A RISK MANAGEMENT VIEW TO INFORMATION SECURITY

Nick Bambos

Stanford University

GameSec 2010 Plenary Talk

Berlin, Nov. 2010

Outline

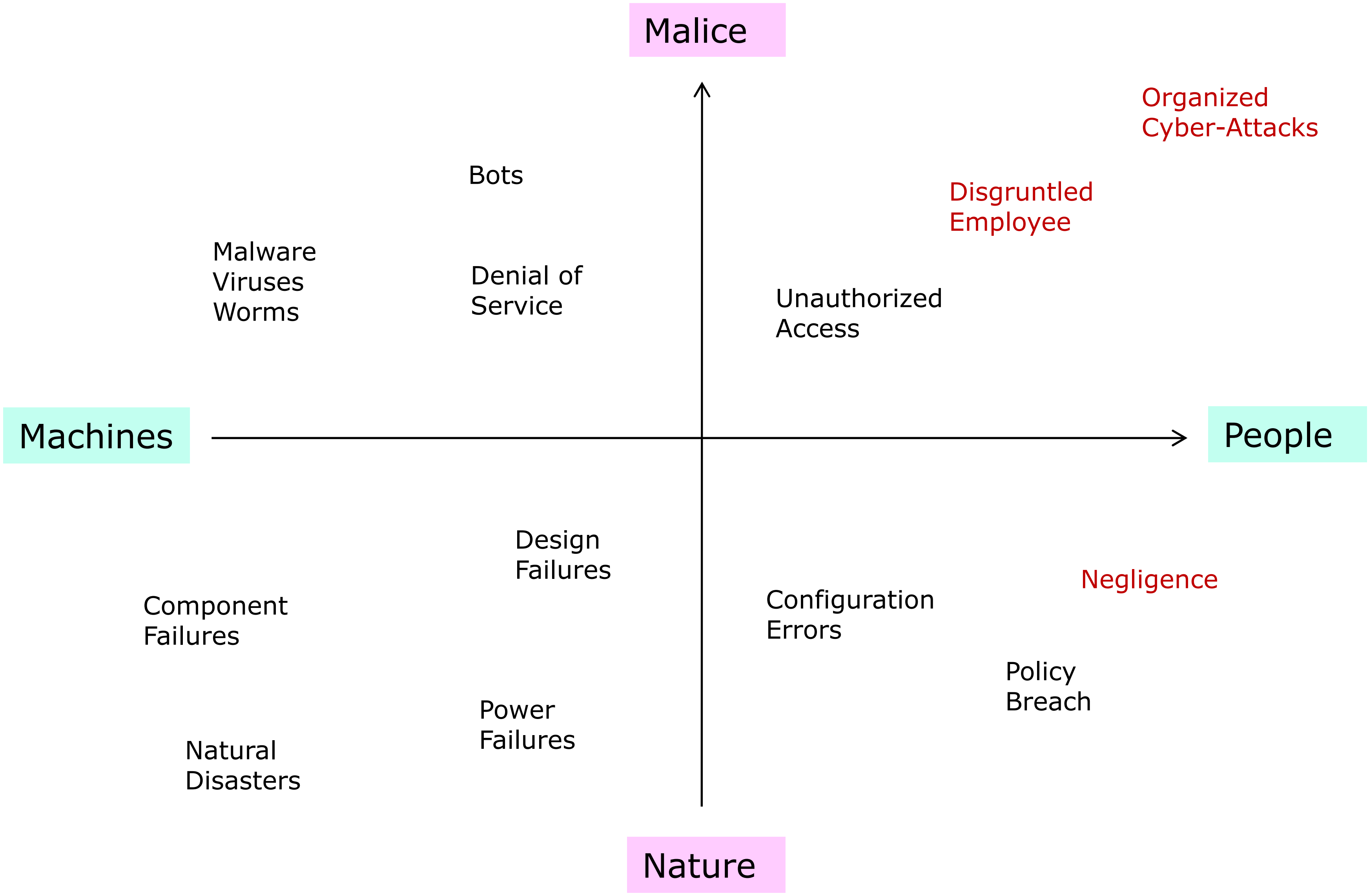
The Case for Corporate IT Risk Management

Some Paradigms and Models for (systematic) Risk Management

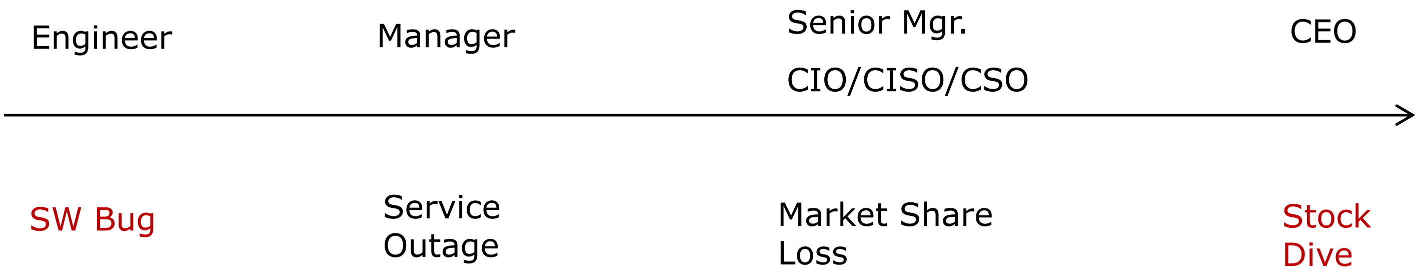
IT Risk Management

Some Observations…

Risk Sources… Scattered, but Equally Important



Risk Nature and Impact – 2 Examples



Engineer

Manager

Senior Mgr.

CIO/CISO/CSO

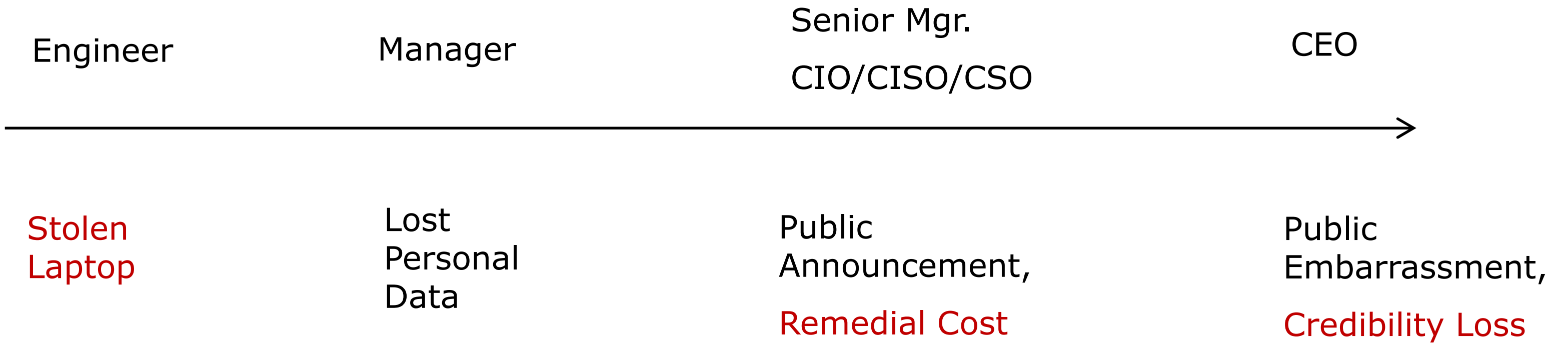
CEO

SW Bug

Service Outage

Market Share Loss

Stock Dive



Engineer

Manager

Senior Mgr.

CIO/CISO/CSO

CEO

Stolen Laptop

Lost Personal Data

Public Announcement,

Remedial Cost

Public Embarrassment,

Credibility Loss

Risks & Decisions

Decision-Maker	Time Scale	Risks	Possible Actions
CIO	Months	Loss of Sensitive Data, Service Outage	Company-Wide Policies, Major Security Investments
Dept. Managers	Days/Hours	Announced Threats, Equip. Theft	Dept. Policies, Change Org. Flow
Engineering	Seconds	Worms, Machine Failures, etc.	Block ports, Isolate Networks, etc.

