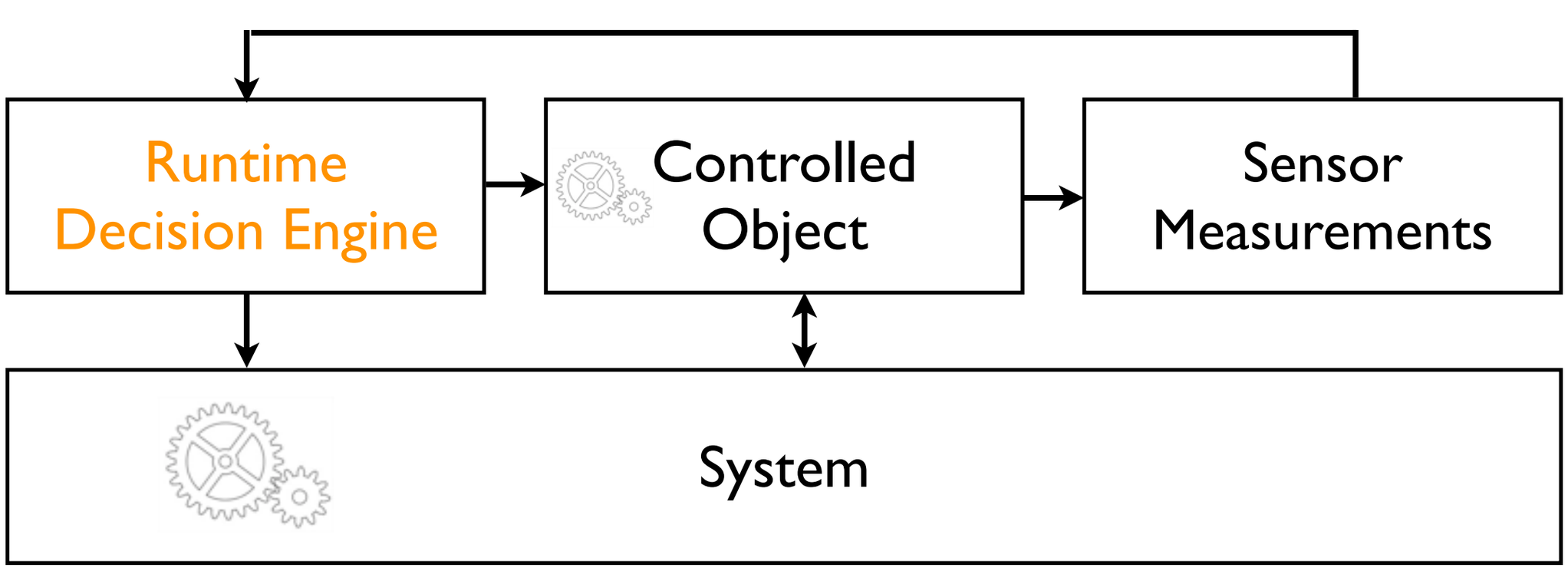
Decision Making Strategies

Comparison of Different Approaches in Autonomic Computing Systems

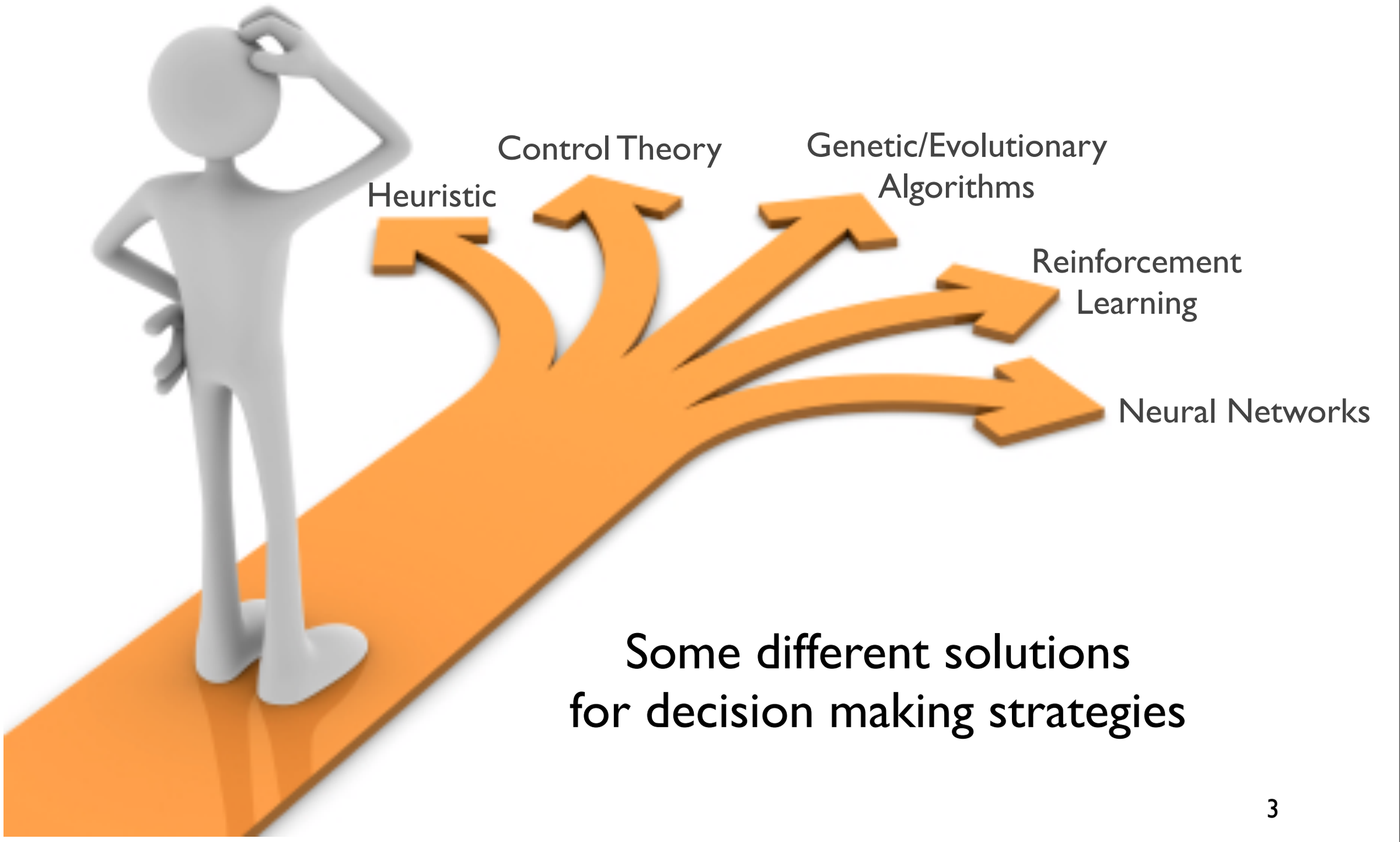
Martina Maggio (mmaggio@mit.edu)

Motivation

- Self-aware and adaptive systems.
- Decision engine to introduce awareness and feedback at runtime.



Decision making strategies



Some different solutions
for decision making strategies

Finding the best solution

- The best solution depends on the problem.
- But we still don't know how to choose.

We need a comparison of different strategies for class of problems, possibly without the need of implementing every strategy.

- Results comparison.

Most works in the literature apply only a single technique to solve a problem. Was that the best one?

Our work toward this goal

- Qualitative comparison.

  Comparing properties and guarantees.

- Quantitative comparison.

  Showing the performance of different techniques on a benchmark suite.

Outline

- Case study.
- Techniques overview.
- Experimental results.
- Conclusions.

The application is heartbeating

- The Heartbeat framework provides a way to instrument an application.
- Performance indicators needed.
- Target level specification (minimum, maximum).

Case study

- An instrumented application.
- Some resources to be allocated.
- Goal:

  Allocate the minimum amount of resources to meet the target performance level.

