USING GAME THEORY TO ASSESS THE STRENGTH OF AN AV SYSTEM AGAINST EVOLVING OFFENCES

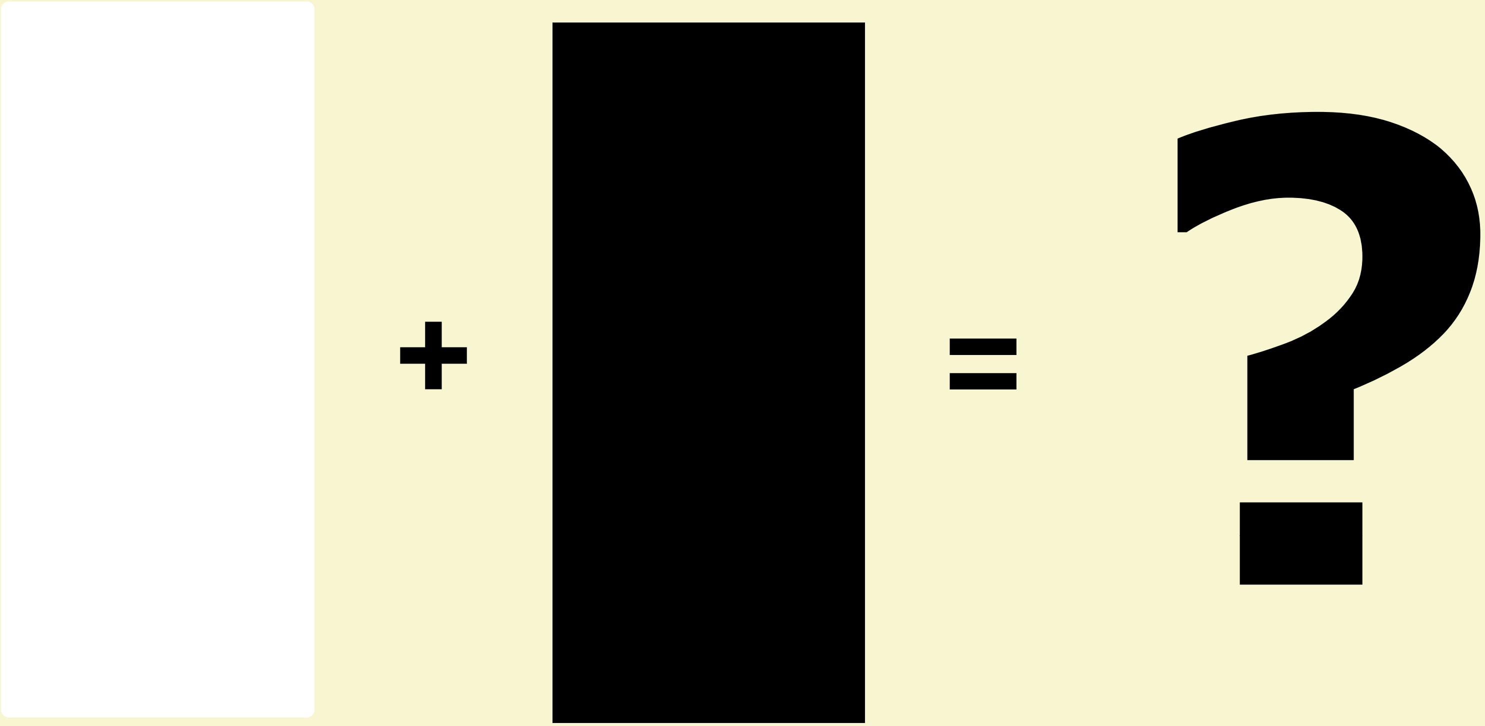
Bin Mai

Northwestern State University

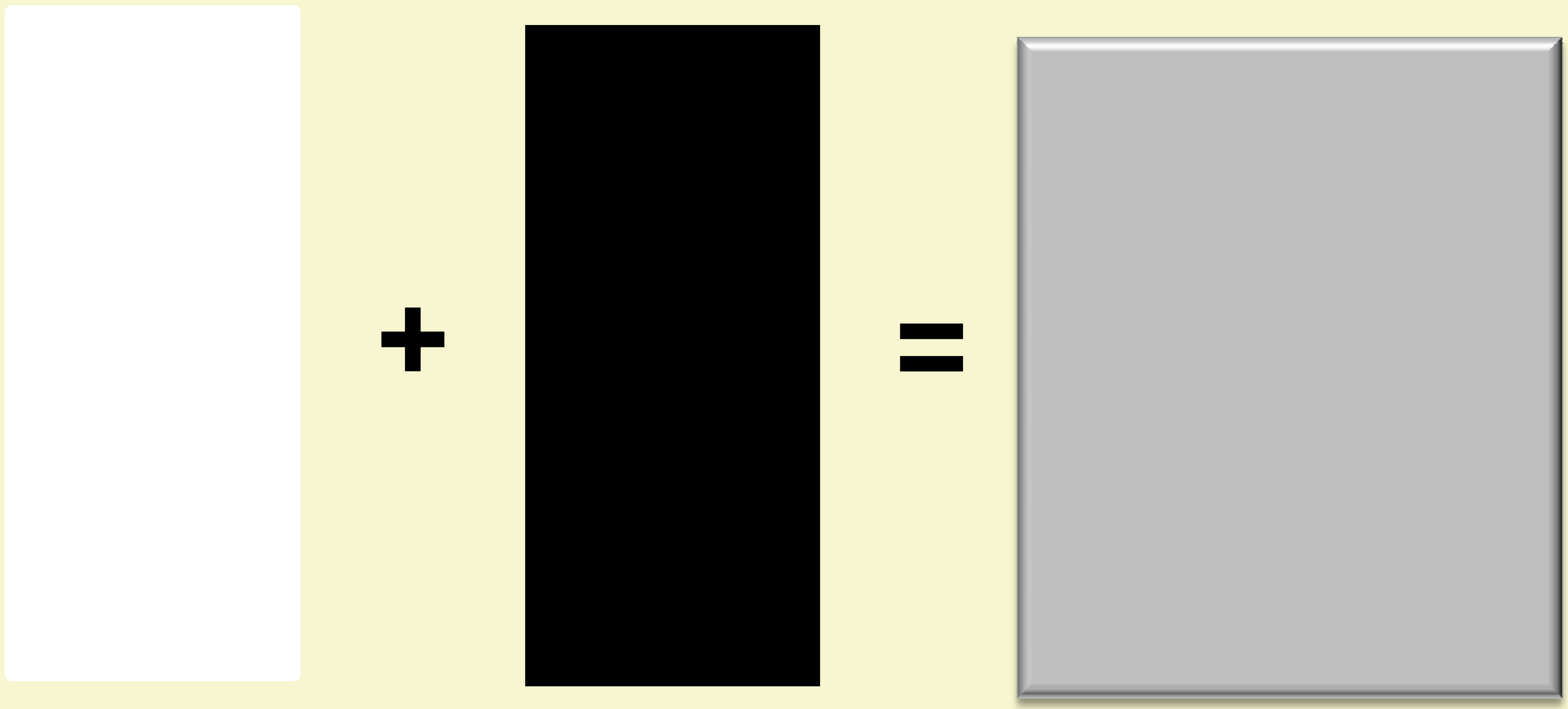
Anshuman Singh, Andrew Walenstein, Arun Lakhotia

