialize so that neither train stops in the passing loop¹⁰. In other situations, it may be appropriate to use the connection structure $\alpha \vee \beta$ at $(0,0)$ (for example if train \mathfrak{T}_i is required to stop at least $S_i > 0$ seconds in the passing loop, $i \in \mathbf{2}$).

6.1.1. *Digression – embedding in a Filippov system.* In the previous example, the network vector field $\mathbf{F}(\mathbf{X}) = \mathbf{V}^{\mathcal{E}(\mathbf{X})}(\mathbf{X})$ is not continuous on \mathbf{M} . It is natural to ask whether there is a connection with Filippov systems [44] – a class originally inspired by engineering applications which allows for discontinuous vector fields. Roughly speaking, a Filippov system is defined by a collection of sufficiently smooth vector fields defined on open subsets of the phase space. It is assumed that the boundaries between these open sets are hypersurfaces. If the vector fields on the open sets close to these surfaces both point towards the surface, then dynamics are determined by the unique convex combination that leaves the hypersurface invariant (see example 4.23) and thus the hypersurface of discontinuity becomes a “sliding surface.”

In an asynchronous network, dynamics on event sets is given explicitly rather than by the conventions used in Filippov systems. However, asynchronous networks can sometimes be locally represented by a Filippov system. We give details below for the passing loop example (we refer to [23] for more details and greater generality). These connections open the possibility of applying methods and results from, for example, the bifurcation theory of Filippov systems to asynchronous networks.

Recall that the semiflow defined by the vector field \mathbf{V} is not continuous on $\overline{E^\alpha \cup E^\beta} \subset \mathbb{R}^2$ (as opposed to $\mathbb{R}_>^2$). However, we can obtain dynamics on $\mathbb{R}_>^2$ as the restriction of Filippov dynamics on \mathbb{R}^2 and the Filippov dynamics has a continuous semiflow on \mathbb{R}^2 . In order to represent dynamics on $\mathbb{R}_>^2$ as the restriction of dynamics of a Filippov system on \mathbb{R}^2 , we define lines $S^\alpha \supset E^\alpha, S^\beta \supset E^\beta$ by $S^\alpha = \{(x_1, x_2) \in \mathbb{R}^2 \mid x_1 = 0\}$, $S^\beta = \{(x_1, x_2) \in \mathbb{R}^2 \mid x_2 = 0\}$. We define a vector field V on $\mathbb{R}^2 \setminus S^\alpha \cup S^\beta$

¹⁰A situation to be avoided in practice.

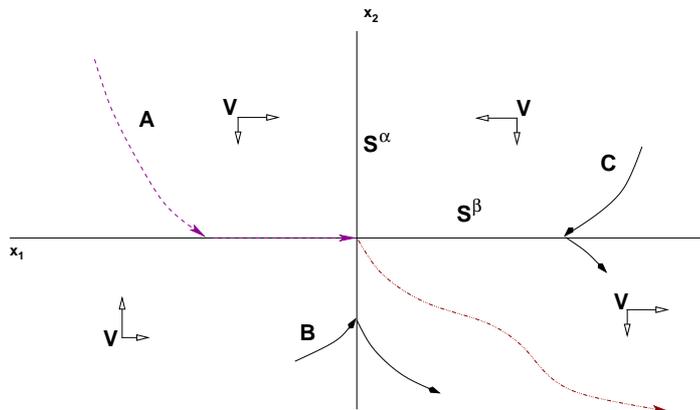


FIGURE 9. Dynamics for the associated Filippov system. Trajectory **A** is unchanged; trajectories **B** and **C** correspond to one train reversing after the other train enters the passing loop.

by

$$V(x_1, x_2) = \begin{cases} (V_1(x_1), V_2(x_2)), & \text{if } x_1 x_2 \leq 0, \\ (V_1(x_1), -V_2(x_2)), & \text{if } x_1 < 0, x_2 < 0, \\ (-V_1(x_1), V_2(x_2)), & \text{if } x_1 > 0, x_2 > 0, \end{cases}$$

see figure 9.

For any point $x \in S^\alpha \cup S^\beta \setminus \{(0, 0)\}$ the discontinuous vector field V defines a Filippov system. For $x \in E^\alpha \cup E^\beta$ the dynamics of the local Filippov system are given by a sliding vector field given by the unique convex combination of vector fields on opposite sides of S^α, S^β that is tangent to the sliding line [44, 22] (see also example 4.23) which evaluate to $\mathbf{V}^\alpha, \mathbf{V}^\beta$. Thus, E^α, E^β become sliding surfaces of the local Filippov systems. If we define $V(0, 0) = (V_1(0), V_2(0))$, then the semiflow of induced by V as a Filippov system is continuous on \mathbb{R}^2 and restricts to the required semiflow on \mathbb{R}_0^2 .

Remark 6.1. Of course, the resulting semiflow on $\mathbb{R}^2 \setminus \mathbb{R}_0^2$ does not have an interpretation in terms of trains on a line with a passing loop. However, the Filippov model is directed towards sliding and control applications, whereas what we are discussing is appropriate for networks where nodes may be turned on and off as thresholds are reached. Unlike for characteristic Filippov systems, the vector field for the original model is uniquely defined at every point of the phase space and the underlying functional structure consists of smooth vector fields defined at every point of the phase space.

6.2. Local times. In the basic model, at most one of the trains *stops* in the passing loop. A more realistic model is to suppose that the loop is a station and that both trains have to spend a specified amount of time in the station. For example, we might require that trains have to spend $S \geq 0$ seconds when they are both stopped in the station. In this case, we can suppose that when the trains have stopped, drivers synchronize their watches and wait for S seconds to pass before restarting. This type of network is an example of a *locally non-autonomous asynchronous network*. More formally, we allow for a local time $\tau \in \mathbb{R}_+$ given by the drivers synchronized watches (or the station clock). We define dynamics on $\mathbb{R}^2 \times \mathbb{R}_+$.

Initialize by $(x_1(0), x_2(0), 0)$, where $x_1(0) < 0 < x_2(0)$, and assume given vector fields V_1, V_2 as in section 6.1 (note that our setup gives well-defined dynamics for *all* $(x_1(0), x_2(0)) \in \mathbb{R}^2$). We define the connection structures α, β, γ by

$$\begin{aligned}\alpha &= \{N_0 \rightarrow N_1\} \\ \beta &= \{N_0 \rightarrow N_2\} \\ \gamma &= \{N_0 \rightarrow N_1 \leftrightarrow N_2 \leftarrow N_0\}\end{aligned}$$

and take the generalized connection structure $\mathcal{A} = \{\alpha, \beta, \gamma, \emptyset\}$.

The event map $\mathcal{E} : \mathbb{R}^2 \times \mathbb{R}_+ \rightarrow \mathcal{A}$ will depend on the local time τ . We define

$$\mathcal{E}((x_1, x_2), \tau) = \begin{cases} \alpha, & \text{if } x_1 = 0, x_2 > 0 \\ \beta, & \text{if } x_1 < 0, x_2 = 0 \\ \gamma, & \text{if } x_1 = x_2 = 0, \tau < S \\ \emptyset, & \text{otherwise} \end{cases}$$

Choose $\omega > 0$ with $\omega \approx 1$. We define an \mathcal{A} -structure \mathcal{F} consisting of τ -dependent vector fields by

$$\begin{aligned}\mathbf{V}^\alpha(x_1, x_2, \tau) &= ((0, V_2(x_2)), 0) \\ \mathbf{V}^\beta(x_1, x_2, \tau) &= ((V_1(x_1), 0), 0) \\ \mathbf{V}^\gamma(x_1, x_2, \tau) &= ((0, 0), \omega) \\ \mathbf{V}^\emptyset(x_1, x_2, \tau) &= ((V_1(x_1), V_2(x_2)), 0)\end{aligned}$$

Network dynamics on $\mathbb{R}^2 \times \mathbb{R}_+$ is then given by the piecewise smooth vector field

$$\mathbf{F}((x_1, x_2), \tau) = \mathbf{V}^{\mathcal{E}((x_1, x_2), \tau)}((x_1, x_2), \tau).$$

Remarks 6.2. (1) The local time component of \mathbf{F} is zero unless both trains are in the passing loop. The local time τ then satisfies the

equation $\tau' = \omega$, with initialization $\tau = 0$. The clock stops when $\tau(t) = S$. Of course, from the modelling point of view we might just as well assume $\omega = 1$ and vary S as necessary.

(2) An observer can always assume a global time. However, the train drivers have no connection – and cannot synchronize watches – until they are both in the passing loop.

(3) As set up, if $x_1(0) > x_2(0) > 0$, then \mathfrak{T}_2 does not stop when it goes through the passing loop (\mathfrak{T}_1 does not go through the passing loop). It is not hard to modify the model to allow \mathfrak{T}_2 to stop for a specified time in the passing loop. Similarly for \mathfrak{T}_1 if $x_2(0) < x_1(0) < 0$. It is also straightforward to modify the model to require, in addition to both trains being stopped together in the passing loop for time S , that \mathfrak{T}_1 must stop for time $S_1 \geq S$ and \mathfrak{T}_2 for time $S_2 \geq S$.

(4) The event sets $\{E^\alpha \mid \alpha \in \mathcal{A}\}$ for this model are semialgebraic subsets of $\mathbb{R}^2 \times \mathbb{R}_+$ and satisfy the amenability condition of definition 4.28 (provided $S > 0$). Their more complex structure reflects the increased logical complexity of the problem. We refer to [42] for the general formalism for locally nonautonomous asynchronous networks.

6.3. Adding some dynamics. In the passing loop example, the trains only “interact” when they are both in the passing loop and the effect on dynamics is minimal. We may vary the example to give a non-trivial dynamic interaction when the trains are in the passing loop. To this end, we additionally require that

- (1) The drivers are both running a non-linear oscillator of approximately the same frequency (initial conditions randomly chosen at the start of the trip).
- (2) When both trains are in the station, the oscillators are cross-coupled so that they eventually approximately synchronize.
- (3) The trains can leave the station when the oscillators have been synchronized to within ε for at least $S > 0$ seconds. We assume $1 \gg \varepsilon > 0$.

We omit the formalisation – which is broadly similar to that of the previous section – and remark that this type of restart event is, for example, applicable to distributed production networks where there are chemical reactions or mixing which take time to complete. We could also couple the dynamics to the train speed – for example by making the train velocity part of a skew product system weakly forced by some internal dynamics with forcing turned off in the passing loop. There is the possibility that the condition of approximation is never met (a deadlock) or takes a very long time to occur (the network dynamics

stalls). Both situations can occur if, for example, the coupling is very weak.

6.4. Initialization, termination and function. We now work towards a more realistic model of the passing loop example that can be embedded in a larger transport network. Assume that the rail line is of finite length and terminated by two stations S_p, S_q . We model by the closed interval $[-p, q]$, where $\infty > p, q > 0$. We continue to assume the passing loop is at the origin $0 \in [-p, q]$. The *initialization set* \mathbb{I}_i for \mathfrak{T}_i is a choice of allowable initial conditions for the train \mathfrak{T}_i , $i \in \mathbf{2}$. We always assume \mathbb{I}_i is a closed subset of $M_i = [-p, q]$. In our case, we take

$$\mathbb{I}_1 = \{-p\} \subset [-p_1, q], \quad \mathbb{I}_2 = \{q\} \subset [-p_1, q].$$

We call $\mathbb{I} = \mathbb{I}_1 \times \mathbb{I}_2$ the *initialization set* for the network.

The *termination set*, or *target set* \mathbb{F}_i , for \mathfrak{T}_i is a choice of allowable final states for the train \mathfrak{T}_i . Again we assume \mathbb{F}_i is closed. We take

$$\mathbb{F}_1 = \{q\} \subset [-p_1, q], \quad \mathbb{F}_2 = \{-p\} \subset [-p_1, q].$$

We call $\mathbb{F} = \mathbb{F}_1 \times \mathbb{F}_2$ the termination (or target) set for the network.

What we are interested in is finding the time it takes for the trains to traverse the rail line: starting in the initialization set \mathbb{I} and ending in the termination set \mathbb{F} . In the simplest case we assume the trains depart from \mathbb{I} simultaneously at time $t = 0$. The times at which the trains reach the end of the line can be computed in terms of p, q , the velocities V_i and the time S both trains are required to be together in the loop. In real-world networks, however, starting times are almost never exact. Thus, a more realistic model for the network is to require stochastic initialization of starting times. For example, the actual time of departure of train \mathfrak{T}_i could be given by $T_i \geq 0$ sampled from an exponential or uniform distribution.

The network we are describing has a *function*: getting from points in \mathbb{I} to points in \mathbb{F} in finite time. The key invariant is the probability distribution of arrival times. For a fixed initialization, the arrival times can be explicitly computed whereas stochastically initialized networks yield nontrivial distribution of arrival times. Knowledge of the probability distribution allows, for example, to estimate the probability of network failure – failure to arrive within a prescribed time. There is also the possibility of deadlocks occurring: one or more of the trains never arrives: the network is unable to complete its function.

Even though the example of two trains in the passing loop is relatively simple, it relates to larger and more complex networks. For example, instead of stopping at its final destination, train \mathfrak{T}_1 may first

have to pass through a second passing loop with a third train \mathfrak{T}_3 . In this situation, the distribution of arrival times for the first network translate into an initial distribution of starting times for the second network. The larger functional network may thus be regarded as a concatenation of two functional networks. The function of the larger network will depend on the functions of the two constituent subnetworks. Iterating the process leads rapidly to functional networks that reflect the complexity of real transport networks. Conversely, it is natural to try to determine the factors or constituents of a large complex functional network.

7. FUNCTIONAL ASYNCHRONOUS NETWORKS AND DEADLOCKS

In the previous section we introduced the idea of a functional asynchronous network. We formalize the concept in this section. We also discuss some of the issues that arise such as deadlocks and stopping. We refer to [42] for greater generality and note that it is easy to allow for local times (as in section 6.2).

In what follows we assume that $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ is a proper asynchronous network with k nodes and semiflow $\Phi = (\Phi_1, \dots, \Phi_k)$.

7.1. Initialization, termination and network function.

Definition 7.1. Let $\mathbb{I}_i, \mathbb{F}_i \subset M_i$, $i \in \mathbf{k}$. The sets $\mathbb{I} = \prod_{i=1}^k \mathbb{I}_i$ and $\mathbb{F} = \prod_{i=1}^k \mathbb{F}_i$ are *initialization and termination sets* for \mathfrak{N} if the following conditions are satisfied.

- (P1) \mathbb{I}, \mathbb{F} are closed nonempty subsets of \mathbf{M} .
- (P2) If $\mathbf{X} \in \mathbb{I}$, then for each $i \in \mathbf{k}$ there exists $t_i(\mathbf{X}) \geq 0$ such that $\Phi_i(\mathbf{X}, t) \in \mathbb{I}_i$ if and only if $t \in [0, t_i(\mathbf{X})]$.
- (P3) If $\mathbf{X} \in \mathbb{I}$ and there exists $t \geq 0$ such that $\Phi_i(\mathbf{X}, t) \in \mathbb{F}_i$, then $t \geq t_i(\mathbf{X})$.

If (P3) applies, let $S_i = S_i(\mathbf{X}) = \inf_{t \geq 0} \{t \mid \Phi_i(\mathbf{X}, t) \in \mathbb{F}_i\}$ denote the *transit time* from \mathbf{X} to \mathbb{F}_i .

Remarks 7.2. (1) Condition (P2) implies that \mathbb{I} contains no compact Φ -invariant sets.

(2) Since \mathbb{F}_i is assumed closed, we have $\Phi_i(\mathbf{X}, S_i) \in \mathbb{F}_i$.

(3) If we insist that $\mathbb{I}_i \cap \mathbb{F}_i = \emptyset$, then (P3) follows with $S_i > t_i(\mathbf{X})$ provided either \mathbb{I}_i or \mathbb{F}_i is compact. However, it will sometimes be useful to take $\mathbb{I}_i = \mathbb{F}_i$. As a general rule, we always assume either $\mathbb{I}_i \cap \mathbb{F}_i = \emptyset$ or $\mathbb{I}_i = \mathbb{F}_i$ and then (P3) is immediate.

(4) In the definition, we do not (yet) require that for *every* $\mathbf{X} \in \mathbb{I}$, $i \in \mathbf{k}$, there exists $S_i \geq 0$ for which $\Phi_i(\mathbf{X}, S_i) \in \mathbb{F}_i$. Moreover, if

$\Phi_i(\mathbf{X}, S_i) \in \mathbb{F}_i$, it may or may not be the case that $\Phi_i(\mathbf{X}, t) \in \mathbb{F}_i$, for $t > S_i$. The point is that S_i gives the time that the state of node N_i first enters \mathbb{F}_i . For some examples, it is natural to have $\Phi_i(\mathbf{X}, t) = \Phi_i(\mathbf{X}, S_i)$ for all $t \geq S_i$ (so the state $\Phi_i(\mathbf{X}, S_i)$ really is a terminal state for the semiflow Φ , not just for the function of transitioning from points in \mathbb{I} to \mathbb{F}_i , $i \in \mathbf{k}$). In other situations the state may continue to evolve under Φ after a possibly nonzero stopping time (see section 6.2).

(5) It is often natural to assume that each phase space M_i has boundary ∂M_i and that $\partial M_i \supset \mathbb{I}_i \cup \mathbb{F}_i$. This is the situation for the passing loop example discussed in the previous section. In some applications \mathbb{F}_i might be a relatively small subset of the boundary ∂M_i . For example, part of the function of the network might be to attain a specific terminating state for each node. More generally, $\mathbb{I}_i, \mathbb{F}_i$ may be hypersurfaces in M_i , possibly with $\partial \mathbb{I}_i, \partial \mathbb{F}_i \neq \emptyset$. In particular, $M_i \setminus \mathbb{I}_i, M_i \setminus \mathbb{F}_i$ may be connected.

Definition 7.3. (Notation and assumptions as above.) Let \mathbb{I}, \mathbb{F} be initialization and termination sets for \mathfrak{N} . A point $\mathbf{X} \in \mathbb{I}$ is Φ -connected to \mathbb{F} if there exists $\mathbf{Y} = (\mathbf{y}_1, \dots, \mathbf{y}_k) \in \mathbb{F}$ and transit times $\mathbf{S} = \mathbf{S}(\mathbf{X}) = (S_1, \dots, S_k) \in \mathbb{R}_+^k$ such that

$$\mathbf{y}_i = \Phi_i(\mathbf{X}, S_i), \quad i \in \mathbf{k}.$$

Remarks 7.4. (1) With the notation of definition 7.3, we say \mathbf{X} is Φ -connected to \mathbf{Y} . If \mathbf{X} is Φ -connected to \mathbf{Y} , then $S_i = S_i(\mathbf{X})$ is always the minimal transit time from \mathbf{X} to \mathbb{F}_i . Setting $\mathbf{S}(\mathbf{X}) = (S_1, \dots, S_k)$ and abusing notation, we often write $\mathbf{Y} = (\mathbf{y}_1, \dots, \mathbf{y}_k) = \Phi(\mathbf{X}, \mathbf{S}(\mathbf{X}))$. We refer to \mathbf{y}_i as the *terminal state* of N_i , $i \in \mathbf{k}$.