Decision making techniques

- Heuristics
- Control-theoretical

  Standard

  Advanced (RLS and Kalman)
- Machine-learning

  Neural network

  Reinforcement learning

Heuristics

- Heuristic solutions
  - \+ Low overhead
  - – No guarantees

Standard control-theoretical

- Standard deadbeat control
  - \+ Performance guarantees
  - \+ Low overhead
  - – Analytical model required
  - – Potentially not capable of handling non-nominalities

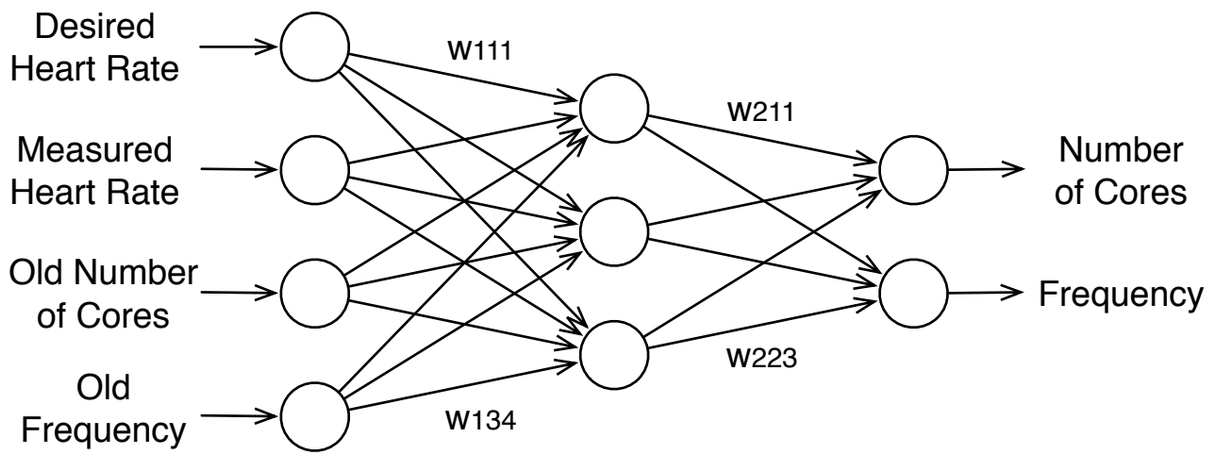
Advanced control-theoretical

- Identification augmentation

  + Performance guarantees
  + Capable of handling non-nominalities
  – Analytical model required
  – High overhead

Neural network

- Learning the best action
  - \+ Capable of reacting to changes in working conditions
  - \+ Very high level model required
  - – No guarantees
  - – Extremely high overhead



Reinforcement learning

- State action reward state action
  - \+ Capable of reacting to changes in working conditions
  - \+ No model needed
  - – No guarantees

Summary

Techniques overview and qualitative comparison.

	Performance Guarantees	Low Overhead	Model not Needed	Reaction to Unseen
Heuristic		✓	✓	
Standard Control	✓	✓		
Advanced Control	✓			✓
Neural Network			✓	✓
Reinforcement Learning			✓	✓

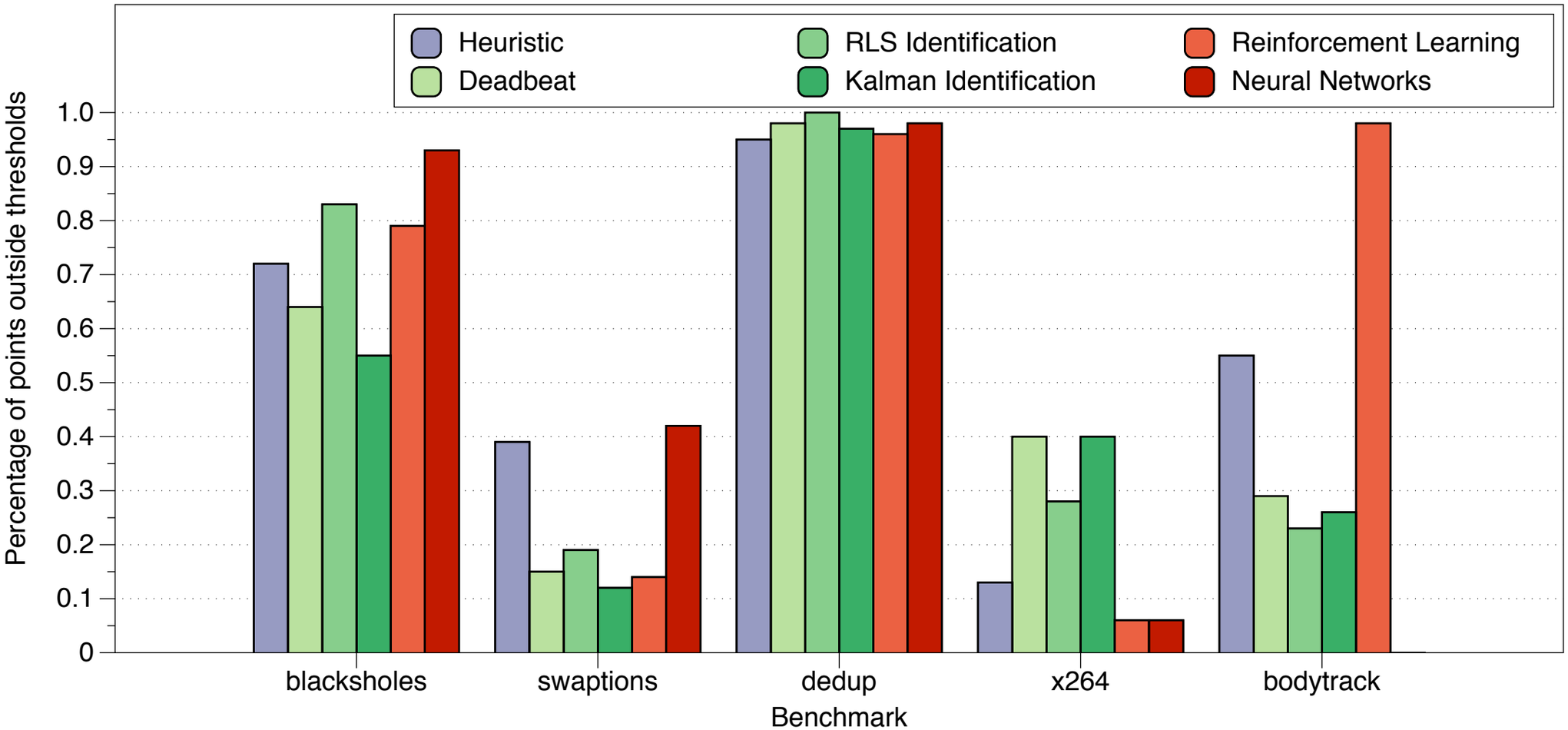
Experimental result

- Different techniques
- 5 Applications chosen within the PARSEC benchmarks
  blackscholes, bodytrack, dedup, swaptions, x264

Results post-processing:

Percentage of Wrong Data Points (WDP)
Integral of the Squared Error (ISE)
Integral of the Squared Error for Wrong Data Points (ISEWDP)
Overhead of decision mechanism

