Probability Analysis of Linear Time Logic Statements on Infinite Walks in Large Random Kripke Structures

Joran van den Bosse

1 July 2018

Abstract

Model checking is concerned to check the functionality of software and hardware systems. Those systems are discretised by labelled, directed graphs, which are called Kripke Structures. Due to the so-called state explosion problem those systems can become too large to be modelled by deterministic graphs This report uses the Erdös-Rényi random graph model to model large systems. Linear Time Logic is used to describe events related to critical states in systems. Probabilities that basic Linear Time Logic statements hold, are computed. Former research showed that first order logic for graphs satisfied a zero-one law. In this paper it is investigated to what extent that result can be extended to Linear Time Logic statements on labelled, directed random graphs.

1 Introduction

When software or hardware systems are used, they are desired to satisfy certain requirements. For instance, it is desirable that no states of the system can be reached that cause the system to crash. Model checking is a technique in computer science that checks whether software or hardware systems satisfy certain properties [1, page 330]. In this report we will be concerned with the probability that events related to critical states occur, such as their reachability.

According to [1, page 331] model checking often considers discrete models of such systems. One method to do this is to describe the system through so-called *Kripke Structures*. These structures are directed graphs where each vertex represents a state in the systems and the directed edges represent state transitions. Each vertex contains a label that describes the properties of a certain state in the system. One of the vertices in the graph is the initial state where a system starts running. An execution of the system is then modelled by a walk through the graph.

Model checking statements are formulated using some specific logic. This paper focuses on statements that are formulated in Linear Time Logic, which is described on [1, page 334-336]. Linear Time Logic statements describe properties of walks in Kripke Structures. For instance, the statement "all vertices in the walk are healthy states" belongs to this logic. In the next section a formal definition of the logic is given. This paper focuses on the probability that all walks in a random graph starting from a fixed starting vertex satisfy a certain Linear Time Logic statement.

As is described on [1, page 343-344] hardware systems can have a lot of parallel components, causing a system to be able to reach a lot of different states. When such systems are modelled by directed graphs, the number of vertices can become large and it can be hard to describe such systems using deterministic graphs. This phenomenon is called the *state explosion problem*.

These large systems can be modelled using Erdös-Rényi random graphs [2]. In this model the number of vertices is a deterministic choice. However, the edges in the graph are selected at random using a given existence probability. This model was created by P. Erdös and A. Rényi around 1960 [2] and has been applied to model networks. In this paper we use it to model the Kripke Structures. In large systems there is a high number of states but the exact phase transitions are unknown. However, the density of phase transitions often can be determined. Therefore the random graph can be a good model to describe these systems.

After P. Erdös and A. Rényi had developed the random graph model a lot of research has been done on this field in order to extend the theory of deterministic graphs to the field of random graphs. For instance, in the theory of deterministic graphs the term *connectivity* is used to describe graphs in which a path exists between any pair of vertices. An example of a connected graph is shown in figure 1. The question arose with what probability structural properties in graphs such as connectivity occurred in random graphs.

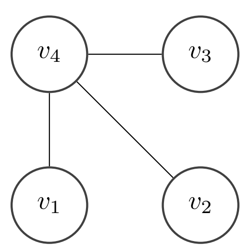

Figure 1: A connected graph

Former research on random graphs has shown that such structural properties of random graphs satisfy zero-one laws [3, 4, 8, 9]. That means that the probability that certain properties appear in a random graph either tend to 0 or to 1 if the number of vertices of a graph grows infinitely large. Whether the probability tends to 0 or 1, depends on the edge existence probability. There exists some threshold value of the edge existence probability at which the structural property appears or disappears. For instance, in [2] it is shown that $p(n) = \frac{\log(n)}{n}$ is a threshold for connectivity. That is, if the edge existence probability is larger than this threshold and the number of vertices is large, it is almost surely connected and if the edge existence probability is lower, the graph is almost surely not connected. Since zero-one laws exist on a whole class of such properties, described by the so-called *first order logic for graphs*, it is possible that some class of Linear Time Logic statements also satisfies a zero-one law.

The aim of this probability analysis is to investigate whether such a zero-one law exists by computing the probability that basic Linear Time Logic statements hold. In section 2 formal definitions of the used structures are given. In section 3 an extension of former research on zero-one laws onto the model checking problem is considered. In section 4 a result on strong connectivity in directed random graph is presented. Then in sections 5 and 6 the probability analysis is executed. As a result of this analysis a further research on zero-one laws in model checking could become of interest.

2 Definitions

Before we start the probability analysis let us first consider some definitions. Let us firstly give a definition for the random graph model, as described by [5, page 2].

Definition 1 (Binomial Directed Random Graph)**.** *Let us consider a simple directed graph that contains n vertices and a real number p, with $p \in (0, 1)$. For each of the $n(n-1)$ ordered pairs of vertices let a directed edge exist between these vertices with probability p and no edge with probability $1 - p$. This graph is considered a* Binomial Directed Random Graph*. Such a graph is denoted by $G(n, p)$.*

For the purpose of the model checking problem no loops are allowed in the graphs. After all in model checking only transitions between different states of a model are of interest. This is a slight difference to the model described by [5, page 2].

Once the vertices and edges of the graph are created according to definition 1, labels are added to the graph. In this paper the only considered labels are the colours red and blue, where each state has exactly one of the two colour labels. The colour blue represents a healthy state and the colour red represents a critical state of a modelled system. Two methods of vertex labelling are considered, a random labelling and a deterministic labelling. These are defined as follows.