(2) If $\mathbf{X} \in \mathbb{I}$ is Φ -connected to \mathbb{F} , this does not imply that the Φ -trajectory through \mathbf{X} meets \mathbb{F} . Even if there exists $s > 0$ such that $\Phi(\mathbf{X}, s) \in \mathbb{F}$, then s and $\Phi(\mathbf{X}, s)$ may not give any of the transit times and terminal states. That is, we may have $S_i < s$ and $\Phi_i(\mathbf{X}, s) \neq \mathbf{y}_i$ for all $i \in \mathbf{k}$. However, if $\Phi_i(\mathbf{X}, t) = \Phi_i(\mathbf{X}, S_i)$ for all $t \geq S_i$ then $\mathbf{X} \in \mathbb{I}$ is Φ -connected to \mathbb{F} if and only if the Φ -trajectory through \mathbf{X} meets \mathbb{F} .

Let \mathbb{I}, \mathbb{F} be initialization and termination sets for \mathfrak{N} and set

$$\mathcal{D}(\mathfrak{N}, \mathbb{I}, \mathbb{F}) = \{\mathbf{X} \in \mathbb{I} \mid \mathbf{X} \text{ is } \Phi\text{-connected to } \mathbb{F}\}.$$

Definition 7.5. (Notation and assumptions as above.) The *transition function* $G_0 : \mathcal{D}(\mathfrak{N}, \mathbb{I}, \mathbb{F}) \subset \mathbb{I} \rightarrow \mathbb{F}$ is defined by

$$G_0(\mathbf{X}) = \Phi(\mathbf{X}, \mathbf{S}(\mathbf{X})), \quad \mathbf{X} \in \mathcal{D}(\mathfrak{N}, \mathbb{I}, \mathbb{F}).$$

Definition 7.6. Let \mathbb{I}, \mathbb{F} be initialization and termination sets for \mathfrak{N} . The triple $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ is a *functional asynchronous network (FAN)*.

The *network function* is transition from points in \mathbb{I} to \mathbb{F} and is represented by the transition function $G_0 : \mathcal{D}(\mathbf{N}) \subset \mathbb{I} \rightarrow \mathbb{F}$. The network function is *achieved* if $\mathcal{D}(\mathbf{N}) = \mathbb{I}$ – that is, if every point in \mathbb{I} is Φ -connected to \mathbb{F} and $G_0 : \mathbb{I} \rightarrow \mathbb{F}$.

Remarks 7.7. (1) For a general FAN \mathbf{N} , the network function may not always be achieved and so $\mathcal{D}(\mathbf{N})$ may be a proper subset of \mathbb{I} . Achieving the network function may be part of the process of network design and involve a mix of dynamics and logical conditions.

(2) In the sequel we regard the achievement of network function as synonymous with the transition function G_0 having domain \mathbb{I} .

7.2. Deadlocks. In this section we address one of the reasons for the failure of a FAN to achieve its function: the presence of deadlocks.

Definition 7.8. A FAN $(\mathfrak{N}, \mathbb{I}, \mathbb{F})$ has a *dynamical deadlock* if there exists a subset $A \subset \mathbf{M}$ such that

- (1) A is compact and semiflow invariant: $\Phi^t(A) = A, t \geq 0$.
- (2) $A \cap (\mathbb{I} \cup \mathbb{F}) = \emptyset$.
- (3) There is a nonempty subset X of $\mathbb{I} \setminus \mathcal{D}(\mathbf{N})$ such that every trajectory through a point of X enters A within finite time.

We refer to A as *deadlock sink*. If X contains an open set, A is a *topological deadlock sink*, and if X has nonzero Lebesgue measure, A is an *observable deadlock sink*.

Remark 7.9. We can strengthen condition (2) of the definition to require that $\pi_i(A) \cap (\mathbb{I}_i \cup \mathbb{F}_i) = \emptyset$, all $i \in \mathbf{k}$. In this case, if a trajectory enters A , then none of the nodes ever reach a terminal state: a *total deadlock*.

Example 7.10. In the example of section 6.3, suppose that a pair of phase oscillators are coupled when both trains are in the passing loop and that dynamics is given by

$$\theta'_i = \omega + k_{ij} \sin(\theta_j - \theta_i), \quad i \in \mathbf{2},$$

where $k_{12}, k_{21} > 0$. We have a dynamical deadlock given by $\theta_2 = \theta_1 + \pi$. Here the initialization sets will be given by $\mathbb{I}_1 = \{-p\} \times \mathbb{T}$, $\mathbb{I}_2 = \{q\} \times \mathbb{T}$ and the deadlock will not be observable. \diamond

The next lemma shows that dynamical deadlocks cannot occur in networks governed by a single set of differential equations.

Lemma 7.11. *A FAN $(\mathfrak{N}, \mathbb{I}, \mathbb{F})$ has no dynamical deadlocks if $\mathcal{E}|\mathbf{M} \setminus (\mathbb{I} \cup \mathbb{F})$ is constant.*

Proof. Suppose that A is a deadlock sink for the FAN. Let \mathbf{F} be the vector field on \mathbf{M} determined by $\mathcal{E}|\mathbf{M} \setminus (\mathbb{I} \cup \mathbb{F})$. Clearly, A is invariant by the flow of \mathbf{F} . Since A has open neighbourhood in $\mathbf{M} \setminus (\mathbb{I} \cup \mathbb{F})$, it follows that no trajectory starting in \mathbb{I} (or $\mathbf{M} \setminus A$) can enter A in finite time, contradicting our assumption that A is a deadlock sink for the FAN. \square

Definition 7.12. Let A be a deadlock sink for the FAN $(\mathfrak{N}, \mathbb{I}, \mathbb{F})$.

- (1) A is a *deadlock* if A consists of a single point (a_1, \dots, a_k) . The deadlock is *total* if $a_i \notin \mathbb{F}_i$ for all $i \in \mathbf{k}$, and *partial* if there exists $i \in \mathbf{k}$ such that $a_i \in \mathbb{F}_i$.
- (2) A is a *livelock* if A is a periodic orbit.

Examples 7.13. (1) The deadlock given in example 7.10 is a livelock – an antiphase (periodic) solution of the phase oscillator pair.

(2) Our next example gives an illustration of how a deadlock can occur because of faulty logic (this is a common programming issue in computer science). Suppose we have a 4-way stop sign at a cross roads. We have the following traffic rules.

- (a) All traffic arriving at the stop sign stops.
- (b) No left or right turns allowed – when a vehicle restarts it proceeds in same direction whence it came.
- (c) Priority is determined by order of arrival – first to arrive (stop), first to leave (restart) – unless more than one vehicle arrives at the stop sign at the same time, in which case priority is given to the car on the right (there is no issue if two vehicles arrive at the same time from opposite directions).

A total deadlock occurs if four cars arrive at the stop sign at the same time. There is no easy way to vary the logic to resolve the deadlock. Provided the traffic is light, the deadlock may be regarded as acceptable¹¹, otherwise it may be preferable (a) to have a two way stop sign, or (b) use traffic lights. Although the deadlock is, in principle, not observable, in practice it is: drivers are only able to approximately judge the time of arrival of vehicles on their right and thus the practical rule is similar to the ϵ -approximation synchronization rule given in section 6.3. The example can be extended to allow N -way stop signs or right and left turns at the intersection. \diamond

The presence of deadlocks can prevent a functional network from completing its function. We discuss some examples of deadlocks in real-world networks and how they can be resolved.

¹¹An example of the *ostrich algorithm* from computer science.

7.2.1. *Resource allocation.* In computer science and distributed systems, deadlocks are characteristically associated with problems in resource allocation. In our terms, this can be interpreted as two (or more) nodes trying to connect to a third node N_t which only allows one connection. For example, in threaded computation data corruption can occur if two threads attempt simultaneous writes to the same memory. The resolution involves (a) the connected node N_s having a *lock* on the target node until the process requiring the connection is finished; (b) a protocol for how to handle the situation when two nodes simultaneously request connection to N_t , (c) prioritisation of connection requests. If the node N_s does not ever get disconnected from N_t , then there will be at least a partial deadlock and if low priority nodes are connected then the system may run slowly. If the time of connection is small then attempts at simultaneous connection may be rare: deadlocks are often very hard to find in complex networks. In terms of event driven dynamics and asynchronous networks, (a,b) are handled by a correctly written event structure which may use local time. After the connection between N_t and N_s is released, a new connection will be made either randomly or consistent with a prioritisation list (or both!). In large complex networks, it may be extremely hard to organise the structure so that there are no deadlocks.

We conclude with a simple example where deadlocks can occur owing to faulty network logic (the example can be interpreted in terms of resource allocation).

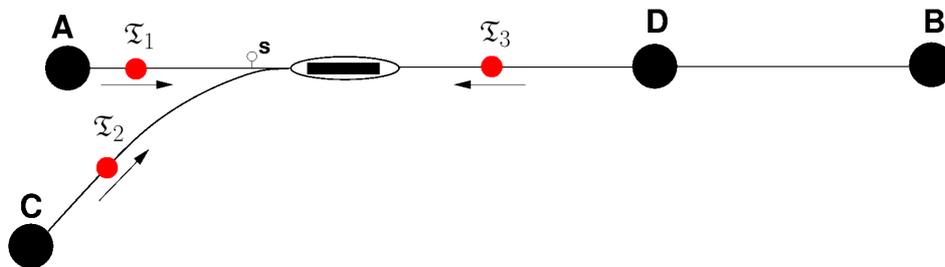


FIGURE 10. Passing loop and junction on a single track rail line with three trains.

7.2.2. *Faulty logic leading to a deadlock.* In figure 10 we show a one track railway line with passing loop, junction, four stations and three trains $\mathfrak{T}_1, \mathfrak{T}_2, \mathfrak{T}_3$. We suppose that \mathfrak{T}_1 starts at A and proceeds to B; \mathfrak{T}_2 starts at C and proceeds to D, \mathfrak{T}_3 starts at B and proceeds to A. Matters are organised so that trains $\mathfrak{T}_1, \mathfrak{T}_2$ start at the same time and \mathfrak{T}_1 normally arrives at the passing loop before \mathfrak{T}_2 . The junction is

controlled by signals in the following way: normally signals are both set to red. If a single train approaches, the signal changes to green on that line and allows the train to proceed. If the trains arrive simultaneously at the signal, priority is given to the train on the line from $A - \mathfrak{T}_1$. Provided \mathfrak{T}_1 arrives at the junction before \mathfrak{T}_2 , there are no problems. However, if \mathfrak{T}_2 arrives first, then \mathfrak{T}_2 enters the passing loop and waits for \mathfrak{T}_3 . When both trains are in the passing loop neither can proceed: \mathfrak{T}_2 cannot proceed because it would block \mathfrak{T}_1 from proceeding to D and \mathfrak{T}_3 cannot proceed because the line to A is blocked by \mathfrak{T}_1 . Hence we have a total deadlock. In a purely deterministic system the deadlock will not occur provided that the departures of $\mathfrak{T}_1, \mathfrak{T}_2$ from A and B are correctly timed.

7.3. To stop or not to stop? Consequences for network function. Given a FAN $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$, our primary interest is in the terminal states, as given by the transition function $G_0 : \mathcal{D}(\mathbf{N}) \rightarrow \mathbb{F}$, and the transition times, as given by the timing function $\mathbf{S} : \mathcal{D}(\mathbf{N}) \rightarrow \mathbb{R}_+^k$. It is natural that nodes stop when they have reached their terminal state. That is, after a node has terminated its state remains fixed, while all other nodes evolve according to the network dynamics. Such stopping may be part of the network dynamics itself (if there is an associated stopping event). In some cases, such as in the passing loop example, stopping is *forced* by reaching the “end of the line” – a trajectory reaches the boundary of the phase space. For general asynchronous networks, if a node is forced to stop, the evolution of the remaining nodes may change or may not be well defined – by stopping the node we may change the network structure. While for many applications, it is straightforward to continue trajectories after a node is forced to stop and we do not change the network function, care is needed: stopping a node may change the network function. We present a simple cautionary example that shows how stopping nodes can not only affect transit times but also create deadlocks.

Example 7.14. Take a three node FAN with $M_i = \mathbb{R}$, $\mathbb{I}_i = \{0\}$, $i \in \mathbf{3}$, $\mathbb{F}_1 = \{1\}$, and $\mathbb{F}_2 = \mathbb{F}_3 = \{2\}$. Define dynamics according to

$$\begin{aligned} x' &= 1 \\ y' &= \begin{cases} 1, & \text{if } y < 1, \text{ or } x \geq 1.5 \\ 0, & \text{if } y = 1, \text{ and } x < 1.5 \end{cases} \\ z' &= \begin{cases} \frac{1}{2}, & \text{if } x < 1.5 \\ 1, & \text{if } x \geq 1.5 \end{cases} \end{aligned}$$

If we continue evolution past the terminal states, then $(0, 0, 0)$ is Φ -connected to $(1, 2, 2)$ and $\mathbf{S}(0, 0, 0) = (1, 2.5, 2.75)$. On the other hand, if we stop evolution of nodes when they reach their terminal state, then $(0, 0, 0)$ is not Φ -connected to \mathbb{F} : N_2 never attains its terminal state and there is a deadlock. Moreover, N_3 now takes time 4 to reach its terminal state. \diamond

The question of stopping nodes becomes an issue when considering, for example, a large transport network (representable as a FAN) that is composed of many subnetworks (each also representable as a FAN). Here we would like to describe the transition and timing functions of the main network in terms of the transition and timing functions of the constituent subnetworks (these will generally have initialization and termination sets which are different from those of the main network). In order to do this, we require that the dynamics of each subnetwork does not depend on dynamics past termination, or equivalently, that stopping of nodes does not change the network function. In the next section we give conditions that enable us to carry through this program.

8. WEAKLY REGULAR AND REGULAR FANS

Our aim in the next two sections is show how we can decompose the dynamics of a FAN of ‘feedforward’ type in terms of the dynamics of subnetworks which are also FANs. In the present section, we shift our focus from the absolute definition of a FAN, as given in definition 7.6, to a more relative definition where we impose geometric and structural conditions on a (sub)network that give dynamics that is closely related to the dynamics of the containing global network. As part of this process, we will need to relax our unrealistic assumption that all nodes, even uncoupled nodes, are started at exactly the same time.

8.1. Geometric FANs. We start with some notational conventions and assumptions that we maintain throughout this section. If $(\mathfrak{N}, \mathbb{I}, \mathbb{F})$ is a FAN, then $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ will be proper asynchronous network with k nodes and well defined semiflow $\Phi = (\Phi_1, \dots, \Phi_k)$. Furthermore, we suppose that $\emptyset \in \mathcal{A}$, set $\mathbf{f}^\emptyset = \mathbf{Z} = (Z_1, \dots, Z_k)$ and assume that \mathbf{Z} determines a smooth flow $\Psi^t = (\psi_1^t, \dots, \psi_k^t)$ on \mathbf{M} (automatic if \mathbf{M} is compact). Since \mathbf{Z} is given by the empty connection structure, ψ_i^t determines a flow on M_i for all $i \in \mathbf{k}$.

Definition 8.1. The FAN $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ is *geometric* if

- (G) For all $i \in \mathbf{k}$, $\mathbb{I}_i, \mathbb{F}_i$ are disjoint nonempty closed hypersurfaces in M_i that separate M_i into nonempty closed connected regions M_i^-, M_i^0, M_i^+ with smooth boundary satisfying

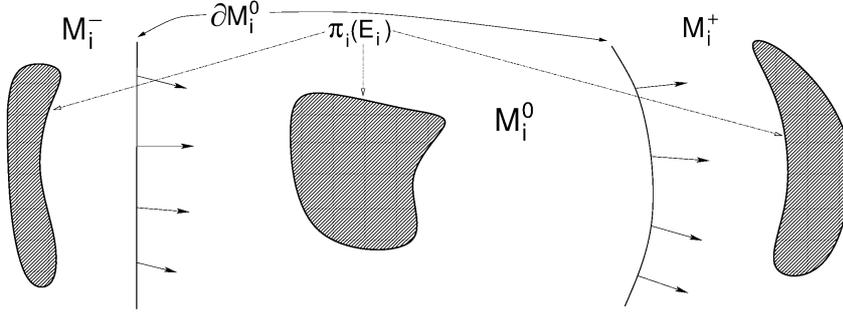


FIGURE 11. Geometry and constraints on node dynamics imposed by conditions (G,T).

- (a) $\partial M_i^- = M_i^- \cap M_i^0 = \mathbb{I}_i$,
- (b) $\partial M_i^+ = M_i^+ \cap M_i^0 = \mathbb{F}_i$.
- (T) For $i \in \mathbf{k}$, Z_i is transverse to $\mathbb{I}_i \cup \mathbb{F}_i$, and is inward pointing for M_i^0 along \mathbb{I}_i , and outward pointing for M_i^0 along \mathbb{F}_i .
- (F) For each $\mathbf{X} \in \mathbb{I}$, and $i \in \mathbf{k}$, there exists a unique smallest $S_i(\mathbf{X}) \in \mathbb{R}_+$ such that $\Phi_i(\mathbf{X}, S_i(\mathbf{X})) \in \mathbb{F}_i$.

See figure 11 and the following remarks for the geometric implication of these conditions. Note that the labels G , T and F refer respectively to *Geometry*, *Transversality* and *Function*.

Remarks 8.2. (1) Condition (G) implies condition (P1); conditions (G,T) imply condition (P2) (with $t_i(\mathbf{X}) = 0$, for all $i \in \mathbf{k}$ and $\mathbf{X} \in \mathbb{I}$). Condition (P3) follows trivially from (F) and (G).

(2) The geometric conditions on the regions M_i^- , M_i^0 , M_i^+ are strong and entail that $\mathbb{I}_i, \mathbb{F}_i$ each separate M_i into two connected components. For periodic problems, such as daily factory inventory oscillations or biological rhythms¹², $\mathbb{I}_i, \mathbb{F}_i$ may not disconnect M_i (for example, if $M_i = \mathbb{T}^n$). This would imply that $M_i^- = M_i^+$ and allow for trajectories to leave and reenter M_i^0 or $M_i \setminus M_i^0$. In a different direction, we may weaken condition (G) by allowing, for example, \mathbb{I}_i to be a closed connected hypersurface with boundary $\partial \mathbb{I}_i \neq \emptyset$. If every point of $\partial \mathbb{I}_i$ is connected to \mathbb{F}_i by a forward trajectory of ψ_i , then we can form a region $M_i^0 \subset M_i$ (with corners) bounded by $\mathbb{I}_i, \mathbb{F}_i$ and the ψ_i -trajectories joining $\partial \mathbb{I}_i$ to \mathbb{F}_i . Now we may need to place restrictions on re-entry: once $\Phi_i(\mathbf{X}, t)$ exits M_i^0 it is not allowed to re-enter M_i^0 .

(3) Condition (T) implies that $Z_i|_{(\mathbb{I}_i \cup \mathbb{F}_i)}$ is non-vanishing, all $i \in \mathbf{k}$.

¹²Thanks to David Rand for pointing these examples out to us.