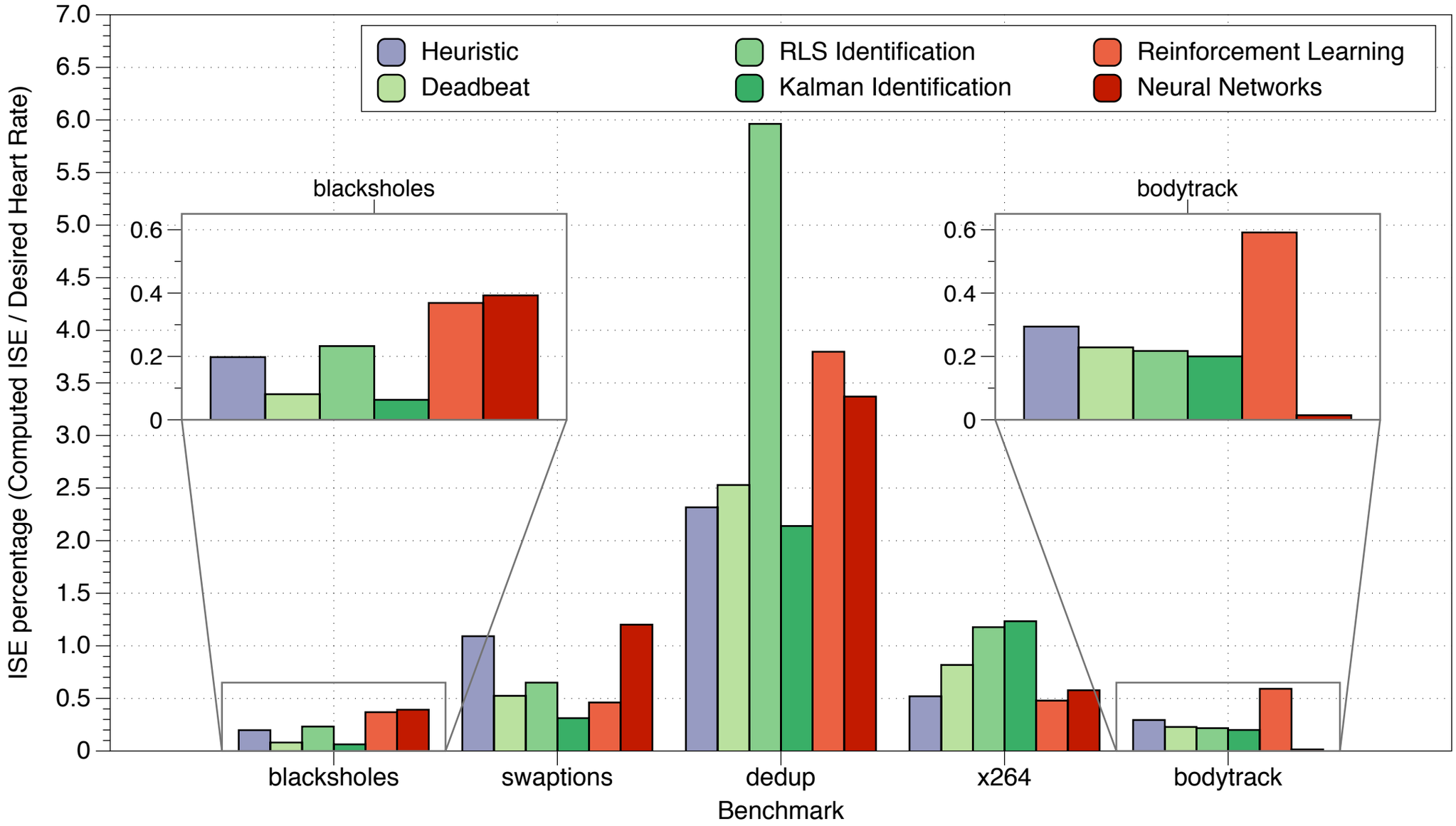
Aggregate results



Experimental results.

WDP with different decision mechanisms
(lower is better)

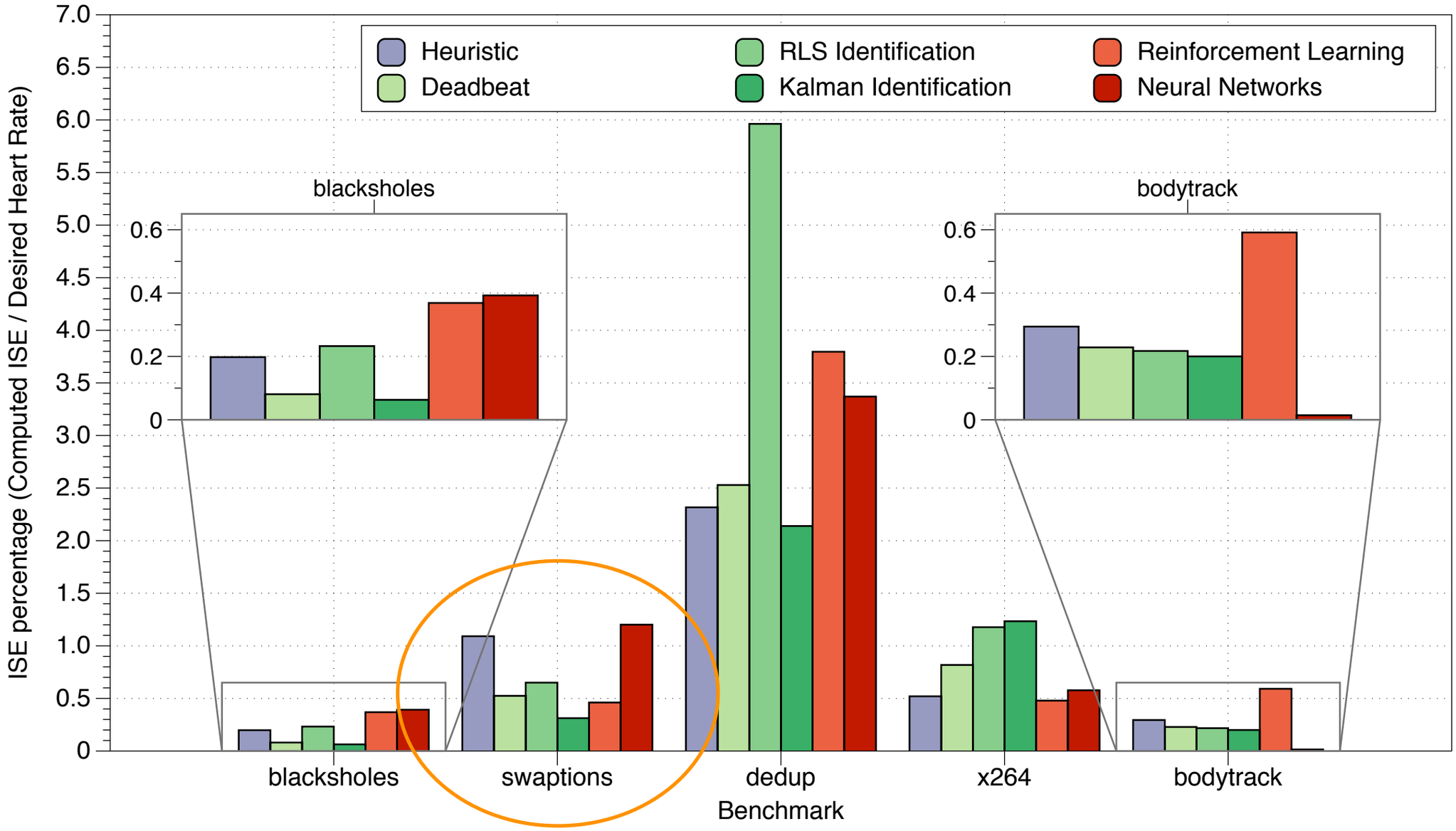
Aggregate results



Experimental results.

ISE with different decision mechanisms
(lower is better)

Aggregate results



Experimental results.

ISE with different decision mechanisms
(lower is better)

Swaptions

- 300 heart beats.
- Minimum threshold: 6 heart beats per second.
- Maximum threshold: 12 heart beats per second.
- Desired goal: 9 heart beats per second.

Experimental results.