University of Louisiana at Lafayette

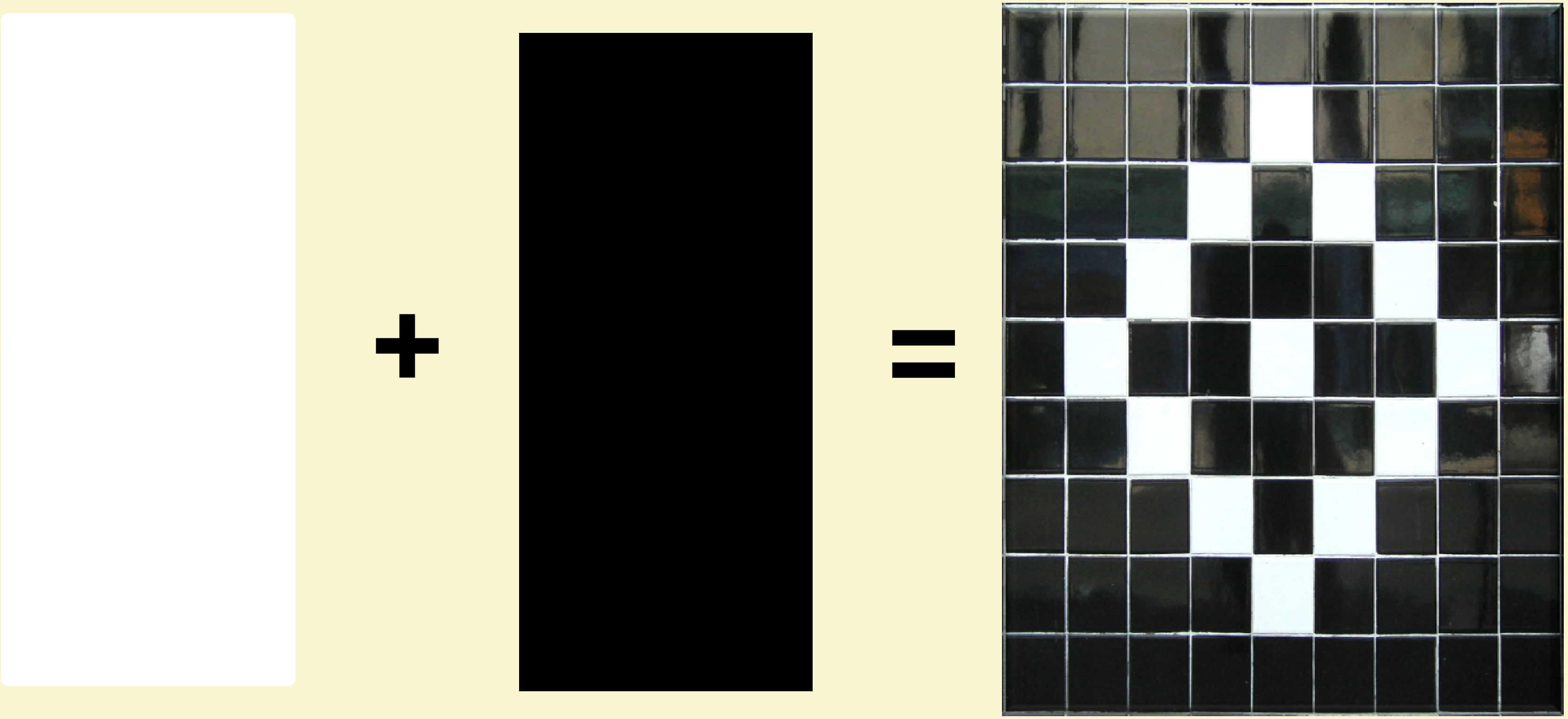
White List + Black List = ?



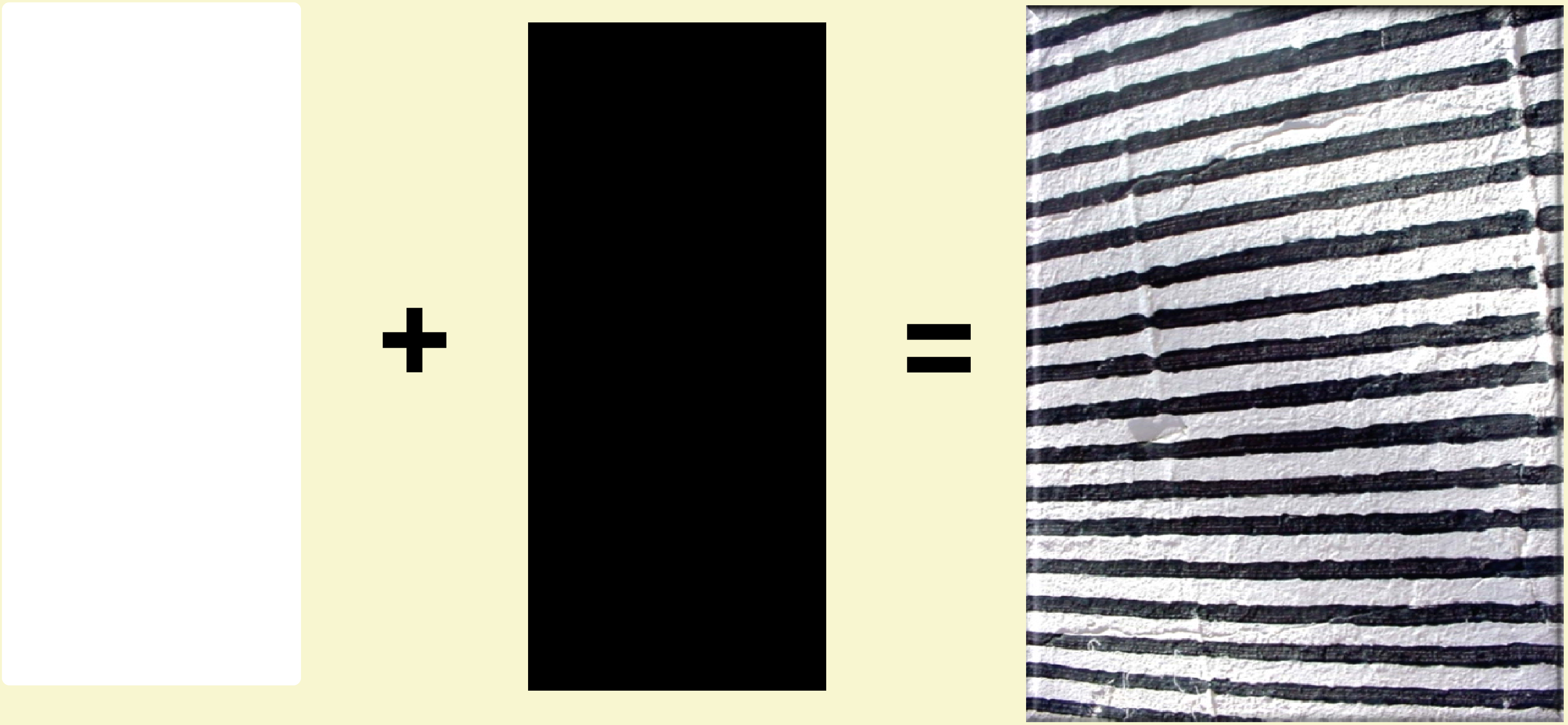
White List + Black List = ?



