PEPA models of Internet worm attacks

Jane Hillston.
LFCS, University of Edinburgh

8th September 2005

Joint work with Jeremy Bradley and Stephen Gilmore

Outline

Outline

Epidemiology

- Internet-based computer infections (worms, viruses, etc) are a major concern, particularly to industry.

Epidemiology

- Internet-based computer infections (worms, viruses, etc) are a major concern, particularly to industry.
- They results in substantive loss of revenue each year as well as shaking user confidence.

Epidemiology

- Internet-based computer infections (worms, viruses, etc) are a major concern, particularly to industry.
- They results in substantive loss of revenue each year as well as shaking user confidence.
- The analogy with the spread of real-organism diseases is easy to see.

Epidemiology

- Internet-based computer infections (worms, viruses, etc) are a major concern, particularly to industry.
- They results in substantive loss of revenue each year as well as shaking user confidence.
- The analogy with the spread of real-organism diseases is easy to see.
- Inspired by the work of others, we have chosen to model such spread with a process algebra

Epidemiology

- Internet-based computer infections (worms, viruses, etc) are a major concern, particularly to industry.
- They results in substantive loss of revenue each year as well as shaking user confidence.
- The analogy with the spread of real-organism diseases is easy to see.
- Inspired by the work of others, we have chosen to model such spread with a process algebra
- ...incorporating timing aspects with actions with duration and scalability by mapping to ODEs.

Epidemiology

- Internet-based computer infections (worms, viruses, etc) are a major concern, particularly to industry.
- They results in substantive loss of revenue each year as well as shaking user confidence.
- The analogy with the spread of real-organism diseases is easy to see.
- Inspired by the work of others, we have chosen to model such spread with a process algebra
- ...incorporating timing aspects with actions with duration and scalability by mapping to ODEs.

PEPA

$$
\begin{aligned}
S &::= (\alpha, r).S \mid S + S \mid A \\
P &::= S \mid P \underset{L}{\bowtie} P \mid P/L
\end{aligned}
$$

PEPA

$$
\begin{array}{rcl}
S & ::= & (\alpha, r).S \mid S + S \mid A \\
P & ::= & S \mid P \bowtie_{L} P \mid P/L
\end{array}
$$

PREFIX: $(\alpha, r).S$ designated first action

PEPA

$$
\begin{array}{rcl}
S & ::= & (\alpha, r).S \mid S + S \mid A \\
P & ::= & S \mid P \underset{L}{\bowtie} P \mid P/L
\end{array}
$$

PREFIX:	$(\alpha, r).S$	designated first action
CHOICE:	$S + S$	competing components (race policy)

PEPA

$$
\begin{array}{rcl}
S & ::= & (\alpha, r).S \mid S + S \mid A \\
P & ::= & S \mid P \underset{L}{\bowtie} P \mid P/L
\end{array}
$$

PREFIX:	$(\alpha, r).S$	designated first action
CHOICE:	$S + S$	competing components (race policy)
CONSTANT:	$A \stackrel{\mathit{def}}{=} S$	assigning names

PEPA

$$
\begin{array}{rcl}
S & ::= & (\alpha, r).S \mid S + S \mid A \\
P & ::= & S \mid P \underset{L}{\bowtie} P \mid P/L
\end{array}
$$

PREFIX:	$(\alpha, r).S$	designated first action
CHOICE:	$S + S$	competing components (race policy)
CONSTANT:	$A \stackrel{\mathit{def}}{=} S$	assigning names
COOPERATION:	$P \underset{L}{\bowtie} P$	$\alpha \notin L$ concurrent activity (*individual actions*) $\alpha \in L$ cooperative activity (*shared actions*)

PEPA

$$
\begin{array}{rcl}
S & ::= & (\alpha, r).S \mid S + S \mid A \\
P & ::= & S \mid P \underset{L}{\bowtie} P \mid P/L
\end{array}
$$

PREFIX:	$(\alpha, r).S$	designated first action
CHOICE:	$S + S$	competing components (race policy)
CONSTANT:	$A \stackrel{\mathit{def}}{=} S$	assigning names
COOPERATION:	$P \underset{L}{\bowtie} P$	$\alpha \notin L$ concurrent activity (*individual actions*) $\alpha \in L$ cooperative activity (*shared actions*)
HIDING:	P/L	abstraction $\alpha \in L \Rightarrow \alpha \rightarrow \tau$

Generating a CTMC

The corresponding Continuous Time Markov Chain (CTMC) is derived automatically from the structured operational semantics which define the language:

Generating a CTMC

The corresponding Continuous Time Markov Chain (CTMC) is derived automatically from the structured operational semantics which define the language:

PEPA
MODEL

Generating a CTMC

The corresponding Continuous Time Markov Chain (CTMC) is derived automatically from the structured operational semantics which define the language:

PEPA MODEL $\xrightarrow{\text{SOS rules}}$

Generating a CTMC

The corresponding Continuous Time Markov Chain (CTMC) is derived automatically from the structured operational semantics which define the language:

PEPA MODEL $\xrightarrow{\text{SOS rules}}$ LABELLED TRANSITION SYSTEM

Generating a CTMC

The corresponding Continuous Time Markov Chain (CTMC) is derived automatically from the structured operational semantics which define the language:

$$\text{PEPA MODEL} \xrightarrow{\text{SOS rules}} \text{LABELLED TRANSITION SYSTEM} \xrightarrow[\text{diagram}]{\text{state transition}}$$