Detailed results

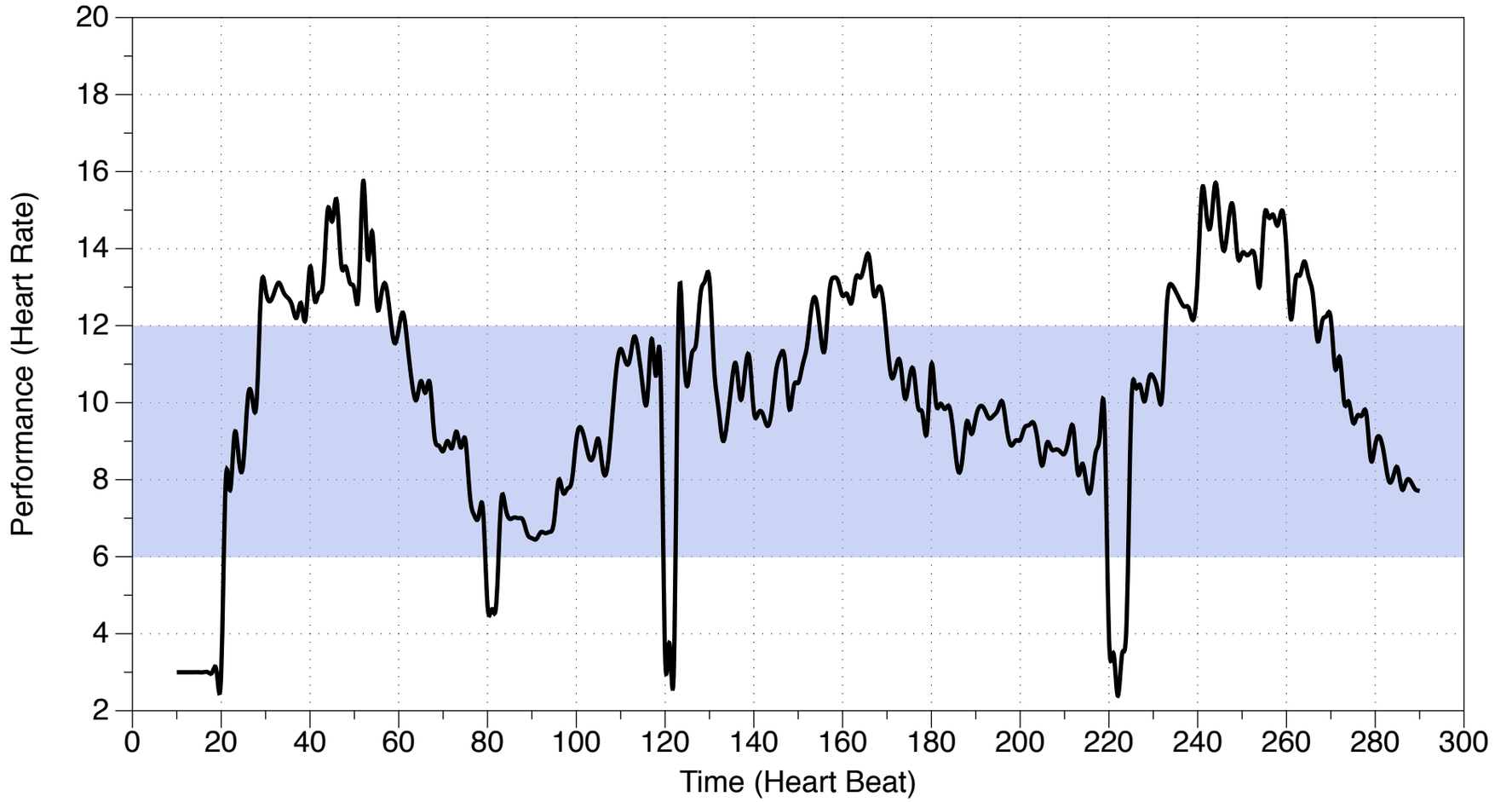
- Swaptions

	ISE	WDP	ISEWDP	Overhead [10^{-9} s]
Heuristic	9.82	0.39	4.40	3.60
Standard Control Deadbeat	4.72	0.15	1.40	20.09
Advanced Control RLS	5.85	0.19	1.38	37.10
Advanced Control Kalman	2.81	0.12	1.86	44.90
Reinforcement Learning	4.15	0.14	1.77	89.80
Neural Network	10.81	0.42	7.27	1410.00

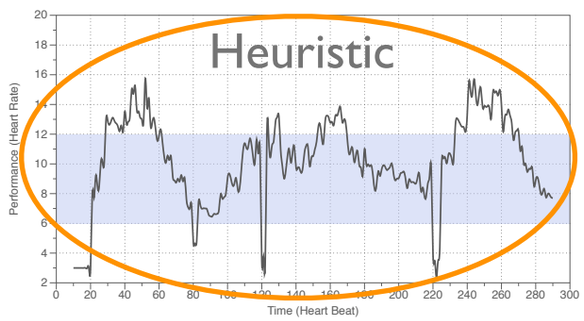
(lower is better)

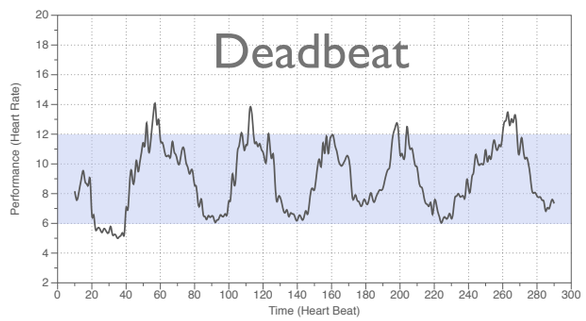
Experimental results.

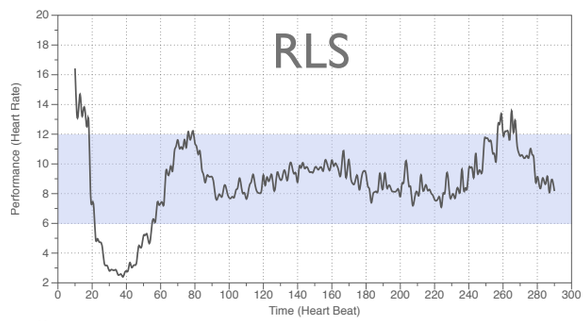
Swaptions - Heuristic

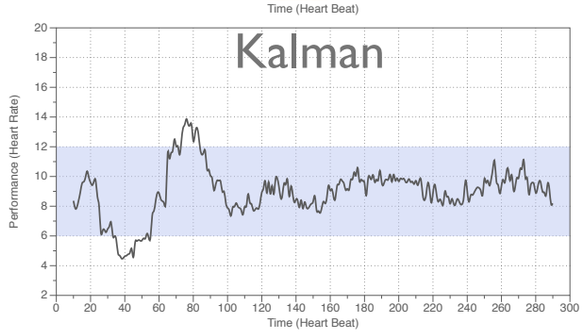


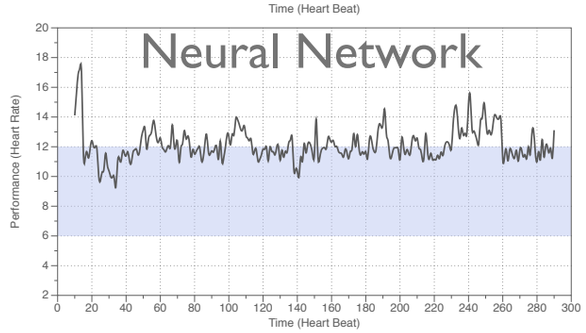
Experimental results.