Approach/View

Management… Strategic (CIO) /Tactical (Dept. Head)

Engineering… Operational

Management – Engineering **Disconnect**

Engineers think in terms of absolute (0-1) security, hardening and redundancy

Managers think in terms risk exposure and loss reduction

Example: Vulnerabilities and Patching

Vulnerabilities:

~ 100 vulnerabilities announced per week!

~ 2 weeks testing, before applying patch!

Why IT Risk Management Now? And How?

Senior executives demand it…

Increasing damages from IT security incidents (~$8B/US)

Increasing spending on IT security (~$80B/US)

Legal requirements creating pressure (Sarbanes-Oxley Act)

Unique problem requirements…

Little agreement on metrics…

Lack of 'tested and approved' concepts and models

Rapidly evolving landscape

Interdependencies create huge complexity

Systematic approach needed…

Assessing Risk (and Probabilities)

High impact events are rare (almost no statistics)

Behavioral (Subjective) Approach – Ask the Manager:

A. Cost / Benefit Game:

Given $100, how would you allocate it to risk factors?

Profile:

Risk factors and their (relative) importance.

B. Threshold Based Game:

Is it more than X, or less?

Tracking Risk

The risk you know… vs. the risk you don’t know…

Nobody likes the “bearer of bad news” … even when true…

How do you know the integrity state of your system?

Ubiquitous problem: **Quickest Detection vs. False Alarm**

Context of Corporate IT Risk Mgt.

Largely qualitative, empirical, **instinctive**

… yet **effective** in various cases (… but not most)

Organizational level… **policies** and procedures (don’t carry around critical data)

Service Level… **controlled access**, authorization, authentication

Application level… **countermeasures** (patching, honeycombs)

Infrastructure level… **redundancy**, overdesign (hot spares, backups)

State-of-the-Art … in Tactical Risk Management

Department Heads

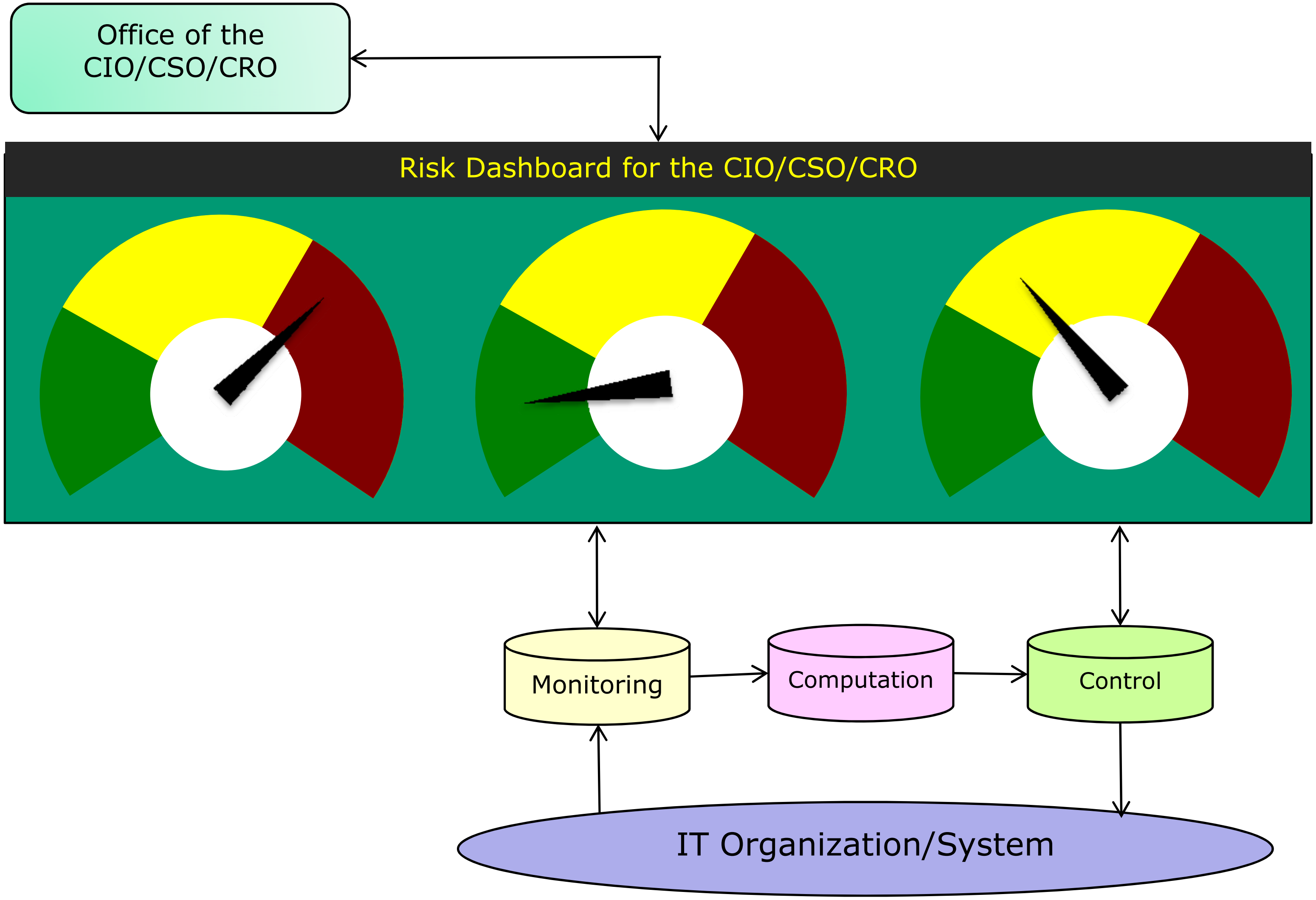
fill out spreadsheet s(templates with fields) periodically
record ‘risk values’ of individual risk elements
capture ‘snapshot’ of perceived risk exposure … in their domain

Central Risk Mgt. Office

exercises best-effort to
identify ‘hit patterns’ across forms
develop big picture of risk exposure
decisions made ~ 10mil

Key Issue… lack of systematic methodology/framework
low resolution global risk visibility
no computation-aided decision support

Risk Monitoring & Decision `Cockpit'



The Goal: The Human in an Agile Decision Loop

Computation-Based *Decision Support* System

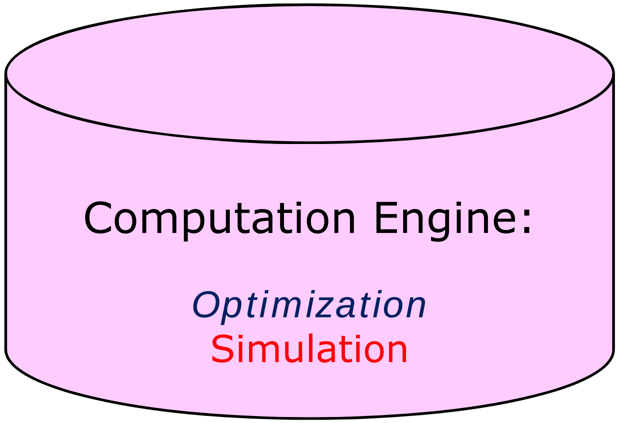
Human Decisions:

Strategic:	90% (long term policy, investments, etc.)
Tactical:	70% (medium term procedures, configurations)
Operational:	30% (short term re-configurations, patching)
Real-Time:	00% (dynamic control)

Computation Engine:

Optimization Module

Simulation Module

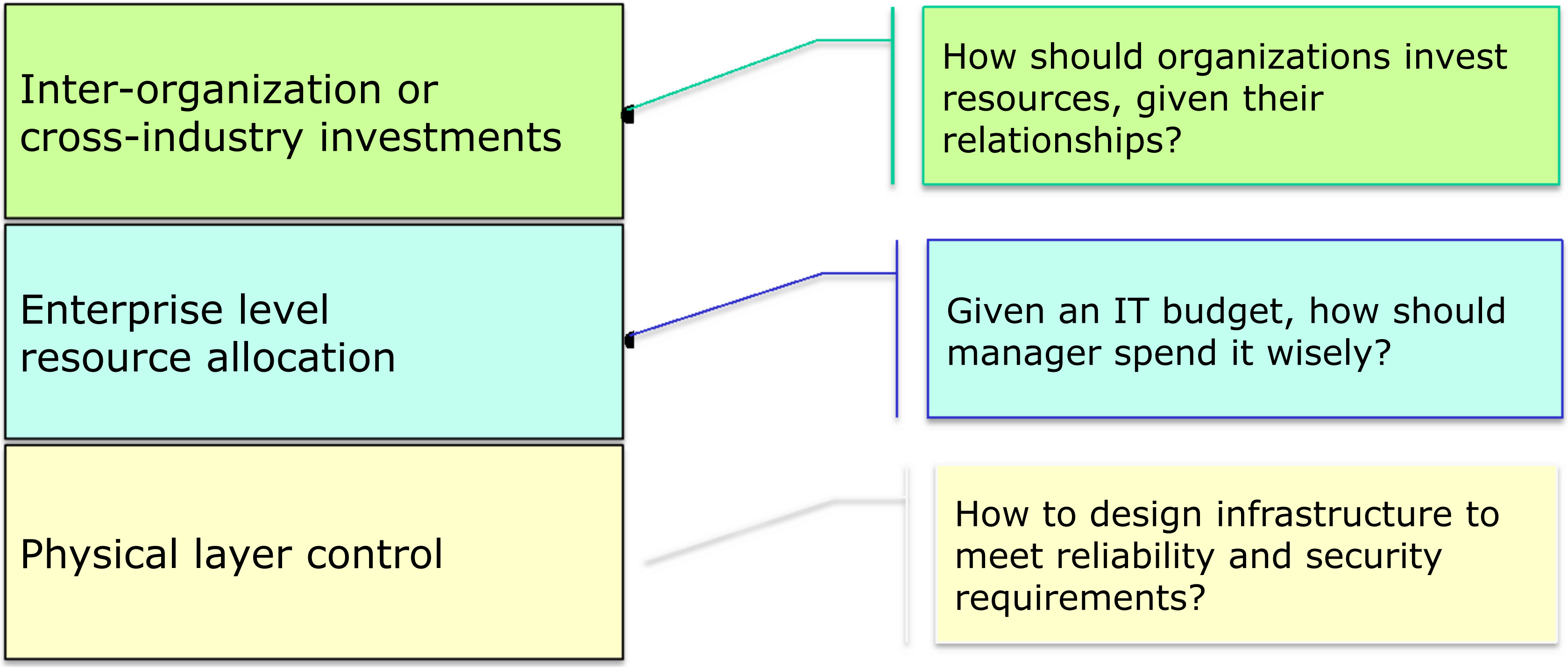


Computation Engine:

Optimization

Simulation

Levels at Issue - Examples



Inter-organization or cross-industry investments

How should organizations invest resources, given their relationships?

Enterprise level resource allocation

Given an IT budget, how should manager spend it wisely?

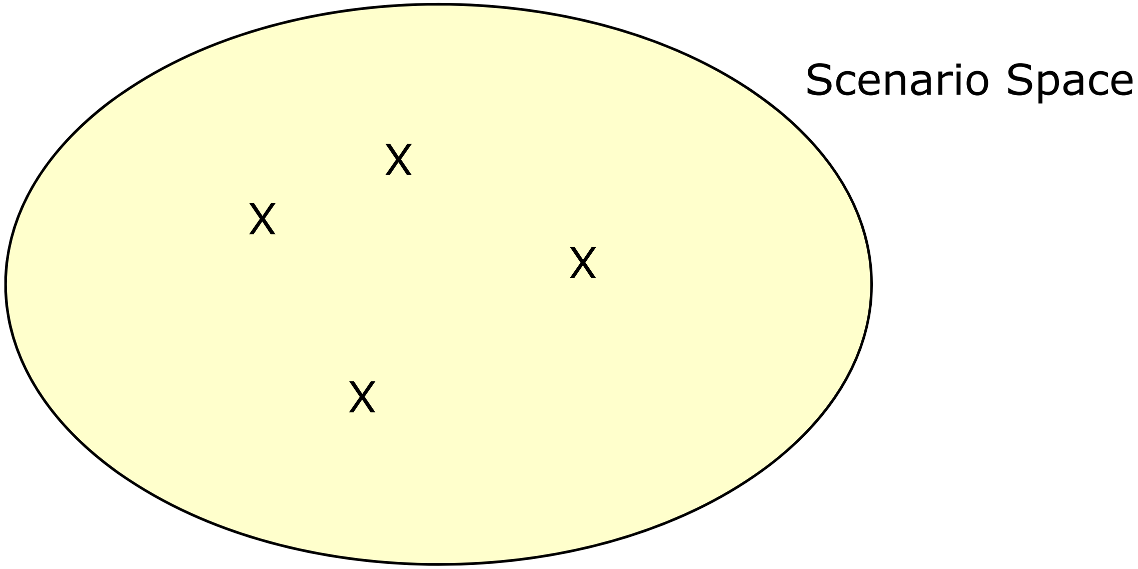
Physical layer control

How to design infrastructure to meet reliability and security requirements?