Definition 2 (Random Vertex Labelling). *Let us consider a random graph* $G(n,p)$. *Choose a fixed real number* q, *with* $q \in (0,1)$. *Each vertex of* $G(n,p)$ *receives a red label with probability* q *and a blue label with probability* $1-q$.

Definition 3 (Deterministic Vertex Labelling). *Let us consider a random graph* $G(n,p)$. *Choose a fixed integer* r, *with* $r > 0$. *If* $n \leq r$ *all vertices of* $G(n,p)$ *receive a red label. If* $n > r$ *the first* r *vertices are labelled red and the remaining vertices receive a blue label.*

The labelled, directed, binomial random graphs that are created using a combination of definition 1 and either definition 2 or 3 are the graphs considered in this paper. In order to apply model checking we consider walks through the graph, as described on [1, page 335].

Definition 4 (Walk). *Let us consider a directed graph* $G(V,E)$, *where* V *is the vertex set and* E *the edge set of the graph. A* finite walk *of length* k *through* G *is a sequence of vertices of* G, *denoted as* $\pi =< \pi_0, \pi_1, ..., \pi_k >$, *such that for all* i *between 0 and* $k-1$ *the edge* (π_i, π_{i+1}) *is in edge set* E. *Here* k *is a finite integer, with* $k > 0$. *An* infinite walk *is a sequence* $\pi =< \pi_0, \pi_1, ... >$ *such that for all positive* i *the edge* (π_i, π_{i+1}) *is in the edge set* E. *Vertex* π_0 *is the* starting vertex *of the walk. The* tail of the walk *starting at vertex* π_j *is the sequence* $\pi^j =< \pi_j, \pi_{j+1}, ..., \pi_k >$ *in case of a finite walk or the sequence* $\pi^j =< \pi_j, \pi_{j+1}, ... >$ *in case of an infinite walk. The* length of a walk, *denoted as* $|\pi|$ *is the integer* k *if the walk is finite or* ∞ *if it is an infinite walk.*

Note that [1] uses the word "path" instead of "walk". This paper stays in line with the choice of words in [6], since that is an elementary course on graph theory. In this paper only infinite walks will be considered and the first generated vertex in a random graph $G(n,p)$ will be the starting vertex of each walk.

A walk through a random graph models the running of a system that is considered by model checking. Now the mathematical framework is given, let us consider the formal definitions of the discrete models of those systems. Those models are called *Kripke Structure*. We will consider the definition as given by [1, page 331-332].

Definition 5 (Kripke Structure). *A* Kripke Structure *over a set of Atomic Propositions, denoted as* AP, *is a system described by the triple* (S,R,I). *Here* S *is a set of* states, R *is a set of* transition relations, *with* $R \subseteq S \times S$, *and* I *is an* interpretation *of the atomic propositions belonging to the states. For each proposition* $p \in AP$ *and for each state* $v \in S$ *either* $p \in I(v)$ *or* $p \notin I(v)$. *Therefore* I *is a function defined on the space* $I : S \to 2^{AP}$.

The labelled, directed random graphs model Kripke Structures. We are concerned with the properties of walks through those graphs. Those properties are described by the language of Linear Time Logic, as described on [1, page 334-336].

Definition 6 (Linear Time Logic)**.** *The* Linear Time Logic *in model checking defines whether a walk* π *in a Kripke Structure meets a statement* ϕ*, denoted as* $\pi \models \phi$*. A statement* ϕ *in the language of* Linear Time Logic *is built using the following grammar rule:*

$$\phi := p|\neg\phi|\phi_1 \vee \phi_2|X(\phi)|F(\phi)|G(\phi)|U(\phi,\psi)|WU(\phi,\psi).$$

Here $X(\phi)$ *is pronounced as* Next ϕ*,* $F(\phi)$ *is pronounced as* Finally ϕ*,* $G(\phi)$ *is pronounced as* Generally ϕ*,* $U(\phi,\psi)$ *is pronounced as* ϕ Until ψ *and* $WU(\phi,\psi)$ *is pronounced as* ϕ Weak Until ψ*. These statements have the following logical interpretation:*

$$\begin{aligned}
\pi \models p &\quad\Leftrightarrow\quad p \in I(\pi_0)\\
\pi \models \neg\phi &\quad\Leftrightarrow\quad \pi \not\models \phi\\
\pi \models \phi_1 \vee \phi_2 &\quad\Leftrightarrow\quad \pi \models \phi_1 \vee \pi \models \phi_2\\
\pi \models X(\phi) &\quad\Leftrightarrow\quad \pi^1 \models \phi\\
\pi \models F(\phi) &\quad\Leftrightarrow\quad \exists k \in \{k|0 \leq k < |\pi|\} : \pi^k \models \phi\\
\pi \models G(\phi) &\quad\Leftrightarrow\quad \forall k \in \{k|0 \leq k < |\pi|\} : \pi^k \models \phi\\
\pi \models U(\phi,\psi) &\quad\Leftrightarrow\quad \exists k \in \{k|0 \leq k < |\pi|\} : (i < k \rightarrow \pi^i \models \phi) \wedge (\pi^k \models \psi)\\
\pi \models WU(\phi,\psi) &\quad\Leftrightarrow\quad \pi \models U(\phi,\psi) \vee \pi \models G(\phi).
\end{aligned}$$

Let us explain this definition with an example. Consider the graph in figure 2. Here some arbitrary Kripke Structure of six vertices is presented. The blue vertices are represented by circles and the red vertices by squares. Let v_1 be the initial state of the system and let us consider the finite walk $\pi =< v_1, v_2, v_4, v_5, v_2, v_1 >$. Since the initial state is blue, the statement $\pi \models blue$ is true. The next vertex in the walk, v_2, is also a blue state. Consequently we have $\pi \models X(blue)$. The fourth vertex in the walk, v_5, is a red state. In this walk a red state is reached, so $\pi \models F(red)$. The walk contains both red and blue vertices. Consequently $\pi \models \neg(G(red) \vee G(blue))$.