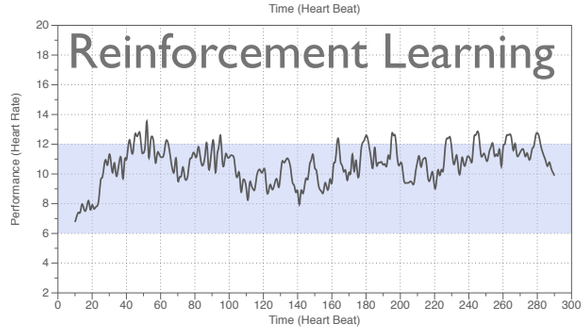




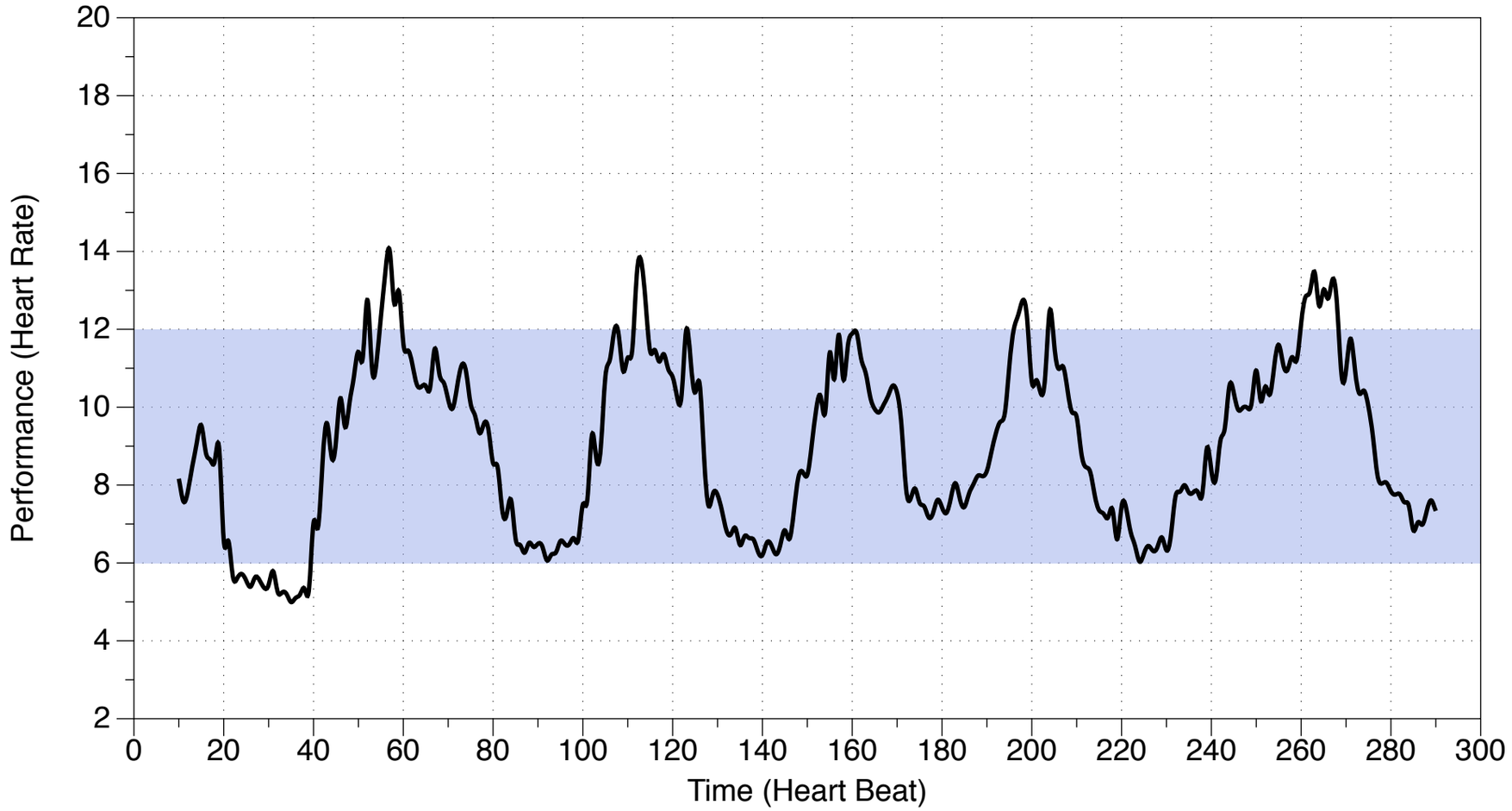




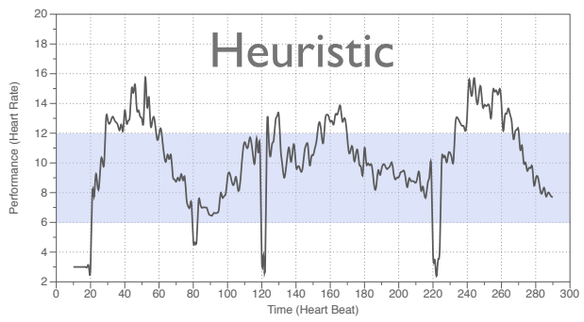


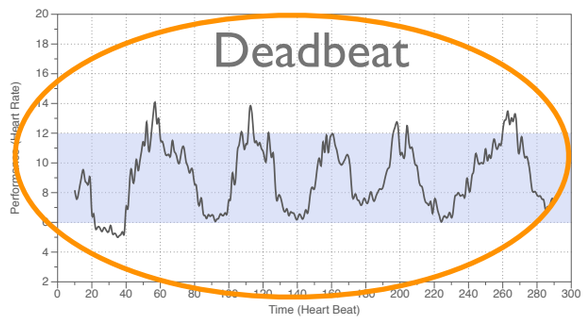


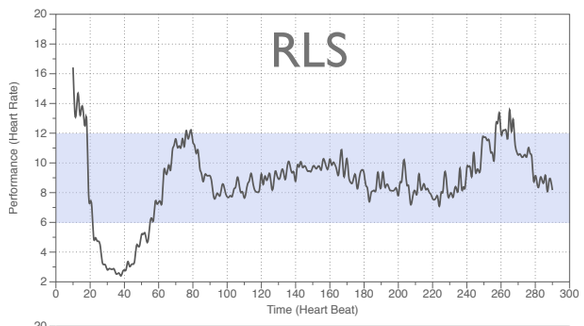
Swaptions - Deadbeat

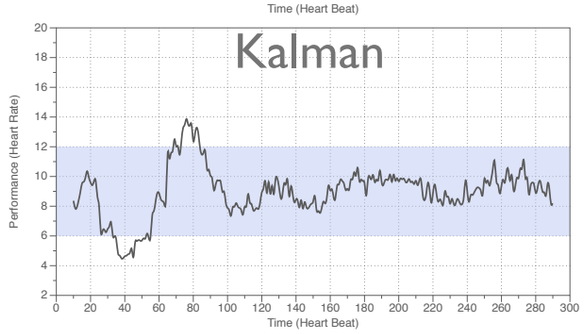


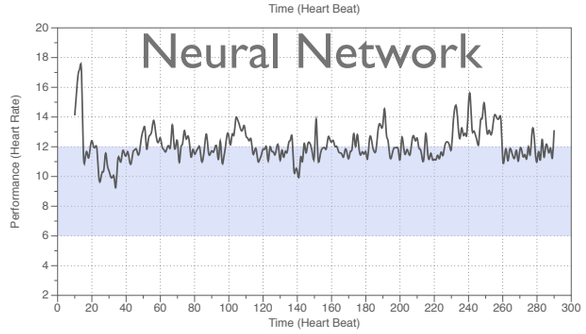
Experimental results.