Multiple levels at issue

Cross-layer concerns

Approach… Need A Few Good Models



Very complex scenario/design space

Spotlight key paradigms and understand canonical models

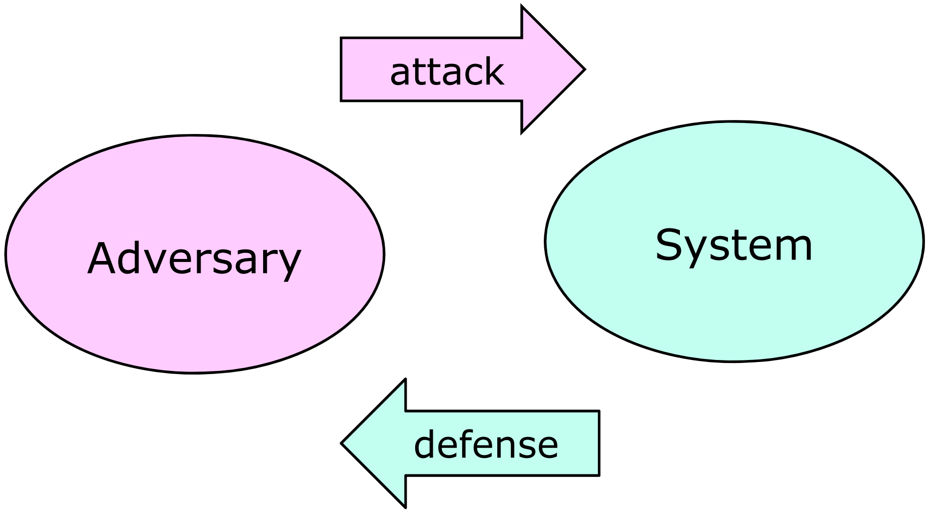
Aim for robust designs

Some Risk Management Paradigms

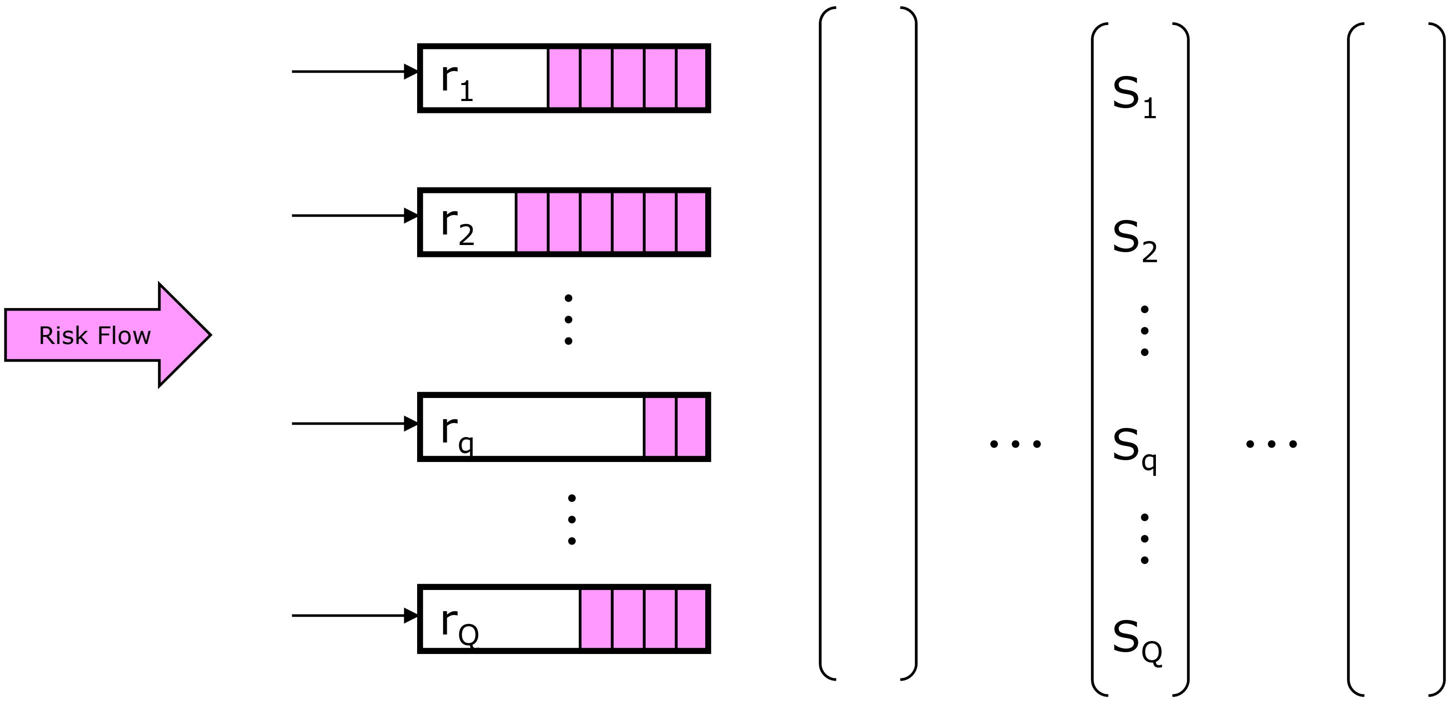
Managing Risk Dynamically

The Adversary vs. Defender Paradigm

(attack intensity vs. defense capacity)



The Basic Model … Note Queueing Analogy



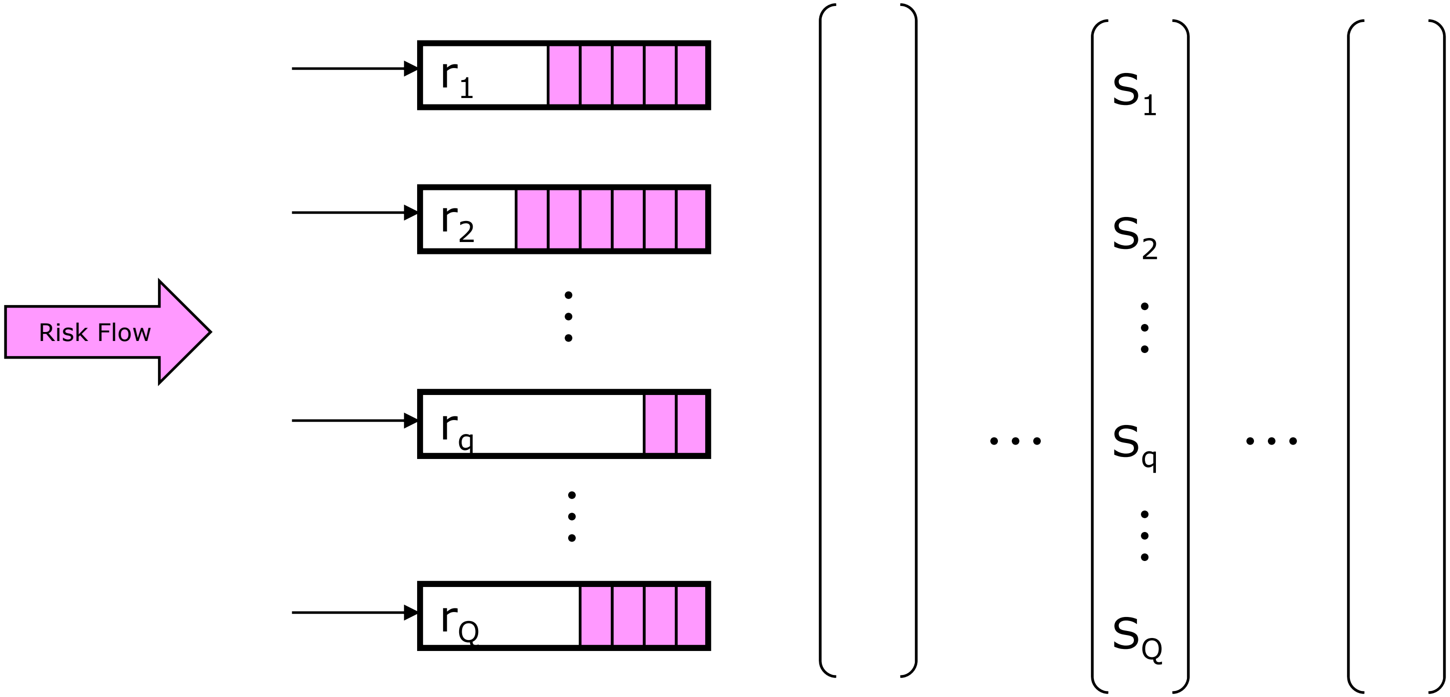
$r = (r_1 \ldots r_q \ldots r_Q)$ risk profile … r_q = risk indicator of node q

S = de-risking vector/mode/configuration/allocation… defense mode

$\mathcal{S}$ = set of all possible derisking vectors

C_S = cost of derisking vector S

The Basic Problem ... Note Queueing Analogy



Problem:

Given risk profile $r = (r_1 \ldots r_q \ldots r_Q)$ at time t,

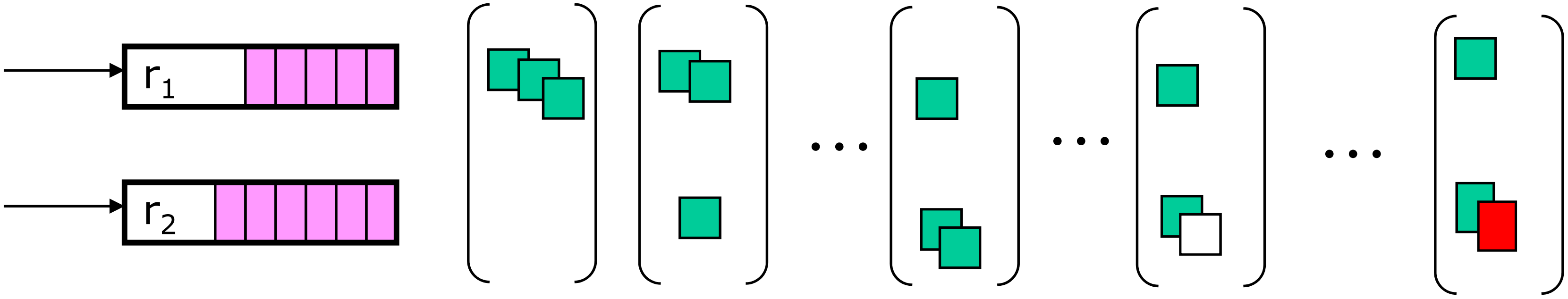
dynamically choose de-risking vector S from $\mathcal{S}$

to max. throughput, min. risk, min. cost, balance risk, etc.

