Mestrado em Engenharia Informática,
Universidade do Minho

# Falsification of Hybrid Programs

PROJETO EM MÉTODOS FORMAIS
DE PROGRAMAÇÃO

**Grupo:**
João Duarte (PG60110)
Luís Silva (PG60279)

# Tema

**O que são?**

Sistemas que combinam eventos discretos (saltos lógicos) com dinâmica contínua (evolução no tempo).

**O problema:**

A complexidade destes sistemas torna a verificação exaustiva impraticável em muitos cenários "black-box".

**A solução:**

Em vez de provar que o sistema está sempre certo, procuramos ativamente por um contra-exemplo que prove que ele falha.

# Sistema de Condução Autónoma: ACC

**Adaptive Cruise Control (ACC):**
Evolução onde o veículo ajusta a velocidade com base na distância ao carro líder.

```
p := 0 ; v := 0 ; pl := 50 ; vl := 10 ;
while tt {
  if safe(p,v,pl,vl)
  then p' = v, v' =  2, pl' = vl, vl' = 0 for 1
  else p' = v, v' = -2, pl' = vl, vl' = 0 for 1
}
```

Figura 1 - Exemplo de um programa de ACC

Neves, R., Proença, J., & Souza, J. (n.d.). An Adequate While-Language for Stochastic Hybrid Computation. Retrieved March 18, 2026, from https://lmf.di.uminho.pt/lbex/neves25b.pdf

# Propriedades de Segurança

**Definição:** Segurança (Safety) garante que "algo de mau nunca acontece".

**Lógica Temporal (LTL):** Usamos o operador G (Globalmente) para definir estados seguros.

**Exemplo no ACC:**

- G(pl > pf): O líder deve estar sempre à frente do seguidor.

# Método 1: Amostragem Aleatória (Random Sampling)

**Conceito:** Sequência temporal de perturbações externas que afetam o sistema, designadas **trajetórias de distúrbio**.

- A posição do líder e do seguidor pode variar, alterando a distância entre ambos;
- O líder pode assumir velocidades diferentes em cada unidade de tempo;
- Se o seguidor mantém apenas uma distância segura fixa para com o líder, uma travagem súbita deste pode levar à colisão.

**Papel no Projeto:** Servir como baseline (ponto de comparação).

**Limitação:** É ineficiente para encontrar falhas "raras" em sistemas muito robustos.

# Método 2: Funções de Custo e Minimização

**A Abordagem Quantitativa:** Medir "o quanto" o sistema satisfaz a especificação.

**Métrica de Miss Distance:**
c(x) = min t (pl(t) - pf(t))

**Minimização (Hill Climbing):** O algoritmo atua como um "adversário", tentando minimizar o custo até que c(x) <= 0 (falsificação).

# Roadmap do Projeto

| 1º Mês | 2º Mês | 3º Mês |
| --- | --- | --- |
| Estudo aprofundado de métodos de falsificação. | Implementação dos algoritmos no motor de simulação e aplicação direta ao caso do ACC. | Generalização para outros problemas e exploração de outras propriedades. |