Generating a CTMC

The corresponding Continuous Time Markov Chain (CTMC) is derived automatically from the structured operational semantics which define the language:

PEPA MODEL $\xrightarrow{\text{SOS rules}}$ LABELLED TRANSITION SYSTEM $\xrightarrow[\text{diagram}]{\text{state transition}}$ CTMC $\mathbf{Q}$

Generating a CTMC

The corresponding Continuous Time Markov Chain (CTMC) is derived automatically from the structured operational semantics which define the language:

$$\text{PEPA MODEL} \xrightarrow{\text{SOS rules}} \text{LABELLED TRANSITION SYSTEM} \xrightarrow[\text{diagram}]{\text{state transition}} \text{CTMC } \mathbf{Q}$$

The states of the CTMC are the distinct syntactic terms which the model may evolve to.

Generating a CTMC

The corresponding Continuous Time Markov Chain (CTMC) is derived automatically from the structured operational semantics which define the language:

$$\text{PEPA MODEL} \xrightarrow{\text{SOS rules}} \text{LABELLED TRANSITION SYSTEM} \xrightarrow[\text{diagram}]{\text{state transition}} \text{CTMC } \mathbf{Q}$$

The states of the CTMC are the distinct syntactic terms which the model may evolve to.

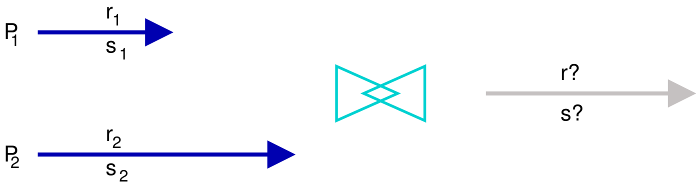
Solving the model has meant finding the steady state probability distribution over the entire state space.

Timed Synchronisation

- The issue of what it means for two timed activities to synchronise is a vexed one....

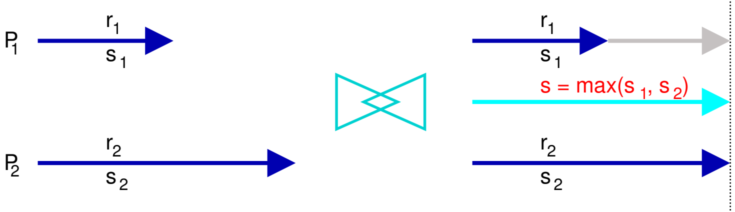
Timed Synchronisation

- The issue of what it means for two timed activities to synchronise is a vexed one....



Timed Synchronisation

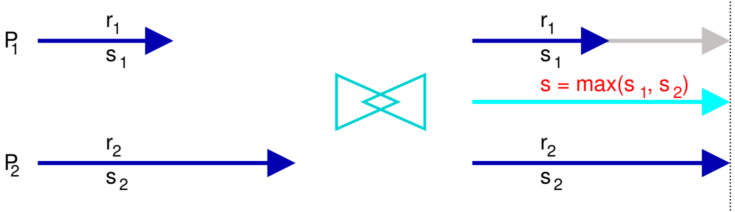
- The issue of what it means for two timed activities to synchronise is a vexed one....



Barrier Synchronisation

Timed Synchronisation

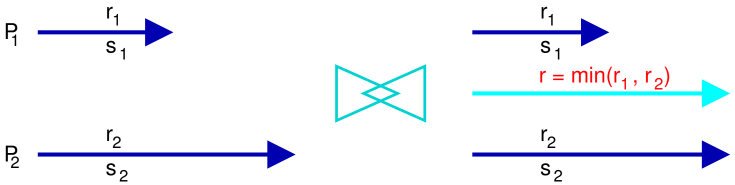
- The issue of what it means for two timed activities to synchronise is a vexed one....



s is no longer exponentially distributed

Timed Synchronisation

- The issue of what it means for two timed activities to synchronise is a vexed one....



bounded capacity: new rate is the minimum of the rates

Cooperation in PEPA

- In PEPA each component has a bounded capacity to carry out activities of any particular type, determined by the apparent rate for that type.

Cooperation in PEPA

- In PEPA each component has a bounded capacity to carry out activities of any particular type, determined by the apparent rate for that type.
- Synchronisation, or cooperation cannot make a component exceed its bounded capacity.

Cooperation in PEPA

- In PEPA each component has a bounded capacity to carry out activities of any particular type, determined by the apparent rate for that type.
- Synchronisation, or cooperation cannot make a component exceed its bounded capacity.
- Thus the apparent rate of a cooperation is the minimum of the apparent rates of the co-operands.

Outline

Internet worm models

We consider three distinct models, taking alternative views of what happens after a computer has been infected.

Internet worm models

We consider three distinct models, taking alternative views of what happens after a computer has been infected.

- In the first model we assume that a patch is applied with the result that the infected machine is no longer infected or susceptible

Internet worm models

We consider three distinct models, taking alternative views of what happens after a computer has been infected.

- In the first model we assume that a patch is applied with the result that the infected machine is no longer infected or susceptible — it is removed from the infection.

Internet worm models

We consider three distinct models, taking alternative views of what happens after a computer has been infected.

- In the first model we assume that a patch is applied with the result that the infected machine is no longer infected or susceptible — it is removed from the infection.
- In the second model we consider the situation when this patch is not permanent, thus allowing the possibility of reinfection.