Lemma 8.3. *Let $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ be a geometric FAN. Then \mathbf{N} has a well-defined transition function $G_0 : \mathbb{I} \rightarrow \mathbb{F}$ and timing function $\mathbf{S} : \mathbb{I} \rightarrow \mathbb{R}_+^k$.*

Proof. Immediate from remarks 8.2(1) and condition (F). \square

8.2. Weakly regular FANs. Given a connection structure $\alpha \in \mathcal{A}$, recall from section 5.4 that $v(\alpha) \subset \mathbf{k}$ denotes the set of linked nodes for α and $V(\mathcal{A}) = \bigcup_{\alpha \in \mathcal{A}} v(\alpha)$. Define

$$E_i = \{\mathbf{X} \in \mathbf{M} \mid i \in v(\mathcal{E}(\mathbf{X}))\}, \quad i \in \mathbf{k},$$

$$E^* = \bigcup_{i \in \mathbf{k}} E_i.$$

Observe that $\mathbf{X} \in E^*$ if and only if at least one node is linked at \mathbf{X} . We have

$$E^\emptyset = \mathbf{M} \setminus E^*.$$

Remark 8.4. In many applications nodes will interact only if they are constrained. We may then think of a constrained node as being in a partially stopped or waiting state. Typically, a node enters a constrained state when certain thresholds are met (a state dependent event) and the node is waiting for interaction and/or synchronization with other nodes. If we require that only constrained nodes are coupled then

- (1) A node is linked iff it is constrained: $i \in v(\alpha) \iff N_0 \rightarrow N_i$.
- (2) If $N_i \rightarrow N_j$, $i, j \in \mathbf{k}$, then $N_0 \rightarrow N_i, N_0 \rightarrow N_j$.

While this assumption is *not* necessary for the theory we develop, it is consistent with the passing loop examples or the threaded programming model. See also remarks 8.26(5).

Before we give our main definition, we need some new notation and a definition. Suppose that A^0, A^-, A^+ are proper disjoint subsets of \mathbf{k} , with $A^0 \cup A^- \cup A^+ = \mathbf{k}$. Observe that at least two of A^0, A^-, A^+ must be nonempty.

If $\sigma \in \{-, +, 0\}$ and $\alpha \in \mathcal{A}$, define $\alpha^\sigma \in M_\bullet(k)$ by

$$(12) \quad \alpha^\sigma = \{j \rightarrow i \in \alpha \mid j \in (A^\sigma)^\bullet, i \in A^\sigma\},$$

Let $\sigma \in \{-, +, 0\}$ and suppose that for $i \in A^\sigma$, we are given an open neighbourhood W_i^σ of M_i^σ in M_i . Set $\mathbf{W}^\sigma = \prod_{i \in A^\sigma} W_i^\sigma$ and define

$$\mathbf{W} = \mathbf{W}^0 \times \mathbf{W}^- \times \mathbf{W}^+ \subset \mathbf{M}.$$

Definition 8.5. (Notation and assumptions as above.) The asynchronous network \mathfrak{N} has *product structure* on \mathbf{W} if for each $\sigma \in \{-, +, 0\}$,

we can find an asynchronous network $\mathfrak{N}^\sigma = (\mathcal{N}^\sigma, \mathcal{A}^\sigma, \mathcal{F}^\sigma, \mathcal{E}^\sigma)$, where \mathcal{N}^σ has nodes $\{N_i \mid i \in A^\sigma\}$ and network phase space \mathbf{W}^σ , such that

$$\mathfrak{N}|\mathbf{W} = \mathfrak{N}^0 \times \mathfrak{N}^- \times \mathfrak{N}^+.$$

If \mathfrak{N} has product structure on \mathbf{W} , it follows from the results of section 5.3 that the asynchronous networks $\mathfrak{N}^0, \mathfrak{N}^-, \mathfrak{N}^+$ are proper (since \mathfrak{N} is assumed proper) and that for all $\mathbf{X} = (\mathbf{X}^0, \mathbf{X}^-, \mathbf{X}^+) \in \mathbf{W}$ we have

$$(13) \quad \mathcal{E}(\mathbf{X}) = \mathcal{E}^0(\mathbf{X}^0) \vee \mathcal{E}^-(\mathbf{X}^-) \vee \mathcal{E}^+(\mathbf{X}^+)$$

$$(14) \quad \mathbf{f}^{\mathcal{E}(\mathbf{X})}(\mathbf{X}) = \mathbf{f}_0^{\mathcal{E}^0(\mathbf{X}^0)}(\mathbf{X}^0) \times \mathbf{f}_-^{\mathcal{E}^-(\mathbf{X}^-)}(\mathbf{X}^-) \times \mathbf{f}_+^{\mathcal{E}^+(\mathbf{X}^+)}(\mathbf{X}^+).$$

Remarks 8.6. (1) With the notation of (12), equation (13) implies that $\mathcal{E}^\sigma(\mathbf{X}^\sigma) = \mathcal{E}(\mathbf{X})^\sigma$, for all $\mathbf{X} = (\mathbf{X}^0, \mathbf{X}^-, \mathbf{X}^+) \in \mathbf{W}$, $\sigma \in \{-, +, 0\}$.

(2) Equation (14) implies that if $i \in A^\sigma$, then the component $f_i^{\mathcal{E}(\mathbf{X})}|\mathbf{W}$ depends only on $\mathbf{X}^\sigma \in \mathbf{W}^\sigma$ and is dependent only on nodes in \mathcal{N}^σ .

Definition 8.7. (Notation as above.) The FAN $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ is *weakly regular* if

- (1) \mathbf{N} is geometric.
- (2) There exist open neighbourhoods V_i of ∂M_i^0 in M_i , $i \in \mathbf{k}$, such that
 - (S1) $\pi_i(E_i) \subset M_i \setminus V_i$, all $i \in \mathbf{k}$.
 - (S2) If A^0, A^-, A^+ are proper disjoint subsets of \mathbf{k} , with $A^0 \cup A^- \cup A^+ = \mathbf{k}$, and $\mathbf{W}^\sigma = \prod_{i \in A^\sigma} (M_i^\sigma \cup V_i)$, $\sigma \in \{0, -, +\}$, then \mathfrak{N} has product structure on $\mathbf{W} = \mathbf{W}^0 \times \mathbf{W}^- \times \mathbf{W}^+$.

Remarks 8.8. (1) Condition (S1) implies that if the state of node N_i is close to $\mathbb{I}_i \cup \mathbb{F}_i$, then the node will be uncoupled. It also follows from (S1) that $\prod_{i \in \mathbf{k}} V_i \subset E^\emptyset$ and so E^\emptyset is a neighbourhood of $\mathbb{I} \cup \mathbb{F}$.

(2) It follows from (S1,S2) and remarks 8.6(2) that $\mathcal{A}^\sigma \subset \mathcal{A}$, $\sigma \in \{0, -, +\}$ (for example, if $\sigma = 0$ and $A^0 \neq \emptyset$, choose $\mathbf{X}^- \in \prod_{i \in A^-} V_i$, $\mathbf{X}^+ \in \prod_{i \in A^+} V_i$).

(2) Let $\sigma, \eta \in \{0, -, +\}$, $\sigma \neq \eta$. Since $\mathfrak{N}|\mathbf{W} = \mathfrak{N}^0 \times \mathfrak{N}^- \times \mathfrak{N}^+$, there are no connections between N_i and N_j , if $i \in A^\sigma$, $j \in A^\eta$.

The next result will be critical for developing the dynamical and structural properties of weakly regular FANs.

Lemma 8.9 (Local product structure). *Let $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ be a weakly regular FAN. Let $A \subset \mathbf{k}$ be a nonempty subset of \mathbf{k} and set $B = \mathbf{k} \setminus A$. Set $\mathbf{M}_B^\pm = \prod_{i \in B} (M_i^- \cup M_i^+)$. If $\mathbf{X} = (\mathbf{X}_A, \mathbf{X}_B) \in \mathbf{M}_A^0 \times \mathbf{M}_B^\pm$, there exists $\delta > 0$ such that for $t \in [0, \delta]$ we may write*

$$\Phi(\mathbf{X}, t) = (\Phi_A(\mathbf{X}_A, t), \Phi_B(\mathbf{X}_B, t)).$$

Proof. Choose $\delta > 0$ so that for $t \in [0, \delta]$, $\Phi_i(\mathbf{X}, t) \in M_i^0 \cup V_i$, all $i \in A$, and $\Phi_j(\mathbf{X}, t) \in M_j^- \cup M_j^+ \cup V_j$, all $j \in B$. It follows from (S2) that we may write

$$\Phi(\mathbf{X}, t) = (\Phi_A(\mathbf{X}_A, t), \Phi_B(\mathbf{X}_B, t)) \in \mathbf{M}_A^0 \times \mathbf{M}_B^\pm, \quad t \in [0, \delta],$$

where $\Phi_A(\mathbf{X}_A, t) = \pi_A \Phi(\mathbf{X}, t)$, $\Phi_B(\mathbf{X}_B, t) = \pi_B \Phi(\mathbf{X}, t)$. (These relations may fail once $\Phi_i(\mathbf{X}, t)$ exits V_i , $i \in \mathbf{k}$.) \square

8.3. Generalized initialization of FANs. We require conditions that will allow us to write the transition and timing function of a FAN in terms of transition and timing functions of constituent subnetworks. We start with an example that indicates some of the problems and the need to work with a more general definition of transition function.

Example 8.10. Let $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ be a weakly regular FAN. Suppose we can identify two FANs $\mathbf{N}^1 = (\mathfrak{N}^1, \mathbb{I}^1, \mathbb{F}^1)$, $\mathbf{N}^2 = (\mathfrak{N}^2, \mathbb{I}^2, \mathbb{F}^2)$ with semiflows Φ^1 and Φ^2 such that

- (1) $\mathbb{I}^1 = \mathbb{I}$, $\mathbb{F}^1 = \mathbb{I}^2$, $\mathbb{F}^2 = \mathbb{F}$.
- (2) $\mathbf{N}^1, \mathbf{N}^2$ are weakly regular.
- (3) $\mathbf{N}^1, \mathbf{N}^2$ are compatible (definition 5.22) and $\mathbf{N} = \mathbf{N}^2 \sqcup \mathbf{N}^1$ (definition 5.26).

Is it possible to express the network function of \mathbf{N} , as given by the transition function G_0 and timing function \mathbf{S} , in terms of the subnetworks $\mathbf{N}^1, \mathbf{N}^2$ and their transition and timing functions G_0^1, G_0^2 and $\mathbf{S}^1, \mathbf{S}^2$? The answer is yes if all nodes of \mathbf{N}^1 terminate simultaneously. That is, for every $\mathbf{X} \in \mathbb{I}^1$, there exists $S^1 > 0$ such that $\mathbf{S}^1(\mathbf{X}) = (S^1, \dots, S^1)$. Then $\mathbf{Y} = G_0^1(\mathbf{X}) = \Phi^1(\mathbf{X}, S^1) \in \mathbb{F}^1$ and we have $G_0(\mathbf{X}) = G_0^2(G_0^1(\mathbf{X}))$ and $\mathbf{S}(\mathbf{X}) = \mathbf{S}^1(\mathbf{X}) + \mathbf{S}^2(\mathbf{Y})$.

In general, nodes of \mathbf{N}^1 will not terminate simultaneously and so we need to allow for general initialization times for \mathbf{N}^2 (corresponding to the termination times for \mathbf{N}^1). That is, we require a generalized transition function $G^2(\mathbf{Y}, \mathbf{S}^1) = (\mathbf{W}, \mathbf{S}^2)$ for \mathbf{N}^2 that gives the terminal states \mathbf{W} and termination times \mathbf{S}^2 in terms of the initial states \mathbf{Y} and initialization times $\mathbf{S}^1(\mathbf{X})$ derived from \mathbf{N}^1 . \diamond

Let $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ be a weakly regular FAN with semiflow Φ . Set $\widehat{\mathbb{I}} = \mathbb{I} \times \mathbb{R}_+^k$, $\widehat{\mathbb{F}} = \mathbb{F} \times \mathbb{R}_+^k$. We refer to $\widehat{\mathbb{I}}$ and $\widehat{\mathbb{F}}$ as generalized initialization and termination sets.

Our first step will be to construct *evolution* and *timing* operators

$$\begin{aligned} \widehat{\Phi} &= (\widehat{\Phi}_i, \dots, \widehat{\Phi}_k) : \widehat{\mathbb{I}} \times \mathbb{R}_+ \rightarrow \mathbf{M} \\ \widehat{\mathbf{S}} &= (\widehat{S}_1, \dots, \widehat{S}_k) : \widehat{\mathcal{D}}(\mathbf{N}) \subset \widehat{\mathbb{I}} \rightarrow \mathbb{R}_+^k \end{aligned}$$

that allow for general initialization times.

We first construct $\widehat{\Phi}$. Let $(\mathbf{X}, \mathbf{T}) \in \widehat{\mathbb{I}}$ and $0 \leq \tau_1 < \tau_2 < \dots < \tau_p$ be chosen so that $\tau_1 = \min \{T_1, \dots, T_k\}$, $\tau_p = \max \{T_1, \dots, T_k\}$ and $\{T_1, \dots, T_k\} = \{\tau_1, \dots, \tau_p\}$. For $\ell \in \mathbf{p}$, define $J_\ell = \{i \in \mathbf{k} \mid T_i = \tau_\ell\}$. For $t \leq \tau_1$, define $\widehat{\Phi}((\mathbf{X}, \mathbf{T}), t) = \mathbf{X}$. For $t \in (\tau_1, \tau_2]$, define $\widehat{\Phi}((\mathbf{X}, \mathbf{T}), t)$ by evolving the variables \mathbf{x}_i , $i \in J_1$, under Φ , keeping the remaining variables fixed. This is well defined by lemma 8.9. Proceeding inductively in the obvious way, using lemma 8.9, we successively switch on the variables associated to J_2, \dots, J_p . For the final step, suppose that $\widehat{\Phi}((\mathbf{X}, \mathbf{T}), \tau_p) = \mathbf{Y}$. Using the properness of \mathfrak{N} , define $\widehat{\Phi}((\mathbf{X}, \mathbf{T}), t) = \Phi(\mathbf{Y}, t - \tau_p)$, $t > \tau_p$.

Lemma 8.11. *Let $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ be weakly regular. The evolution operator $\widehat{\Phi} : \widehat{\mathbb{I}} \times \mathbb{R}_+ \rightarrow \mathbf{M}$ constructed above is well defined and continuous in forward time. Moreover, if $\mathbf{0} = (0, \dots, 0) \in \mathbb{R}_+^k$, then*

$$\widehat{\Phi}((\mathbf{X}, \mathbf{0}), t) = \Phi(\mathbf{X}, t), \text{ for all } t \in \mathbb{R}_+.$$

Proof. Immediate from the constructions preceding the lemma. \square

Remarks 8.12. (1) The trajectories of $\widehat{\Phi}$ may be given as solutions of a differential equation. Indeed, if \mathbf{F} is the network vector field and $H(t)$ denotes the Heaviside step function (with $H(0) = 1$), the trajectory $\widehat{\Phi}((\mathbf{X}, \mathbf{T}), t)$, $t \geq 0$, is the solution $\mathbf{X}(t)$ of the non-autonomous system

$$\mathbf{x}'_i(t) = H(T_i - t)F_i(\mathbf{X}(t)), \quad i \in \mathbf{k}.$$

The argument given prior to lemma 8.11 implies that solutions (in the sense of definition 4.24) exist provided that $\mathbf{X} \in \mathbb{I}$.

(2) It is generally *not* possible to extend $\widehat{\Phi}$ to $(\mathbf{M} \times \mathbb{R}_+^k) \times \mathbb{R}_+$.

For $i \in \mathbf{k}$, let \mathcal{D}_i be the subset of $\widehat{\mathbb{I}}$ consisting of (\mathbf{X}, \mathbf{T}) for which there exists a (minimal) $\widehat{S}_i = \widehat{S}_i(\mathbf{X}, \mathbf{T}) \geq T_i$ such that $\widehat{\Phi}_i((\mathbf{X}, \mathbf{T}), \widehat{S}_i) \in \mathbb{F}_i$. Set $\widehat{\mathcal{D}}(\mathbf{N}) = \bigcap_{i \in \mathbf{k}} \mathcal{D}_i$ and define the *timing function* $\widehat{\mathbf{S}} : \widehat{\mathcal{D}}(\mathbf{N}) \rightarrow \mathbb{R}_+^k$ by $\widehat{\mathbf{S}}(\mathbf{X}, \mathbf{T}) = (\widehat{S}_1(\mathbf{X}, \mathbf{T}), \dots, \widehat{S}_k(\mathbf{X}, \mathbf{T}))$. Since \mathbf{N} is weakly regular, $\widehat{\mathcal{D}}(\mathbf{N}) \supset \mathbb{I}$ (we identify \mathbb{I} with the subset $\{(\mathbf{X}, \mathbf{0}) \mid \mathbf{X} \in \mathbb{I}\}$ of $\widehat{\mathbb{I}}$).

Definition 8.13. (Notations and assumptions as above.) If $\widehat{\mathcal{D}}(\mathbf{N}) = \widehat{\mathbb{I}}$, we define the *generalized transition function* $G : \widehat{\mathbb{I}} \rightarrow \widehat{\mathbb{F}}$ by

$$G(\mathbf{X}, \mathbf{T}) = ((\widehat{\Phi}_1((\mathbf{X}, \mathbf{T}), \widehat{S}_1), \dots, \widehat{\Phi}_k((\mathbf{X}, \mathbf{T}), \widehat{S}_k)), \widehat{\mathbf{S}}(\mathbf{X}, \mathbf{T})),$$

where $\widehat{\mathbf{S}}(\mathbf{X}, \mathbf{T}) = (\widehat{S}_1, \dots, \widehat{S}_k)$ is given by the timing function.

Remark 8.14. For $(\mathbf{X}, \mathbf{T}) \in \widehat{\mathbb{I}}$ with $\mathbf{T} = (T, \dots, T)$, $T \in \mathbb{R}_+$ we have $G(\mathbf{X}, \mathbf{T}) = (G_0(\mathbf{X}), \mathbf{S}(\mathbf{X}) + \mathbf{T})$.

Definition 8.15. (Notations and assumptions as above.) A weakly regular FAN is *regular* if $\widehat{\mathcal{D}}(\mathbf{N}) = \widehat{\mathbb{I}}$.

Remark 8.16. If $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ is weakly regular, we write $\widehat{\mathbf{N}} = (\mathfrak{N}, \widehat{\mathbb{I}}, \widehat{\mathbb{F}})$ to emphasise that we require initialization from $\widehat{\mathbb{I}} = \mathbb{I} \times \mathbb{R}_+^k$ rather than at time zero from \mathbb{I} . In particular, the FAN $\widehat{\mathbf{N}}$ achieves its network function (that is, \mathbf{N} is regular) if every point of $\widehat{\mathbb{I}}$ is $\widehat{\Phi}$ -connected to $\widehat{\mathbb{F}}$.

Example 8.17. The example of two trains on a single track railway line with a passing loop and stations described in section 6.4 admits a generalized transition function $G : \mathbb{I} \times \mathbb{R}_+^2 \rightarrow \mathbb{F} \times \mathbb{R}_+^2$.

8.4. Hidden deadlocks.

Definition 8.18. Let $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ be a weakly regular FAN with semiflow Φ . A compact Φ -invariant set $A \subset \mathbf{M}^0$ is a *hidden deadlock* if

- (1) $A \cap \mathbb{F} = \emptyset$.
- (2) A is a deadlock for the FAN $\widehat{\mathbf{N}}$.

Remarks 8.19. (1) Condition (2) of the definition is equivalent to there existing $(\mathbf{X}, \mathbf{T}) \in \widehat{\mathbb{I}}$ and $t > 0$ such that $\widehat{\Phi}((\mathbf{X}, \mathbf{T}), t) \in A$. Note that by the Φ -invariance of A , $\pi_i(A) \cap \mathbb{I}_i = \emptyset$, all $i \in \mathbf{k}$. Hence, once a $\widehat{\Phi}$ -trajectory has entered A , the subsequent evolution of the nodes is given by Φ and so the $\widehat{\Phi}$ -trajectory cannot leave A .