In order to show the difference between "Until" and "Weak Until", consider a second walk $\mu =< v_1, v_2, v_1, v_6 >$. In the walk π there is a vertex at which the property "red" is true, vertex v_5, and for which all preceding vertices are blue. Therefore $\pi \models U(blue, red)$. However, the walk μ only contains blue vertices. Since no red vertex is reached, we have that $\mu \not\models U(blue, red)$. In this case the difference between "until" and "weak until becomes obvious. Since all vertices are blue we have $\mu \models G(blue)$. If we apply the definition of "weak until" we see that $\mu \models WU(blue, red)$, which is differs from the "until" function.

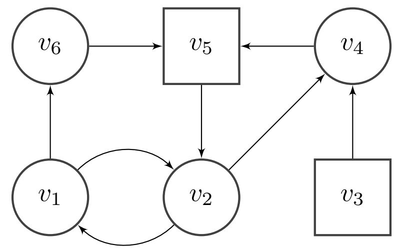

Figure 2: A Kripke Structure of six states

3 Zero-One Law

Let us consider a binomial undirected random graph which is defined similarly as in Definition 1 with the difference that now edges are undirected. On [3, page 98-99] the definition of the First Order Logic for Graphs is given.

Definition 7 (First Order Logic for Graphs)**.** *A statement ϕ in the language of* First Order Logic for Graphs *may contain the following elements: vertices, the equality sign (=) to state that two mentioned vertices are the same vertex in the graph, the adjacency sign ($\sim$) to state that two vertices are adjacent, the logical connectives $\wedge$, $\vee$, $\neg$ and $\rightarrow$ and the quantifiers $\forall$ and $\exists$. Here the $=$ and $\sim$ sign are assumed symmetric and $\sim$ is assumed antireflexive.*

Independently [8] and [9] showed that statements formulated in First Order Logic for Graphs satisfy a zero-one law. That is, if the number of vertices goes to infinity the probability that such a statement is true goes to either 0 or 1.

Since that result applies for all statements in an entire logic the question arises whether the Linear Time Logic statements from model checking also satisfy such a zero-one law. If it were possible to write Linear Time Logic statements in the form of First Order Logic for Graphs a zero-one law would immediately have been proven to exist.

However, the Linear Time Logic statements causes problems when we try to rewrite statements into the First Order Logic for Graphs. Firstly, Linear Time Logic statements rely on the vertex labelling. The first order logic can only refer to vertices in general. In order to refer to vertices with certain properties, in this case the colours red and blue, sets have to be introduced. Allowing sets requires us to consider Second Order Logic, which does allow sets [7, page 143]. Then a set could be defined for each atomic proposition. For instance the statement $G(red)$ is true if every vertex that is reachable from the starting vertex, is in the set of red vertices. In first order logic such sets are not allowed.

The reason why first order logic does not allow vertex labels, is indirectly explained in [3]. It is stated there that first order logic statements hold up to isomorphism. Since there exist graphs that posses automorphisms this prevents us from defining Linear Time Logic statements as first order logic statements.

(a) v_1 is blue and v_2 is red

(b) v_1 is red and v_2 is blue

Figure 3: Two graphs that have equal structure for first order logic

For instance, consider the graph of two vertices where there exists an edge in both directions and consider the starting vertex to be blue and the second vertex to be red, as displayed in figure 3a. Compare this to a similar graph where only the colour labels have changed (figure 3b). Let π be the infinite walk starting at initial state v_1 and then following the only outgoing edge from each state: $\pi =< v_1, v_2, v_1, v_2, ... >$. Since these two graphs are isomorphic, the statement $\pi \models red$ would be either true or false for both graphs if it could be rewritten as a statement from the First Order Logic for Graphs. Since the statement is actually false in the first graph and true in the second graph, this is not a first order logic statement.

A second problem is that the random colouring makes it uncertain how many vertices will be red and how many will be blue. Logical statements have to be finite. The statement "generally blue" is only true for a walk if no red vertex is reached. If it is uncertain how many red vertices exist, no finite statement can be made that states that no red vertices are reached. Likewise, the existence of a k-cycle, a cycle of k vertices, in an undirected random graph is a first order logic statement, while the existence of a cycle in general is not [3]. In order to apply logic on graphs to statements on walks we therefore are restricted to a deterministic colouring.

Thirdly, Kripke Structures are modelled by directed graphs instead of undirected graphs. In the definition of the First Order Logic for Graphs adjacency is assumed to be symmetric. However, in a directed graph symmetry is not a necessary condition for adjacency. If vertices v and w are in a directed graph it is possible that the edge (v, w) exists while the edge (w, v) does not. It should be investigated whether the proofs of the theorems considering the zero-one laws in the First Order Logic for Graphs depend on the symmetry argument before they can be applied.