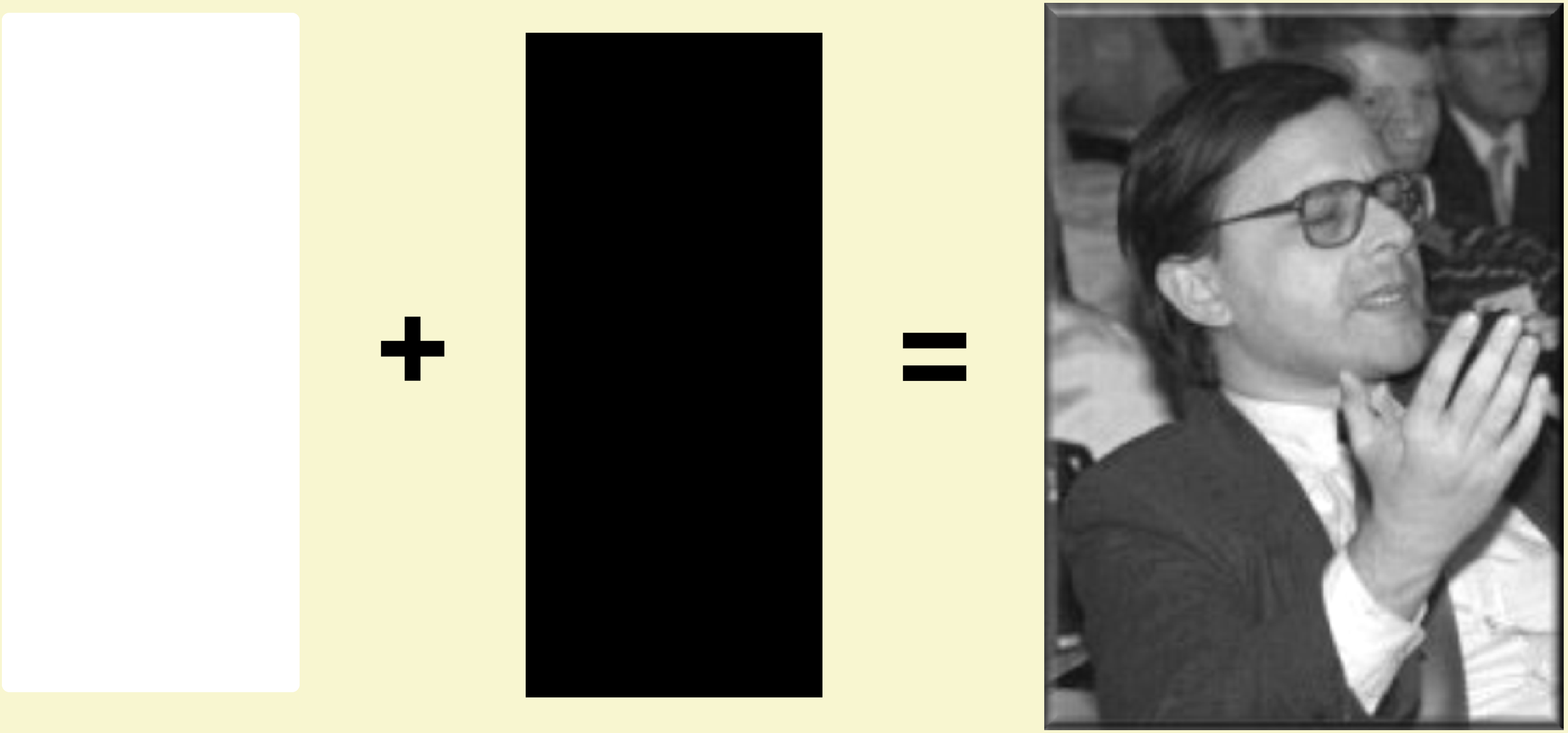
White List + Black List = ?



White List + Black List = ?



White List + Black List = ?



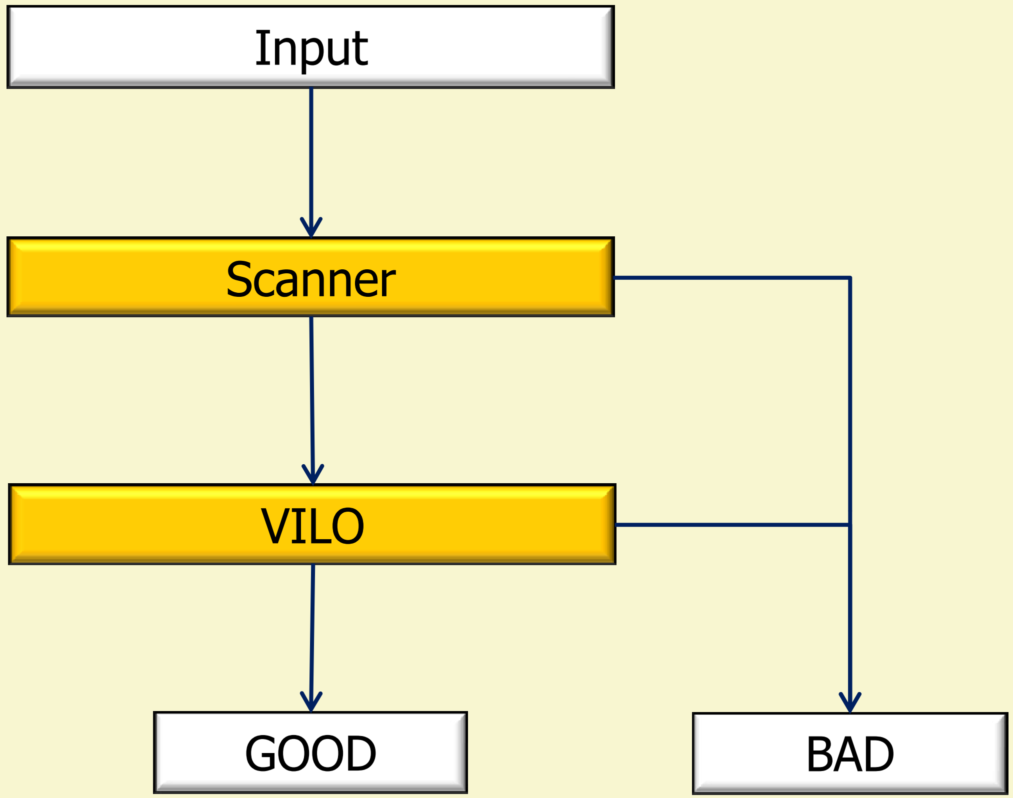
Composition Experiences

- Choices reflect composition ideas in AV
  - Which choices do we use?
- We were experimenting with composition
  - We were building tools for analysis and detection
  - Could compose and integrate in variety of ways
- Ran into some interesting questions along the way…

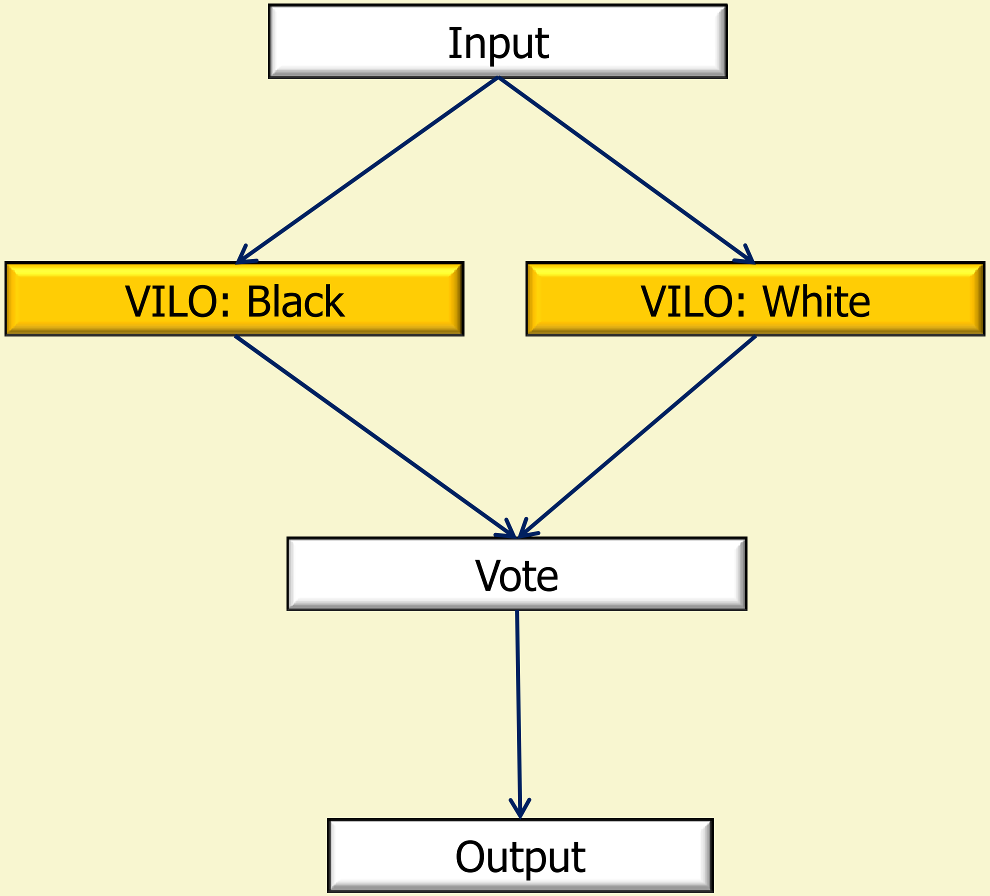
Vilo Composition

- Vilo:
  - Automated classifier / detector
    - Analyzes binaries
    - Constructs generic matcher
  - Applications
    - Black list: use malware for training
    - White list: use benign for training
- Question:
  - How to integrate Vilo into AV systems?