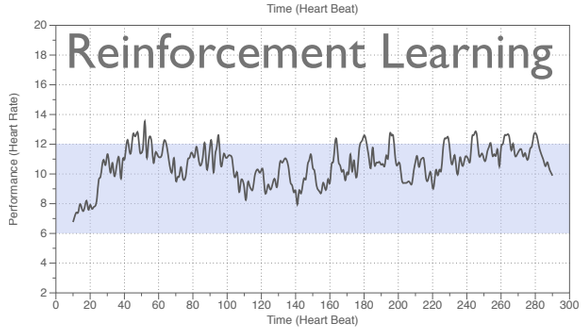




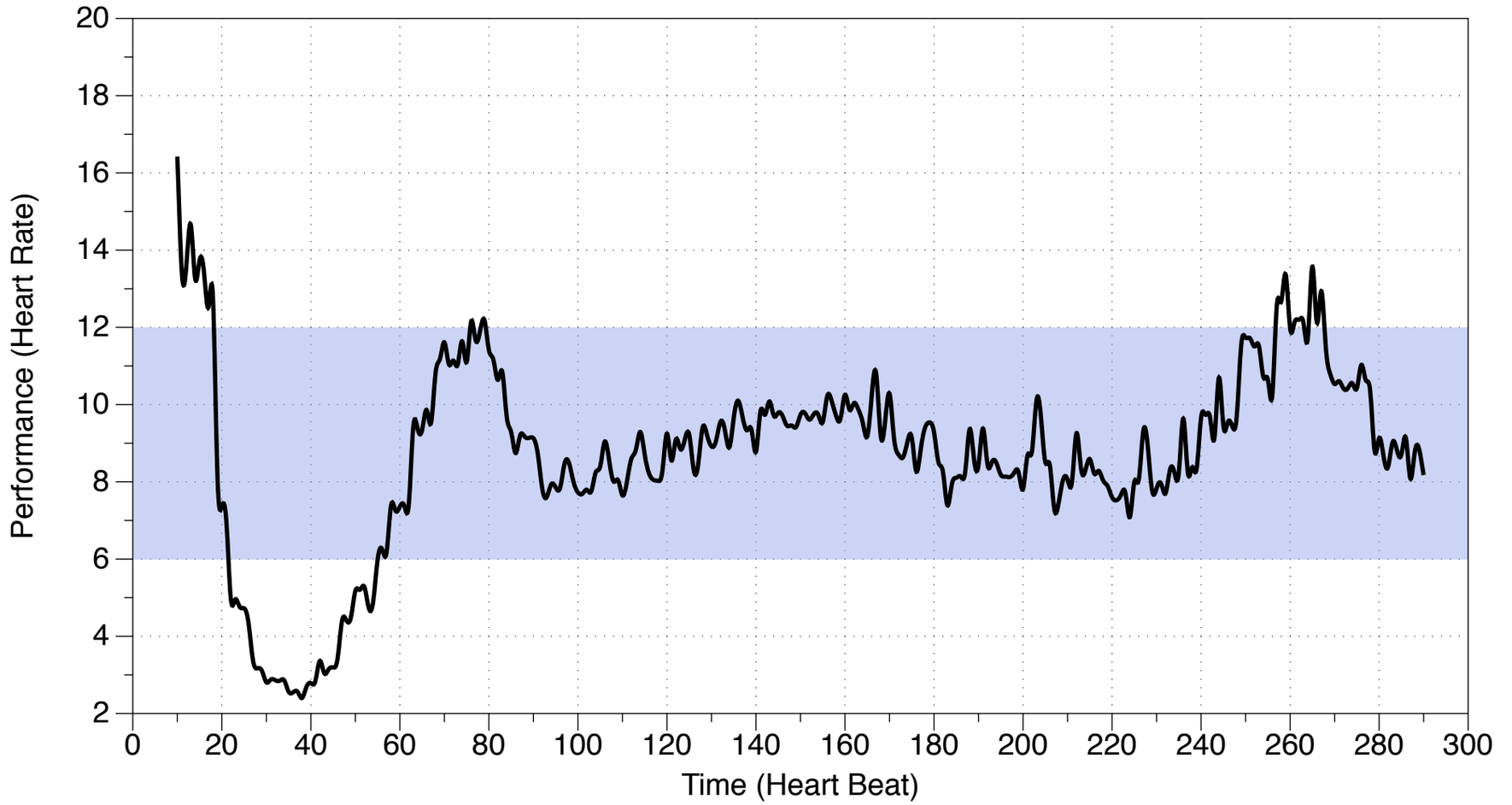




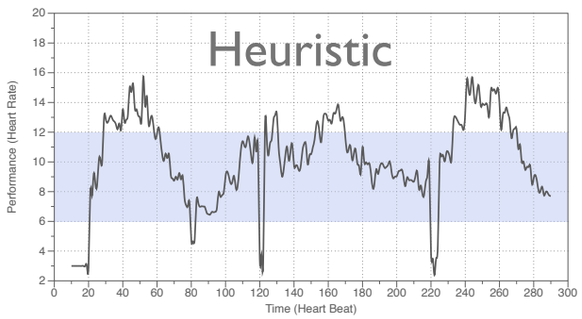


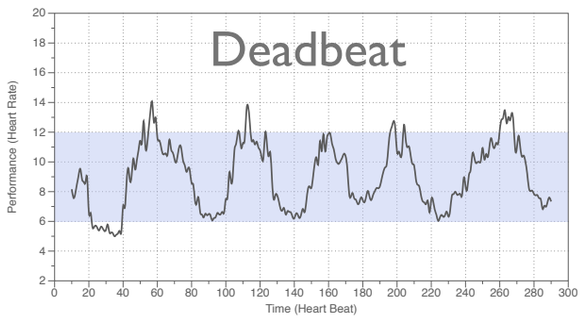


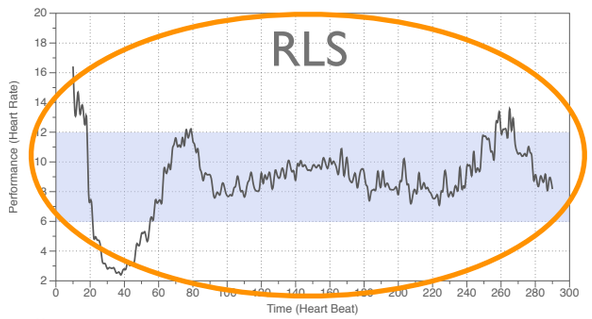
Swaptions - Deadbeat + RLS

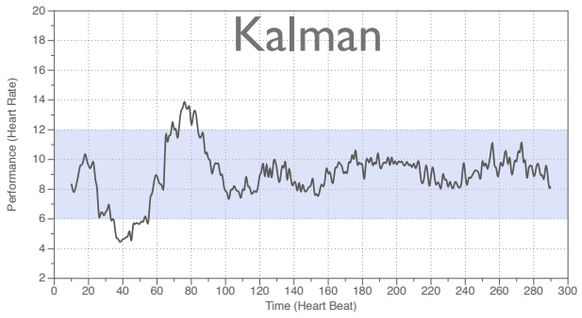


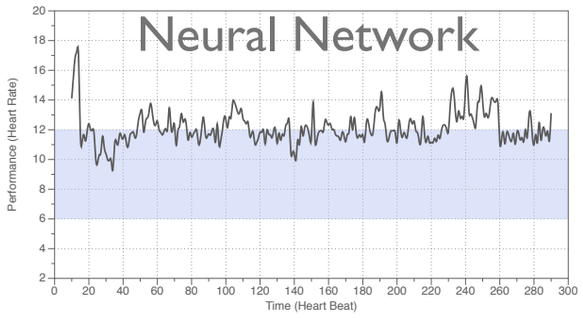
Experimental results.