Since Linear Time Logic statements cannot be written as First Order Logic statements the proven theorems about zero-one laws cannot be directly applied. Nevertheless it is still possible that Linear Time Logic statements still satisfy some zero-one law. To investigate this, in this paper a probability analysis has been done.

4 Strong Connectivity in Binomial Direct Random Graphs

While the most results on random graphs consider undirected random graphs, [5] provides a useful result on directed random graphs. This result can be used in the probability analysis. After all, if in our case the graphs turn out to be almost surely strongly connected, all states are reachable from the initial state. In other words, reachability comes down to the analysis of components.

In [5] a threshold for strong connectivity is found. The following result states that a constant edge existence probability is greater than the given probability threshold. It is then quickly deduced that in this case graphs are almost surely strongly connected.

Theorem 1. *Let $G(n,p)$ be a Binomial Directed Random Graph and let the edge appearance probability p be constant. Then the probability that $G(n,p)$ is strongly connected goes to 1 as $n\infty$.*

Proof. Let $\mathcal{S}$ be the property that a directed graph is strongly connected. According to [5] $\widehat{p} = \frac{\ln n + c}{n}$ is a threshold for strongly connectedness. That is, strongly connectedness satisfies a zero-one law except if the chosen probability p is the same order as $\hat{p}$. In [5] it is shown that if for some random graph $G(n,p)$ we have that $p >> \hat{p}$, which holds if $\lim_{n\to\infty} \frac{p}{\hat{p}} > \infty$, then $\lim_{n\to\infty} \mathbb{P}(\mathbb{G}(n,p) \in \mathcal{S}) = 1$. Now, let p be constant. In that case we have:

$$\begin{aligned}
\lim_{n\to\infty} \frac{p}{\widehat{p}} &= \lim_{n\to\infty} \frac{pn}{\ln n + c} \\
&= \lim_{n\to\infty} \frac{p}{\left(\frac{1}{n}\right)} \\
&= \lim_{n\to\infty} pn \\
&= \infty.
\end{aligned}$$

So when p is constant then we have that $p >> \hat{p}$. Therefore a directed random graph with constant p is strongly connected with probability 1 for $n \to \infty$. □

In the following sections cover computations on basic Linear Time Logic Statements in random graphs with fixed edge probability p. Firstly we consider the Random Vertex Colouring of Definiton 2, secondly we consider the Deterministic Vertex Colouring of Definition 3. In those computations this result can be applied.

5 Random Colouring

In this section the probabilities that all walks starting at a fixed starting vertex meet some basic Linear Time Logic statement are computed. The graph is assumed to have a random vertex colouring. The first generated vertex by random graph $G(n,p)$ is considered to be the starting vertex and is denoted by v_1. Since we now focus on properties of all walks starting at the same initial state we use the following notation:

$$v \models \phi \quad \Leftrightarrow \quad \forall \pi \text{ s.t. } \pi_0 = v : \pi \models \phi.$$

Here v is any state in the Kripke Structure and π is an infinite walk. Since we denote our initial state as v_1, that state will be plugged into the statement above.

5.1 Generally Red and Generally Blue

Firstly consider the functions $G(red)$ and $G(blue)$. The probabilities that are computed are $\mathbb{P}(v_1 \models G(red))$ and $\mathbb{P}(v_1 \models G(blue))$. To compute the probability that these events occur let us use Theorem 1 that $G(n,p)$ is almost surely, that is with probability 1, strongly connected if $n \to \infty$. In a strongly connected graph each vertex can be reached from the starting vertex. Therefore the events $v_1 \models G(red)$ and $v_1 \models G(blue)$ are only true if respectively all vertices are red or all vertices are blue. As a result we have the following results:

$$\mathbb{P}(v_1 \models G(red)) = q^n$$
$$\mathbb{P}(v_1 \models G(blue)) = (1-q)^n.$$

Since we let the number of vertices grow infinitely large these probabilities vanish: $\mathbb{P}(v_1 \models G(red)) \to 0$ and $\mathbb{P}(v_! \models G(blue)) \to 0$ as $n \to \infty$.

5.2 Finally Red and Finally Blue

Secondly let us compute the probabilities that the event "finally red" or "finally blue" occurs on each walk starting at v_1. By definition the event $v_1 \models F(red)$ is true if for all walks starting at v_1 there exists a $k \geq 0$ such that $\pi^k \models red$. The first step of the computation of $\mathbb{P}(v_1 \models F(red)$ is conditioning on the colour of v_1. After all, if v_1 is red the event $\pi^0 \models red$ is true and thus we have found a nonnegative k such that $\pi^k \models red$ and such that for all i with $0 \leq i < k$ we have that $\pi^i \models blue$. The second condition is a result from the fact that

$$\{i \,|\, 0 \leq i < 0, i \in \mathbb{Z}\} = \emptyset.$$

However if v_1 is blue the statement $v_1 \models F(red)$ is only true if there is no possibility of infinitely repeating the same blue cycle. After all the probability that all vertices are blue equals $(1-q)^n$, which vanishes as $n \to \infty$. Thus if no path of blue vertices to a blue cycle exists, almost surely a red vertex is reached eventually and the statement is true.