Composition: Vilo in Sequence



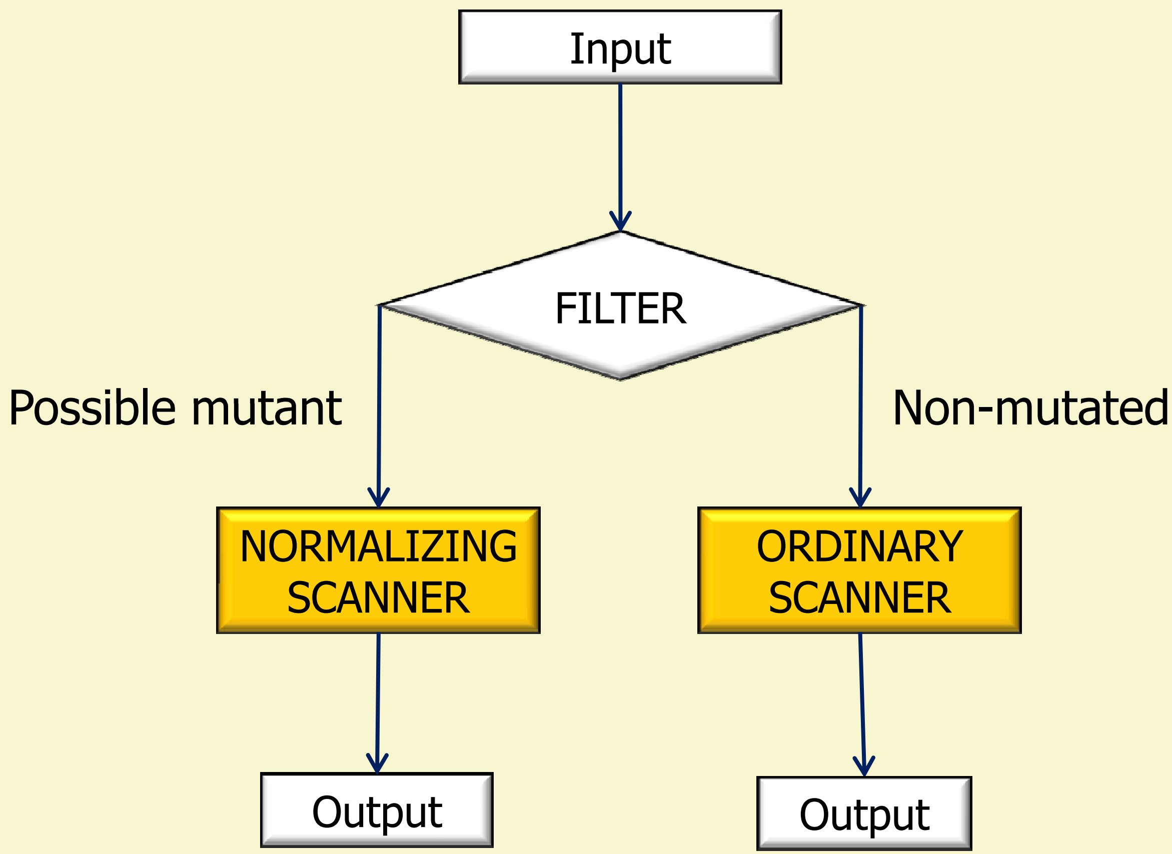
Composition: Vilo in Parallel



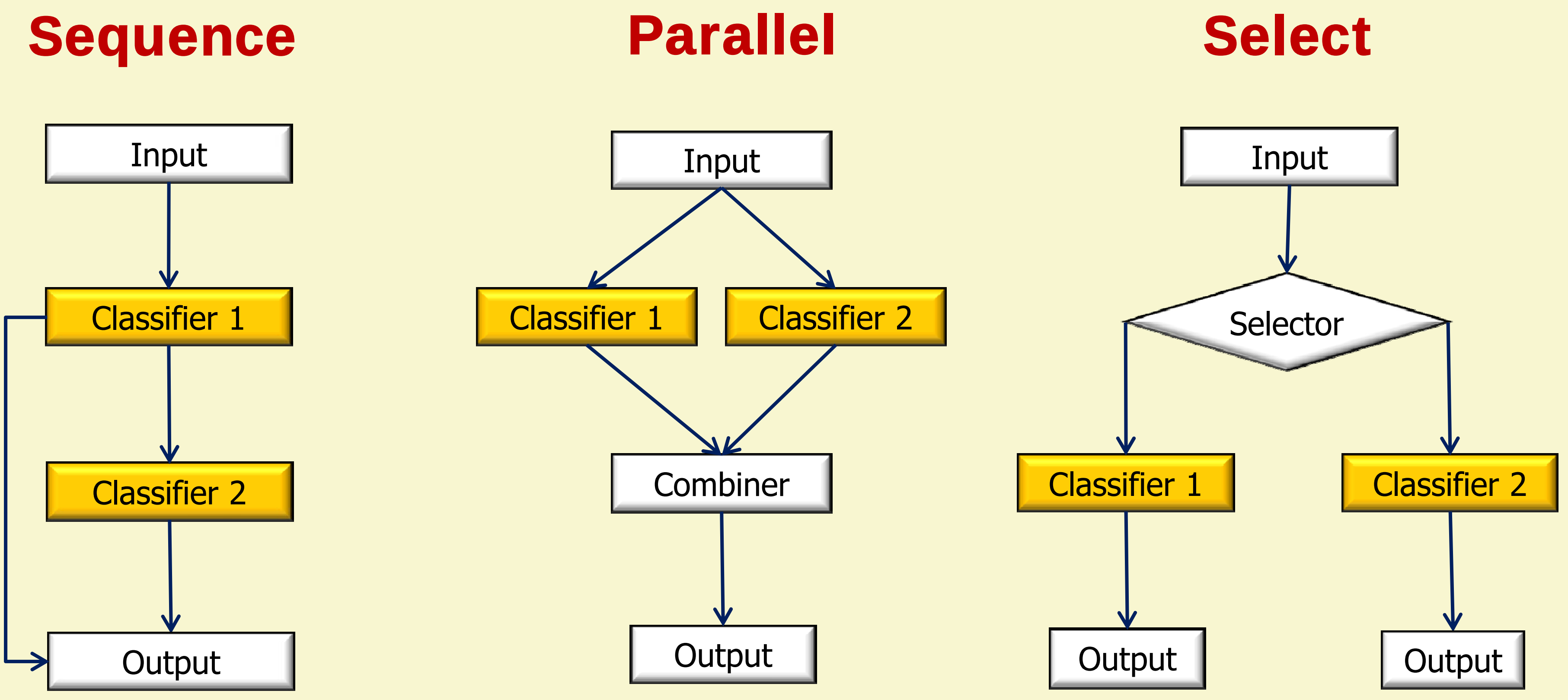
Normalizer/Filter Composition

- Mutant Normalizer:
  - Converts mutants/variants to a single form
  - Perform ordinary scan on normalized form
  - Concern:
    - Relatively expensive
  - Solution:
    - Apply only on files likely to be mutants
    - We developed fast filter to identify likely mutants
- Question:
  - How to integrate Normalizer/Filter into AV system?