(2) Since $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ is assumed weakly regular, a hidden deadlock can never be a deadlock of \mathbf{N} .

In general terms, deadlocks are obstructions to weak regularity; hidden deadlocks are obstructions to regularity.

Examples 8.20. (1) A regular FAN has no hidden deadlocks.

(2) Referring to figure 12, consider two trains travelling in the opposite direction on a line which contains a single track segment $[-0.5, 0.5]$ and stations at ± 2 . We associate a collision event with the single track segment which results in both trains stopping. Train \mathfrak{T}_1 starts at $\{-2\} = \mathbb{I}_1$ and proceeds with velocity 0.5; train \mathfrak{T}_2 starts at $\{2\} = \mathbb{I}_2$ and proceeds with velocity -2.0 . The termination sets are $\mathbb{F}_1 = \{2\}$, $\mathbb{F}_2 = \{-2\}$. The trajectory γ of figure 12, represents the case where both trains start at time $t = 0$. At $t = 1.25$, \mathfrak{T}_1 is at point -1.375 , and \mathfrak{T}_2 is at $-0.5 > -1.375$. Hence there is no collision (deadlock) and network function is achieved. On the other hand if \mathfrak{T}_2 starts at time $T_2 = 3$ and \mathfrak{T}_1 at time $T_1 = 0$, the trains will collide at the origin at time $t = 4$ (the trajectory ν of figure 12) and the network is not regular. \diamond

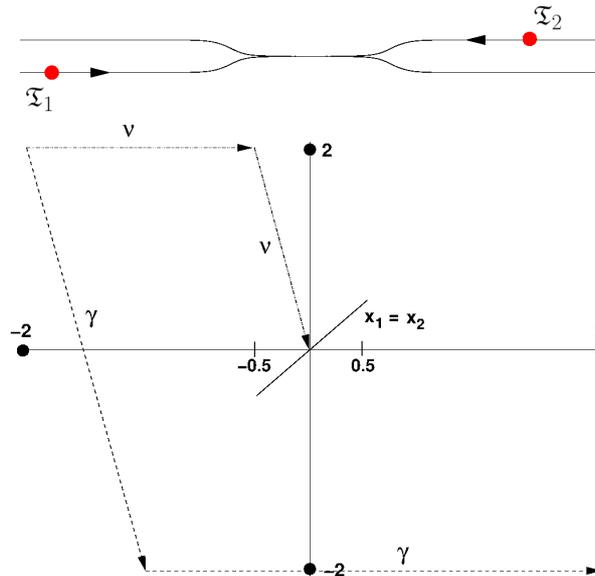


FIGURE 12. Two trains on a partially single track line.

8.5. Weakly regular FANs of simple type. The definitions of regularity and weak regularity for a FAN involve many details not directly related to the construction of the generalized transition function. We show that given a weakly regular FAN \mathbf{N} , we can construct a simpler variant \mathbf{N}^c which has the same generalized transition function. Roughly speaking, we are making the transition from viewing the FAN as a (possible) subnetwork of a larger network (relative viewpoint) to an absolute viewpoint (the FAN is not contained in a larger network).

Definition 8.21. A geometric FAN $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ is of *simple type* if for all $i \in \mathbf{k}$, we can choose open neighbourhoods V_i of ∂M_i^0 such that

$$(S1)^c \quad \pi_i(E_i) \subset M_i^0 \setminus V_i.$$

(S2)^c (a) The event map $\mathcal{E}(\mathbf{x}_1, \dots, \mathbf{x}_k)$ is locally constant as a function of $\mathbf{x}_i \in M_i^\sigma \cup V_i$, where $\sigma \in \{+, -\}$.

(b) For all $i \in \mathbf{k}$, $f_i^{\mathcal{E}(\mathbf{X})}(\mathbf{X}) = Z_i(\mathbf{x}_i)$ if $\mathbf{x}_i \in M_i^- \cup M_i^+ \cup V_i$.

Remark 8.22. Condition (S2)^c relates to the concepts of structural decomposability and elementary type (see section 5). If (S2a)^c holds, there are no dependencies of the event map on the state of a node N_i once its state has exited M_i^0 .

Lemma 8.23. *If the FAN $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ is of simple type, then \mathbf{N} is weakly regular. We say \mathbf{N} is weakly regular of simple type. If, in addition, \mathbf{N} is regular, we say \mathbf{N} is regular of simple type.*

Proof. Obviously $(S1)^c$ implies $(S1)$ and $(S1,S2)^c$ imply $(S2)$. \square

We need some new notation before stating our next result. Suppose that $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ is a weakly regular FAN. For $\mathbf{X} \in \mathbf{M}$, define mutually disjoint subsets $A^0(\mathbf{X}), A^\pm(\mathbf{X})$ of \mathbf{k} by

$$\begin{aligned} A^0(\mathbf{X}) &= \{i \in \mathbf{k} \mid \mathbf{x}_i \in M_i^0 \setminus \partial M_i^0\}, \\ A^-(\mathbf{X}) &= \{i \in \mathbf{k} \mid \mathbf{x}_i \in M_i^-\}, \\ A^+(\mathbf{X}) &= \{i \in \mathbf{k} \mid \mathbf{x}_i \in M_i^+\}. \end{aligned}$$

We have $\mathbf{k} = A^0(\mathbf{X}) \cup A^-(\mathbf{X}) \cup A^+(\mathbf{X})$ for all $\mathbf{X} \in \mathbf{M}$.

For each $i \in \mathbf{k}$, fix $\mathbf{x}_i^- \in \mathbb{I}_i$, and $\mathbf{x}_i^+ \in \mathbb{F}_i$. Given $\mathbf{X} \in \mathbf{M}$, define $\tilde{\mathbf{X}} \in \mathbf{M}$ by

$$\tilde{\mathbf{x}}_i = \begin{cases} \mathbf{x}_i^-, & \text{if } i \in A^-(\mathbf{X}), \\ \mathbf{x}_i, & \text{if } i \in A^0(\mathbf{X}), \\ \mathbf{x}_i^+, & \text{if } i \in A^+(\mathbf{X}). \end{cases}$$

Define the event map $\mathcal{E}^c : \mathbf{M} \rightarrow \mathcal{A}$ by

$$\mathcal{E}^c(\mathbf{X}) = \mathcal{E}(\tilde{\mathbf{X}}), \quad \mathbf{X} \in \mathbf{M}.$$

Define $\mathcal{A}^c = \{\mathcal{E}^c(\mathbf{X}) \mid \mathbf{X} \in \mathbf{M}\} \subset \mathcal{A}$ and $\mathcal{F}^c = \{\mathbf{f}^\alpha \mid \alpha \in \mathcal{A}^c\} \subset \mathcal{F}$. Note that, by weak regularity, \mathcal{E}^c , and hence \mathcal{A}^c and \mathcal{F}^c , do not depend on the specific choice of \mathbf{x}_i^\pm used for the definition of $\tilde{\mathbf{X}}$. Let \mathfrak{N}^c denote the asynchronous network $(\mathfrak{N}, \mathcal{A}^c, \mathcal{F}^c, \mathcal{E}^c)$.

Theorem 8.24. *The asynchronous network \mathfrak{N}^c is proper and $\mathbf{N}^c = (\mathfrak{N}^c, \mathbb{I}, \mathbb{F})$ is a weakly regular FAN of simple type. If we denote the evolution operator of \mathbf{N}^c by $\hat{\Phi}^c$ then for all $((\mathbf{X}, \mathbf{T}), t) \in \hat{\mathbb{I}} \times \mathbb{R}_+$ we have*

$$(15) \quad \hat{\Phi}_i^c((\mathbf{X}, \mathbf{T}), t) = \hat{\Phi}_i((\mathbf{X}, \mathbf{T}), t), \text{ if } \hat{\Phi}_i((\mathbf{X}, \mathbf{T}), t) \in M_i^0.$$

If \mathbf{N} is regular then so is \mathbf{N}^c and both FANs have the same generalized transition function.

Proof. The properness of \mathfrak{N}^c , weak regularity of \mathbf{N}^c and (15) are immediate from the construction of \mathbf{N}^c . The remaining statements follow from (15). \square

Definition 8.25. We refer to the FAN $\mathbf{N}^c = (\mathfrak{N}^c, \mathbb{I}, \mathbb{F})$ given by theorem 8.24 as the *core* of $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ and write $\mathbf{N}^c = \text{core}(\mathbf{N})$.

We conclude this section with some additional remarks and comments about the conditions of definitions 8.1 and 8.7.

Remarks 8.26. (1) In the sequel it will sometimes be convenient to take $\mathbb{I}_i = \mathbb{F}_i$ for some indices i . We then have $M_i^0 = \mathbb{I}_i = \mathbb{F}_i$ and require that the vector field Z_i point from M_i^- to M_i^+ .

(2) Typically, different choices of initialization and termination sets satisfying (G,T,F) will be isotopic (by the flow of the uncoupled node). Thus, if $\mathbb{I}_i, \mathbb{I}'_i$ are initialization sets for M_i , there will exist a smooth map $\xi : \mathbb{I}_i \rightarrow \mathbb{R}$ such that $\mathbb{I}'_i = \{\psi_i^{\xi(x)}(x) \mid x \in \mathbb{I}_i\}$. Similarly for the termination hypersurfaces \mathbb{F}_i . We use this property later.

(3) We do not require that the flow ψ_i^t maps \mathbb{I}_i to \mathbb{F}_i – for example, the vector fields Z_i may have equilibria in M_i^0 . However, interactions with other nodes will then be needed for the state of N_i to reach the termination set \mathbb{F}_i . If N_i is never coupled to other nodes, then (F) implies that ψ_i^t maps \mathbb{I}_i to \mathbb{F}_i .

(4) We assumed for definitions 8.1, 8.7 that $\partial M_i = \emptyset$. It is not hard to remove this restriction. We briefly sketch the details. First, in definition 8.7, we need to assume $V_i \supset \partial M_i$, $i \in \mathbf{k}$ – this is automatic if $\partial M_i = \mathbb{I}_i \cup \mathbb{F}_i$. Next add an open collar [62, Chapter 4, §6] C^i to ∂M_i and thereby enlarge M_i to $M_i^c = M_i \cup C^i$. We do this so that M_i^c is without boundary (necessarily M_i^c is not compact). Extend Z_i to M_i^c so that Z_i has a well defined flow ψ_i and then trivially extend event maps to $\mathbf{M}^c = \prod_{i \in \mathbf{k}} M_i^c$ so that we can take $V_i \subset C^i$ in definition 8.7.

(5) In remark 8.4, we commented on the possibility of assuming that nodes could interact if and only if they were constrained – that is, in a waiting state. This is often natural when we consider regularity and allow for generalized initialization times. If regularity holds but nodes are not constrained, then nodes N_i, N_j can only interact if their starting times T_i, T_j lie in a narrow range – that is, for interaction at time t , we need $\widehat{\Phi}_i((\mathbf{X}, \mathbf{T}), t) \in M_i^0 \setminus V_i$ and $\widehat{\Phi}_j((\mathbf{X}, \mathbf{T}), t) \in M_j^0 \setminus V_j$. If $|T_i - T_j|$, is sufficiently large there will be no interaction.

8.6. Concatenating regular networks of simple type. Assume given a fixed set of nodes $\mathcal{N} = \{N_0, N_1, \dots, N_k\}$ and network phase space $\mathbf{M} = \prod_{i \in \mathbf{k}} M_i$. Suppose that $\mathbf{N}^a = (\mathfrak{N}^a, \mathbb{I}^a, \mathbb{F}^a)$, $a \in \mathbf{2}$, are regular FANs of simple type. Following definition 8.7, we label so that $M_i = M_i^{a,-} \cup M_i^{a,0} \cup M_i^{a,+}$, where $M_i^{a,-} \cap M_i^{a,0} = \mathbb{I}_i^a$, and $M_i^{a,0} \cap M_i^{a,+} = \mathbb{F}_i^a$, $i \in \mathbf{k}$, $a \in \mathbf{2}$.

If $\alpha \in \mathcal{A}^a$, label the associated admissible vector field by $\mathbf{f}^{a,\alpha}$, $a \in \mathbf{2}$. For $a \in \mathbf{2}$, let $E^{a,*}$ denote the union of the event sets E_i^a for $i \in \mathbf{k}$ and E_α^a be the event set corresponding to $\alpha \in \mathcal{A}^a$, $a \in \mathbf{2}$.

Lemma 8.27. (*Notation and assumptions as above.*) Suppose that

$$(1) \mathbf{f}^{1,\emptyset} = \mathbf{f}^{2,\emptyset} \stackrel{\text{def}}{=} \mathbf{f}^\emptyset.$$

$$(2) \mathbb{F}^1 = \mathbb{I}^2,$$

If $\alpha \in \mathcal{A}^1 \cap \mathcal{A}^2$, then we may choose an admissible vector field \mathbf{f}^α such that $\mathbf{f}^\alpha|_{E^{a,\alpha}} = \mathbf{f}^{a,\alpha}$, $a \in \mathbf{2}$.

Proof. By (S1)^c, there is a connection $N_j \rightarrow N_i$ at $(\mathbf{x}_1, \dots, \mathbf{x}_k) \in \mathbf{M}$ only if we can choose $a \in \mathbf{2}$, such that $\mathbf{x}_i \in M_i^{a,0}$, $\mathbf{x}_j \in M_j^{a,0}$. It follows that if $\alpha \in \mathcal{A}^1 \cap \mathcal{A}^2$, then $\overline{E^{1,\alpha}} \cap \overline{E^{2,\alpha}} = \emptyset$. Now define \mathbf{f}^α by extending $\mathbf{f}^{1,\alpha}|_{\overline{E^{1,\alpha}}}$, $\mathbf{f}^{2,\alpha}|_{\overline{E^{2,\alpha}}}$ smoothly to \mathbf{M} . \square

If the conditions of lemma 8.27 hold, then $\{\mathfrak{N}^1, \mathfrak{N}^2\}$ is compatible and we let $\mathfrak{N} = \bigsqcup_{a \in \mathbf{2}} \mathfrak{N}^a$ be the amalgamation as defined in definition 5.26.

Proposition 8.28. *(Notation and assumptions as above.) Suppose that $\mathbf{N}^a = (\mathfrak{N}^a, \mathbb{I}^a, \mathbb{F}^a)$, $a \in \mathbf{2}$, are regular FANs of simple type satisfying*

- (1) $\mathfrak{N}^a = (\mathcal{N}, \mathcal{A}^a, \mathcal{F}^a, \mathcal{E}^a)$, $a \in \mathbf{2}$.
- (2) $\mathbf{f}^{a,\emptyset} = \mathbf{f}^\emptyset$, $a \in \mathbf{2}$.
- (3) $\mathbb{F}^1 = \mathbb{I}^2$,

We have a well-defined regular FAN $\mathbf{N}^2 \diamond \mathbf{N}^1 = (\mathfrak{N}, \mathbb{I}^1, \mathbb{F}^2)$, where $\mathfrak{N} = \bigsqcup_{a \in \mathbf{2}} \mathfrak{N}^a = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$. Moreover, the generalized transition function G for $\mathbf{N}^2 \diamond \mathbf{N}^1$ is the composite $G^2 \circ G^1$. We call $\mathbf{N}^2 \diamond \mathbf{N}^1$ the concatenation of \mathbf{N}^1 and \mathbf{N}^2 .

Proof. It is immediate from the definitions that $(\mathfrak{N}, \mathbb{I}^1, \mathbb{F}^2)$ satisfies the conditions for a weakly regular FAN of simple type and so has an evolution operator $\widehat{\Phi}$. It remains to be shown that $\widehat{\Phi}$ can be written in terms of the evolution operators $\widehat{\Phi}^a$ of \mathbf{N}^a , $a \in \mathbf{2}$. Given $((\mathbf{X}, \mathbf{T}), t) \in \widehat{\mathbb{I}} \times \mathbb{R}_+$, let $(\mathbf{Y}, \widehat{\mathbf{S}}^1) = G^1(\mathbf{X}, \mathbf{T})$, $\widehat{\mathbf{S}}^1 = (\widehat{S}_1^1, \dots, \widehat{S}_k^1)$. Since $\mathbf{N}^1, \mathbf{N}^2$ are of simple type, it follows that for all for all $i \in \mathbf{k}$ we have

$$\widehat{\Phi}_i((\mathbf{X}, \mathbf{T}), t) = \begin{cases} \widehat{\Phi}_i^1((\mathbf{X}, \mathbf{T}), t), & t \leq \widehat{S}_i^1 \\ \widehat{\Phi}_i^2((\mathbf{Y}, \widehat{\mathbf{S}}^1), t), & t \geq \widehat{S}_i^1. \end{cases}$$

Hence $G = G^2 \circ G^1$. \square

Remarks 8.29. (1) We emphasize that proposition 8.28 requires conditions (S1)^c and (S2)^c. In particular, if (S2)^c fails, the concatenation may not be proper even though the amalgamation $\bigsqcup_{a \in \mathbf{2}} \mathfrak{N}^a$ is well defined as an asynchronous network.

(2) The operation of concatenation is *not* commutative. Even if we redefine the initialization and termination sets so as to make $\mathbf{N}^1 \diamond \mathbf{N}^2$ well defined, we generally have $\mathbf{N}^2 \diamond \mathbf{N}^1 \neq \mathbf{N}^1 \diamond \mathbf{N}^2$.

(3) As constructed, $\mathbf{N} = \mathbf{N}^1 \diamond \mathbf{N}^2$ is of simple type. If \mathbf{N} is regular but not of simple type, we can replace \mathbf{N} by $\text{core}(\mathbf{N})$. It may then be possible to write $\text{core}(\mathbf{N}) = \mathbf{N}^1 \diamond \mathbf{N}^2$, where $\mathbf{N}^1, \mathbf{N}^2$ are of simple type.

This suffices for applications to the original FAN \mathbf{N} as \mathbf{N} and $\text{core}(\mathbf{N})$ have the same generalized transition function by theorem 8.24. Note also that if $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$, and $\text{core}(\mathbf{N}) = \mathbf{N}^1 \diamond \mathbf{N}^2$, then for $a \in \mathbf{2}$, the FAN $\tilde{\mathbf{N}}^a = (\mathfrak{N}, \mathbb{I}^a, \mathbb{F}^a)$ will be regular, though not of simple type, and have transition function G^a .

8.7. Forced stopping. Let $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ be a weakly regular FAN with associated timing function $\mathbf{S} = (S_1, \dots, S_k) : \mathbb{I} \rightarrow \mathbb{R}_+$. In this section we address the issue of *forced stopping* raised in section 7.3. The results of this section are not used elsewhere in the paper and may be omitted at first reading.

We first construct a new FAN $\mathbf{N}^* = (\mathfrak{N}^*, \mathbb{I}, \mathbb{F})$ that provides a simple minimal model for the network function of \mathbf{N} and for which nodes are stopped when they reach their terminal state.

We start by formalizing the forced stopping of nodes (this can easily be framed in terms of an explicit stopping constraint). Let A be a nonempty subset of \mathbf{k} and ξ_A be the connection structure $\{N_0 \rightarrow N_i \mid i \in A\}$. Define $\mathcal{A}^s = \mathcal{A} \cup \{\xi_A \mid A \subset \mathbf{k}\}$. We define a new event map $\mathcal{E}^s : \mathbf{M} \rightarrow \mathcal{A}^s$. Let $\mathbf{X} \in \mathbf{M}$. If $\mathbf{x}_i \in M_i^+$, $i \in A$, and $\mathbf{x}_i \notin M_i^+$, $i \notin A$, define

$$\mathcal{E}^s(\mathbf{X}) = \mathcal{E}(\mathbf{X}) \vee \xi_A.$$

For each $\alpha^s = \alpha \vee \xi_A \in \mathcal{A}^s$, define $\mathbf{f}^{\alpha^s} = (f_1, \dots, f_k)$ by

$$f_i(\mathbf{X}) = \begin{cases} 0, & \text{if } i \in A \\ f_i^\alpha(\mathbf{X}), & \text{if } i \notin A. \end{cases}$$

Define $\mathcal{F}^s = \{\mathbf{f}^{\alpha^s} \mid \alpha^s \in \mathcal{A}^s\}$.