This can be demonstrated using the Kripke Structure in figure 2 from section 2. There a blue cycle exists containing v_1, the initial vertex, and v_2. This allows the existence of walk $\pi =< v_1, v_2, v_1, v_2, ... >$, which is a cycle that only contains blue vertices. Consequently, $\pi \not\models F(red)$ and therefore $v_1 \not\models F(red)$. However, if edge (v_2, v_1) did not exist, each infinite walk would inevitably reach vertex v_5, which is coloured red. Then all walks starting at v_1 would meet $F(red)$.

The aim now is to show that v_1 is almost surely part of some blue cycle and therefore the event $v_1 \models F(red)$ almost surely is false given that v_1 is blue. The probability that there exists an outgoing edge from v_1 to any other edge equals p by definition of $G(n,p)$. Similarly, an incoming edge to v_1 from an arbitrary vertex exists with probability p. Thus the probability that both an outgoing and incoming edge exists to the same vertex equals p^2. Moreover the probability that any vertex is blue equals $1-q$. Therefore the probability of both an outgoing and incoming edge occurring from v_1 to any other arbitrary vertex and that the other vertex is blue, equals $p^2(1-q)$. Let us define the random variable B_n as the total number of blue vertices with both an outgoing and incoming edge from v_1. Let us define the indicator random variables I_j as follows:

$$I_j = \begin{cases} 1 & \text{if edges } (v_1, v_j) \text{ and } (v_j, v_1) \text{ exist and } v_j \text{ is blue} \\ 0 & \text{else} \end{cases}$$

We have shown in the previous paragraph that

$$I_j \sim \text{Bernoulli}(p^2(1-q)).$$

Now consider our definition of B_n:

$$B_n = \sum_{j=2}^{n} I_j.$$

Since the distribution of the indicators is independent of j, we have that

$$B_n \sim \text{Bin}(n-1, p^2(1-q)).$$

We can now deduce that $\mathbb{P}(B_n = 0) = (1-p^2(1-q))^{n-1}$. Since p and q are fixed probabilities strictly between 0 and 1 the number $(1-p^2(1-q)$ is also strictly between 0 and 1. As a result $\mathbb{P}(B_n = 0) \to 0$ as $n \to \infty$. In other words, given that v_1 is blue it is almost surely one of two vertices of some blue cycle in $G(n,p)$. Therefore given that v_1 is blue the statement $v_1 \models F(red)$ is almost surely false.

Now consider the required probability that the statement ”finally red” is true can be deduced using conditioning on the colour of v_1:

$$\begin{aligned}\lim_{n\to\infty}\mathbb{P}(v_1 \models F(red)) &= \lim_{n\to\infty}(\mathbb{P}(v_1 \models F(red) \mid v_1 \models red)\mathbb{P}(v_1 \models red) \\ &+ \mathbb{P}(v_1 \models F(red) \mid v_1 \models blue)\mathbb{P}(v_1 \models blue)) \\ &= 1 \cdot q + 0 \cdot (1-q) \\ &= q.\end{aligned}$$

In short, the probability that the statement ”finally red” is true for all walks starting at v_1, equals q. Similarly it can be deduced that ”finally blue” is true for all walks starting at v_1 with probability $1-q$.

Note that the stochastic element of model checking only appears in the creation of the graph, not in the selection of walks. Intuitively one could expect these probabilities to be equal to 1. If we consider the graph to be a Markov chain and we would like to know whether an arbitrary walk would finally reach a red vertex, then we would indeed end up with probability 1. However, we are interested in the probability that every walk reaches a red vertex finally. This difference is similar to the use of a software package. Let a robot take arbitrary decisions. Then the system will almost surely end up in a critical state. However, a human user could understand the package and know how to avoid the system to collapse, which resembles a blue cycle. This causes the probability of interest to be lower than 1.

5.3 Next Red and Next Blue

The third set of statements to consider are ”next red” and ”next blue” By definition the event $v_1 \models X(red)$ is true if the event $\pi^1 \models red$ is true for all walks starting at v_1. To be certain that the second vertex of each walk starting at v_1 is red, we have to consider the probability that v_1 has no outgoing edges to blue vertices. The probability that an outgoing edge exists to any arbitrary vertex equals p and the probability that such a neighbour is blue equals $1-q$. Therefore similarly to previous computation a random variable B_n can be defined. Only this time this time the random variable is the number of outgoing edges to blue vertices. Therefore we have that

$$B_n \sim \text{Bin}(n-1, p(1-q)).$$

This yields the following result:

$$\mathbb{P}(B_n = 0) = (1 - p(1-q))^n \to 0 \text{ as } n \to \infty.$$

Therefore $\mathbb{P}(v_1 \models X(red) \to 0$ as $n \to \infty$ and similarly $\mathbb{P}(v_1 \models X(blue) \to 0$ as $n \to \infty$.

5.4 Red Until Blue and Blue Until Red

The next considered statements are $v_1 \models U(red, blue)$ and $v_1 \models U(blue, red)$. The statement "red until blue" is true for all walks if there exists a $k \geq 0$ such that $\pi^k \models blue$ and for all i such that $0 \leq i < k$ we have that $\pi^i \models red$.

These statements appear to be similar to "finally blue" and "finally red". In fact, those statements turn out to be equivalent.