Composition: Filter/Normalizer



Composition Choices



Composition: a Way of Life

- Multiple identification methods in AV scanners
  - Crypto checksum
  - X-ray scanning
  - Static behaviour-based patterns
  - Emulation-based behavior matching
  - Heuristic analysis

  …
- Common to be combined in AV system
  – Including composition using selection logic

Important Questions

- Compositions raise important questions
  - How do we know a composition is any good?
  - How do we tune the composed system for optimal performance?
  - If two classifiers are optimally tuned, will their composition also be optimal?
- Game theory can yield insight into such composition problems

Some Interesting Results

- For a particular selector/classifier architecture it can be shown that:
  - Adding another classifier to a scanner will not always make a scanner better
    - better *only if* the cost of "stealthing" the selector is above a specific threshold
  - AV designer is always advised to deter "stealthing" the selector
    - by increasing the spread in the detection rates of the classifiers

Steps To Enlightenment

- Introduce general game theory approach
- Show application to two common architectures
- Derive analysis of game strategies

Game Theory: General Approach

- Game Theory:
  - Aid to analyze strategic choices of adversaries
- Basic idea:
  - Model adversary interaction as game
  - Associate payoffs (costs/benefits) to outcome
  - Mathematical manipulation to analyze strategies
    - Search for optimal strategies

Game Theory Modeling Process

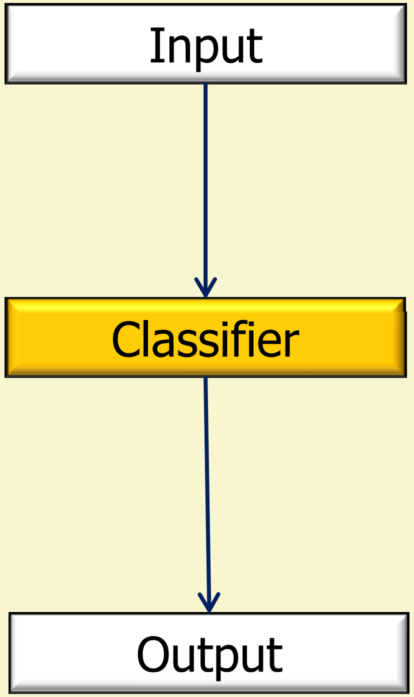
- Four steps
  1. Identify agents
  2. Identify game parameters
  3. Develop game tree
  4. Analyze tree to compute expected payoffs

- Two step introduction using two games:
  - Simple: One classifier
  - Complex: Selector/Classifier

Step 1: Agents & Roles

- Normal User (NU)
  - Presents uninfected files to the system
  - Wants to use system to derive positive utility
- Malware Author (MA)
  - Presents infected files to system
  - Wants to attack system to derive positive utility
- Security Analyst (SA)
  - Attempts to provide optimal system, including:
    - Detect and thwart MA's malware
    - Minimize AV system's total cost

Step 1: Game



- Simple game setup
  - Single classifier

Step 2: Classifier Parameter

- Many classifiers have a tunable parameter
  - Parameter trades off between
    - P_D: true positive rate
    - P_F: false positive rate
  - ROC curve shows relation between P_D and P_F

ROC Curve

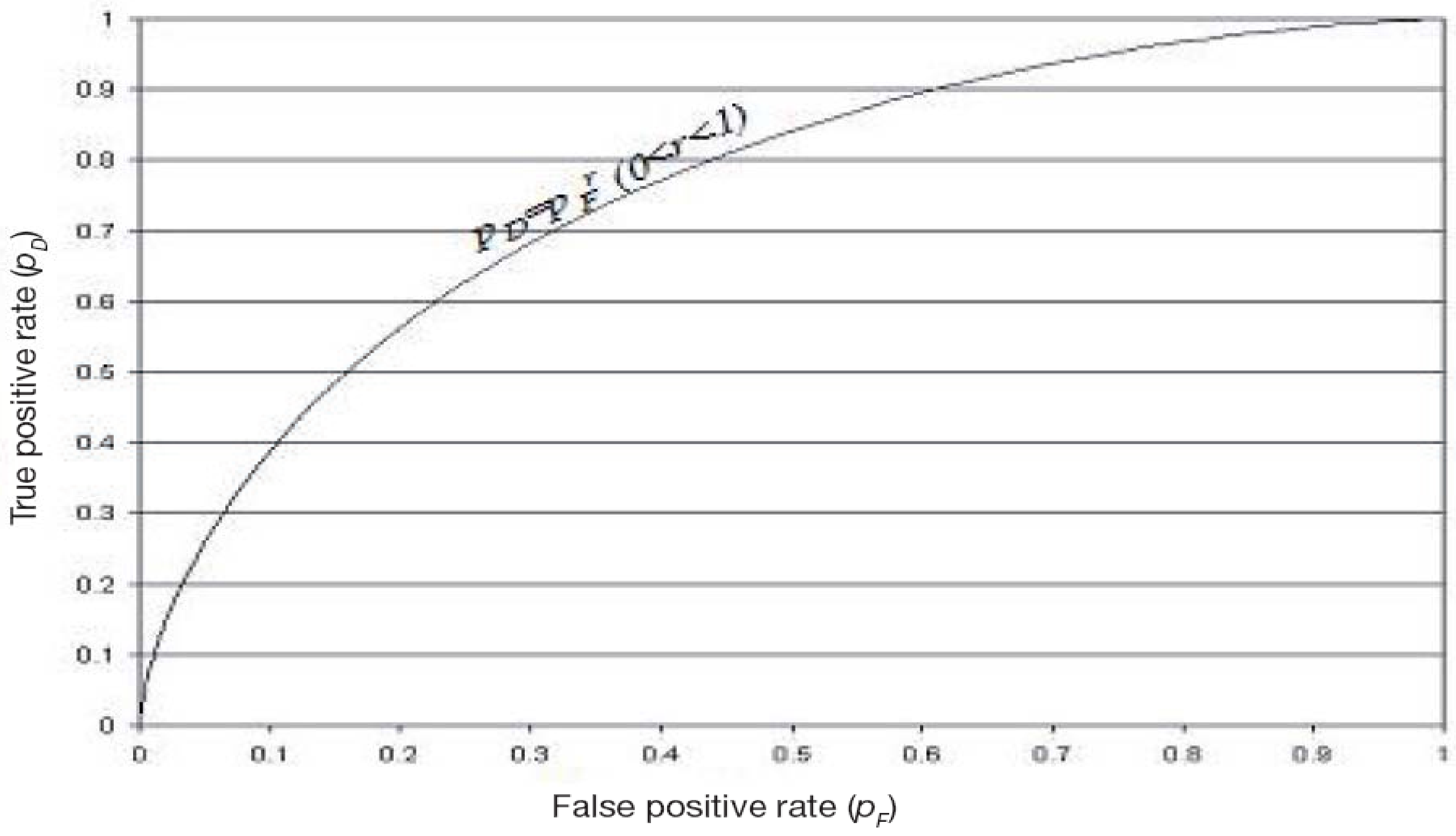


Figure 2: A ROC curve

Step 2: Classifier Parameter

- Parameters are settings the SA can control
  - These define the SA's moves in the game
  - Need to choose parameter to model ROC
- Typical ROC curves follow power function
    - $P_D = P_F^{\,r}$, $0 < r < 1$
  - *r* can be used as model parameter
    - i.e., SA chooses *r* as part of the game strategy