If our focus is purely on network function (of the network or subnetwork), then dynamics of the node N_i on the factors $M_i^- \setminus \mathbb{I}_i$, $M_i^+ \setminus \mathbb{F}_i$ is not of interest. We define the asynchronous network $\mathfrak{N}^* = (\mathcal{N}^*, \mathcal{A}^*, \mathcal{F}^*, \mathcal{E}^*)$ by requiring that \mathfrak{N}^* and \mathfrak{N} have the same node sets ($\mathcal{N}^* = \mathcal{N}$) but take the network phase space of \mathfrak{N}^* to be $\mathbf{M}^0 = \prod_{i \in \mathbf{k}} M_i^0$. Define \mathcal{E}^* by restriction of \mathcal{E}^s to \mathbf{M}^0 and set $\mathcal{A}^* = \mathcal{A}^s$ (equal to $\mathcal{E}^*(\mathbf{M}^0)$), and $\mathcal{F}^* = \{\mathbf{f}^\alpha \mid \mathbf{M}^0 \mid \alpha \in \mathcal{A}^*\}$.

Proposition 8.30. *(Notation and assumptions as above.) The asynchronous network \mathfrak{N}^* is proper, with well-defined semiflow $\Phi^* : \mathbf{M}^0 \times \mathbb{R}_+ \rightarrow \mathbf{M}^0$ and for all $\mathbf{X} \in \mathbf{M}^0$, $i \in \mathbf{k}$, we have*

$$(16) \quad \Phi_i^*(\mathbf{X}, t) = \begin{cases} \Phi_i(\mathbf{X}, t), & \text{provided } t \leq S_i, \\ \Phi_i(\mathbf{X}, S_i), & \text{if } t \geq S_i. \end{cases}$$

In particular,

- (1) \mathbf{N}^* is a FAN, that is, \mathbb{I}, \mathbb{F} satisfy conditions (P1–3) for \mathfrak{N}^* .

(2) \mathbf{N} and \mathbf{N}^* have the same transition and timing functions.

Proof. A routine computation based, as usual, on lemma 8.9. \square

Remarks 8.31. (1) Proposition 8.30 may fail if \mathbf{N} does not satisfy the structural conditions (S1,S2) for weak regularity. Indeed, $\mathfrak{N}^* = (\mathcal{N}^*, \mathcal{A}^*, \mathcal{F}^*, \mathcal{E}^*)$ may not be proper; even if it is, (16) may fail.

(2) In many cases, \mathbf{N}^* gives a natural description of network function. For subnetworks, we need to take account of the global structure of the containing network – not just the network structure determined by restriction. Proposition 8.30 allows us to view both subnetwork and network as FANs where stopping occurs when termination sets are reached.

Proposition 8.30 shows that for a weakly regular FAN $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$, stopping nodes at termination does not change network function. This property also holds for generalized initialization.

Proposition 8.32. *Let $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ be weakly regular. If we let $\widehat{\Phi}^*$ denote the evolution operator for \mathbf{N}^* , then*

(1) $\widehat{\Phi}^* : \widehat{\mathbb{I}} \times \mathbb{R}_+ \rightarrow \mathbf{M}^0$ is well defined and continuous in forward time.

(2) For all $i \in \mathbf{k}$, $((\mathbf{X}, \mathbf{T}), t) \in \widehat{\mathbb{I}} \times \mathbb{R}_+$, we have

$$\widehat{\Phi}_i^*((\mathbf{X}, \mathbf{T}), t) = \widehat{\Phi}_i((\mathbf{X}, \mathbf{T}), t), \text{ if } \widehat{\Phi}_i((\mathbf{X}, \mathbf{T}), t) \in M_i^0.$$

(3) If we denote the timing operator for $\widehat{\Phi}^*$ by $\widehat{\mathbf{S}}^*$, then $\widehat{\mathbf{S}} = \widehat{\mathbf{S}}^*$ and the operators have common domain $\widehat{\mathcal{D}}(\mathbf{N})$.

If \mathbf{N} is regular, then \mathbf{N} and \mathbf{N}^* have identical generalized transition and timing functions.

Proof. Another application of lemma 8.9. \square

Remarks 8.33. (1) For weakly regular FANs with generalized initialization, stopping of nodes upon completion has no effect on the dynamics of the other nodes whose states are in M_i^0 . As a result, the evolution operator $\widehat{\Phi}^*$ suffices for the analysis of network function even if nodes may not terminate for some generalized initial conditions.

(2) If \mathbf{N} is regular, every point $\widehat{\mathbb{I}}$ is $\widehat{\Phi}$ -connected to $\widehat{\mathbb{F}}$ and the $\widehat{\Phi}^*$ -trajectory of every point in $\widehat{\mathbb{I}}$ meets $\widehat{\mathbb{F}}$ (note our use of the notation $\widehat{\mathbf{N}} = (\mathfrak{N}, \widehat{\mathbb{I}}, \widehat{\mathbb{F}})$ for a weakly regular FAN, remark 8.16).

9. A FACTORIZATION THEOREM FOR A CLASS OF FANs

In this section we revisit the classical reductionist approach of analysing a network in terms of its individual components. We do this for a class

of feed-forward FANs that are built from regular FANs of simple type. Our main result gives a spatiotemporal factorization of the dynamics of FANs in this class in terms of the dynamics of the component FANs. In particular, we generalise proposition 8.28 and show the generalized transition function can be written as a composition of the generalized transition functions of the component constituent FANs. The result allows us to break up the analysis of dynamics of a FAN into a study of dynamics of the component FANs together with the analysis of transitions between the components – a mix of analysis, algebra and numerics.

For the remainder of this section we assume all regular FANs are of simple type. This will be no loss of generality for our main application as, by theorem 8.24, we can always replace a regular FAN \mathbf{N} by $\text{core}(\mathbf{N})$.

9.1. Primitive, stably primitive, trivial and basic FANs. We easily extend the definition of a product of asynchronous networks given in section 5 to FANs. Specifically, if $\mathbf{N}^a = (\mathfrak{N}^a, \mathbb{I}^a, \mathbb{F}^a)$, $a \in \mathbf{q}$, are FANs (with disjoint node sets), we define

$$\prod_{a \in \mathbf{q}} \mathbf{N}^a = \left(\prod_{a \in \mathbf{q}} \mathfrak{N}^a, \prod_{a \in \mathbf{q}} \mathbb{I}^a, \prod_{a \in \mathbf{q}} \mathbb{F}^a \right).$$

We may now define our basic building blocks.

Definition 9.1. Let $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ be a regular FAN with p nodes.

- (1) \mathbf{N} is *trivial* if $\mathbf{N} = \prod_{\ell \in \mathbf{p}} \mathbf{S}^\ell$, where \mathbf{S}^ℓ has one node for all $\ell \in \mathbf{p}$.
- (2) \mathbf{N} is *primitive* if $p > 1$ and
 - (a) \mathbf{N} is indecomposable.
 - (b) \mathbf{N} cannot be written as a concatenation of two regular FANs.
- (3) \mathbf{N} is *stably primitive* if $p \geq 3$ and there is a primitive FAN \mathbf{P} and a trivial FAN $\mathbf{S} = \prod_{\ell \in \mathbf{s}} \mathbf{S}^\ell$ such that $\mathbf{N} = \mathbf{P} \times \mathbf{S}$.

We call a trivial, primitive or stably primitive FAN *basic*. If \mathbf{N} is a basic FAN, let \mathbf{P} denote the primitive factor (if there is one), and $V(\mathbf{P})$ denote the set of nodes associated with the primitive factor.

Remark 9.2. Let \mathbf{N} be a stably primitive FAN with p nodes and primitive factor \mathbf{P} . The generalized connection structure \mathcal{A} for \mathbf{N} consists of at least one nontrivial connection structure. However, there may be nodes N_i , $i \in \mathbf{P}$, that are never linked: $i \notin V(\mathcal{A})$. That is, $V(\mathcal{A})$ may be a proper subset of $V(\mathbf{P})$. Since \mathbf{P} is indecomposable, this means that certain values of $\mathbf{x}_i \in M_i^0$ may result in connections between other nodes being switched on or off. In particular, a stably primitive FAN

need not be of elementary type (definition 5.18) though results on independence of stably primitive FANs along the lines of lemma 5.21 hold (see remarks 5.19(3)).

Let $\Lambda = \{\mathbf{N}^a = (\mathfrak{N}^a, \mathbb{I}^a, \mathbb{F}^a) \mid a \in \mathbf{q}\}$ be a compatible family of basic FANs sharing a common set of k nodes \mathcal{N} and network phase space \mathbf{M} . It follows from section 5.4 that the amalgamation $\mathfrak{N} = \bigsqcup_{a \in \mathbf{q}} \mathfrak{N}^a$ is a well defined asynchronous network.

For each $i \in \mathbf{k}$, define $\mathbf{q}(i) = \{a \in \mathbf{q} \mid i \in V(\mathcal{A}^a)\}$ and let $k(i) \in \mathbf{k}^\bullet$ be the number of elements in $\mathbf{q}(i)$. If N_i is never linked for any of the FANs \mathbf{N}^a , then $k(i) = 0$ is zero.

Definition 9.3. The family Λ is *spatiotemporally ordered* if there exist an initialization set \mathbb{I} , termination set \mathbb{F} , and a partial order \prec on \mathbf{q} inducing a total order $<_i$ on $\mathbf{q}(i)$ for all $i \in \mathbf{k}$, such that if $a_1^i <_i \dots <_i a_{k(i)}^i$ is the ordering of $\mathbf{q}(i)$ induced by \prec , then

- (1) $\mathbb{F}_i^{a_j^i} = \mathbb{I}_i^{a_{j+1}^i}$, $1 \leq j < k(i)$.
- (2) $\mathbb{I} = \prod_{i \in \mathbf{k}} \mathbb{I}_i$ satisfies $\mathbb{I}_i^{a_1^i} = \mathbb{I}_i$ if $k(i) > 1$, and \mathbb{I}_i is an initialization set for nodes N_i , with $k(i) \leq 1$.
- (3) $\mathbb{F} = \prod_{i \in \mathbf{k}} \mathbb{F}_i$ satisfies $\mathbb{F}_i^{a_{k(i)}^i} = \mathbb{F}_i$ if $k(i) > 1$, and \mathbb{F}_i is a termination set for nodes N_i , with $k(i) \leq 1$.

Remarks 9.4. (1) If $i, j \in V(\mathcal{A}^a) \cap V(\mathcal{A}^b)$, then $a, b \in \mathbf{q}(i) \cap \mathbf{q}(j)$ and $a <_i b$ if and only if $a <_j b$. However, a, b may be adjacent in $\mathbf{q}(i)$ but not adjacent in $\mathbf{q}(j)$ – there may exist $c \in \mathbf{q}(j) \setminus \mathbf{q}(i)$ such that $a <_j c <_j b$.

(2) The way to think of definition 9.3 is that there is a given FAN $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ such that \mathfrak{N} is the amalgamation of a compatible family $\Lambda = \{\mathbf{N}^a \mid a \in \mathbf{q}\}$. Under suitable conditions, it may be possible to change the initial and termination sets of the components of Λ so as to satisfy the requirements of definition 9.3. This will entail, for example, setting all the initial sets \mathbb{I}_i^a to \mathbb{I}_i , and all of the terminating sets \mathbb{F}_i^a to \mathbb{F}_i , if N_i is a trivial factor for all of the components of Λ . In practice, this is often easy using remarks 8.26(1,2).

We illustrate the definition of spatiotemporal ordering with some examples.

Examples 9.5. (1) Suppose we are given 6 nodes and a compatible family of three stably primitive FANs \mathbf{N}^a , with corresponding primitive factors \mathbf{P}^a , $a \in \mathbf{3}$. We take the amalgamation $\mathbf{N} = \bigsqcup_{a \in \mathbf{3}} \mathbf{N}^a$ and a choice \mathbb{I}, \mathbb{F} of initialization and termination set for \mathbf{N} . Referring to figure 13, we have shown initialization and termination sets node-by-node

as well as the nodes associated to each primitive component \mathbf{P}^a . For example, nodes N_2, N_3, N_4 are associated with \mathbf{P}^1 and the remaining nodes are part of the trivial factor for \mathbf{N}^1 .

For this example, the partial order on $\mathbf{3} = \mathbf{3}(3)$ is the usual total order $1 < 2 < 3$.

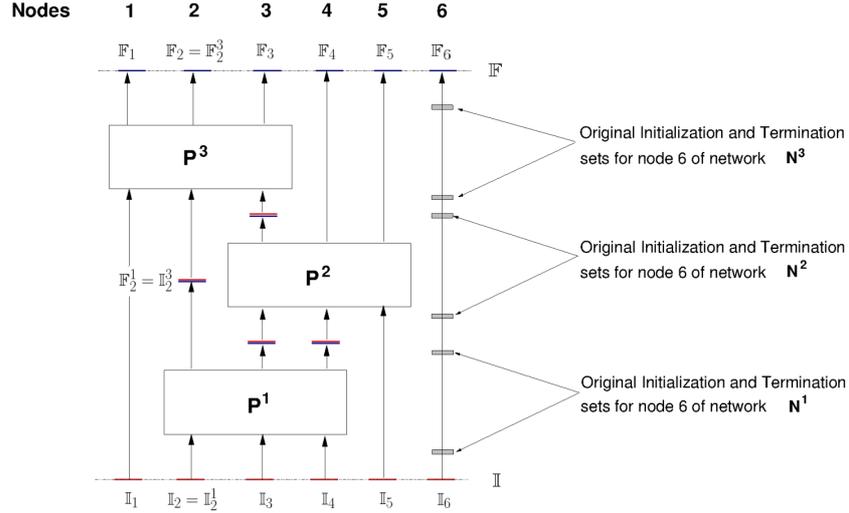


FIGURE 13. A spatiotemporally ordered FAN with 6 nodes.

Observe that node N_6 corresponds to a trivial factor for \mathbf{N}^a , $a \in \mathbf{3}$. Provided that the uncoupled dynamics on M_6 provide an isotopy from \mathbb{I}_6 to \mathbb{I}_6^a and \mathbb{F}_6 to \mathbb{F}_6^a , all $a \in \mathbf{3}$ (see remarks 8.26(2)), we can take $\mathbb{I}_6^a = \mathbb{I}_6$, $\mathbb{F}_6^a = \mathbb{F}_6$, $a \in \mathbf{3}$. We refer to figure 13 where we have indicated initial choices for initialization and termination sets for the components. Of course, if say $\mathbb{I}_6^1 = \mathbb{I}_6$, $\mathbb{F}_6^1 = \mathbb{F}_6$ or $\bigcup_a M_{a,i}^0$ covers M_6^0 , we only have to ensure that $\mathbb{I}_6, \mathbb{F}_6$ are hypersurfaces transverse to the flow on M_6 . We apply the same logic for node N_5 . Here node N_5 corresponds to a trivial factor of \mathbf{N}^1 and \mathbf{N}^3 . Now we want to take $\mathbb{I}_5^2 = \mathbb{I}_5$, $\mathbb{F}_5^2 = \mathbb{F}_5$. We can ignore the initialization sets for the remaining nodes. The same approach applies to node N_1 . Finally, we consider the most complicated case: node N_3 . Here to achieve the spatiotemporal order, we need to choose initialization and termination sets so that (1) \mathbf{N}^a , $a \in \mathbf{3}$ satisfy regularity with these choices, and (2) $\text{Int}(M_{a,3}^0) \cap \text{Int}(M_{b,3}^0) = \emptyset$, $a \neq b$. We match according to the rules: $\mathbb{I}_3^1 = \mathbb{I}_3$, $\mathbb{F}_3^1 = \mathbb{I}_3^2$, $\mathbb{F}_3^2 = \mathbb{I}_3^3$, $\mathbb{F}_3^3 = \mathbb{F}_3$.

Notice that changing the initialization and termination sets does not affect compatibility. However, we need to be careful about regularity

if we shrink any of the domains $M_{a,i}^0$.

(2) For our second example, we consider an event sequence for a 9 node FAN built from stably primitive FANs, $\mathbf{N}^a, \dots, \mathbf{N}^h$. We assume the components comprise a spatiotemporally ordered compatible family. Let $\mathbf{q} = \{a, \dots, h\} \approx \mathbf{8}$. Just as in our first example, we only indicate primitive components of FANs in the diagram. The schematic indicates that \mathbf{N}^a and \mathbf{N}^b are independent whereas \mathbf{N}^g depends on both \mathbf{N}^a and \mathbf{N}^b .

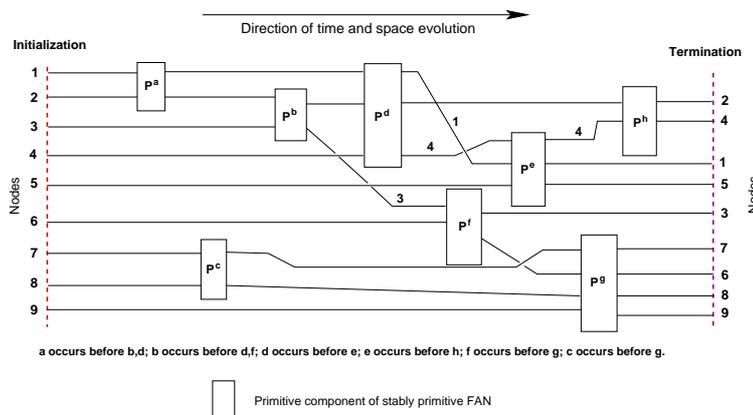


FIGURE 14. Event sequence for a 9 node FAN.

The partial order on \mathbf{q} is uniquely determined by the relations

$$a \prec b, d; b \prec d, f; d \prec e, h; e \prec h; f \prec g; c \prec g.$$

No relation holds between a and c or d and e . This implies that nodes N_1, N_2 can interact according to \mathbf{N}^a before or after nodes N_7, N_8 interact according to \mathbf{N}^c . Similar comments hold for N_1, N_4, N_5 and N_3, N_6 in relation to \mathbf{N}^d and \mathbf{N}^e .

The total order induced on $\mathbf{8}(2)$ is $a <_2 b <_2 d <_2 h$. Similar computations hold for remaining nodes.

(3) For our final example, we show a compatible family of 3 basic FANs which is not spatiotemporally ordered. Referring to figure 15, observe that $\mathbf{q}(2)$ has the ordering $b <_2 c <_2 a$ which is incompatible with the ordering $a <_1 b$ of $\mathbf{q}(1)$. Hence there exists no partial order of \mathbf{q} which induces the correct orders on $\mathbf{q}(1), \mathbf{q}(2)$. Spatiotemporal ordering implies that there are no ‘feedback’ loops between distinct basic sets: each node has at most one bite at an ‘event cherry’.