Internet worm models

We consider three distinct models, taking alternative views of what happens after a computer has been infected.

- In the first model we assume that a patch is applied with the result that the infected machine is no longer infected or susceptible — it is removed from the infection.
- In the second model we consider the situation when this patch is not permanent, thus allowing the possibility of reinfection.
- The model considers a worm which instigates a distributed denial of service attack

Internet worm models

We consider three distinct models, taking alternative views of what happens after a computer has been infected.

- In the first model we assume that a patch is applied with the result that the infected machine is no longer infected or susceptible — it is removed from the infection.
- In the second model we consider the situation when this patch is not permanent, thus allowing the possibility of reinfection.
- The model considers a worm which instigates a distributed denial of service attack — an infected computer, which has not been patched, may either infect another computer or launch an attack on a pre-defined victim computer.

Internet worm models

We consider three distinct models, taking alternative views of what happens after a computer has been infected.

- In the first model we assume that a patch is applied with the result that the infected machine is no longer infected or susceptible — it is removed from the infection.
- In the second model we consider the situation when this patch is not permanent, thus allowing the possibility of reinfection.
- The model considers a worm which instigates a distributed denial of service attack — an infected computer, which has not been patched, may either infect another computer or launch an attack on a pre-defined victim computer.

In all the models we assume that the infection must pass over a network, which can sustain M independent concurrent connections.

Model 1

The Susceptible-Infective-Removed model.

$$
\begin{aligned}
S &= (infectS, \top).I \\
I &= (infectI, \beta).I + (patch, \gamma).R \\
R &= stop \\
Net &= (infectI, \top).Net' \\
Net' &= (infectS, \beta).Net \\
Sys &= (S[N] \parallel I) \underset{L}{\bowtie} Net[M]
\end{aligned}
$$

where $L = \{infectI, infectS\}$.

Model 1

The Susceptible-Infective-Removed model.

$$
\begin{aligned}
S &= (infectS, \top).I \\
I &= (infectI, \beta).I + (patch, \gamma).R \\
R &= stop \\
Net &= (infectI, \top).Net' \\
Net' &= (infectS, \beta).Net \\
Sys &= (S[N] \parallel I) \underset{L}{\bowtie} Net[M]
\end{aligned}
$$

where $L = \{infectI, infectS\}$.

Model 1

The Susceptible-Infective-Removed model.

$$
\begin{aligned}
S &= (infectS, \top).I \\
I &= (infectI, \beta).I + (patch, \gamma).R \\
R &= stop \\
Net &= (infectI, \top).Net' \\
Net' &= (infectS, \beta).Net \\
Sys &= (S[N] \mid\mid I) \underset{L}{\bowtie} Net[M]
\end{aligned}
$$

where $L = \{infectI, infectS\}$.

Model 2

The Susceptible-Infective-Removed-Reinfection model.

$$
\begin{aligned}
S &= (infectS, \top).I \\
I &= (infectI, \beta).I + (patch, \gamma).R \\
R &= (unsecure, \mu).S \\
Net &= (infectI, \top).Net' \\
Net' &= (infectS, \beta).Net \\
Sys &= (S[100] \mid\mid I) \underset{L}{\bowtie} Net[M]
\end{aligned}
$$

where $L = \{infectI, infectS\}$.

Model 2

The Susceptible-Infective-Removed-Reinfection model.

$$
\begin{aligned}
S &= (\mathit{infectS}, \top).I \\
I &= (\mathit{infectI}, \beta).I + (\mathit{patch}, \gamma).R \\
R &= (\mathit{unsecure}, \mu).S \\
\mathit{Net} &= (\mathit{infectI}, \top).\mathit{Net}' \\
\mathit{Net}' &= (\mathit{infectS}, \beta).\mathit{Net} \\
\mathit{Sys} &= (S[100] \parallel I) \underset{L}{\bowtie} \mathit{Net}[M]
\end{aligned}
$$

where $L = \{\mathit{infectI}, \mathit{infectS}\}$.

Model 3

The Susceptible-Infective-Removed-Attack model.

$$
\begin{aligned}
S &= (infectS, \top).I \\
I &= (infectI, \beta).I + (attack, \lambda).I + (patch, \gamma).R \\
R &= stop \\
Net &= (infectI, \top).Net' \\
Net' &= (infectS, \beta).Net \\
A &= (attack, \top).A' \\
A' &= (recover, \mu).A \\
Sys &= ((S[N] \parallel I) \underset{L}{\bowtie} Net[M]) \underset{L'}{\bowtie} A[T]
\end{aligned}
$$

where $L = \{infectI, infectS\}$, $L' = \{attack\}$.

Model 3

The Susceptible-Infective-Removed-Attack model.

$$
\begin{array}{rcl}
S & = & (\mathit{infectS}, \top).I \\
I & = & (\mathit{infectI}, \beta).I + (\mathit{attack}, \lambda).I + (\mathit{patch}, \gamma).R \\
R & = & \mathit{stop} \\
\mathit{Net} & = & (\mathit{infectI}, \top).\mathit{Net}' \\
\mathit{Net}' & = & (\mathit{infectS}, \beta).\mathit{Net} \\
A & = & (\mathit{attack}, \top).A' \\
A' & = & (\mathit{recover}, \mu).A \\
\mathit{Sys} & = & ((S[N] \parallel I) \underset{L}{\bowtie} \mathit{Net}[M]) \underset{L'}{\bowtie} A[T]
\end{array}
$$

where $L = \{\mathit{infectI}, \mathit{infectS}\}$, $L' = \{\mathit{attack}\}$.