Step 2: Outcomes and Payoffs

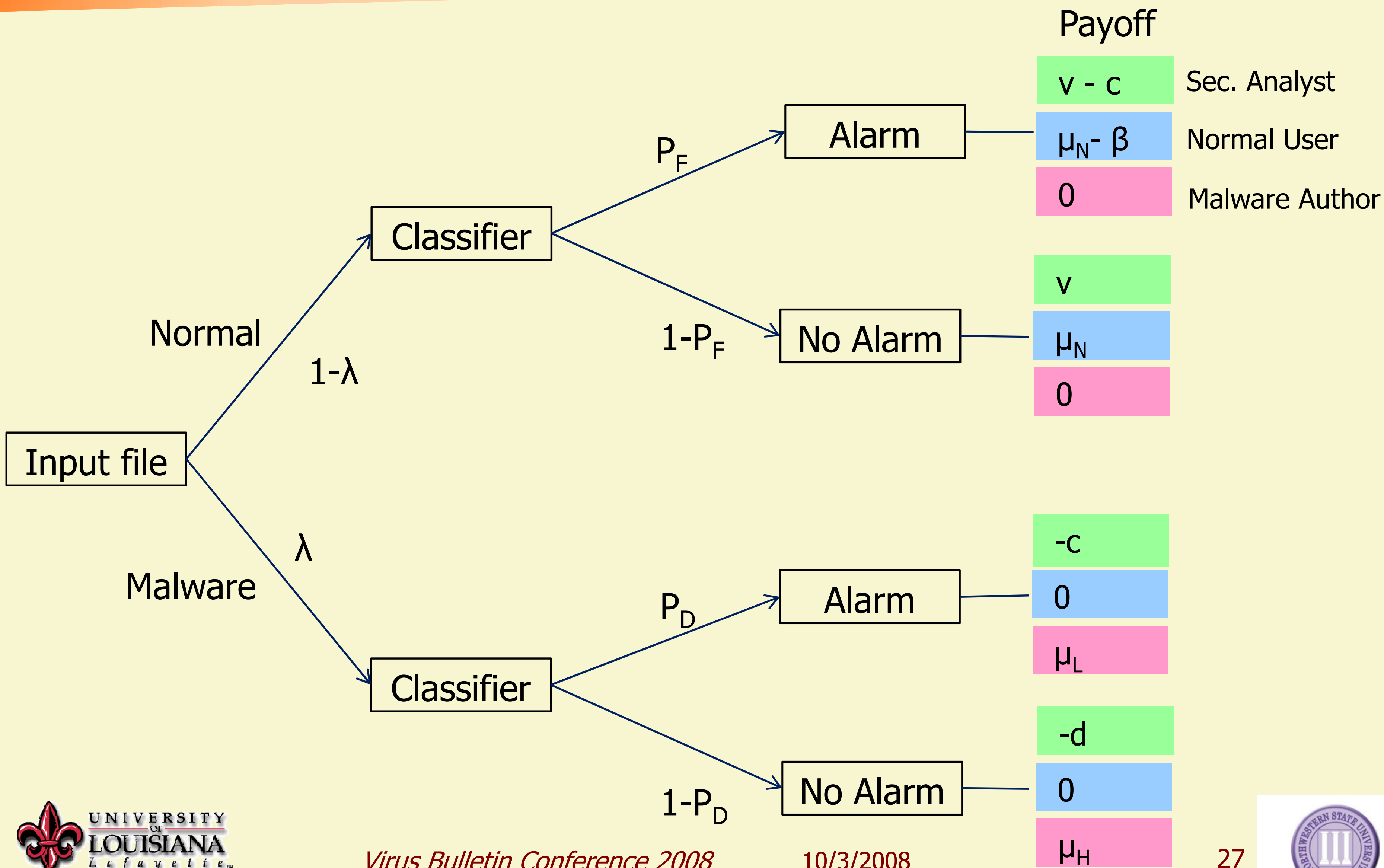
- Actual payoffs for agents are their *benefits less costs* given a particular game outcome
- Define the game based on analysis of SA, MA, and NU payoffs

Step 2: Outcomes and Payoffs

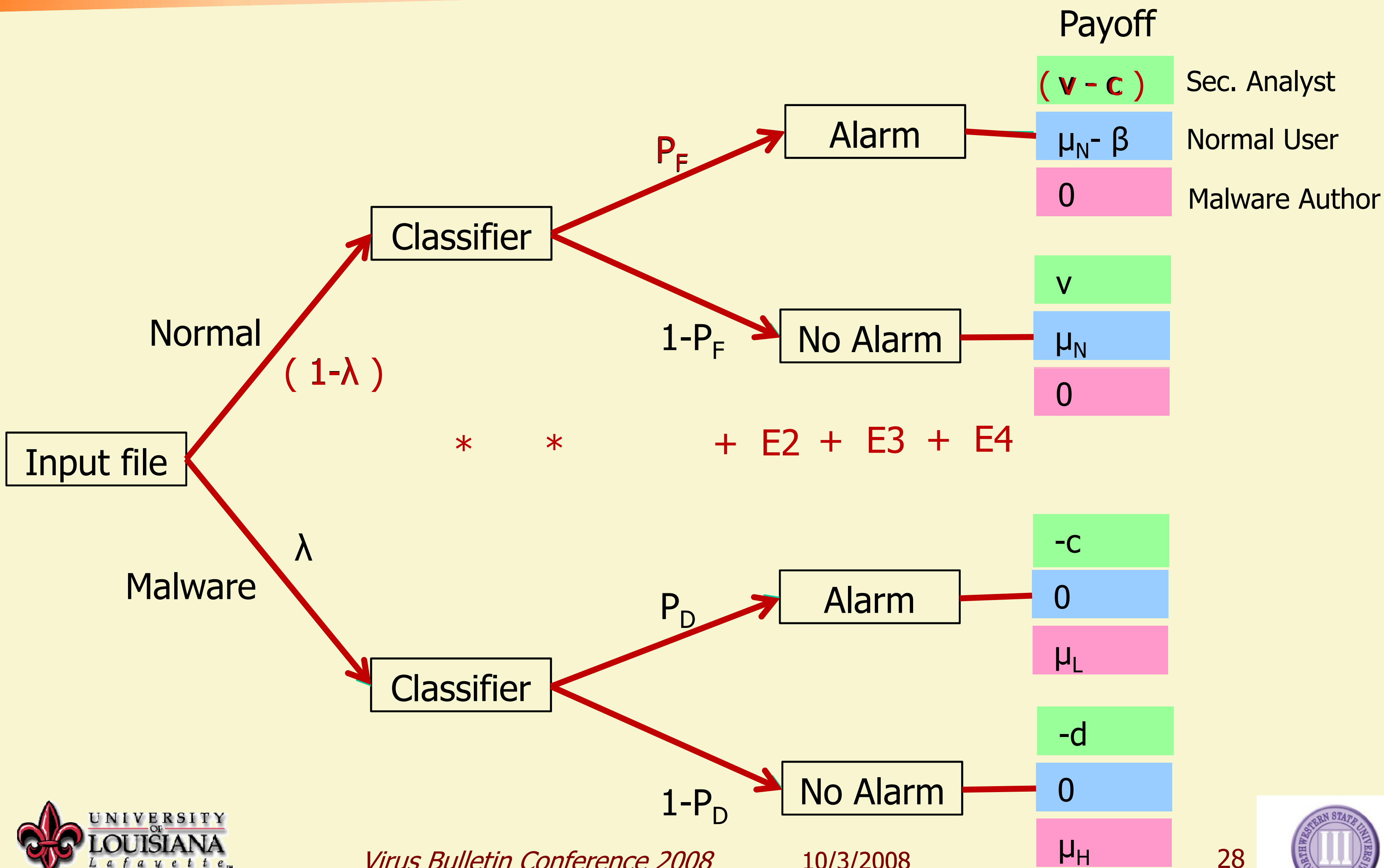
Outcomes		Benefits SA	Benefits NU	Benefits MA	Costs SA	Costs NU	Costs MA
Normal	Alarm	v	μ_N		c	β	
	No Alarm	v	μ_N				
Malware	Alarm			μ_L	c		
	No Alarm			μ_H	d		

MA damages SA if malware gets through

Step 3: Construct Game Tree



Step 4: Expected Payoff Analysis



Step 4: Expected Payoff Analysis

- The expected payoffs for all agents:

  NU: $(\mu_N - \beta)p_F + \mu_N(1-p_F) = \mu_N - \beta p_F$

  MA: $\mu_H(1-p_D) + \mu_L p_D$

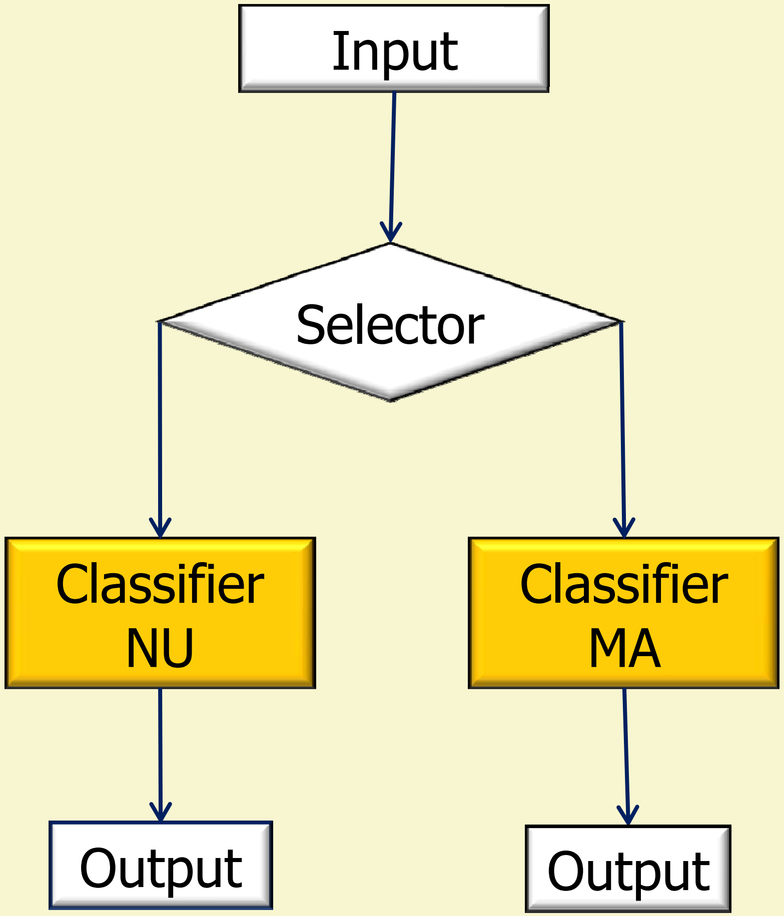
  SA: $v - (d+v)\lambda - c(1-\lambda)p_F + (d-c)\lambda p_D$

Step 4: Strategy Analysis

- Optimal solution for SA:

$$p_D = \left(\frac{\xi}{r}\right)^{\frac{r}{r-1}} \text{ where } \xi = \frac{c}{d-c} \times \frac{1-\lambda}{\lambda}$$

Selector Game



- SA
  - Sets up system
  - Tunes it
- MA
  - Chooses anti-AV technique
  - Sends cloaked file
- Question:
  - Optimal parameters?