A Simple Example …

Risk Profile ~ vulnerabilities (number/severity) to be patched on each node

Allocate 3 de-risking agents/workers to 2 nodes at risk



🟥 fast

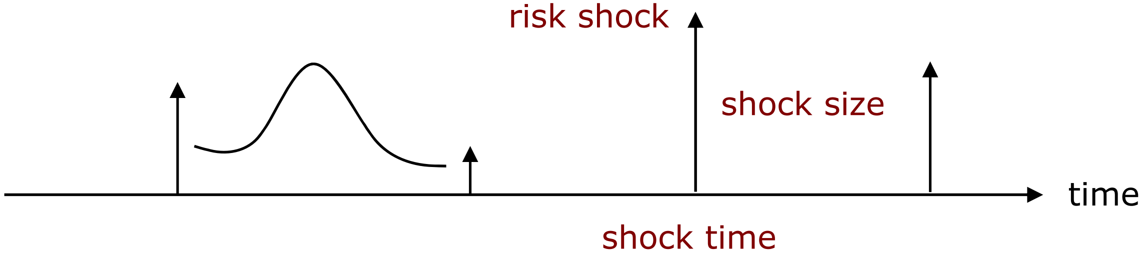
🟩 normal

□ slow

… in general … *any set* of de-risking vectors

Risk Flow, Load & Throughput

Risk Flow into node q



Risk Load $\rho = (\rho_1, \rho_2, \ldots, \rho_q, \ldots \rho_Q)$ … long-term avg. risk rate/intensity

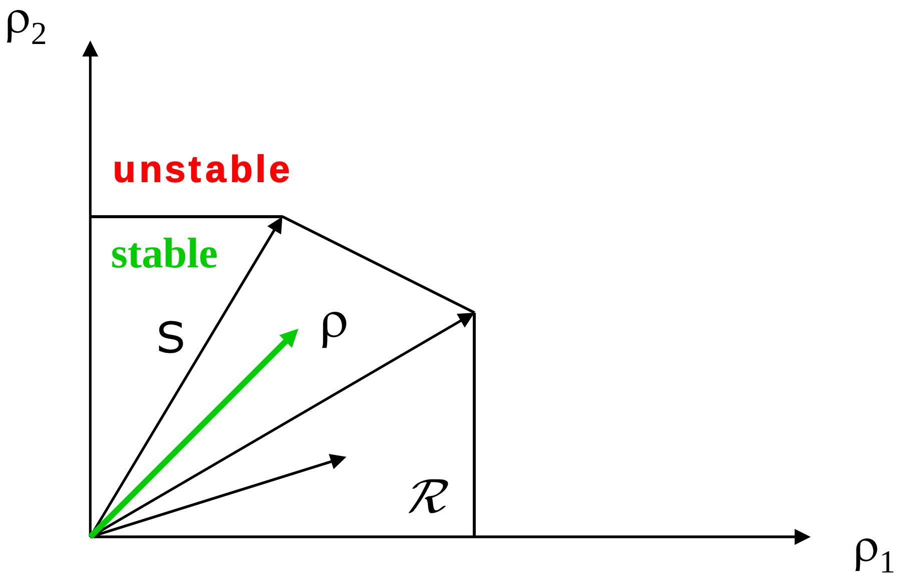
{cumulative risk into queue q in (0, t) } / t → ρ_q … as t →∞

Throughput … risk in-flow rate = risk out-flow rate (clearance rate)

… flow conservation

r(t) / t → 0 … as t → ∞

Throughput … Risk Mitigation Region



$R = \{\ \rho :\quad \rho \leq\ \Sigma_{S \in S}\ \phi_S\ S\ \ldots\quad \text{for}\ \ \text{some}\ \phi_S > 0\ \text{with}\ \ \Sigma\ \phi_S = 1\ \}$

Cone Policies Maximize `Protection’

Cone Policy… when risk profile r, choose S to maximize projection on **B**r

max <S, **B**r>   over S in *S*

maximizes throughput

for *any* fixed matrix **B** that is

positive-definite, symmetric and has

negative/zero off-diagonal elements

… **universally** on all adversarial traces

MWM algorithm … when **B=I**

Rich family of policies… (~ Q^2 matrix parameters to tweak and tune)

Extremely robust schedules

Simple`geometric’ operation

Simple Principle… and Robust Solution

Rule-of-Thumb:

Simply **align** *defense* profile to … current *risk/attack* profile

Robustness:

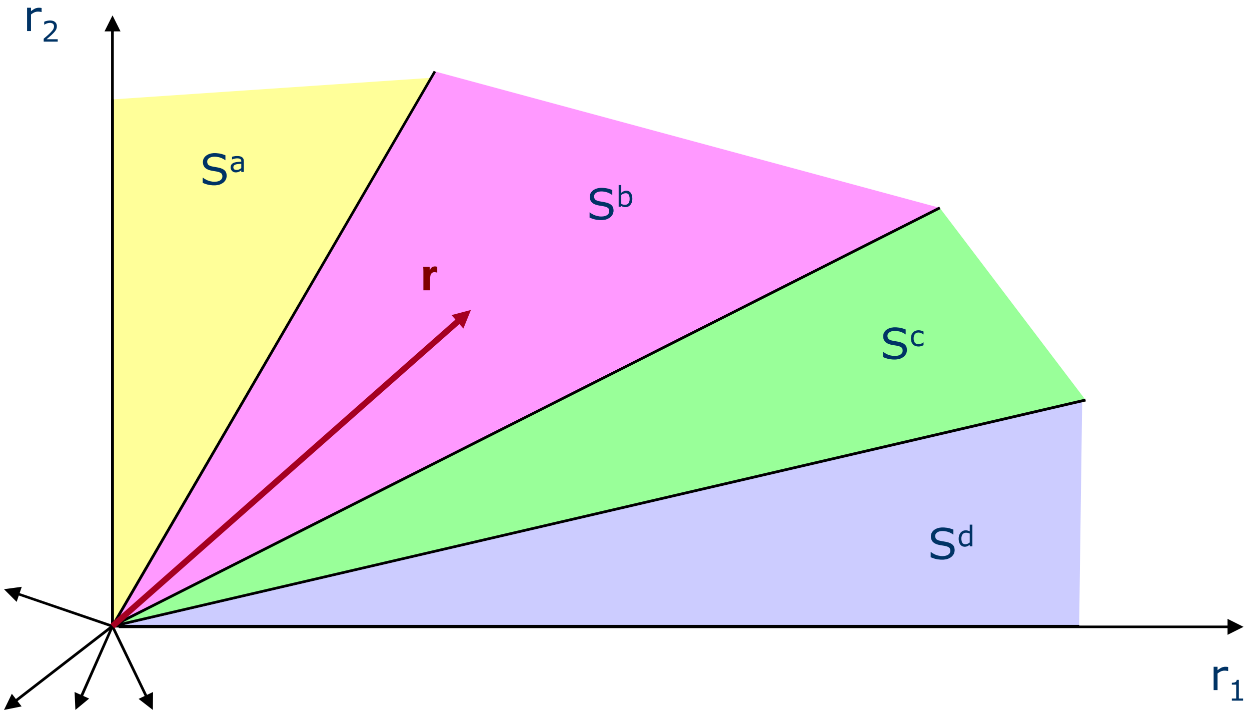
Avoids risk saturation even under

very ‘rough’ risk profile tracking (delayed, intermittent, erroneous)

very `sluggish’ defense response

Geometry… of Cone Policies

When risk profile r, choose S to maximize <S,**B**r> over all S in *S*



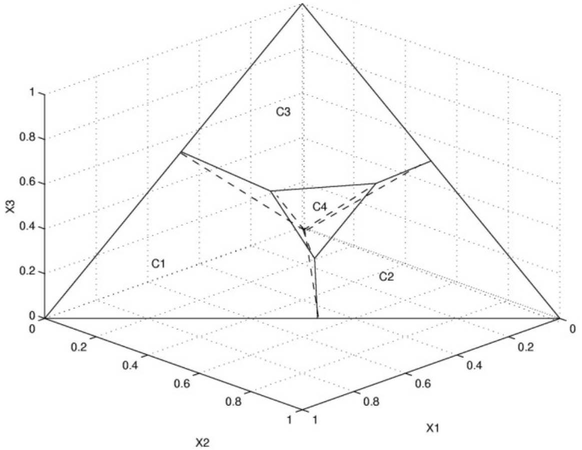
When risk profile r in cone C,

choose S = S(C) corresponding to that cone

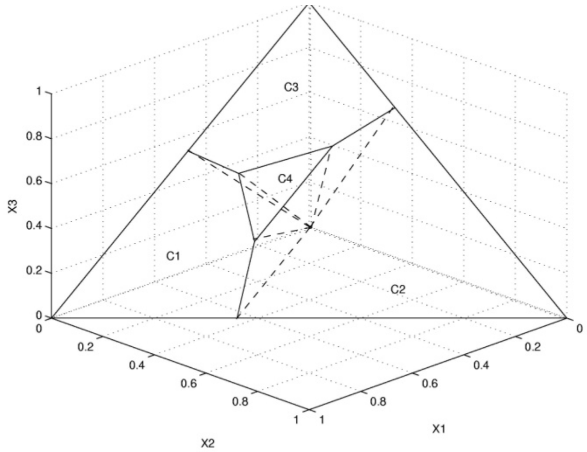
A 3-Node Example

S1=(9,0,0) / S2=(0,8,0) / S3=(0,0,8) / S4=(3,4,3)