Model 3

The Susceptible-Infective-Removed-Attack model.

$$
\begin{aligned}
S &= (infectS, \top).I \\
I &= (infectI, \beta).I + (attack, \lambda).I + (patch, \gamma).R \\
R &= stop \\
Net &= (infectI, \top).Net' \\
Net' &= (infectS, \beta).Net \\
A &= (attack, \top).A' \\
A' &= (recover, \mu).A \\
Sys &= ((S[N] \parallel I) \underset{L}{\bowtie} Net[M]) \underset{L'}{\bowtie} A[T]
\end{aligned}
$$

where $L = \{infectI, infectS\}$, $L' = \{attack\}$.

Model 3

The Susceptible-Infective-Removed-Attack model.

$$
\begin{aligned}
S &= (infectS, \top).I \\
I &= (infectI, \beta).I + (attack, \lambda).I + (patch, \gamma).R \\
R &= stop \\
Net &= (infectI, \top).Net' \\
Net' &= (infectS, \beta).Net \\
A &= (attack, \top).A' \\
A' &= (recover, \mu).A \\
Sys &= ((S[N] \parallel I) \underset{L}{\bowtie} Net[M]) \underset{L'}{\bowtie} A[T]
\end{aligned}
$$

where $L = \{infectI, infectS\}$, $L' = \{attack\}$.

Model 3

The Susceptible-Infective-Removed-Attack model.

$$
\begin{aligned}
S &= (infectS, \top).I \\
I &= (infectI, \beta).I + (attack, \lambda).I + (patch, \gamma).R \\
R &= stop \\
Net &= (infectI, \top).Net' \\
Net' &= (infectS, \beta).Net \\
A &= (attack, \top).A' \\
A' &= (recover, \mu).A \\
Sys &= ((S[N] \parallel I) \underset{L}{\bowtie} Net[M]) \underset{L'}{\bowtie} A[T]
\end{aligned}
$$

where $L = \{infectI, infectS\}$, $L' = \{attack\}$.

Outline

Introduction

Internet worm models

Continuous Approximation

Quantified analysis

Conclusions

Performance evaluation: new mathematical structures

For a generation, performance modellers have seen their choices as being:

- Closed form analytical models;

Performance evaluation: new mathematical structures

For a generation, performance modellers have seen their choices as being:

- Closed form analytical models;
- Simulations; or

Performance evaluation: new mathematical structures

For a generation, performance modellers have seen their choices as being:

- Closed form analytical models;
- Simulations; or
- Numerical solution of continuous time Markov chains (CTMC)

Performance evaluation: new mathematical structures

For a generation, performance modellers have seen their choices as being:

- Closed form analytical models;
- Simulations; or
- Numerical solution of continuous time Markov chains (CTMC)

The major limitations of the CTMC approach are the state space explosion problem and the reliance on exponential distributions.

New mathematical structures: differential equations

- Use a more abstract state representation rather than the CTMC complete state space.

New mathematical structures: differential equations

- Use a more abstract state representation rather than the CTMC complete state space.
- No longer aim to calculate the probability distribution over the entire state space of the model.

New mathematical structures: differential equations

- Use a more abstract state representation rather than the CTMC complete state space.
- No longer aim to calculate the probability distribution over the entire state space of the model.
- Assume that these state variables are subject to continuous rather than discrete change.

New mathematical structures: differential equations

- Use a more abstract state representation rather than the CTMC complete state space.
- No longer aim to calculate the probability distribution over the entire state space of the model.
- Assume that these state variables are subject to continuous rather than discrete change.

Only appropriate for some models, but results are promising in those cases.

New mathematical structures: differential equations

- Use a more abstract state representation rather than the CTMC complete state space.
- No longer aim to calculate the probability distribution over the entire state space of the model.
- Assume that these state variables are subject to continuous rather than discrete change.

Only appropriate for some models, but results are promising in those cases. large numbers of repeated components

Differential equations from PEPA models

- In a PEPA model the state at any current time is the local derivative or state of each component of the model.
- When we have large numbers of repeated components it can make sense to represent the state of the system as the count of the current number of each possible local derivative or component type.
- We can approximate the behaviour of the model by treating the number of each component type as a continuous variable, and the state of the model as a whole as the set of such variables.
- The evolution of each such variable can then be described by an ordinary differential equation (assuming rates are deterministic).

Differential equations from PEPA models

- In a PEPA model the state at any current time is the local derivative or state of each component of the model.
- When we have large numbers of repeated components it can make sense to represent the state of the system as the count of the current number of each possible local derivative or component type.
- We can approximate the behaviour of the model by treating the number of each component type as a continuous variable, and the state of the model as a whole as the set of such variables.
- The evolution of each such variable can then be described by an ordinary differential equation (assuming rates are deterministic).

Differential equations from PEPA models

- In a PEPA model the state at any current time is the local derivative or state of each component of the model.
- When we have large numbers of repeated components it can make sense to represent the state of the system as the count of the current number of each possible local derivative or component type.
- We can approximate the behaviour of the model by treating the number of each component type as a continuous variable, and the state of the model as a whole as the set of such variables.
- The evolution of each such variable can then be described by an ordinary differential equation (assuming rates are deterministic).