◇

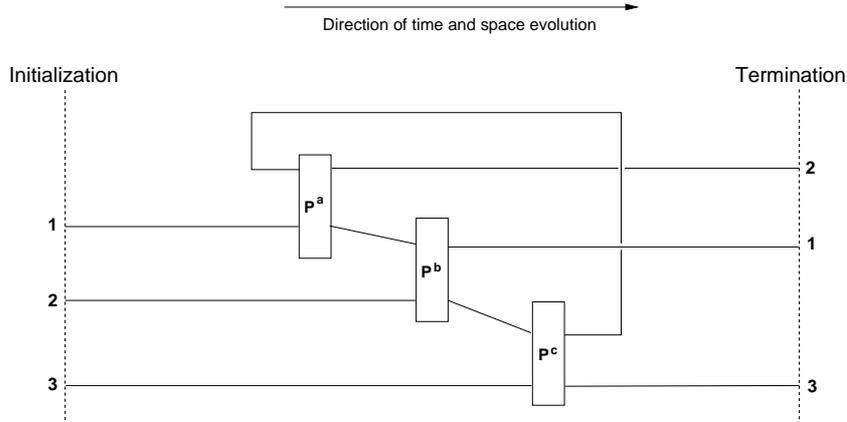


FIGURE 15. Breakdown of spatiotemporal ordering in a 3 node FAN

9.2. FANs of feedforward type and factorization.

Definition 9.6. The FAN $(\mathfrak{N}, \mathbb{I}, \mathbb{F})$ is of *feedforward type* if there exists a spatiotemporally ordered compatible family $\Lambda = \{\mathbf{N}^a \mid a \in \mathbf{q}\}$ of basic FANs, sharing a common node set \mathcal{N} and network phase space \mathbf{M} , such that $\mathfrak{N} = \bigsqcup_{a \in \mathbf{q}} \mathfrak{N}^a$ (the initialization and termination sets $\mathbb{I}, \mathbb{F}, \mathbb{I}^a, \mathbb{F}^a$, $a \in \mathbf{q}$, satisfy the conditions of definition 9.3).

Proposition 9.7. *If the FAN \mathbf{N} is of feedforward type, then \mathbf{N} is regular and has a well-defined generalized transition function. In particular, \mathbf{N} is proper (and amenable, if the components of the defining family Λ are all amenable).*

Proof. It suffices to prove that \mathbf{N} has a generalized transition function since \mathbf{N} obviously satisfies the conditions for weak regularity except possibly condition (F). We prove by induction on q . The result is obviously true if $q = 1$. Suppose we have proved the result for $q - 1$, $q > 1$. Pick a minimal element a_1 of \mathbf{q} with respect to the partial order \prec . We define initialization and termination sets $\mathbb{I}(1)$ and $\mathbb{F}(1)$ for \mathbf{N}^{a_1} by $\mathbb{I}(1) = \mathbb{I}$, and

$$\mathbb{F}(1)_i = \begin{cases} \mathbb{I}_i, & i \notin V(\mathbf{P}^{a_1}) \\ \mathbb{F}_i^{a_1}, & i \in V(\mathbf{P}^{a_1}) \end{cases}$$

Note that if $\mathbf{q}(i) = \{a_1\}$, then $\mathbb{F}(1)_i = \mathbb{F}_i$. Let Λ' be the family obtained from Λ by removing \mathbf{N}^{a_1} . Then the family Λ' is spatiotemporally ordered with \mathbb{I} replaced by $\mathbb{F}(1)$ and \mathbb{F} unchanged. The result follows by the inductive hypothesis applied to \mathbf{N}^{a_1} and Λ' . Note that in the

cases where $\mathbb{I}_i = \mathbb{F}_i$, the transition function fixes the component (cf. remarks 8.26(1)). \square

Example 9.8. (Notation and assumptions of examples 9.5(1).) We illustrate the proof of proposition 9.7 for examples 9.5(1). We have $\mathbb{F}(1)_i = \mathbb{I}_i$, for $i \notin \mathbf{P}^1$, $\mathbb{F}(1)_i = \mathbb{F}_i^1$, $i \in V(\mathbf{P}^1)$. Set $\Lambda' = \{\mathbf{N}^2, \mathbf{N}^3\}$. For \mathbf{N}^1 , we take initialization and termination sets $\mathbb{I}(1) = \mathbb{I}$ and $\mathbb{F}(1)$ and on Λ' we take the initialization and termination structures given by $\mathbb{F}(1)$ and \mathbb{F} respectively. We define the generalized transition function $G^1 : \widehat{\mathbb{I}(1)} \rightarrow \widehat{\mathbb{F}(1)}$ for \mathbf{N}^1 by

$$G^1((\mathbf{x}_1, \dots, \mathbf{x}_6), (T_1, \dots, T_6)) = ((\mathbf{y}_1, \dots, \mathbf{y}_6), (\widehat{S}_1, \dots, \widehat{S}_6)),$$

where $\mathbf{y}_i = \mathbf{x}_i$, $\widehat{S}_i = T_i$, $i \notin V(\mathbf{P}^1)$, and the remaining variables are given by the generalized transition function H^1 of \mathbf{P}^1 . That is,

$$H^1((\mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4), (T_2, T_3, T_4)) = ((\mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_4), (\widehat{S}_2, \widehat{S}_3, \widehat{S}_4)).$$

We now repeat the process defining $\mathbb{I}(2) = \mathbb{F}(1)$, and $\mathbb{F}(2)$ by

$$\mathbb{F}(2)_i = \begin{cases} \mathbb{I}(2)_i, & i \notin V(\mathbf{P}^2) \\ \mathbb{F}_i^2, & i \in V(\mathbf{P}^2) \end{cases}$$

This defines the initialization and termination structure for the family \mathbf{N}^2 and associated generalized transition function $G^2 : \widehat{\mathbb{I}(2)} \rightarrow \widehat{\mathbb{F}(2)}$. For the final step, we take $\mathbb{I}(3) = \mathbb{F}(2)$ and $\mathbb{F}(3) = \mathbb{F}$ (this integrates up the simple node N_6) and generalized transition function $G^3 : \widehat{\mathbb{I}(3)} \rightarrow \widehat{\mathbb{F}(3)}$. Finally, the generalized transition function for \mathbf{N} is the composite $G^3 \circ G^2 \circ G^1$. \diamond

We continue with our previous notational assumptions. Suppose that the FAN $(\mathfrak{N}, \mathbb{I}, \mathbb{F})$ is of feedforward type with associated spatiotemporally ordered compatible family $\Lambda = \{\mathbf{N}^a \mid a \in \mathbf{q}\}$ of basic FANs.

Extend the partial order \prec on \mathbf{q} to a total order $<$ on \mathbf{q} .

Example 9.9. There are several different total orders that can be defined on the family Λ of examples 9.5. For example,

$$\begin{aligned} a < c < b < d < e < h < f < g \\ c < a < b < f < g < d < e < h \end{aligned}$$

(Each total ordering extending \prec will correspond to a factorisation of the generalized transition function for the feedforward FAN.) \diamond

Theorem 9.10 (Factorization theorem). *Let \mathbf{N} be a FAN of feedforward type with generalized transition function G and associated spatiotemporally ordered compatible family $\Lambda = \{\mathbf{N}^a \mid a \in \mathbf{q}\}$ of basic FANs.*

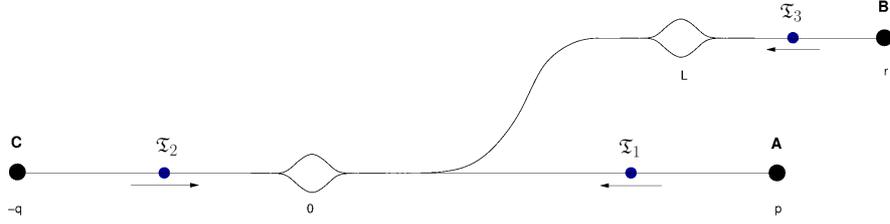


FIGURE 16. Three trains going through two passing loops. We require that once train \mathfrak{T}_2 has traversed the first passing loop it will continue on the branch line towards the second passing loop.

Assume no trivial factors of \mathbf{N} . Every extension of the partial order \prec on \mathbf{q} gives rise to a factorization

$$\mathbf{N} = \mathbf{N}^{a_q} \diamond \dots \diamond \mathbf{N}^{a_1}$$

with generalized transition function

$$G = G^{a_q} \circ \dots \circ G^{a_1}$$

where the $a_q > \dots > a_1$ is the total order on \mathbf{q} and the nontrivial part of the generalized transition function G^{a_j} is the generalized transition function of the primitive component of \mathbf{N}^{a_j} .

If \mathbf{N} contains trivial factors, the associated generalized transition functions can be inserted at any point in the composition.

Proof. The same arguments as used in the proof of proposition 9.7. \square

To conclude, we give an example to illustrate theorem 9.10.

Example 9.11. We consider a FAN $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ that models three trains $\mathfrak{T}_1, \mathfrak{T}_2, \mathfrak{T}_3$ passing through two passing loops, see figure 16. The state x_i of \mathfrak{T}_i is given by its position on the real line $M_i = \mathbb{R}$, $i \in \mathbf{3}$. The passing loops are located at $0, L$ and stations A, B, C are at $p, -q, r$ respectively, where $L, p, q, r > 0$. The trains $\mathfrak{T}_1, \mathfrak{T}_3$ start at stations A and B and travel with velocities $v_1 < v_3 < 0$ respectively. The train \mathfrak{T}_2 travels with velocity $v_2 > 0$ and starts at station C . It is required that \mathfrak{T}_1 has to go through the passing loop at 0 to pass train \mathfrak{T}_2 , and that train \mathfrak{T}_2 then has to traverse the passing loop at L to pass train \mathfrak{T}_3 . The trains $\mathfrak{T}_1, \mathfrak{T}_3$ terminate at C , and \mathfrak{T}_2 terminates at B .

We take node set $\mathcal{N} = \{N_0, N_1, N_2, N_3\}$ and network phase space $\mathbf{M} = \mathbb{R}^3$. Let $\alpha_i = N_0 \rightarrow N_i$, $i \in \mathbf{3}$. We define the generalized connection structure $\mathcal{A} = \{\emptyset, \alpha_1, \alpha_2, \alpha_3, \alpha_1 \vee \alpha_3, \alpha_2 \vee \alpha_3\}$. We define

the \mathcal{A} -structure \mathcal{F} by

$$\begin{aligned} \mathbf{f}^\emptyset &= (v_1, v_2, v_3), & \mathbf{f}^{\alpha_1} &= (0, v_2, v_3), \\ \mathbf{f}^{\alpha_2} &= (v_1, 0, v_3), & \mathbf{f}^{\alpha_3} &= (v_1, v_2, 0), \\ \mathbf{f}^{\alpha_1 \vee \alpha_3} &= (0, v_2, 0), & \mathbf{f}^{\alpha_2 \vee \alpha_3} &= (v_1, 0, 0). \end{aligned}$$

We define the event map $\mathcal{E} : \mathbf{M} \rightarrow \mathcal{A}$ by

$$\mathcal{E}(x_1, x_2, x_3) = \begin{cases} \alpha_1 & \text{if } x_1 = 0, x_2 < 0, \\ \alpha_2 & \text{if } x_1 > 0, x_2 = 0 \text{ or } x_2 = L, x_3 > L, \\ \alpha_3 & \text{if } x_2 < L, x_3 = L, \\ \alpha_1 \vee \alpha_3 & \text{if } x_1 = 0, x_2 < 0, x_3 = L, \\ \alpha_2 \vee \alpha_3 & \text{if } x_1 < 0, x_2 = 0, x_3 = L, \\ \emptyset & \text{otherwise.} \end{cases}$$

These definitions define the asynchronous network $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$. We obtain the FAN $\mathbf{N} = (\mathfrak{N}, \mathbb{I}, \mathbb{F})$ modelling the train network by taking initialization and termination sets \mathbb{I}, \mathbb{F} defined by $\mathbb{I}_1 = \{p\}$, $\mathbb{F}_1 = \{-q\}$, $\mathbb{I}_2 = \{-q\}$, $\mathbb{F}_2 = \{r\}$, $\mathbb{I}_3 = \{r\}$, $\mathbb{F}_3 = \{-q\}$.

For the factorization of dynamics, we identify two stably primitive components, \mathbf{N}^a (describing dynamics in the first passing loop), and \mathbf{N}^b (describing dynamics in the second passing loop). We define \mathfrak{N}^a by $\mathcal{A}^a = \{\emptyset, \alpha_1, \alpha_2\}$, $\mathcal{F}^a = \{\mathbf{f}_a^{\alpha_1} = \mathbf{f}^{\alpha_1}, \mathbf{f}_a^{\alpha_2} = \mathbf{f}^{\alpha_2}, \mathbf{f}_a^\emptyset = \mathbf{f}^\emptyset\}$, and take

$$\mathcal{E}^a(x_1, x_2, x_3) = \begin{cases} \alpha_1 & \text{if } x_1 = 0, x_2 < 0, \\ \alpha_2 & \text{if } x_1 > 0, x_2 = 0 \\ \emptyset & \text{otherwise.} \end{cases}$$

For \mathfrak{N}^b we take $\mathcal{A}^b = \{\emptyset, \alpha_2, \alpha_3\}$, $\mathcal{F}^b = \{\mathbf{f}_b^{\alpha_2} = \mathbf{f}^{\alpha_2}, \mathbf{f}_b^{\alpha_3} = \mathbf{f}^{\alpha_3}, \mathbf{f}_b^\emptyset = \mathbf{f}^\emptyset\}$,

$$\mathcal{E}^b(x_1, x_2, x_3) = \begin{cases} \alpha_2 & \text{if } x_2 = L, x_3 > L, \\ \alpha_3 & \text{if } x_2 < L, x_3 = L, \\ \emptyset & \text{otherwise.} \end{cases}$$

The family $\{\mathfrak{N}^a, \mathfrak{N}^b\}$ is compatible and we have $\mathfrak{N} = \mathfrak{N}^a \sqcup \mathfrak{N}^b$. It remains to define the initialization and termination sets for $\mathfrak{N}^a, \mathfrak{N}^b$. We take $\mathbb{I}_i^c = \mathbb{I}_i$, $c \in \{a, b\}$, $i \in \mathbf{3}$, except that $\mathbb{I}_2^b = \{L/2\}$ (any point in $(0, L)$ would do). Similarly, we take $\mathbb{F}_i^c = \mathbb{F}_i$, $c \in \{a, b\}$, $i \in \mathbf{3}$, except that $\mathbb{F}_2^a = \{L/2\}$. With these definitions, $\mathbf{N}^a = (\mathfrak{N}^a, \mathbb{I}^a, \mathbb{F}^a)$ and $\mathbf{N}^b = (\mathfrak{N}^b, \mathbb{I}^b, \mathbb{F}^b)$ are stably primitive FANs, $\{\mathbf{N}^a, \mathbf{N}^b\}$ is a spatiotemporally ordered compatible family, and \mathbf{N} is a FAN of feedforward type. If we denote the primitive components of $\mathbf{N}^a, \mathbf{N}^b$ by $\mathbf{P}^a, \mathbf{P}^b$ respectively, then $V(\mathbf{P}^a) = \{1, 2\}$, and $V(\mathbf{P}^b) = \{2, 3\}$.

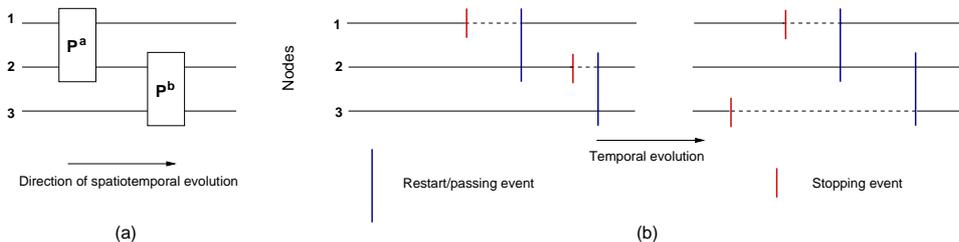


FIGURE 17. (a) Schematic indicating the primitive components of \mathbf{N} . (b) Temporal evolution for individual nodes based on different starting times. Dotted lines indicate stopped nodes, vertical bars either stopping or restarting/passing events. The stopping events may occur in any order in contrast to the restarting events.

Applying theorem 9.10, the generalized transition function for \mathbf{N} can be written as a composition of the generalized transition functions G^a for \mathbf{N}^a and G^b for \mathbf{N}^b . Note that G^a is the identity in the third component, G^b is the identity in the first component. Different initializations yield different trajectories as depicted in figure 17(b). \diamond

10. STOCHASTIC ASYNCHRONOUS NETWORKS

Definition 10.1. An *event process* for an asynchronous network structure $(\mathcal{N}, \mathcal{A}, \mathcal{F})$ is a state dependent stochastic process $\mathcal{E}_{(t, \mathbf{x})}$ taking values in \mathcal{A} .

In the most general case there are no restrictions on the process $\mathcal{E}_{(t, \mathbf{x})}$: there may be (stochastic) dependence on time $t \in \mathbb{R}^+$, pure space dependence ($\mathcal{E}_{(t, \mathbf{x})} = \mathcal{E}(\mathbf{X})$), or both. If $\mathcal{E}_{(t, \mathbf{x})}$ is independent of time, then the event process reduces to an event map $\mathcal{E} : \mathbf{M} \rightarrow \mathcal{A}$.

Definition 10.2. A *stochastic asynchronous network* $\mathfrak{N} = (\mathcal{N}, \mathcal{A}, \mathcal{F}, \mathcal{E})$ consists of an asynchronous network structure $(\mathcal{N}, \mathcal{A}, \mathcal{F})$ together with an event process $\mathcal{E} = \mathcal{E}_{(t, \mathbf{x})}$.

Of course, additional assumptions are needed for a stochastic asynchronous network to give well defined dynamics. In some cases it is possible to represent stochastic asynchronous networks as random dynamical systems where there is a body of established theory (see [13] and section 10.3 for an example). Here we concentrate on examples where there are no difficulties defining the dynamics with the aim of illustrating dynamical phenomena that can arise in stochastic asynchronous networks. We highlight applications to numerics as well as hint at some of the novel dynamics that can occur.

10.1. Passing loop revisited: stochastic events and network stalls. Recall the basic example from section 6: two trains, $\mathfrak{T}_1, \mathfrak{T}_2$ go in opposite directions on a single track railway line with a passing loop. In contrast to the purely deterministic case, we now add stochastic effects to the system (see also [42]).

Consider an event process \mathcal{E}_t (for simplicity, assume no state dependence). For example, suppose we are given a homogeneous Poisson process $P(t)$ with small rate parameter $0 < \lambda \ll 1$. This defines a sequence $0 \leq t_1 < t_2 < \dots$ where $\mathbb{E}(t_{k+1} - t_k) = \lambda^{-1} \gg 1, k \in \mathbb{N}$. At each time t_j we randomly choose one of the trains $\mathfrak{T}_1, \mathfrak{T}_2$ with equal probability and then add the connection $N_0 \rightarrow N_i$, where \mathfrak{T}_i is the chosen train – the train \mathfrak{T}_i is then stopped. We assume that the connection $N_0 \rightarrow N_i$ persists for a time given by a renewal process. For example, we could assume the stopping time is modelled by a possibly truncated exponential distribution with parameter $\mu \gg 1$. This setup models breakdowns on the line. Observe that if there is a breakdown before both trains have entered the passing loop then the network will come to a complete stop when the working train enters the passing loop and will only restart when the breakdown is removed: we say the network is *stalled*. Otherwise there will be a *partial stall* – the working train will keep running.