Theorem 2. *Consider a Kripke Structure with initial state v_1. The statements $v_1 \models F(red)$ is true if and only if the statement $v_1 \models U(blue, red)$ is true. Similarly, the statement $v_1 \models F(blue)$ is true if and only if the statement $v_1 \models U(red, blue)$ is true.*

Proof. Let us prove the equivalence between $F(red)$ and $U(blue, red)$. The other equivalence is proved similarly.
Let us assume that $v_1 \models F(red)$. Consider an arbitrary infinite walk π with $\pi_0 = v_1$. By definition some k exists such that vertex π_k is red. Let us now consider the following set:

$$R = \{k \mid \pi^k \models red\}.$$

Now let us consider the minimum of set R:

$$k^* = \min_{k \in R} k.$$

Then we have that $\pi^{k^*} \models red$ and if $0 \leq i < k^*$ we have that $\pi^i \models blue$. Therefore we have that $\pi \models U(blue, red)$. Since π is an arbitrary walk starting at v_1 we have that $v_1 \models U(blue, red)$.
Conversely, assume that $v_1 \models U(blue, red)$. Let us consider an arbitrary walk π such that $\pi_0 = v_1$. By definition there exists a k such that $\pi^k \models red$. Therefore we also have that $\pi \models F(red)$ and consequently we have that $v_1 \models F(red)$. $\square$

An immediate result of theorem 2 is that $\mathbb{P}(v_1 \models U(red, blue)) \to 1 - q$ and likewise that $\mathbb{P}(v_1 \models U(blue, red)) \to q$ as $n \to \infty$. These are the required results.

Note that in general the statements $\pi \models F(\phi)$ and $\pi \models U(\psi, \phi)$ are not equivalent. For instance, let us consider the Kripke Structure in figure 2. Let $\pi =< v_1, v_6, v_5, v_2, v_1, v_2, v_1, v_2, ... >$ and compare the statements $\pi \models F(red)$ and $\pi \models U(red, red)$. The first statement is true, because vertex v_5 is red. However, the initial vertex v_1 is blue. Therefore $\pi \not\models U(red, red)$ in this case.

5.5 Red Weak Until Blue and Blue Weak Until Red

The final basic formulas of the Linear Time Logic are $WU(red, blue)$ and $WU(blue, red)$. These statements turn out to be true regardless of the considered walk. It will be shown why this is the case.

Theorem 3. *Let us consider a Kripke Structure and any arbitrary walk π. Then the statements $\pi \models WU(red, blue)$ and $\pi \models WU(blue, red)$ are true.*

Proof. Let us prove the theorem for the statement $\pi \models WU(red, blue)$. The second statements follows similarly.
The statement is true for an arbitrary walk π if one of two options is true. The first option is that there exists a $k \geq 0$ such that $\pi^k \models blue$ and for all i such that $0 \leq i < k$ it holds that $\pi^i \models red$. The second option is that for all $k \geq 0$ it holds that $\pi^k \models red$. The aim is to show that always one of the options is true. Let us consider a walk for which the second option is false. Then there has to exist a $k \geq 0$ such that $\pi^k \models blue$. Let us pick the smallest k such that $\pi^k \models blue$. This k can be picked in the definition of $U(red, blue)$. Therefore the first option is true.
Conversely, if the first option is false then no k can be found such that $\pi^k \models blue$. As a result $G(red)$ is true, which is our second option.
Since one of both statements has to be true, the statement $WU(red, blue)$ is true for all walks. □

By theorem 3 it is guaranteed that $\mathbb{P}(v_1 \models WU(red, blue)) \to 1$ and similarly $\mathbb{P}(v_1 \models WU(blue, red)) \to 1$ as $n \to \infty$.

These were all the basic functions of the Linear Time Logic. It can be seen that the probability that all walks satisfy a basic Linear Time Logic statement either satisfies a zero-one law or solely depends on the colour of the starting vertex. This partly confirms the conjecture that all Linear Time Logic Statements satisfy some zero-one law. In order to investigate this possibility further, let us consider some embedded statements.

5.6 Finally Generally Red and Finally Generally Blue

Firstly, consider the statements $FG(red)$ and $FG(blue)$. All walks satisfy these statements if for all walks there exists an $i \geq 0$ such that for all $k \geq i$ it respectively holds that $\pi^k \models red$ or $\pi^k \models blue$. That means that each walk reaches a point from which all vertices in the tail of the walk have the same colour. Since by Theorem 1 the graph is almost surely strongly connected if n is large, this can only be true if all n vertices have the same colour. All vertices are red with probability q^n and all vertices are blue with probability $(1-q)^n$. Since those probabilities go to 0 as $n \to \infty$ it holds that $\mathbb{P}(v_1 \models FG(red)) \to 0$ and $\mathbb{P}(v_1 \models FG(blue)) \to 0$ as $n \to \infty$.

5.7 Generally Finally Red and Generally Finally Blue

Secondly consider the reverse embedding: $GF(red)$ and $GF(blue)$. These statements hold for all walks if for all $i \geq 0$ there exists a $k \geq i$ such that respectively $\pi^k \models red$ or $\pi^k \models blue$. Let us focus on $GF(red)$, as $GF(blue)$ is shown similarly. Since $\mathbb{P}(v_1 \models X(blue) \to 0$ there almost surely exists an edge from v_1 to a blue vertex. In the computation of "Finally blue" it was shown that the probability that v_1 is not contained in any 2-cycle of blue vertices goes to 0. Since that proof was not restricted to any specific vertex as only general properties of the graph were used, it can likewise be shown that the blue neighbour of v_1 is almost surely contained in some blue cycle. Therefore a walk exist of which the tail only contains blue vertices. As a result $\mathbb{P}(v_1 \models GF(red)) \to 0$ and $\mathbb{P}(v_1 \models GF(blue)) \to 0$ as $n \to \infty$.