Differential equations from PEPA models

- In a PEPA model the state at any current time is the local derivative or state of each component of the model.
- When we have large numbers of repeated components it can make sense to represent the state of the system as the count of the current number of each possible local derivative or component type.
- We can approximate the behaviour of the model by treating the number of each component type as a continuous variable, and the state of the model as a whole as the set of such variables.
- The evolution of each such variable can then be described by an ordinary differential equation (assuming rates are deterministic).

Differential equations from PEPA models

- The PEPA definitions of the component specify the activities which can increase or decrease the number of components exhibited in the current state.
- The cooperations show when the number of instances of another component will have an influence on the evolution of this component.

Differential equations from PEPA models

- The PEPA definitions of the component specify the activities which can increase or decrease the number of components exhibited in the current state.
- The cooperations show when the number of instances of another component will have an influence on the evolution of this component.

Differential equations from PEPA models

- The PEPA definitions of the component specify the activities which can increase or decrease the number of components exhibited in the current state.
- The cooperations show when the number of instances of another component will have an influence on the evolution of this component.

Derivation of the system of ODES representing the PEPA model then proceeds via an activity matrix which keeps track of the impact of each activity type on each component type.

Outline

Introduction

Internet worm models

Continuous Approximation

Quantified analysis

Conclusions

Model 1: Susceptible-Infective-Removed model

$$
\begin{aligned}
S &= (infectS, \top).I \\
I &= (infectI, \beta).I + (patch, \gamma).R \\
R &= stop \\
Net &= (infectI, \top).Net' \\
Net' &= (infectS, \beta).Net \\
Sys &= (S[N] \mid\mid I) \underset{L}{\bowtie} Net[M]
\end{aligned}
$$

where $L = \{infectI, infectS\}$.

Which form of synchronisation?

In this model (and the others) the cooperations are all of the form *active-passive*, i.e. one component governs the rate of the activity and the other just passively witnesses the activity.

Which form of synchronisation?

In this model (and the others) the cooperations are all of the form *active-passive*, i.e. one component governs the rate of the activity and the other just passively witnesses the activity.

These cooperations each involve the network and we assume that a computer (susceptible or invective) can attach to any of the available network connections.

Which form of synchronisation?

In this model (and the others) the cooperations are all of the form *active-passive*, i.e. one component governs the rate of the activity and the other just passively witnesses the activity.

These cooperations each involve the network and we assume that a computer (susceptible or invective) can attach to any of the available network connections.

In terms of Jeremy's classification yesterday, this means we use the passive synchronisation scheme in the ODEs.

Mapping to an ODE

$$
\begin{aligned}
\frac{\mathrm{d}v_{11}(t)}{\mathrm{d}t} &= -\beta I_{11}(t)v_{22}(t) \\
\frac{\mathrm{d}v_{12}(t)}{\mathrm{d}t} &= -\gamma v_{12}(t) + \beta I_{11}(t)v_{22}(t) \\
\frac{\mathrm{d}v_{13}(t)}{\mathrm{d}t} &= \gamma v_{12}(t) \\
\frac{\mathrm{d}v_{21}(t)}{\mathrm{d}t} &= -\beta I_{21}(t)v_{12}(t) + \beta I_{11}(t)v_{22}(t) \\
\frac{\mathrm{d}v_{22}(t)}{\mathrm{d}t} &= -\beta I_{11}(t)v_{22}(t) + \beta I_{21}(t)v_{12}(t)
\end{aligned}
$$

where $v_{11} \leftrightarrow S,\ v_{12} \leftrightarrow I,\ v_{13} \leftrightarrow R,\ v_{21} \leftrightarrow Net,\ v_{22} \leftrightarrow net'$.

Mapping to an ODE

$$
\begin{aligned}
\frac{\mathrm{d}v_{11}(t)}{\mathrm{d}t} &= -\beta I_{11}(t) v_{22}(t) \\
\frac{\mathrm{d}v_{12}(t)}{\mathrm{d}t} &= -\gamma v_{12}(t) + \beta I_{11}(t) v_{22}(t) \\
\frac{\mathrm{d}v_{13}(t)}{\mathrm{d}t} &= \gamma v_{12}(t) \\
\frac{\mathrm{d}v_{21}(t)}{\mathrm{d}t} &= -\beta I_{21}(t) v_{12}(t) + \beta I_{11}(t) v_{22}(t) \\
\frac{\mathrm{d}v_{22}(t)}{\mathrm{d}t} &= -\beta I_{11}(t) v_{22}(t) + \beta I_{21}(t) v_{12}(t)
\end{aligned}
$$

where $v_{11} \leftrightarrow S,\ v_{12} \leftrightarrow I,\ v_{13} \leftrightarrow R,\ v_{21} \leftrightarrow Net,\ v_{22} \leftrightarrow net'$.

Model 1: experiments

We assume a susceptible population of $N = 1000$ computers and a network capable of sustaining up to $M = 200$ simultaneous concurrent connections.

Model 1: experiments

We assume a susceptible population of $N = 1000$ computers and a network capable of sustaining up to $M = 200$ simultaneous concurrent connections.

We assume that the system starts with one infected computer.