The notion of a functional network extends to stochastic asynchronous network. For given sets of initial and terminal conditions, function is determined by the distribution of completion times. In terms of the example above, construct a functional network by introducing stations as in section 6.4. The possibility of network stalls can increase the expected arrival time thereby potentially rendering arrival within a reasonable time impossible. This situation could be seen as an *latent deadlock*. In contrast to deterministic systems where deadlocks can be avoided by good network design, stochastically induced deadlocks in a stochastic FAN may be hard to find and analyze. Many times, breakdowns in network function are handled by a network reset – for example, at the end of a 24 hour working cycle.

At the same time, adding stochasticity may sometimes be beneficial for network function. Consider the example of two trains travelling in the same direction with the same velocity on a double track line that has a single track segment (as in example 8.20(2)) with a station immediately before the line becomes single track. Suppose both trains have to wait at the station together for a specified amount of time. In this case, randomly choosing the start times will avoid the otherwise inevitable collision. In a similar manner, stochastic initialization times may prevent deadlock at a four-way stop, example 7.13(2).

10.2. Numerical solution of differential equations. We consider an asynchronous network associated to a diffusively coupled network of k nodes with additive input structure. Let $Z(k)$ denote the space of $k \times k$ 0-1 matrices (we allow diagonal elements to be nonzero). Suppose we are given a stochastic process $C(t) = (c_{ij}(t))$ with values in $Z(k)$. If we assume that the probability that $C = Z$ is nonzero for all $Z \in Z(k)$, then C defines an event process $\mathcal{E}_t : \mathbf{M} \rightarrow \mathcal{A} = Z(k)$. Define dynamics on \mathcal{N} by

$$(17) \quad \mathbf{x}'_i = c_{ii}G_i(\mathbf{x}_i) + \sum_{s \in \mathbf{k}, s \neq i} c_{is}G_{is}(\mathbf{x}_s - \mathbf{x}_i), \quad i \in \mathbf{k}.$$

Under reasonable conditions on $C(t)$ (for example, Poisson statistics), (17) will almost surely have unique solutions for each $\mathbf{X}_0 \in \mathbf{M}$ which are piecewise smooth and depend continuously on \mathbf{X}_0 [44]. Examples of such a system are blinking systems [18, 58, 59] where $C(t)$ is constant on intervals. Depending on the properties of the stochastic process $C(t)$ and the time scales for (17), it may be possible to take an ensemble average for (17) and thereby reduce to a deterministic system which gives the ‘average’ dynamics for (17).

We look at a simple discretization of such an ODE. Suppose that \mathcal{N} is governed by the differential equations

$$(18) \quad \mathbf{x}'_i = G_i(\mathbf{x}_i) + \sum_{s \in \mathbf{k}, s \neq i} k_{ij}G(\mathbf{x}_s - \mathbf{x}_i), \quad i \in \mathbf{k},$$

For simplicity assume that $k_{ij} \in (0, 1]$ with at least one coefficient $k_{ij} = 1$. Take a discretization of (18) (say Euler or Runge-Kutta 4th order). If we denote the time step by δ , then the discretization may be written in the form

$$(19) \quad \mathbf{x}_i^{n+1} = \mathbf{x}_i^n + \mathbf{G}_i(\delta, \mathbf{x}_i^n) + \sum_{s \in \mathbf{k}} k_{ij} \mathbf{G}(\delta, \mathbf{x}_s^n, \mathbf{x}_i^n), \quad i \in \mathbf{k}, n \geq 0.$$

Now run as a discrete asynchronous network with the k_{ij} following a Bernoulli process with probability k_{ij} ($k_{ij}(n) = 1$ with probability k_{ij}). It is a simple consequence of the ergodic theorem (strong law of large numbers) that as $\delta \rightarrow 0$, the solution of (19) converges to the true solution of (18). Typically the computation in large networks will run significantly faster than the deterministic iteration. As an example, we computed the time in an all-to-all coupled system of 1000 phase oscillators, with nonlocal coupling given by the exponential kernel ($\kappa = 4$ in [1, §III]). In this case the average probability is approximately 0.43. If we ignore the overheads caused by using a random number generator,

it is simple to compute that the time taken for the probabilistic computation should be approximately 45% of the time for the deterministic computation. In practice, using both threaded and non-threaded computation, we found the probabilistic computation took about 58% of the time of the deterministic computation (20,000 iterations, Runge-Kutta 4).

10.3. Dynamics on a stochastic discrete asynchronous network & commingled basins of attraction. We conclude this section by briefly describing a class of discrete stochastic asynchronous networks that exhibit novel dynamics involving time dependent clustering and synchronization as well as what we call *comingled basins of attraction*. This work is discussed in detail in [42].

We assume a network \mathcal{N} with k nodes, N_1, \dots, N_k , each with phase space \mathbb{T} . Let $\mathcal{E} = (\mathbf{E}(n))$ be a random sequential event process such that each $\mathbf{E}(n)$ is a 0-1 matrix $(E_{ij}(n))$ defining a directed connection structure on \mathcal{N} with no self-connections. Here we will be imprecise about the properties of \mathcal{E} except to remark that

- (1) \mathcal{E} is typically state dependent (depends on $\boldsymbol{\theta} \in \mathbb{T}^k$).
- (2) The components $E_{ij}(n)$ are not independent random variables.

We consider a discrete stochastic iteration on \mathbb{T}^k of the form

$$(20) \quad \theta_i^{n+1} = \omega + \theta_i^n + \sum_{j \in \mathbf{k}} E_{ij}(n) g(\theta_j^n - \theta_i^n), \quad j \in \mathbf{k}, \quad n \geq 0,$$

where $0 < \omega \ll 1$, and $g : \mathbb{R} \rightarrow \mathbb{R}$ is 1-periodic and odd (for example $g(\theta) = a \sin(2\pi\theta) + b \sin(4\pi\theta)$). This system can be modelled as a random dynamical system with 2^{k^2-k} symbols (each symbol represents a directed connection structure). Under mild conditions on g and \mathcal{E} , almost surely the empty connection structure will be chosen infinitely often. Since \mathcal{E} is state dependent, the frequency of connections between nodes i and j may depend strongly on i, j .

The deterministic iteration $F : \mathbb{T}^k \rightarrow \mathbb{T}^k$ defined by

$$\theta_i^{n+1} = \omega + \theta_i^n + \sum_{j \in \mathbf{k}} g(\theta_j^n - \theta_i^n), \quad i \in \mathbf{k}, \quad n \geq 0,$$

is S_k -equivariant. Given $p \in \mathbb{N}$, $\alpha \in \mathbb{T}$, define $\alpha^p = (\alpha, \dots, \alpha) \in \Delta_p \subset \mathbb{T}^p$. If we choose $p, q \in \mathbb{N}$, such that $p + q = k$, then the subspace

$$V_{p,q} = \{(\alpha^p, \beta^q) \mid \alpha, \beta \in \mathbb{T}\} \subset \Delta_p \times \Delta_q \subset \mathbb{T}^k,$$

is F -invariant and so, by the S_k -equivariance of F , $\gamma V_{p,q}$ is F -invariant for all $\gamma \in S_k$. This result is obviously *not* true for the stochastic iteration (20). Yet it may be possible to find a subset Σ of $V_{p,q}$ such that

$\gamma\Sigma$ is invariant under the stochastic iteration for all $\gamma \in S_k$. Indeed, suppose that $g(\tau) = 0$ for some $\tau \in (0, 1)$, $\tau \neq \frac{1}{2}$ (since g is assumed odd, we always have $g(\frac{1}{2}) = 0$ and $g(1 - \tau) = 0$). If $\theta_i - \theta_j \in \{0, \pm\tau\}$, then $\sum_{j \in \mathbf{k}} E_{ij}(n)g(\theta_j^n - \theta_i^n) = 0$ whatever the values of $E_{ij}(n)$. It follows that if we define

$$\tilde{V}_{p,q} = \{(\theta^p, (\tau + \theta)^q) \mid \theta \in \mathbb{T}\} \subset V_{p,q} \subset \mathbb{T}^k,$$

then $\gamma\tilde{V}_{p,q}$ is invariant by the stochastic iteration (20) for all $\gamma \in S_k$, $p, q \in \mathbb{N}$, $p + q = k$. We remark that the set $\gamma\tilde{V}_{p,q}$ is an S^1 -orbit under the phase shift symmetry $(\theta_1, \dots, \theta_k) \mapsto (\theta_1 + \psi, \dots, \theta_k + \psi)$ and that deterministic and stochastic iterations are equivariant with respect to phase shift symmetry. Hence $\tilde{V}_{p,q}$ is a relative fixed point set [39, §8.1] for the stochastic iteration.

For certain values of p, q and choice of map g , it is easily shown that $\gamma\tilde{V}_{p,q}$ may be an attractor for the deterministic iteration, all $\gamma \in S_k$, $p + q = k$: a total of $\binom{k}{p}$ attractors (note that the stability of $\tilde{V}_{p,q}$ may be different from that of $\tilde{V}_{q,p}$). Remarkably, this may also be true for the stochastic iteration (20) though now the basins of attraction may overlap, or even be equal (up sets of measure zero). We regard the basins of attraction as ‘comingled’ rather than intermingled [9, 80]. We illustrate with an example that can easily be tested numerically and exhibits some of the complex dynamical phenomena that can occur.

Example 10.3 (Comingled basins of attraction). Suppose

$$g(\psi) = -0.06 \sin(2\pi\psi) + 0.0336 \sin(4\pi\psi).$$

We have $g(\pm\tau) = 0$, where $\tau = \frac{1}{2\pi} \cos^{-1}(\frac{25}{28})$. In this case, if we take p close to $k/2$, $k \geq 4$, and an appropriate stochastic process \mathcal{E} (details in [42]), then $\tilde{V}_{p,k-p}$ can be an attractor for the stochastic iteration (20). More precisely, for almost all initial conditions $\theta_0 \in \mathbb{T}^k$, the stochastic iteration (20) will converge almost surely to $\gamma\tilde{V}_{p,k-p}$, where p is close to $k/2$, and $\gamma \in S_k$ (p, γ will be random variables with statistics depending on $\theta_0 \in \mathbb{T}^k$).

We give two numerical illustrations of the phenomenon of comingled basins. For the numerics, we use a slight elaboration of the stochastic process \mathcal{E} by turning \mathcal{N} into a ‘spiking’ network. To do this, assume given a state dependent probability function $p : \mathbb{T} \rightarrow [0, 1]$ defined on each node. For each iteration of (20), node N_i will fire (a spike) with probability $p(\theta_i)$, $i \in \mathbf{k}$. We define a new stochastic process \mathcal{E}_1 by applying \mathcal{E} only to the nodes which have fired. The random iteration (20) defined by \mathcal{E}_1 is an example of a place-dependent random

dynamical system [97, 17] (it is not of the standard type described by Arnold [13]).

We define the *pulse* and *bell* probability functions p_1, p_2 on $[0, 1]$ by

$$p_1(\theta) = \begin{cases} p_l & \text{if } \theta - \frac{1}{2} > \frac{d}{2}, \\ p_u & \text{if } \theta - \frac{1}{2} \leq \frac{d}{2}, \end{cases} \quad p_2(\theta) = 16\theta^2(1 - \theta)^2,$$

where $0 \leq p_l < p_u \leq 1$ and $d \in (0, \frac{1}{2}]$. When we run the random iteration (20) using \mathcal{E}_1 , we look at the statistics of spikes according to two protocols:

(AS) We register all spikes.

(CS) We only register a spike at node N_i if the out-degree $\sum_j E_{ji} > 0$.

The spiking statistics convey information about which nodes have fired but without identifying the nodes to which they connect.

We describe a scheme for the visualization and identification of clustering that can occur in networks with this mix of random and deterministic dynamics. We use the firing dynamics to drive a set of contractive cocycles so as to capture the average firing behaviour.

We identify each node of the network with the vertex of a regular octagon, centred at the origin of $\mathbb{R}^2 \approx \mathbb{C}$. Denote the coordinates of N_j by Z_j , where $Z_j = \exp((j - 1)\pi i/4)$, $j \in \mathbf{8}$. We associate with each node N_j the contraction mapping $f_j : \mathbb{C} \rightarrow \mathbb{C}$ with fixed point Z_j defined by

$$f_j(z) = \frac{1}{2}(z + Z_j).$$

Take as initial point $z_0 = 0 \in \mathbb{C}$. Suppose we have constructed the sequence z_0, z_1, \dots, z_n after $m \geq n$ steps of the stochastic iteration (20). At the $(m + 1)$ th step of the iteration, suppose that the nodes N_{j_1}, \dots, N_{j_s} fire (if no nodes fire, do nothing, go to the next iteration). Define

$$z_{n+1} = \frac{1}{s} \sum_{i=1}^s f_{j_i}(z_n).$$

At least numerically, (z_n) converges to an attractor with associated invariant measure. Note that this scheme emphasizes the node firing (observables) rather than the ‘internal’ (hidden) dynamics of the nodes. The attractor and measure reflect statistical properties of the firing patterns and yield useful information about the node dynamics (for example, synchronization and clustering statistics). For the example we describe the internal dynamics is simple but the firing patterns observed are complex.

In figure 18, we show the results of four iterations (each of 175,000,000 iterates and using the same initial conditions $\theta_1 = 0.33$, $\theta_2 = 0.436265$,

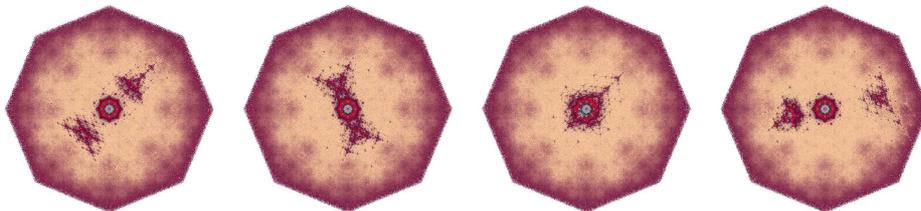


FIGURE 18. Patterns of synchronization in firings using pulse probability function.

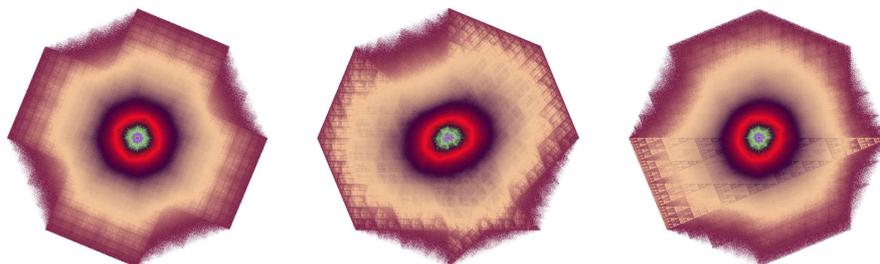


FIGURE 19

$\theta_3 = 0.306$, $\theta_4 = 0.28$, $\theta_5 = 0.57$, $\theta_6 = 0.38$, $\theta_7 = 0.51$, $\theta_8 = 0.4$), pulse probability (with $d = 0.2$, $p_l = 0.01$, $p_u = 0.98$), and protocol (CS). Reading from left to right, the figures correspond to the synchronization patterns *abaabbba*, *ababbaba*, *abaababa*, and *aaaababb*. The last two patterns correspond to synchronization into a (5,3) cluster, the other two a (4,4) cluster.

Use of a different probability function can lead to more transparent patterns. In figure 19, we show the results of three iterations but now with bell probability, and protocol (AS). Reading from left to right, the figures correspond to the synchronization patterns *aabbaabb*, *ababbbaa* and *abbabbba*. The last pattern corresponds to synchronization into a (5,3) cluster, the other two into (4,4) clusters. In this case, it is easy to read the synchronization pattern from the figure.

If we break the invariant subspace structure (for example, by taking $g(\theta) = -0.06 \sin(2\pi\psi) + 0.0336 \sin(4\pi\psi + \beta)$, $\beta \neq 0$, or by making the frequencies ω node dependent) there may be random switching between the sets $\gamma\tilde{V}_{4,4}$, $\gamma \in X$ (see [41] for a video of this phenomenon).

The same phenomena are seen for all $k \geq 4$ (see [41] for examples and note that the range of attractors depends on properties of \mathcal{E}).

Similar phenomena to what we have described above have previously been described by Kaneko in the context of coupled logistic maps with noise added [65, 66]. \diamond

11. CONCLUDING COMMENTS & OUTSTANDING PROBLEMS

Our overall aim has been to outline a mathematical framework for asynchronous networks and event driven dynamics that enables the analysis of network dynamics that cannot be satisfactorily modelled by classical systems of analytic differential equations. Examples and applications are motivated by problems in physics, computer science, biology and engineering. In particular, there are connections with Filippov systems (mostly considered in the engineering literature). Classes of functional networks, which have relations with control theory, have previously been considered in a neuroscientific context [24, 89, 90, 26].

Is it possible to understand the global dynamics of large networks? The classical reductionist approach in terms of nodes – understanding the dynamics of the whole in terms of the dynamics of the nodes – fails even for ‘simple’ synchronous networks. We have shown that for functional asynchronous networks a different reductionist approach is feasible: rather than the nodes themselves acting as building blocks, the dynamics consist of basic spatiotemporal events that involve two or more nodes. This approach cannot work if we attempt to approximate an asynchronous network by a synchronous network. In other words, in order to understand the global dynamics of asynchronous networks, one needs to work with, rather than against, the inherent nonsmoothness.

Many questions and open problems remain. We sketch a few representative mathematical questions for stochastic and functional asynchronous networks.

In section 9 we defined a primitive FAN. As part of a description of FANs of feedforward type, it would be useful to have a structural classification of primitive FANs with a small number of nodes and an identification of the primitive FANs which appear most frequently in applications. From a practical point of view, given a time series output from a functional asynchronous network, is it possible to reconstruct the basic events, possibly by searching for suitable temporal correlation? Turning to bifurcation questions, the factorization of dynamics theorem relies on (a) compatibility of a family – non-intersecting event sets (condition (G1) of weak regularity), (b) structural conditions on the event map (condition (G2) of weak regularity), (c) spatiotemporal order (condition (G3) of weak regularity). Breakdown of any one of these conditions is a natural place to study bifurcation. For example,

the introduction of a weak feedback loop breaking the spatiotemporal order. Such breakdown is likely to be part of evolutionary adaptation and refinement that leads to a more complex and better optimized system. Closely related is the breakdown of regularity and the appearance of hidden deadlocks. In all four cases, bifurcation through evolutionary refinement and optimization of a system carries the possibility of a penalty in terms of the inherent system stability and robustness.

In section 10, we gave an example of a discrete network that exhibited the phenomenon of comingled basins of attraction. Issues here involve finding a general theory for handling the phenomenon, quantifying bifurcation when the invariant subspace structure is broken (for example, by allowing for frequency variation in example 10.3 – numerics indicate phenomena reminiscent of cycling chaos [32, 14], [37, Chapter 7]), and the case of continuous dynamics.

For some applications, it may be appropriate to replace the termination hypersurface used in the definition of a FAN by a proper closed subset that is reached for a particular time initialization, say $\mathbf{0} = (0, \dots, 0) \in \mathbb{R}_+^k$. There is the question of how the target sets may vary and bifurcate as we increase the range of possible initialization times. This question is of direct relevance to applications: initialization at $\mathbf{0}$ can be seen as ‘synchronized initialization’ and network function may break down if the initialization times are too spread out. If we have an FAN with a generalized transition function that spreads the termination times out on average (and termination times yield initialization times for another FAN) then after a certain amount of repetition network function may break down. In terms of a transportation network this could be seen as propagation of delays. In fact, real-world transportation networks are typically approximately synchronized on a daily basis through a nightly ‘reset’. More generally, for realistic applications it is usually natural to assume the initialization times, and other starting time events, follow a statistical law and obtain the corresponding statistical law of the termination times. Here it may be helpful to have a structured random dynamical systems model of a functional asynchronous network.