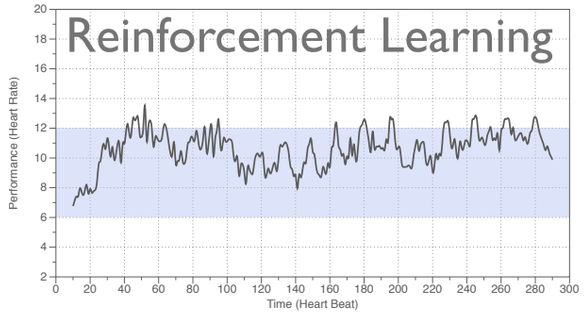




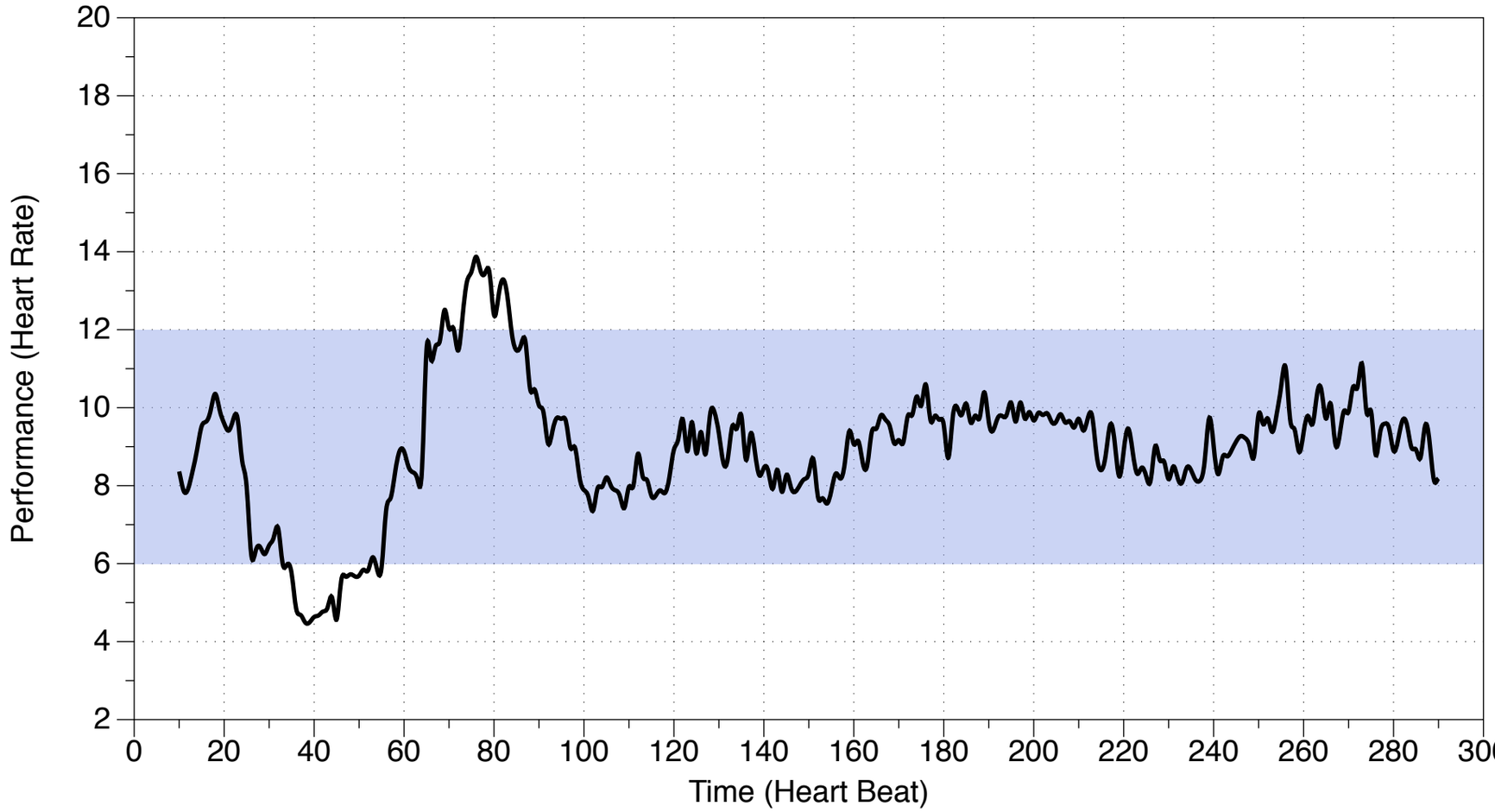


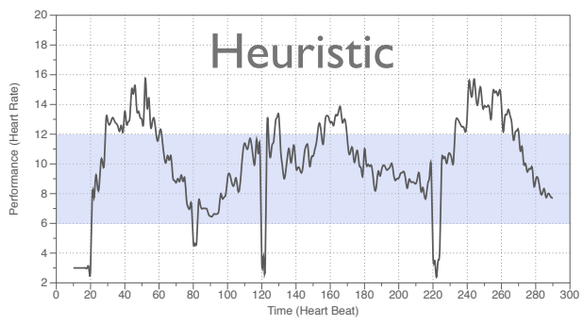


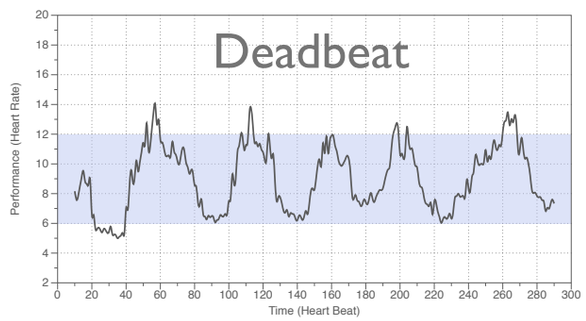


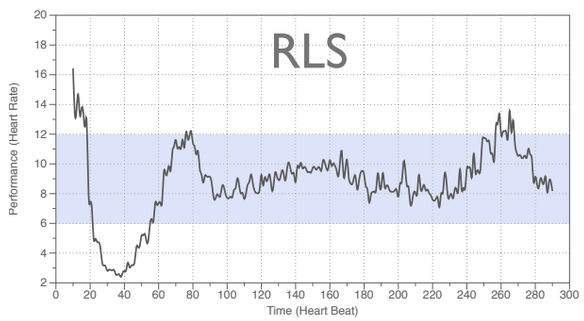


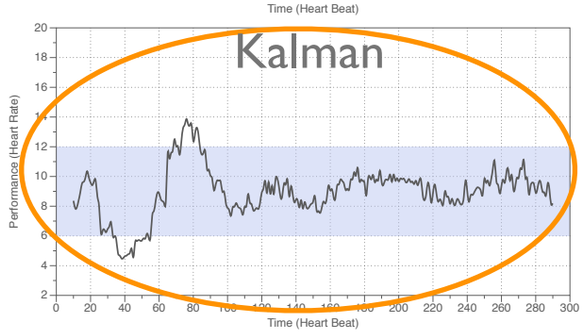
Swaptions - Deadbeat + Kalman filter

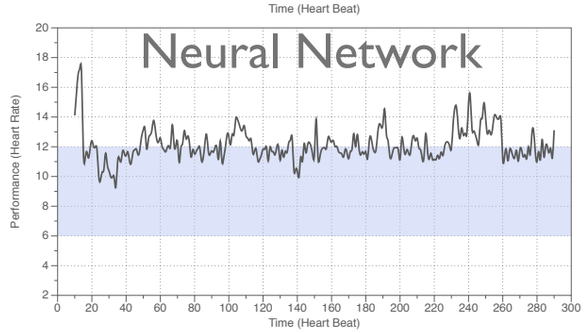


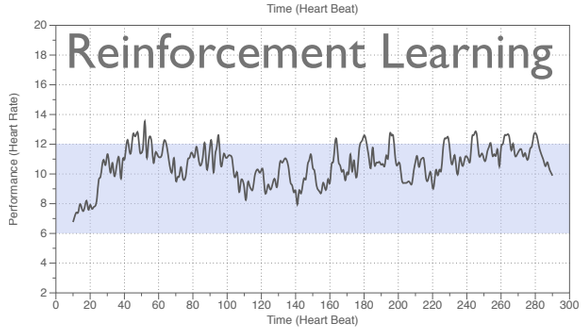