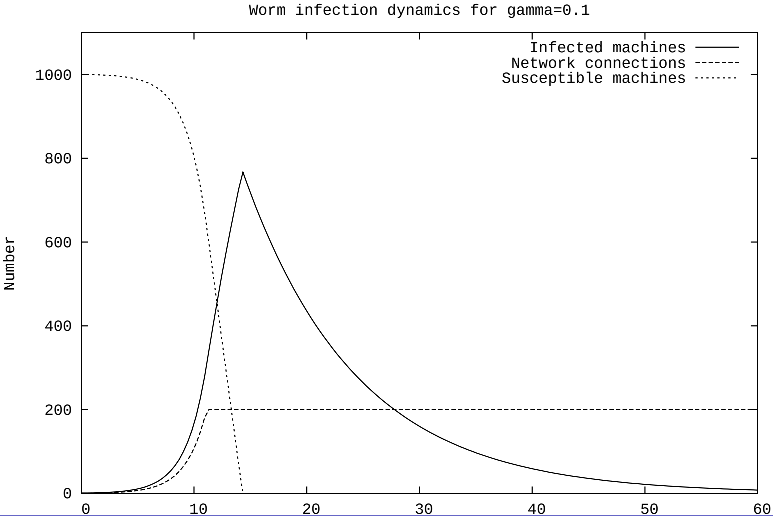
Model 1: experiments

We assume a susceptible population of $N = 1000$ computers and a network capable of sustaining up to $M = 200$ simultaneous concurrent connections.

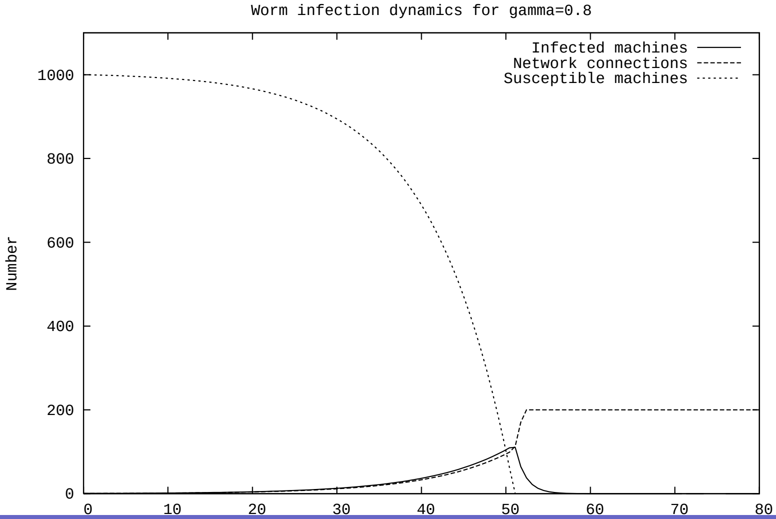
We assume that the system starts with one infected computer.

In the first experiment we varied the rate at which the patch is applied, γ, representing different (human) response rates to the infection.

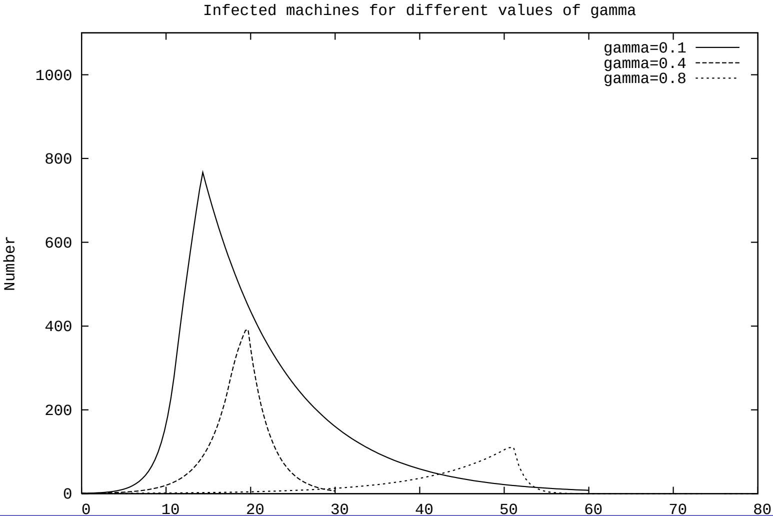
Model 1: $\gamma = 0.1$



Model 1: $\gamma = 0.8$



Model 1: Number of infected machines as γ increases



Model 2: Susceptible-Infective-Removed-Reinfection model

$$
\begin{aligned}
S &= (\mathit{infectS}, \top).I \\
I &= (\mathit{infectI}, \beta).I + (\mathit{patch}, \gamma).R \\
R &= (\mathit{unsecure}, \mu).S \\
\mathit{Net} &= (\mathit{infectI}, \top).\mathit{Net}' \\
\mathit{Net}' &= (\mathit{infectS}, \beta).\mathit{Net} \\
\mathit{Sys} &= (S[N] \parallel I) \underset{L}{\bowtie} \mathit{Net}[M]
\end{aligned}
$$

where $L = \{\mathit{infectI}, \mathit{infectS}\}$.

Mapping to an ODE

$$
\begin{aligned}
\frac{\mathrm{d}v_{11}(t)}{\mathrm{d}t} &= -\beta I_{11}(t)v_{22}(t) + \mu v_{13}(t) \\
\frac{\mathrm{d}v_{12}(t)}{\mathrm{d}t} &= -\gamma v_{12}(t) + \beta I_{11}(t)v_{22}(t) \\
\frac{\mathrm{d}v_{13}(t)}{\mathrm{d}t} &= -\mu v_{13}(t) + \gamma v_{12}(t) \\
\frac{\mathrm{d}v_{21}(t)}{\mathrm{d}t} &= -\beta I_{21}(t)v_{12}(t) + \beta I_{11}(t)v_{22}(t) \\
\frac{\mathrm{d}v_{22}(t)}{\mathrm{d}t} &= -\beta I_{11}(t)v_{22}(t) + \beta I_{21}(t)v_{12}(t)
\end{aligned}
$$

where $v_{11} \leftrightarrow S,\ v_{12} \leftrightarrow I,\ v_{13} \leftrightarrow R,\ v_{21} \leftrightarrow Net,\ v_{22} \leftrightarrow net'$.