Acknowledgements. CB would like to thank Marc Timme at the Max Planck Institute for Dynamics and Self-Organization for continuing hospitality. MF would like to express his gratitude to the Mathematics Department at Rice University, where much of this work was

done, for providing a warm, friendly and very supportive working environment. Many thanks also to Manchester University for their hospitality over the years and to Steve Furber for his penetrating insights and questions.

REFERENCES

- [1] D M Abrams & S H Strogatz. ‘Chimera States in a Ring of Nonlocally Coupled Oscillators’, *Int. Jnr. of Bifurcation and Chaos* **16**(1) (2000), 21–37.
- [2] D M Abrams & S H Strogatz. ‘Chimera States for Coupled Oscillators’, *Phys. Rev. Lett.* **93**(17), (2004).
- [3] J Acebrón, L Bonilla, C Pérez Vicente, F Ritort & R Spigler. ‘The Kuramoto model: A simple paradigm for synchronization phenomena’, *Rev. Mod. Phys.* **77**(1), (2005).
- [4] N Agarwal & M Field. ‘Dynamical equivalence of networks of coupled dynamical systems I: asymmetric inputs’, *Nonlinearity* **23** (2010), 1245–1268.
- [5] N Agarwal & M Field. ‘Dynamical equivalence of networks of coupled dynamical systems II: general case’, *Nonlinearity*, **23** (2010), 1269–1289.
- [6] M Aguiar & A P S Dias. ‘Minimal coupled cell networks’, *Nonlinearity* **20** (2007), 193–219.
- [7] M Aguiar, A P S Dias, P Ashwin, & M Field. ‘Dynamics of coupled cell networks: synchrony, heteroclinic cycles and inflation’, *Journal of Nonlinear Science*, **21**(2) (2011), 271–323.
- [8] M Aldana, S Coppersmith, & L P Kadanoff. ‘Boolean dynamics with random couplings’, in *Perspectives and Problems in Nonlinear Science* (2003), 23–89.
- [9] J C Alexander, J A Yorke, Z You, & I Kan. ‘Riddled basins’, *Int. J Bifur Chaos* **2** (1992), 795–813.
- [10] U Alon. *An Introduction to Systems Biology. Design Principles of Biological Circuits* (Chapman & Hall/CRC, Boca Raton, 2007).
- [11] P W Anderson. ‘More is different’, *Science* **177**(4047) (1972), 393–396.
- [12] D V Anosov. ‘Stability of equilibrium positions in relay systems’, *Automation Remote Control* **20** (1959), 135–149.
- [13] L Arnold. *Random Dynamical Systems* (Springer, 1998)
- [14] P Ashwin & A M Rucklidge. ‘Cycling chaos; its creation, persistence and loss of stability in a model of nonlinear magnetoconvection’, *Physica D* **24** (1998), 134–154.
- [15] P Ashwin & J Swift. ‘The dynamics of n weakly coupled identical oscillators’, *Journal of Nonlinear Science* **2** (1992), 69–108.
- [16] J Barhen & V Protopescu. ‘Preventing Computational Chaos in Asynchronous Neural Networks’, *Proc. of Applied Informatics* (2003), 1290–1295.
- [17] M F Barnsley, S G Demko, J H Elton, & J S Geronimo. ‘Invariant measures for Markov processes arising from iterated function systems with place-dependent probabilities’, *Ann. Inst. H. Poincaré Probab. Stat.* **24** (1988), 367–397 (Erratum: **25**, 1989, 589–590).
- [18] I Belykh, V Belykh, & M Hasler. ‘Blinking model and synchronization in small-world networks with a time-varying coupling’, *Physica D*, 195(1-2) (2004), 188–206.

- [19] I Belykh, M di Bernardo, J Kurths, & M Porfiri. ‘Evolving dynamical networks’, *Physica D*, **267** (2014), 1–6.
- [20] M Bennett, M F Schatz, H Rockwood, & K Wiesenfeld. ‘Huygens’s clocks’, *Proc. Royal Soc. A: Mathematical, Physical and Engineering Sciences* **458**(2019) (2002), 563–579.
- [21] M di Bernardo, C J Budd, & A R Champneys, ‘Normal form maps for grazing bifurcations in n -dimensional piecewise-smooth dynamical systems’, *Physica D* **160** (2001), 222–254.
- [22] M di Bernardo, C J Budd, A R Champneys, & P Kowalczyk. *Piecewise-smooth Dynamical Systems* (Springer, Applied Mathematical Science 163, London, 2008).
- [23] C Bick. ‘Local representation of Asynchronous networks by Filippov systems’, in preparation.
- [24] C Bick & M Rabinovich. ‘Dynamical Origin of the Effective Storage Capacity in the Brain’s Working Memory’, *Phys. Rev. Lett.* **103**(21) (2009),
- [25] C Bick, M Timme, D Paulikat, D Rathlev, & P Ashwin. ‘Chaos in Symmetric Phase Oscillator Networks’, *Phys. Rev. Lett.* **107**(24) (2011), 244101.
- [26] C Bick & E A Martens, ‘Controlling Chimeras’. *New Journal of Physics*, **17**(3) (2015), 033030.
- [27] J Bochnak, M Coste, & M-F Roy. *Real algebraic geometry* (Springer, 1998).
- [28] A N Burkitt, M Gilson, & J L van Hemmen. ‘Spike-timing-dependent plasticity for neurons with recurrent connections’, *Biol. Cybern.* **96** (2007), 533–546.
- [29] N Caporale & Y Dan. ‘Spike timing dependent plasticity: a Hebbian learning rule’ *Annu. Rev. Neurosci.* **31** (2008), 25–36.
- [30] C Gershenson. ‘Classification of Random Boolean Networks’, in *Artificial Life VIII* (Standish, Abbass, Bedau (eds), MIT Press) (2002), 1–8.
- [31] R David & H Alla. *Discrete, Continuous, and Hybrid Petri Nets* (Springer-Verlag, 2010).
- [32] M Dellnitz, M Field, M Golubitsky, A Hohmann, & J Ma. ‘Cycling Chaos’, *Intern. J. Bifur. & Chaos* **5**(4) (1995), 1487–1501 (also appeared in: *IEEE Trans. Circuits & Syst.* **42** (10) (1995), 821–823).
- [33] A P S Dias & I Stewart. ‘Linear equivalence and ODE-equivalence for coupled cell networks’, *Nonlinearity* **18** (2005), 1003–1020.
- [34] F Dörfler & F Bullo. ‘Synchronization and transient stability in power networks and non-uniform Kuramoto oscillators’, *SIAM Journal on Control and Optimization* **50** (3) (2012), 1616–1642.
- [35] F Dörfler, M Chertkov, & F Bullo. ‘Synchronization in Complex Oscillator Networks and Smart Grids’, *PNAS* **110**(6) (2013), 2005–2010.
- [36] G B Ermentrout & N Kopell. ‘Multiple pulse interactions and averaging in systems of coupled neural oscillators’, *J. Math. Biol.* **29** (1991), 193–217.
- [37] M Field. *Dynamics, Bifurcation and Symmetry*, Pitman Research Notes in Mathematics, **356**, 1996.
- [38] M J Field. ‘Combinatorial Dynamics’, *Dynamical Systems* **19** (2004), 217–243.
- [39] M J Field. *Dynamics and Symmetry* (Imperial College Press Advanced Texts in Mathematics — Vol. 3, 2007.)
- [40] M J Field. ‘Heteroclinic networks in homogeneous and heterogeneous identical cell systems’, preprint 2014.

- [41] M J Field. ‘Dynamics on Asynchronous Networks’ (Unpublished talk given in 2013 and video available at URL: math.rice.edu/~mjf8/research/networks/Asynchronous/).
- [42] M Field & C Bick. ‘Asynchronous Networks’, in preparation.
- [43] R Genov & G Cauwenberghs. ‘Kerneltron: Support Vector “Machine” in Silicon’, *IEEE Trans. on neural networks* **14**(5) (2003), 1426–1434.
- [44] A F Filippov. *Differential Equations with Discontinuous Righthand Sides* (Kluwer Academic Publishers, 1988).
- [45] G Filatrella, A H Nielsen, & N F Pederson. ‘Analysis of a power grid using a Kuramoto-like model’, *Eur. Phys. J. B* **61** (2008), 485.
- [46] W Gerstner, R Kempter, L J van Hemmen & H Wagner. ‘A neuronal learning rule for sub-millisecond temporal coding’, *Nature* **383** (1996), 76–78.
- [47] W Gerstner & W Kistler. *Spiking Neuron Models* (Cambridge University Press, 2002).
- [48] C Gibson, K Wirthmüller, A A du Plessis, & E Looijenga. *Topological stability of smooth mappings* (Springer Lect. Notes in Math. **552**, 1976).
- [49] M Gilson, A N Burkitt, & J L van Hemmen. ‘STDP in recurrent neuronal networks’, *Frontiers in Comp. Neuroscience* **4** (2010).
- [50] M Golubitsky, I Stewart, P L Buono, & J J Collins. ‘Symmetry in locomotor central pattern generators and animal gaits’, *Nature* **401** (1999) 693–695.
- [51] M Golubitsky, M Pivato, & I Stewart. ‘Interior symmetry and local bifurcation in coupled cell networks’. *Dynamical Systems* **19** (2004), 389–407.
- [52] M Golubitsky & I Stewart. ‘Nonlinear dynamics of networks: the groupoid formalism’. *Bull. Amer. Math. Soc.* **43** (2006), 305–364.
- [53] M Golubitsky, I Stewart, & A Török. ‘Patterns of Synchrony in Coupled Cell Networks with Multiple Arrows’, *SIAM J. Appl. Dynam. Sys.* **4**(1) (2005), 78–100.
- [54] A Gomez-Exposito, AJ Conejo, & C Canizares (editors). *Electric Energy Systems: Analysis and Operation*, (CRC Press, Taylor & Francis, Boca Raton, Florida, 2009),
- [55] T E Goroehowski, M Di Bernardo, & C S Grierson. ‘Evolving dynamical networks: A formalism for describing complex systems’, *Complexity* **17**(3) (2012), 18–25.
- [56] C Gotsman, E Shamir, & D Lehmann. ‘Asynchronous Dynamics of Random Boolean Networks (Extended Abstract)’, *IEEE International Conference on Neural Networks, 1988* **1**, 1–7.
- [57] W M Haddad, V Chellaboina, & Sergey G Nersesov. *Impulsive and Hybrid Dynamical Systems: Stability, Dissipativity, and Control* (Princeton University Press, 2006).
- [58] M Hasler, V Belykh, & I Belykh. ‘Dynamics of Stochastically Blinking Systems. Part I: Finite Time Properties’, *SIAM J. Appl. Dynam. Sys.*, **12**(2), 1007–1030.
- [59] M Hasler, V Belykh, & I Belykh. ‘Dynamics of Stochastically Blinking Systems. Part II: Asymptotic Properties’, *SIAM J. Appl. Dynam. Sys.*, **12**(2), 1031–1084.
- [60] M Herlihy & N Shavit. ‘The Topological Structure of Asynchronous Computability’, *J. of the ACM*, **46**(6) (1999), 858–923.
- [61] B Heidergott, G J Olsder & J van der Woude. *Max Plus at Work* (Princeton Series in Applied Mathematics, 2006).

- [62] M W Hirsch. *Differential Topology* (Springer Graduate Texts in Math., **33**, Springer-Verlag, New York, 1976).
- [63] P Holme & J Saramäki. ‘Temporal Networks’, *Physics Reports*, **519**(3), 97–125.
- [64] E M Izhikevich. ‘Polychronization: Computation with Spikes’, *Neural Computation*, **18**(2) (2006), 245–282.
- [65] K Kaneko. ‘Dominance of Milnor attractors in globally coupled dynamical systems with more than 7 ± 2 degrees of freedom’, *Phys. Rev. E* **66** (2002),
- [66] K Kaneko. ‘Prevalence of Milnor Attractors and Chaotic Itinerancy in ‘High’-dimensional Dynamical Systems’, In *Synchronization: Theory and Application*, NATO Sci. Ser. II Math. Phy. Chem, Dordrecht, The Netherlands, Kluwer Acad. Publ. **109** (2003), 65–77.
- [67] N Kashtan & U Alon. Spontaneous evolution of modularity and network motifs, *PNAS* **102** (39) (2005), 13773–13778.
- [68] D Kempe & F McSherry. ‘A decentralized algorithm for spectral analysis’, *J. of Comp. and Sys. Sc.* **74** (2008), 70–83.
- [69] J Kleinberg. ‘Complex networks and decentralized search algorithms’, *Proc. Int. Conf. Math., Madrid, Spain* (2006), 1019–1044.
- [70] M Krystek, K Kurowski, A Oleksiak, & K Rządca. ‘Comparison of centralized and decentralized scheduling algorithms using GSSIM simulation environment’, *Grid Computing: Achievements and Prospects* (Springer, 2008), 185–196.
- [71] P Kundur. *Power System Stability and Control* (McGraw-Hill Inc, New York, 1994).
- [72] Y Kuramoto. ‘Self-entrainment of a population of coupled nonlinear oscillators’, (International Symposium on Mathematical Problems in Theoretical Physics, ed H Araki, Springer Lecture Notes in Physics **39**, Springer, New York, 1975), 420–422.
- [73] Y Kuramoto. *Chemical Oscillations, Waves, and Turbulence* (Springer, Berlin, 1984).
- [74] Yu A Kuznetsov, S Rinaldi, & A Gagnani. ‘One-parameter bifurcations in planar Filippov systems’, *Int. J. of Bifur. and Chaos* **13**(8) (2003), 2157–2188.
- [75] L Lamport. ‘Time, Clocks, and the Ordering of Events in a Distributed System’, *Comm. of the ACM*, **21**(7) (1978), 558–565.
- [76] H B Lawson. ‘Foliations’, *Bull. AMS* **80**(3) (1974), 369–418.
- [77] W Lu, F M Atay, & J Jost. ‘Chaos synchronization in networks of coupled maps with time-varying topologies’, *Eur. Phys. J. B* **63** (2008), 399–406.
- [78] J N Mather. ‘Stratifications and mappings’, *Proceedings of the dynamical systems conference, Salvador, Brazil* (ed M Peixoto) (Academic Press, New York, 1973), 195–223.
- [79] S-W Mei & J-Q Zhu. ‘Mathematical and Control Scientific Issues of Smart Grid and Its Prospects’, *Acta Automatica Sinica* **39**(2) (2013), 119–131.
- [80] I Melbourne & A Windsor. ‘A C^∞ diffeomorphism with infinitely many intermingled basins’, *Erg Thy Dynam Sys* **25** (2005), 1951–1959.
- [81] N Minorsky. *Theory of nonlinear control systems* (McGraw-Hill, New York, 1969).
- [82] A Morrison, M Diesmann & W Gerstner. ‘Phenomenological models of synaptic plasticity based on spike timing’, *Biol. Cyber.* **98** (2008), 459–478.

- [83] E Nijholt, B Rink, & J Sanders. ‘Graph fibrations and symmetries of network dynamics’, preprint, 2014.
- [84] J P Pade and T Pereira. ‘Improving Network Structure can lead to Functional Failures’, *Sci. Rep.* **5** 9968 (2015), doi: 10.1038/srep09968.
- [85] C A Pinto & M Golubitsky. ‘Central pattern generators for bipedal locomotion’, *J. Math. Biol.*, **53** (2006) 474–489.
- [86] A Pikovsky, M Rosenblum, & J Kurths. *Synchronization: A Universal Concept in Nonlinear Sciences* (Cambridge University Press, 2003).
- [87] C Petermann, S B Amor, & A Bui. ‘A complex system approach for a reliable Smart Grid modeling’, *Adv. in Knowledge-Based and Intelligent Information and Engineering Systems* (M Grana *et al.* (Eds), IOS Press, 2012), 149–158.
- [88] L A Plana, S B Furber, S Temple, M Khan, Y Shi, J Wu, & S Yang. ‘A GALs Infrastructure for a Massively Parallel Multiprocessor’, *IEEE Design & Test of Computers* **24**(5) (2007), 454–463.
- [89] M I Rabinovich & P Varona. ‘Robust transient dynamics and brain functions’, *Frontiers in computational neuroscience*, **5**(4) (2011).
- [90] M I Rabinovich, V S Afraimovich, C Bick & P Varona. ‘Information flow dynamics in the brain’, *Physics of Life Reviews*, **9**(1) (2012), 51–73.
- [91] J G Restrepo, E Ott, & B R Hunt. ‘Emergence of synchronization in complex networks of interacting dynamical systems’, *Physica D* **224** (2006), 114–122.
- [92] B Rink & J A Sanders. ‘Coupled cell networks and their hidden symmetries’, *SIAM J. Math. Anal.* **46**(2) (2014), 1577–1609.
- [93] B Rink & J A Sanders. ‘Coupled cell networks: semigroups, Lie algebras and normal forms’, *Trans. Amer. Math. Soc.*, to appear.
- [94] M Rohden, A Sorge, M Timme, & D Witthaut. ‘Self-Organized Synchronization in Decentralized Power Grids’, *Phys. Rev. Lett.* **109** (2012).
- [95] H Sakaguchi & Y Kuramoto. ‘A solvable active rotator model showing phase transition via mutual entrainment’, *Prog. Theor. Phys.* **76** (1986), 576–581.
- [96] D Šišlák, J Samek, & M Pěchouček. ‘Decentralized Algorithms for Collision Avoidance in Airspace’, *Proc. of the 7th int. joint conf. on Autonomous agents and multiagent systems – Vol. 2* (2008), 543–550.
- [97] O Stenflo. ‘Uniqueness of invariant measures for place-dependent random iterations’, *IMA Vol. Math. Appl.* **132** (2002), 13–32.
- [98] S H Strogatz. ‘From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators’, *Physica D* **143** (2000), 1–20.
- [99] S H Strogatz. *Sync: The Emerging Science of Spontaneous Order* (Penguin, 2004).
- [100] A S Tanenbaum & M van Steen. *Distributed Systems: Principles and Paradigms* (Prentice Hall, 2nd edition, 2006).
- [101] K Weisenfeld, P Colet, & S Strogatz. ‘Frequency locking in Josephson arrays: Connection with the Kuramoto model’, *Phys. Rev. E* **57**(2) (1998), 1563–1569.
- [102] H Whitney. ‘Analytic extensions of differentiable functions defined on closed subsets’, *Trans AMS* **36** (1934), 63–89.
- [103] D Witthaut & M Timme. ‘Braess’s paradox in oscillator networks, desynchronization and power outage’, *New J. of Physics* **14** (2012), 083036.

E-mail address: c.bick@exeter.ac.uk

DEPARTMENT OF MATHEMATICS, RICE UNIVERSITY MS-136, 6100 MAIN ST.,
HOUSTON, TX 77005, USA

Current address: Department of Mathematics, University of Exeter, Exeter EX4
4QF, UK

E-mail address: `mikefield@gmail.com`, `Michael.Field@imperial.ac.uk`

DEPARTMENT OF MATHEMATICS, RICE UNIVERSITY MS-136, 6100 MAIN ST.,
HOUSTON, TX 77005, USA

Current address: Department of Mathematics, Imperial College, SW7 2AZ, UK