5.8 Red Until Generally Finally Red

Finally, consider one larger embedded statement: $U(red, GF(red)$. This statement is true if for all walks some $k \geq 0$ exists such that for all i such that $0 \leq i < k$ it holds that $\pi^i \models red$ and that for all $a \geq k$ there exists a $b \geq a$ such that $\pi^b \models red$. One requirement of this long statement is that the tail of the walk cannot only contain blue vertices. As was shown previously the starting vertex almost surely has a blue neighbour that is contained in some 2-cycle with only blue vertices. Therefore $\mathbb{P}(v_1 \models U(red, GF(red)) \to 0$ as $n \to \infty$.

As can be seen a if some arbitrary embedded statements are selected the probability that all walks satisfy the statements either goes to 0 or 1. Therefore the possibility of a zero-one law on Linear Time Logic statements still exists.

6 Deterministic Colouring

Let us now focus on the deterministic vertex colouring. Again the aim is to compute the probability that all walks starting at v_1 satisfy some basic Linear Time Logic statement. The difference with previous section is the colouring method. Since the number of red vertices is finite in this section, this fraction of red vertices will vanish. With the random colouring it is expected that a fraction q of all vertices are red, with $q \in (0, 1)$. Here that fraction is $\frac{r}{n}$, with r. This goes to 0 as n becomes infinitely large. This results in some differences compared to the random colouring.

6.1 Generally Red and Generally Blue

Let us again start with the functions $G(red)$ and $G(blue)$. According to Theorem 1 the random graph is almost surely strongly connected if n becomes large. Based on Defenition 3 some integer r is fixed such that the first r vertices are red and the remaining vertices are blue. When $n \to \infty$ but r is fixed it is certain that $n > r$. Therefore blue vertices will appear in the graph. Since the graph is strongly connected there will almost surely exist a walk containing a blue vertex. Consequentely, $\mathbb{P}(v_1 \models G(red)) \to 0$ and $\mathbb{P}(v_1 \models G(blue)) \to 0$ as $n \to \infty$.

6.2 Finally Red and Finally Blue

Secondly, consider the statements $F(red)$ and $F(blue)$. When the random colouring method was applied the probability that all walks satisfied one of these statements depended on the colour of the starting vertex. In this case by assumption the starting vertex is red. Therefore $\mathbb{P}(v_1 \models F(red)) \to 1$ as $n \to \infty$.

On the other hand all walks meet $F(blue)$ if all walks at some point have to reach a vertex other than the first r vertices. The statement is therefore false if in the first r vertices some red cycle exists that can be reached from the starting vertex. The probability that such a walk exists is equal to the probability that in the finite random graph $G(r, p)$ a cycle exists that can be reached from vertex v_1.

This probability is hard to compute. Therefore this probability will be denoted with the symbol P. Therefore we have $\mathbb{P}(v_1 \models F(blue)) \to 1 - P$. However, it is possible to derive an upper and lower boundary for $1 - P$.

If v_1 has no outgoing vertices to any other red vertex, it is certain that the second vertex in each walk is blue. Each edge appears independently with probability p. Therefore, the probability that no edge exists from v_1 to any other red vertex equals $(1-p)^{r-1}$. This is a lower boundary for $\mathbb{P}(v_1 \models F(blue))$.

To determine an upper boundary, let us define R_r as the number of 2-cycles containing v_1 and only containing red vertices. Similarly to section 5.2 it can be derived that

$$\mathbb{P}(R_r = 0) = (1 - p^2 q)^{r-1}.$$

As a result we have that $P \geq (1 - p^2 q)^{r-1}$. This results in the following boundary for the probability that all walks from v_1 meet $F(blue)$:

$$(1-p)^{r-1} \leq \mathbb{P}(v_1 \models F(blue)) \leq 1 - (1 - p^2 q)^{r-1}.$$

Note that this inequality only holds if $r > 1$ and $n \to \infty$. If $r = 1$ then only v_1 is red. As a result, in that case we have that $\mathbb{P}(v_1 \models F(blue)) \to 1$. And n is chosen infinitely large in order to almost surely have outgoing edges from each red vertex to blue vertices. That is a consequence of theorem 1.

In order to determine the exact value of this probability one could simulate some $G(r, p)$ graphs to check which fraction of graphs satisfy the required property. However, that is not of interest in this report. After all, it has been determined that this probability is bounded to some value in the interval (0,1). In other words, this probability does not satisfy a zero-one law due to the finite size of the subgraph induced by all red vertices, which is $G(r, p)$. However, if one would prefer to compute the exact probability, one can use the following property.