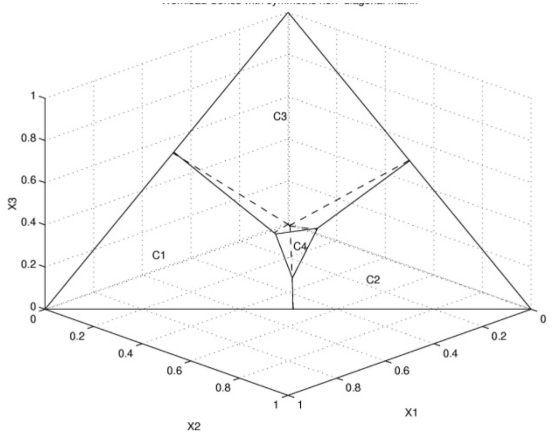
B=[1,0,0; 0,1,0; 0,0,1]



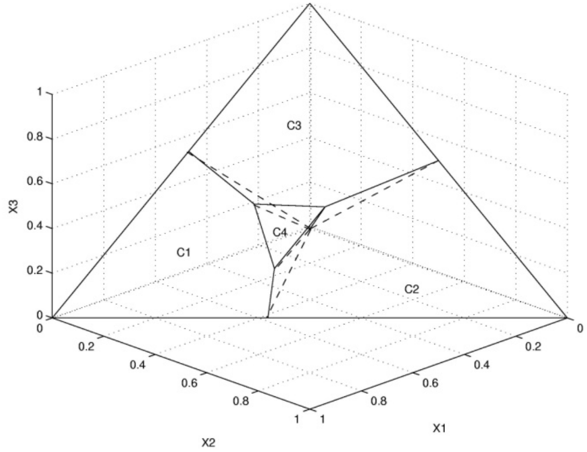
B=[1,0,0; 0,2,0; 0,0,1]



B=[1,-0.5,0; -0.5,1,0; 0,0,1]

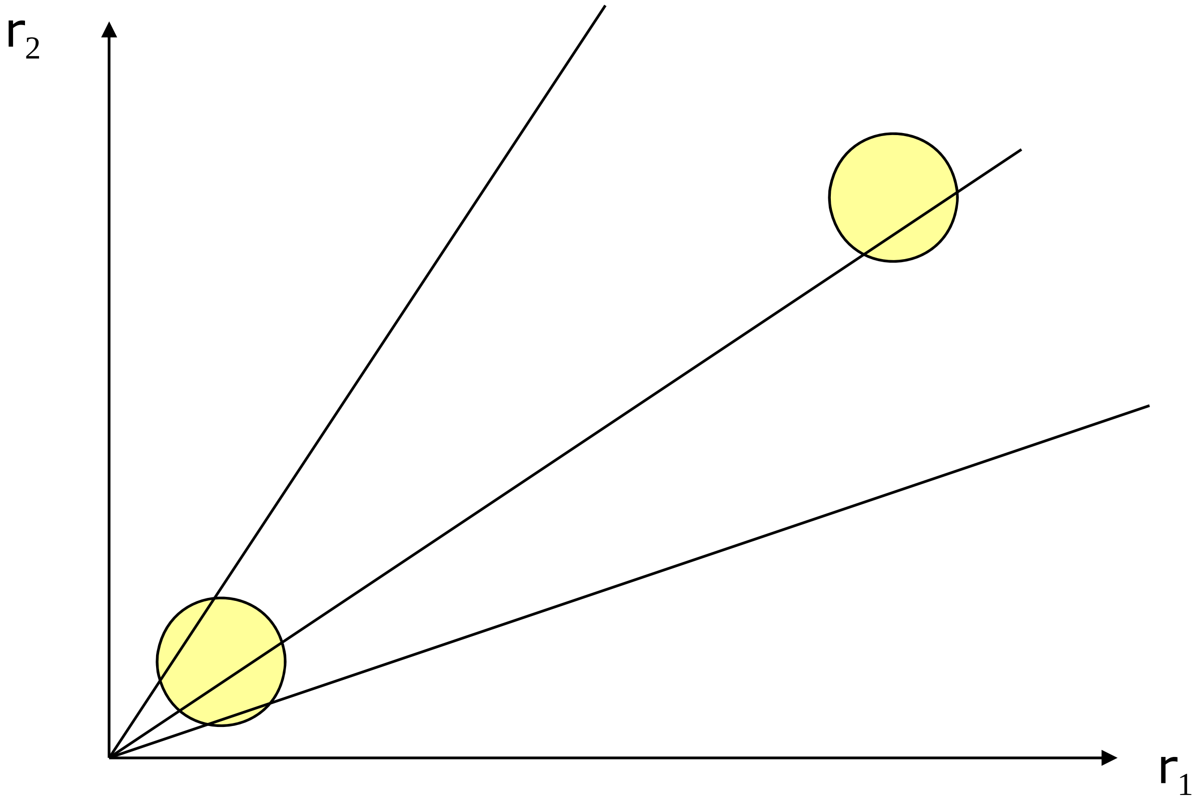


B=[1,-0.5,0; 0,1,0; 0,0,1]