Model 2: experiments

We assume a susceptible population of $N = 1000$ computers.

Model 2: experiments

We assume a susceptible population of $N = 1000$ computers.

We assume that the system starts with one infected computer.

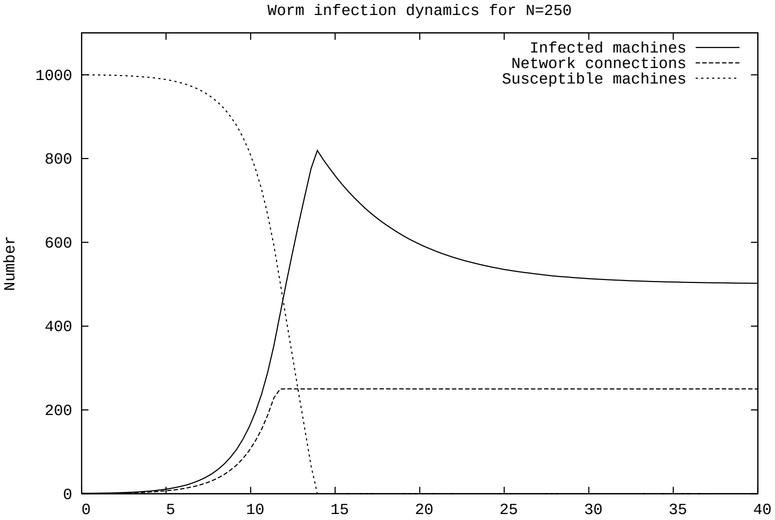
Model 2: experiments

We assume a susceptible population of $N = 1000$ computers.

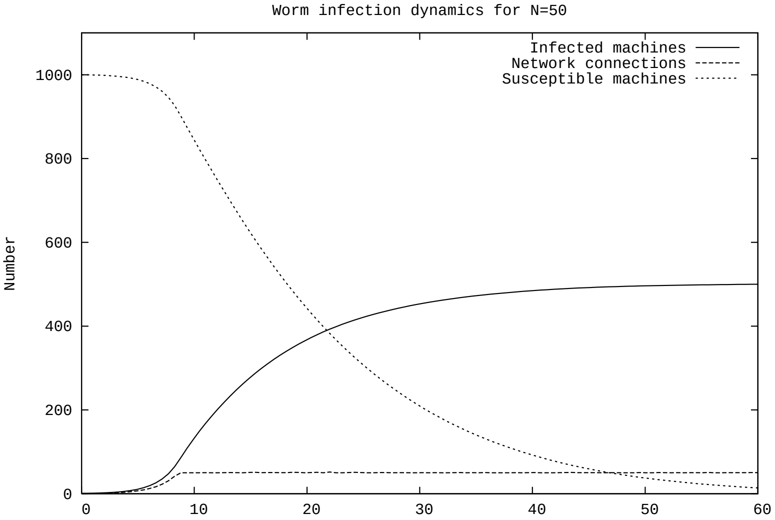
We assume that the system starts with one infected computer.

In this experiment we varied the network capacity, i.e. M. This restricts the medium over which the infection is transmitted.

Model 2: $N = 250$


Worm infection dynamics for N=250
Infected machines
Network connections
Susceptible machines
Number
1000
800
600
400
200
0
0
5
10
15
20
25
30
35
40

Model 2: $N = 50$


Worm infection dynamics for N=50
Infected machines
Network connections
Susceptible machines
Number
1000
800
600
400
200
0
0
10
20
30
40
50
60

Model 3: Susceptible-Infective-Removed-Attack model

$$
\begin{aligned}
S &= (\mathit{infectS}, \top).I \\
I &= (\mathit{infectI}, \beta).I + (\mathit{attack}, \lambda).I + (\mathit{patch}, \gamma).R \\
R &= \mathit{stop} \\
\mathit{Net} &= (\mathit{infectI}, \top).\mathit{Net}' \\
\mathit{Net}' &= (\mathit{infectS}, \beta).\mathit{Net} \\
A &= (\mathit{attack}, \top).A' \\
A' &= (\mathit{recover}, \mu).A \\
\mathit{Sys} &= ((S[N] \parallel I) \underset{L}{\bowtie} \mathit{Net}[M]) \underset{L'}{\bowtie} A[T]
\end{aligned}
$$

where $L = \{\mathit{infectI}, \mathit{infectS}\}$, $L' = \{\mathit{attack}\}$.