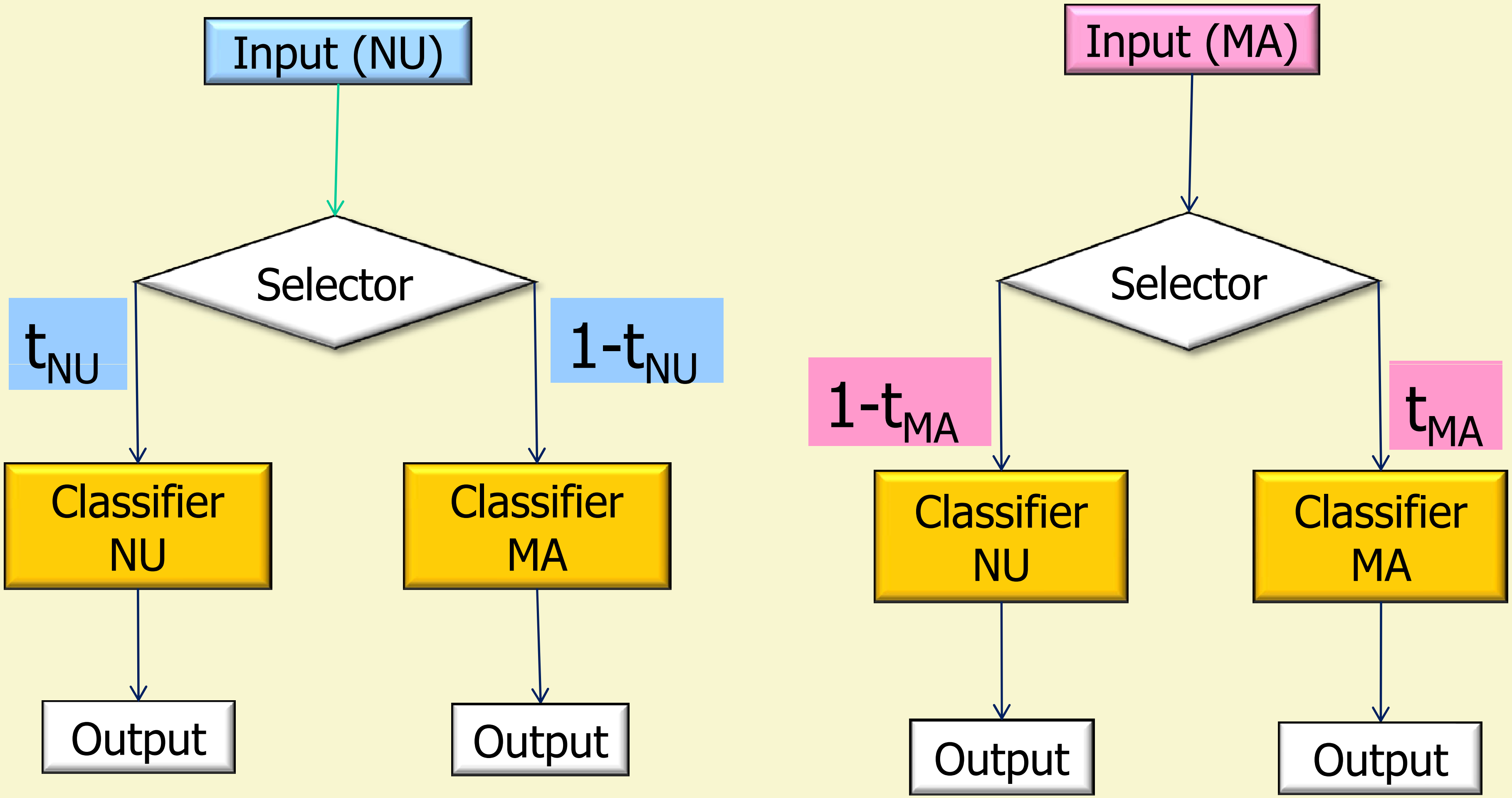
Game Theory Steps - Revisited

- Four steps
  1. Identify agents
  2. Identify game parameters
  3. Develop game tree
  4. Analyze tree to compute expected payoffs

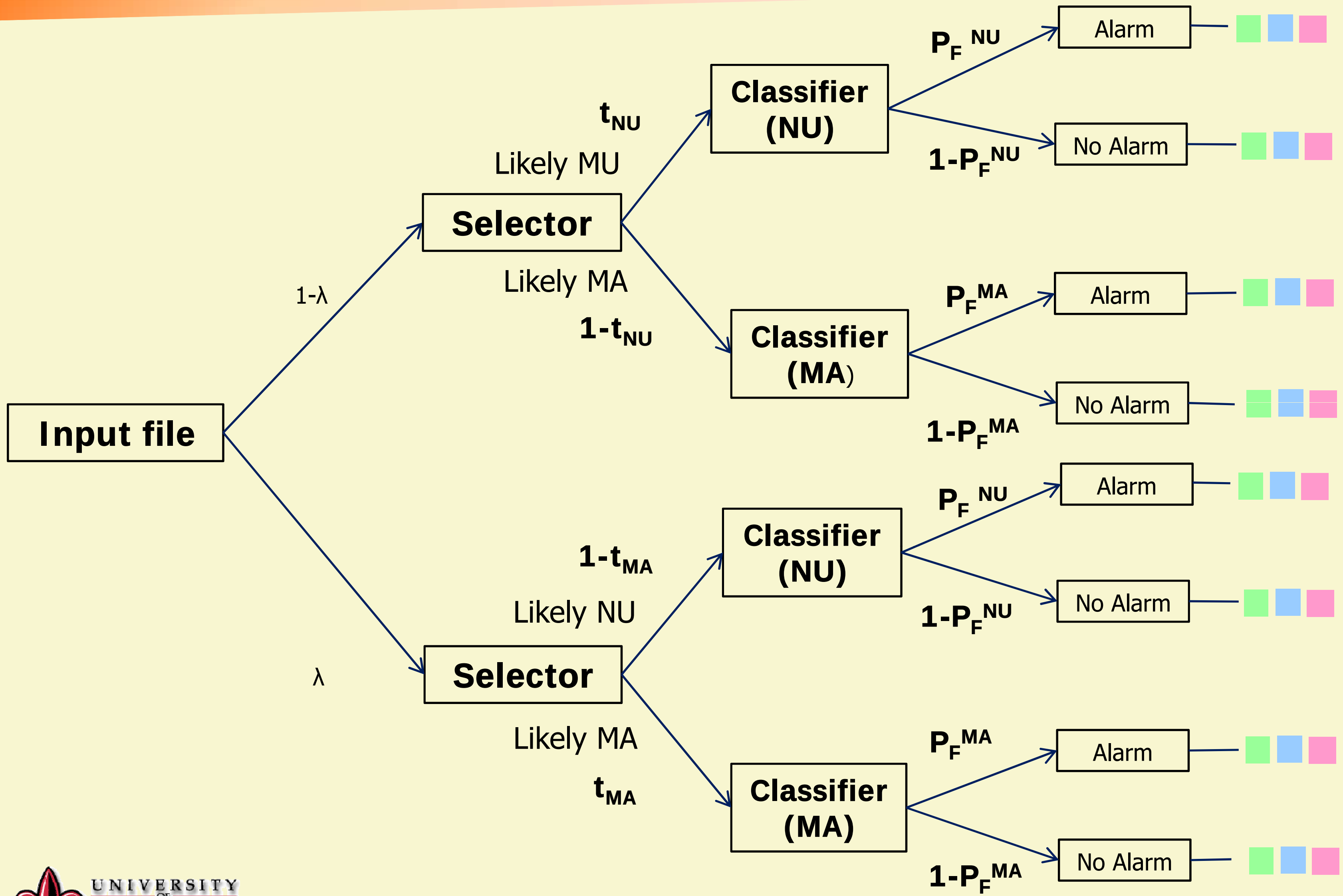
Step 2: Outcomes and Payoffs (Δ)