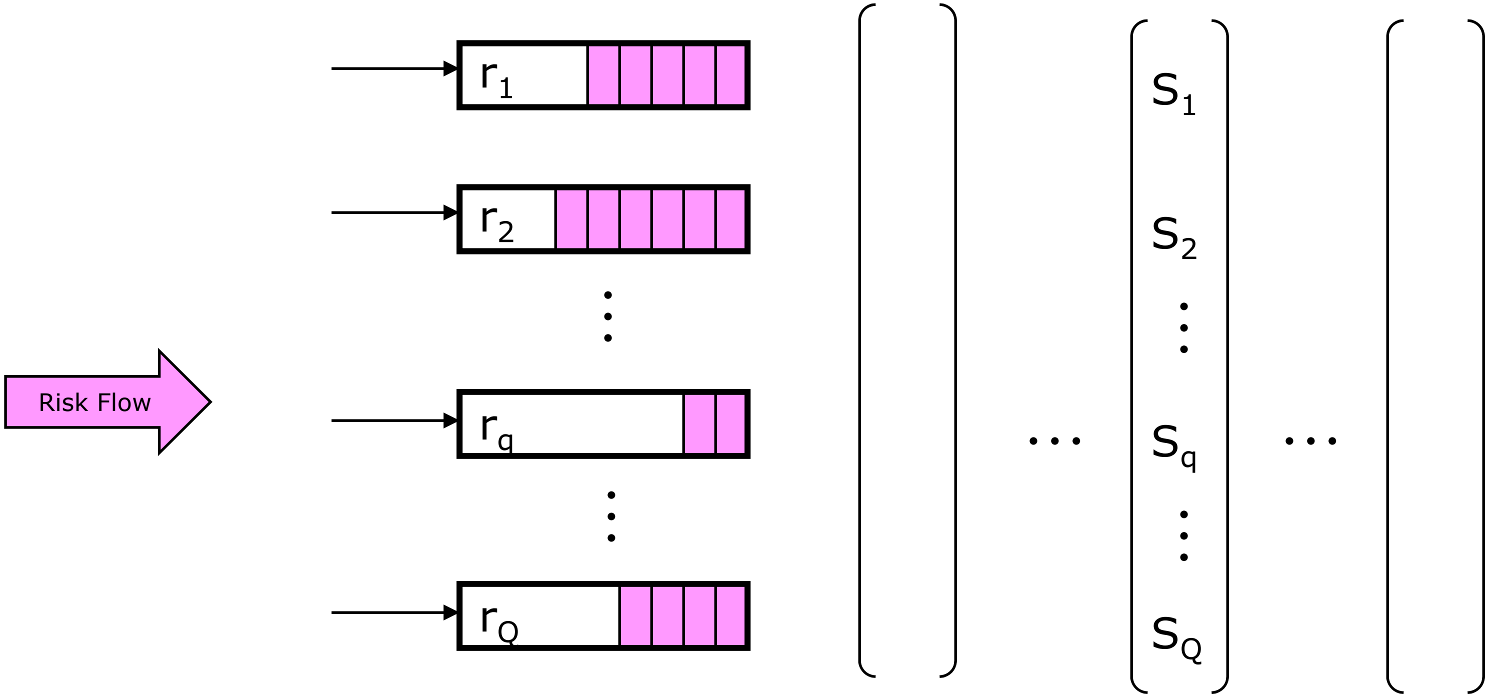
Local Search

Assume bound on ‘risk jumps’



Have to search only neighbor cones … fewer as risk profile grows! … **Local Search**

The Basic Model … Risk vs. Cost



r = (r_1 … r_q … r_Q) risk profile

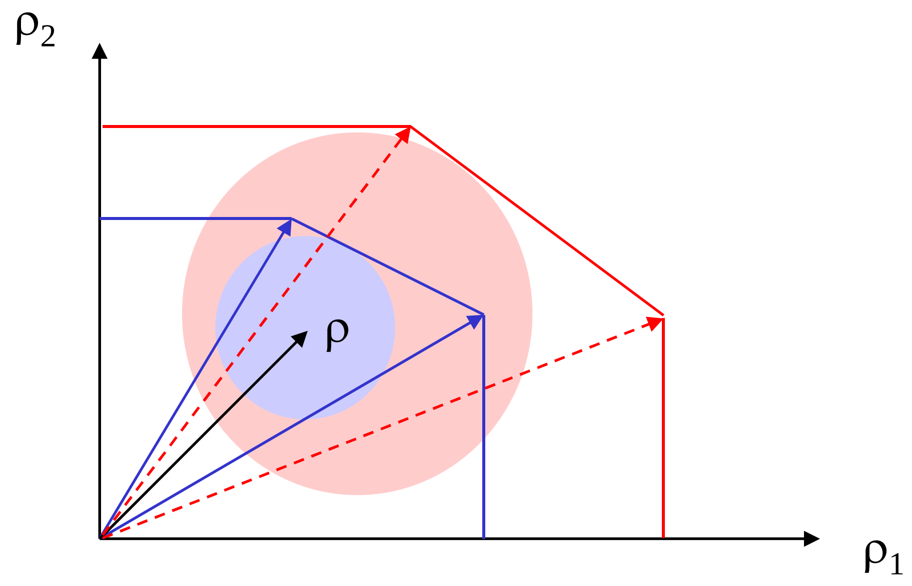
S = de-risking vector

C_S = cost of de-risk vector S

Core Issue… dynamically choose S to **minimize risk + resource cost…**

… dynamic programming formulation

Risk vs. Cost Control … Key Idea



Activating more/less expensive de-risk vectors… adjusts the capacity space

Still need to manage risk excursions beyond stability…

Allocating Protection & Recovery Resources

Which nodes/links should be hardened?

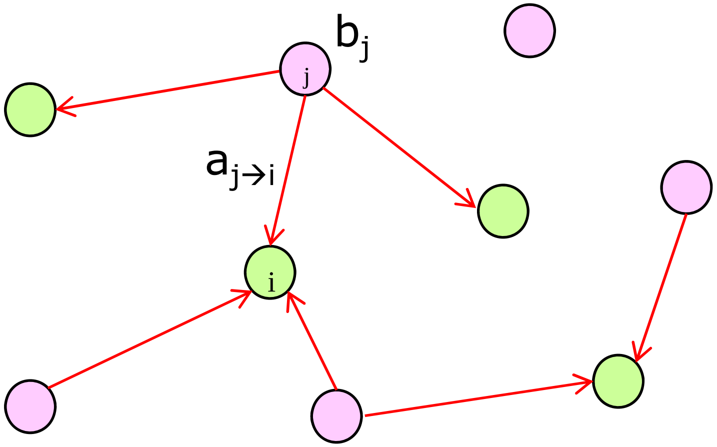
Network Topology Matters!

Network `Epidemics’

$r = (r_1 \ldots r_i \ldots r_j \ldots r_N)$ risk profile … r_q = risk indicator of node q

$r_i = 1$ … node i infected (`risky’ … compromised)

$r_i = 0$ … node i healthy (derisked … secure)



Stochastics of r_i : Markov chain with

$0 \to 1$ … with infection rate $\Sigma_{\{j:\, r_j=1\}}\, a_{j\to i}$

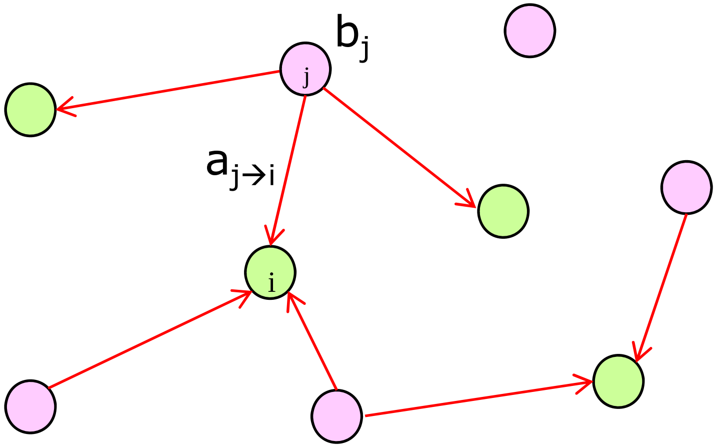
$1 \to 0$ … with recovery rate $b_i > 0$

…. hits r = 0 (all clear) with prob. 1

Speed of Risk Clearance

$dP_t(r)/dt \; = [\mathbf{A}\text{-}\mathbf{B}]\; P_t(r)$

$\mathbf{A} = \{a_{j\rightarrow i}\}$ and $\mathbf{B} = \mathrm{diag}\{b_i\} > 0$



Lower *spectral radius* $\varphi(\mathbf{A}\text{-}\mathbf{B})$ →

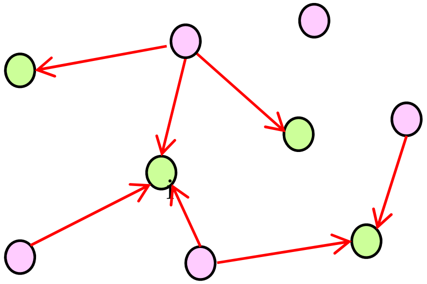
more aggressive derisking →

shorter time to risk clearance

Allocating Resources… Key Idea

Protection resources x (link hardening)

decrease infection rates $\mathbf{A}(\mathbf{x}) = \{a_{j\rightarrow i}(x)\}$



Recovery resources y (node resilience)

increase recovery rates $\mathbf{B}(\mathbf{y})=\mathrm{diag}\{b_i(y)\}$

Given protection-recovery resource budget B(x,y) < B

… maximize the risk clearance speed (spectral radius)