Mapping to an ODE

$$
\begin{aligned}
\frac{\mathrm{d}v_{11}(t)}{\mathrm{d}t} &= -\beta I_{11}(t)v_{22}(t) \\
\frac{\mathrm{d}v_{12}(t)}{\mathrm{d}t} &= -\gamma v_{12}(t) + \beta I_{11}(t)v_{22}(t) \\
\frac{\mathrm{d}v_{13}(t)}{\mathrm{d}t} &= \gamma v_{12}(t) \\
\frac{\mathrm{d}v_{21}(t)}{\mathrm{d}t} &= -\beta I_{21}(t)v_{12}(t) + \beta I_{11}(t)v_{22}(t) \\
\frac{\mathrm{d}v_{22}(t)}{\mathrm{d}t} &= -\beta I_{11}(t)v_{22}(t) + \beta I_{21}(t)v_{12}(t) \\
\frac{\mathrm{d}v_{31}(t)}{\mathrm{d}t} &= -\lambda I_{31}(t)v_{12}(t) + v_{32}(t)\mu \\
\frac{\mathrm{d}v_{32}(t)}{\mathrm{d}t} &= -v_{32}(t)\mu + \lambda I_{31}(t)v_{12}(t)
\end{aligned}
$$

$v_{11} \leftrightarrow S$, $v_{12} \leftrightarrow I$, $v_{13} \leftrightarrow R$, $v_{21} \leftrightarrow Net$, $v_{22} \leftrightarrow net'$, $v_{31} \leftrightarrow A$, $v_{32} \leftrightarrow A'$.

Model 3: experiments

We assume a susceptible population of $N = 1000$ computers, a network capacity of $M = 200$

Model 3: experiments

We assume a susceptible population of $N = 1000$ computers, a network capacity of $M = 200$

We assume that the system starts with one infected computer, and that the target of the attack has 100 ports on which it can accept connections.

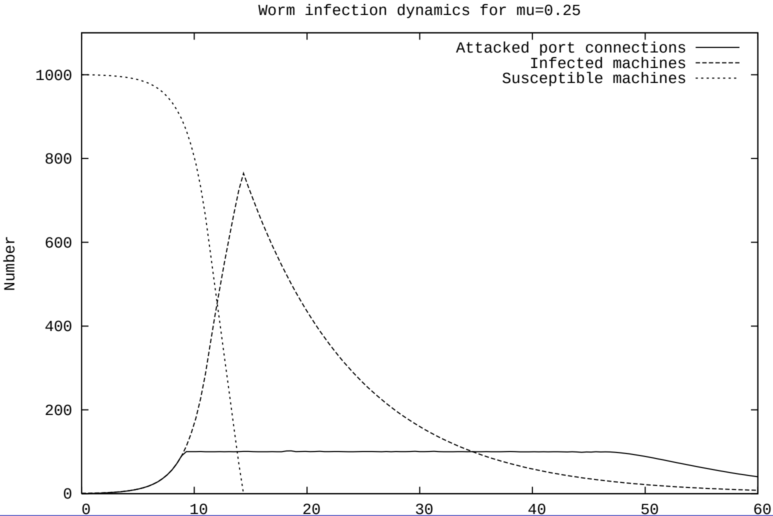
Model 3: experiments

We assume a susceptible population of $N = 1000$ computers, a network capacity of $M = 200$

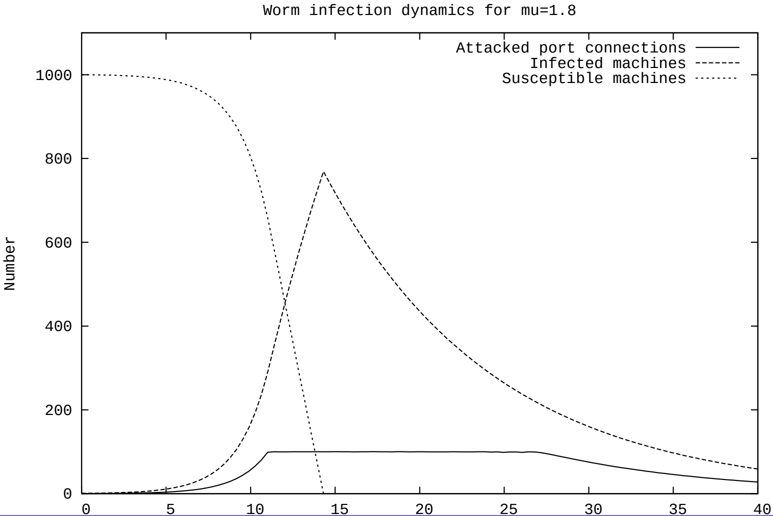
We assume that the system starts with one infected computer, and that the target of the attack has 100 ports on which it can accept connections.

In this experiment we varied the rate μ at which a port timeouts and becomes usable again in the attacked machine.

Model 3: $\mu = 0.25$



Model 3: $\mu = 1.8$


Worm infection dynamics for mu=1.8
Attacked port connections
Infected machines
Susceptible machines
Number
0
200
400
600
800
1000
0
5
10
15
20
25
30
35
40

Outline

Conclusions

ODEs are great!

Conclusions

ODEs are great!

- We could evaluate small systems using the CTMC semantics but not with realistic populations

Conclusions

ODEs are great!

- We could evaluate small systems using the CTMC semantics but not with realistic populations
- We could construct the ODEs directly (eg. [Nicol et al]) but using the process algebra gives a more accessible model, and one which is amenable to other analyses such as model checking.

Conclusions

ODEs are great!

- We could evaluate small systems using the CTMC semantics but not with realistic populations
- We could construct the ODEs directly (eg. [Nicol et al]) but using the process algebra gives a more accessible model, and one which is amenable to other analyses such as model checking.
- For these models there are still many experiments to be considered and variations to the models which could be made.