Theorem 4. *Let us consider a random graph $G(n, p)$ with deterministic vertex colouring and let n be infinitely large. Consider all possible finite walks of r starting at vertex v_1. Then for each walk π of these finite walks it holds that $\pi \models F(blue)$ if and only if it holds that $v_1 \models F(blue)$.*

Proof. Assume that $v_1 \models F(blue)$. Then there exists no infinite walk that only contains red vertices. This means that no cycle of red vertices exists which can be reached with a path starting at v_1. Let us consider the longest walk starting at v_1 that only contains red vertices. Let this be $\pi =< v_1, \pi_1, \pi_2, ..., \pi_k >$. Since no red vertex can be reached that is in a red cycle, all vertices in π are unique. Since there exist r red vertices, we have that $k \leq r-1$. Consequently, if we have an arbitrary walk of length r that starts at v_1, it is longer than the longest walk starting at v_1 that only contains red vertices. Therefore, it contains at least one blue vertex.
Conversely, assume that $v_1 \not\models F(blue)$. Then there exists an infinite walk that only contains red vertices. Now consider the first $r + 1$ vertices of this walk. This sequence is a walk of length r with only red vertices. Therefore not all walks of length r contain at least one blue vertex. □

Theorem 4 reduces the investigation of all infinite walks starting at v_1 to the investigation of all finite walks of length r. Thus with an algorithm based on breadth first search it is possible to check whether for some generated graph $G(r, p)$ it holds that $v_1 \models F(blue)$. However, no simulation is done for this report.

6.3 Next Red and Next Blue

Thirdly consider the statements $X(red)$ and $X(blue)$. All walks meet these statements if v_1 respectively has no outgoing edges to blue or red vertices. The probability that no outgoing edges from v_1 to another red vertex exists, is finite: $(1-p)^{r-1}$. This is a result of the fact that only a finite number of vertices receives a red label, namely vertex v_1 up to v_r. Since all remaining vertices are blue, the number of blue vertices does become infinitely large as $n \to \infty$. The probability that v_1 has no outgoing edges to blue vertices equals $(1 - p)^{n-r}$, which goes to 0 as $n \to \infty$. In short, $\mathbb{P}(v_1 \models X(red)) \to (1 - p)^{r-1}$ and $\mathbb{P}(v_1 \models X(blue)) \to 0$ as $n \to \infty$.

6.4 Red Until Blue an Blue Until Red

The following statements that are considered, are $U(red, blue)$ and $U(blue, red)$. For these functions we can again use theorem 2. Consequently, it holds that $\mathbb{P}(v_1 \models U(red, blue)) \to 1 - P$ and $\mathbb{P}(v_1 \models U(blue, red)) \to 1$.

6.5 Red Weak Until Blue and Blue Weak Until Red

The final basic statements are $WU(red, blue)$ and $WU(blue, red)$. Theorem 3 showed that for all walks these statements have to be true. Therefore these probabilities do not depend on the chosen colouring. Thus we have the result that $\mathbb{P}(v_1 \models WU(red, blue)) \to 1$ and $\mathbb{P}(v_1 \models WU(blue, red)) \to 1$.

Now all the basic Linear Time Logic statements have been covered both using the random vertex colouring and the deterministic colouring. The results are summarised in table 1. The most striking difference between the results using the different colouring methods, is the fact that the finite number of red states in the deterministic colouring method resulted in finite probabilities for certain model checking statements, while the probabilities of the random colouring either satisfied a zero-one law or were fully determined by the colour of v_1. Paradoxically the deterministic colouring was introduced in order to do an attempt to write Linear Time Logic statements in terms of First Order Logic statements. If that was possible, the results on zero-one laws as proved for First Order Logic might have been applicable to the Linear Time Logic statements of model checking. However the fixing of the number of red vertices caused the opposite. It let some probabilities depend on finite structures in the graph. Therefore some probabilities did not satisfy the zero-one law.

Statement	Random Colouring	Deterministic Colouring
$\mathbb{P}(v_1 \models G(red))$	0	0
$\mathbb{P}(v_1 \models G(blue))$	0	0
$\mathbb{P}(v_1 \models F(red))$	q	1
$\mathbb{P}(v_1 \models F(blue))$	$1-q$	$1-P$
$\mathbb{P}(v_1 \models X(red))$	0	0
$\mathbb{P}(v_1 \models X(blue))$	0	$(1-p)^r$
$\mathbb{P}(v_1 \models U(red, blue))$	$1-q$	$1-P$
$\mathbb{P}(v_1 \models U(blue, red))$	q	1
$\mathbb{P}(v_1 \models WU(red, blue))$	1	1
$\mathbb{P}(v_1 \models WU(blue, red))$	1	1

Table 1: Probabilities of basic Linear Time Logic Statements

7 Conclusion

In directed random graphs with constant edge existence probability and random vertex labelling the probability that all walks starting from an initial state satisfy a basic Linear Time Logic statement either equals 0 or 1 or it solely depends on the colour of the initial state. This raises the conjecture that some zero-one law exists for Linear Time Logic. Since the results on zero-one laws for first order logic statements cannot be applied to Linear Time Logic this topic deserves further research.

References

[1] Marcus Müllller-Olm, David Schmidt and Bernhard Steffen: *Model-Checking, A Tutorial Introduction.*

[2] P. Erdös and A. Rényi: *On the Evolution of Random Graphs.*

[3] Saharon Shelah and Joel Spencer: *Zero-One Laws for Sparse Random Graphs.*

[4] Joel Spencer: *The Strange Logic of Random Graphs.*

[5] Alasdair J. Graham and David A. Pike: *A Note on Thresholds and Connectivity.*

[6] J. A. Bondy and U. S. R. Murty: *Graph Theory With Applications*

[7] Dirk van Dalen: *Logic and Structure.*

[8] R. Fagin: *Probabilities on finite models*, *J. Symbolic Logic* chapter 41 (page 50-58).

[9] Y. V. Glebskii, D. I. Kogan, M. I. Liogonkii and V. A. Talanov: *Range and degree of realizability of formulas in the restricted predicate calculus*, *Cyberneti* chapter 5 (page 142-154).