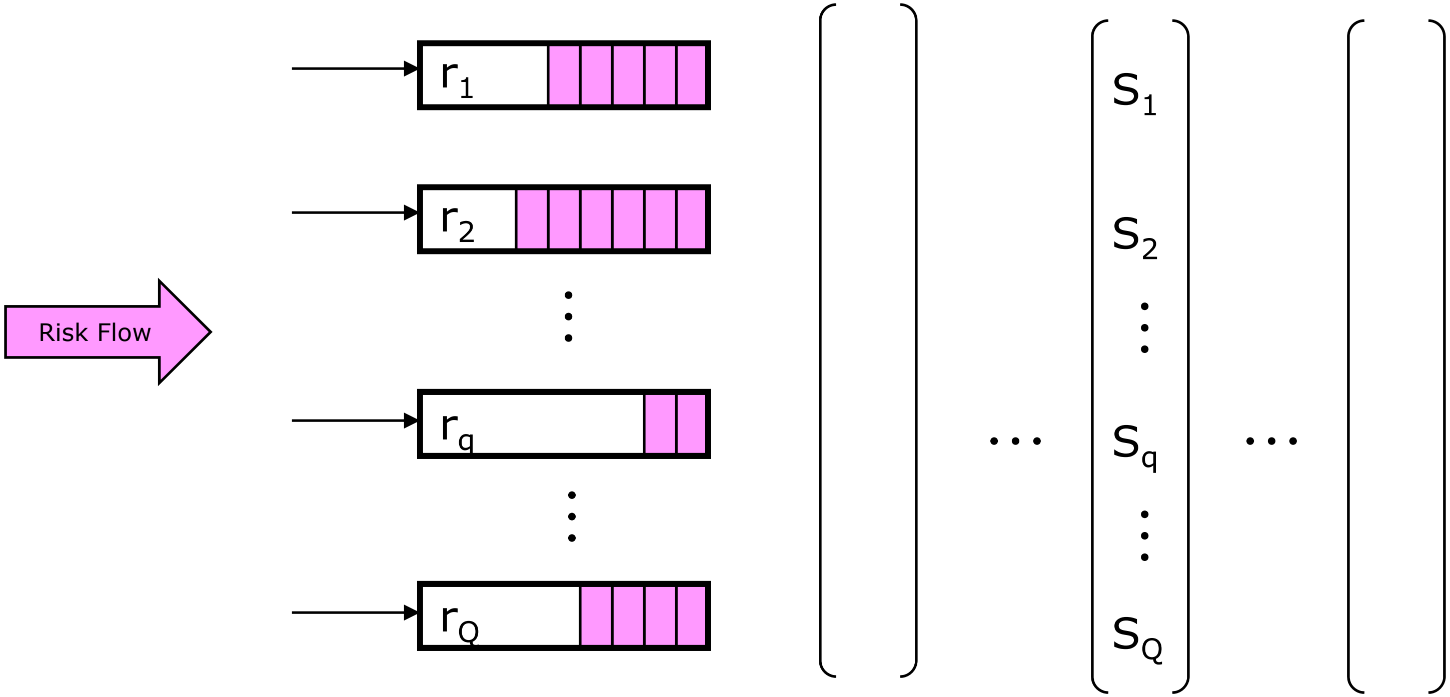
Given target risk clearance speed (spectral radius),

… minimize total protection-recovery resource budget B

For certain convex functions , problems can be solved using geometric programs, semi-definite programs, etc. via eigenvalue optimization techniques.

Maintaining Acceptable Risk Levels

The Basic Model ... again



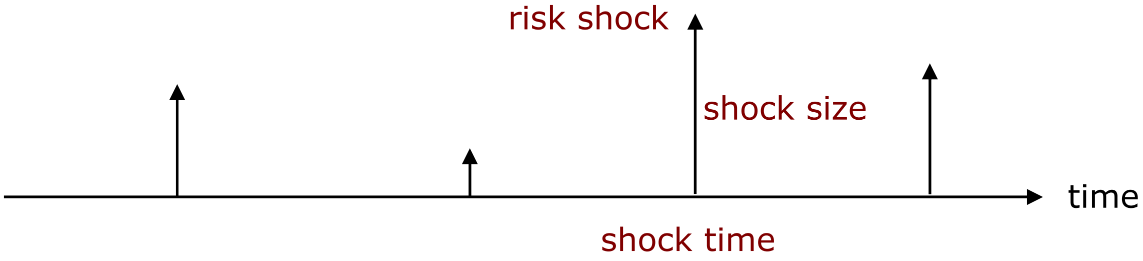
$r = (r_1 \ldots r_q \ldots r_Q)$ risk profile ... r_q = risk indicator of node q

S = de-risking vector/mode/configuration/allocation

$\mathcal{S}$ = set of all possible derisking vectors

C_S = cost of derisking vector S

Markovian Setup …



Risk flows = independent Poisson

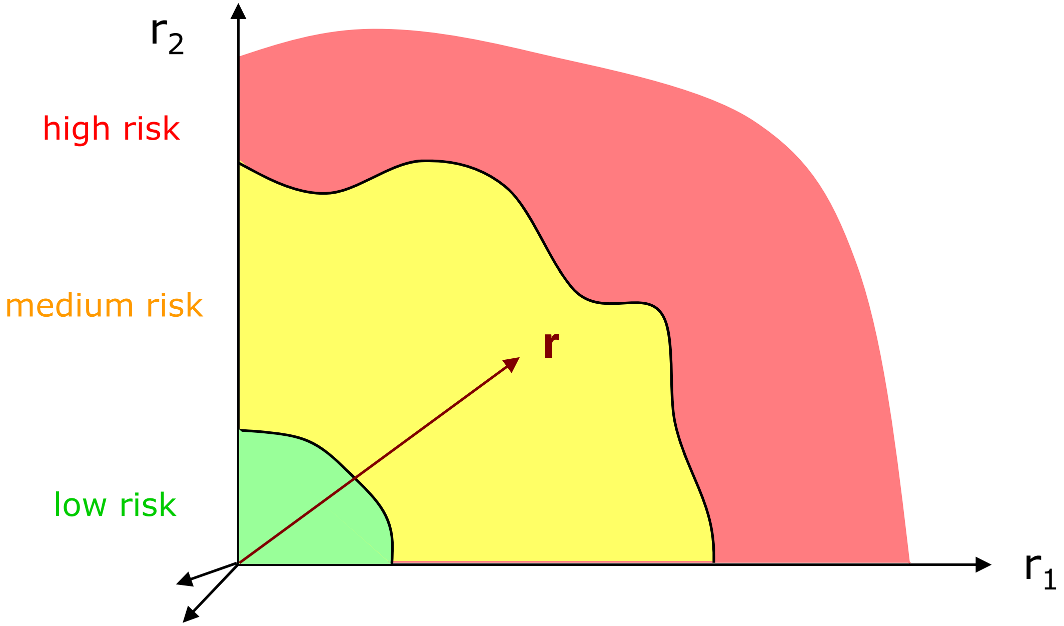
Shock Sizes = i.i.d. exponential (cont. time) or 1 (discrete time)

De-Risking Vectors $S=(S_1 \ldots S_q \ldots S_Q)$ with

S_q = risk drain rate at node q

… controlled Markov chain

Risk Surfaces and Regions/Sets



Three related objectives:

When at risk profile r,

choose de-risking vector S to

- min. time to green or
- max. time to red
- max. prob. of getting to green before red

(if S were kept fixed … which is not !)

Three Related Controls

Min. time to green… S*(r) = argmin L(r, S) over S

L(r, S) = E[time to green | start at r, use S throughout]

Max. time to red… S*(r) = argmax H(r, S) over S

H(r, S) = E[time to red | start at r, use S throughout]

Get to green before red… S*(r) = argmax P(r, S) over S

P(r, S) = Prob[hit green before red | start at r, use S throughout]

Note… L(r, S), H(r, S), P(r, S)

can be explicitly computed in Markovian setup,

but have complexity issues…

In Conclusion…

IT Risk Mitigation is

- already critical need and of rapidly growing importance (& complexity)
- at infancy (little agreement even on risk metrics…)
- highly qualitative (and instinctive) today
- quantitative methods at very early stage

There is need for

- risk ‘analytics’
- computation(sim/opt)-based decision support systems
- development of risk mgt. ‘Cockpit’

Thank You!