Outcomes		Benefits			A	Costs		
		SA	NU	MA		SA	NU	MA
Normal	Alarm	v	μ_N			c	β	
	No Alarm	v	μ_N					
Malware (normal)	Alarm			μ_L		c		
	No Alarm			μ_H		d		
Malware (stealth)	Alarm			μ_L		c		Δ
	No Alarm			μ_H		d		Δ

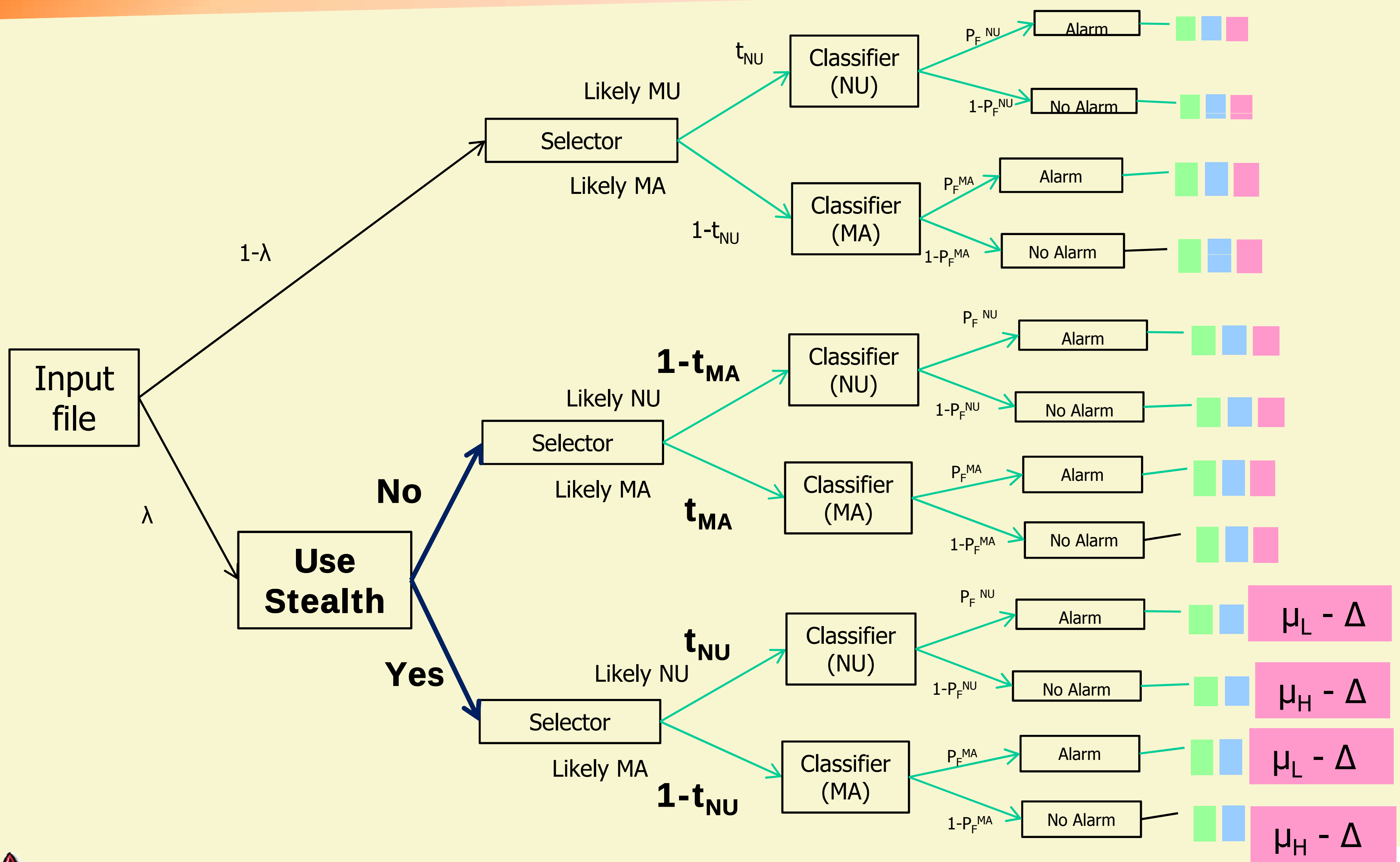
Step 2: Parameters (t_{NU} and t_{MA})



Step 3: Game Tree (no Stealth)



Step 3: Game Tree (w/ Stealth)



Step 4: Strategy Analysis

- Game tree analysis yields expected payoffs

NU: $t_{NU}((\mu_N-\beta)p_F^{NU}+\mu_N(1-p_F^{NU}))+(1-t_{NU})(p_F^{MA}(\mu_N-\beta)+(1-p_F^{MA})\mu_N)$

MA if not using anti-AV:

$(1-t_{MA})(\mu_l p_D^{NU}+\mu_h(1-p_D^{NU}))+t_{MA}(\mu_l p_D^{MA}+\mu_h(1-p_D^{MA}))$

MA if using anti-AV:

$(1-t_{NU})((\mu_l-\Delta)p_D^{MA}+(\mu_h-\Delta)(1-p_D^{MA}))+t_{NU}((\mu_l-\Delta)p_D^{NU}+(\mu_h-\Delta)(1-p_D^{NU}))$

SA: *p(selected as normal, alarm) * [p(normal file | selected as normal, alarm)*v – c] + p(selected as normal, no alarm) * [p(normal file | selected as normal, no alarm)*v – p(malware | selected as normal, no alarm)*d] + p(selected as malware, alarm) * [p(normal file | selected as malware, alarm)*v – c] + p(selected as malware, no alarm) * [p(normal file | selected as malware, no alarm)*v – p(malware | selected as malware, no alarm)*d]*[1]

Step 4: Optimal Solutions

- Optimal solution for SA:

  If $\Delta \geq$ Threshold(r) then $p_D^{MA} = f(r)$ and $p_D^{NU} = g(r)$

  If $\Delta <$ Threshold(r) then $p_D^{MA} - p_D^{NU} = \dfrac{\Delta}{(\mu_H - \mu_L)(t_{MA} + t_{NU} - 1)}$

  and $(1 - t_{NU})(p_D^{MA})^{\frac{1-r}{r}} + t_{NU}(p_D^{NU})^{\frac{1-r}{r}} = \dfrac{r}{\xi}$

Where

$$Threshold(r) = (t_{MA} + t_{NU} - 1)(f(r) - g(r))(\mu_H - \mu_L)$$

$$f(r) = \left(\frac{\xi}{r} \times \frac{1 - t_{NU}}{t_{MA}}\right)^{\frac{r}{r-1}} \text{ and } g(r) = \left(\frac{\xi}{r} \times \frac{t_{MA}}{1 - t_{MA}}\right)^{\frac{r}{r-1}}$$

Step 4: Insights

- For a particular selector/classifier architecture it can be shown that:
  - Adding another classifier to a scanner will not always make a scanner better
    - better *only if* the cost of "stealthing" the selector is above a specific threshold
  - AV designer is always advised to deter "stealthing" the selector
    - by increasing the spread in the detection rates of the classifiers
- Have equations for setting parameters

Conclusions

- Introduced way to analyse AV systems using Game Theory
- Showed it may lead to interesting, possibly counter-intuitive results
- Mathematically derive optimal configurations

Thanks!