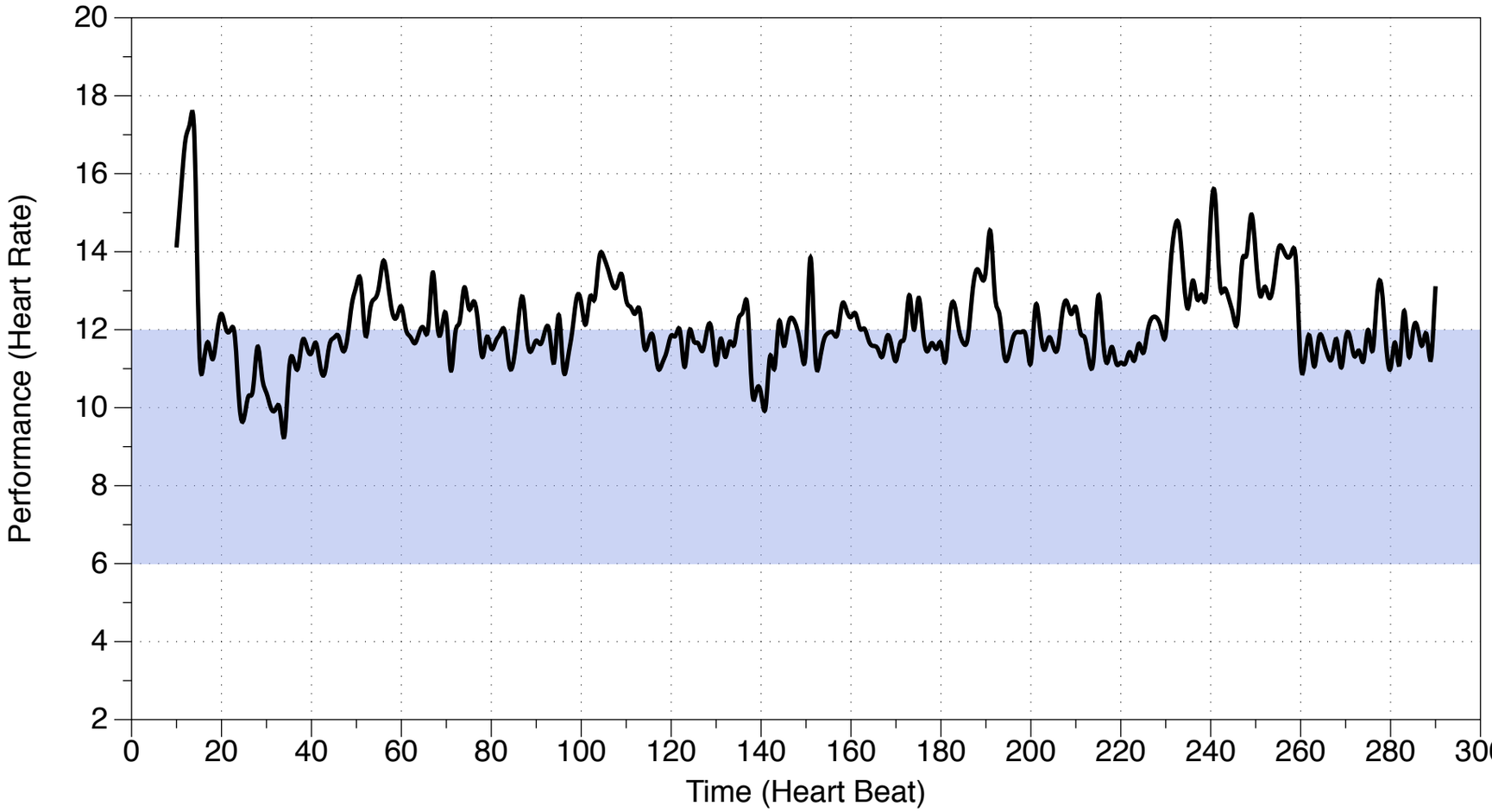


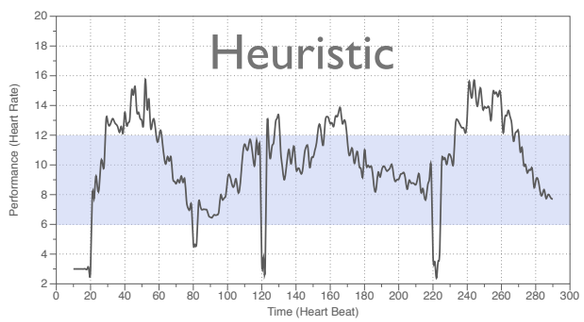


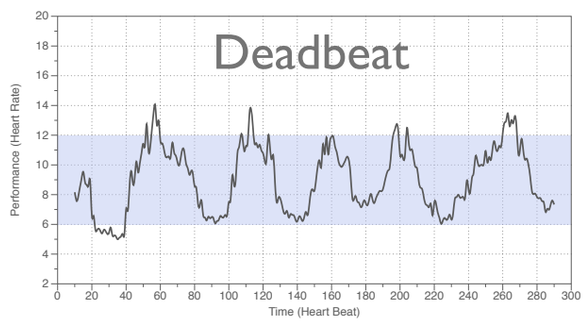


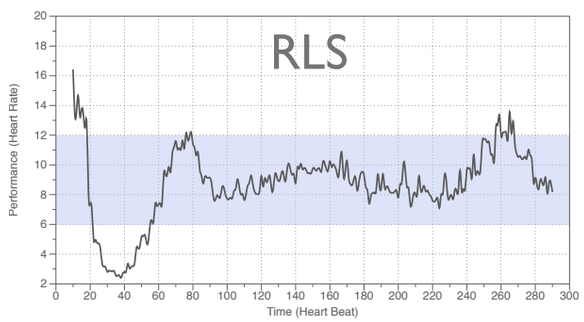
Experimental results.

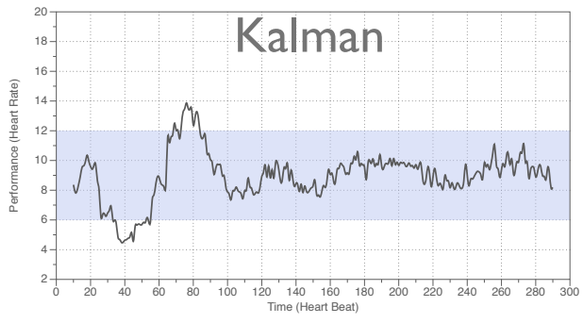
Swaptions - Neural network

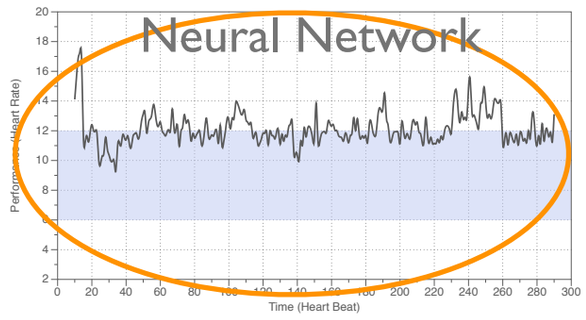


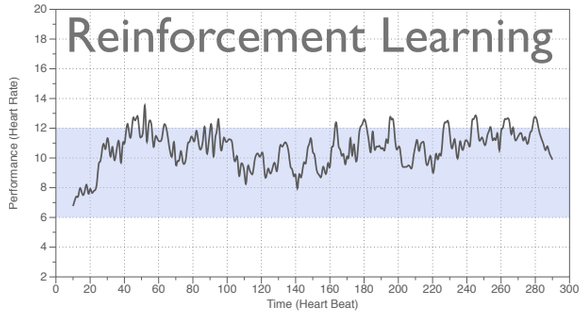






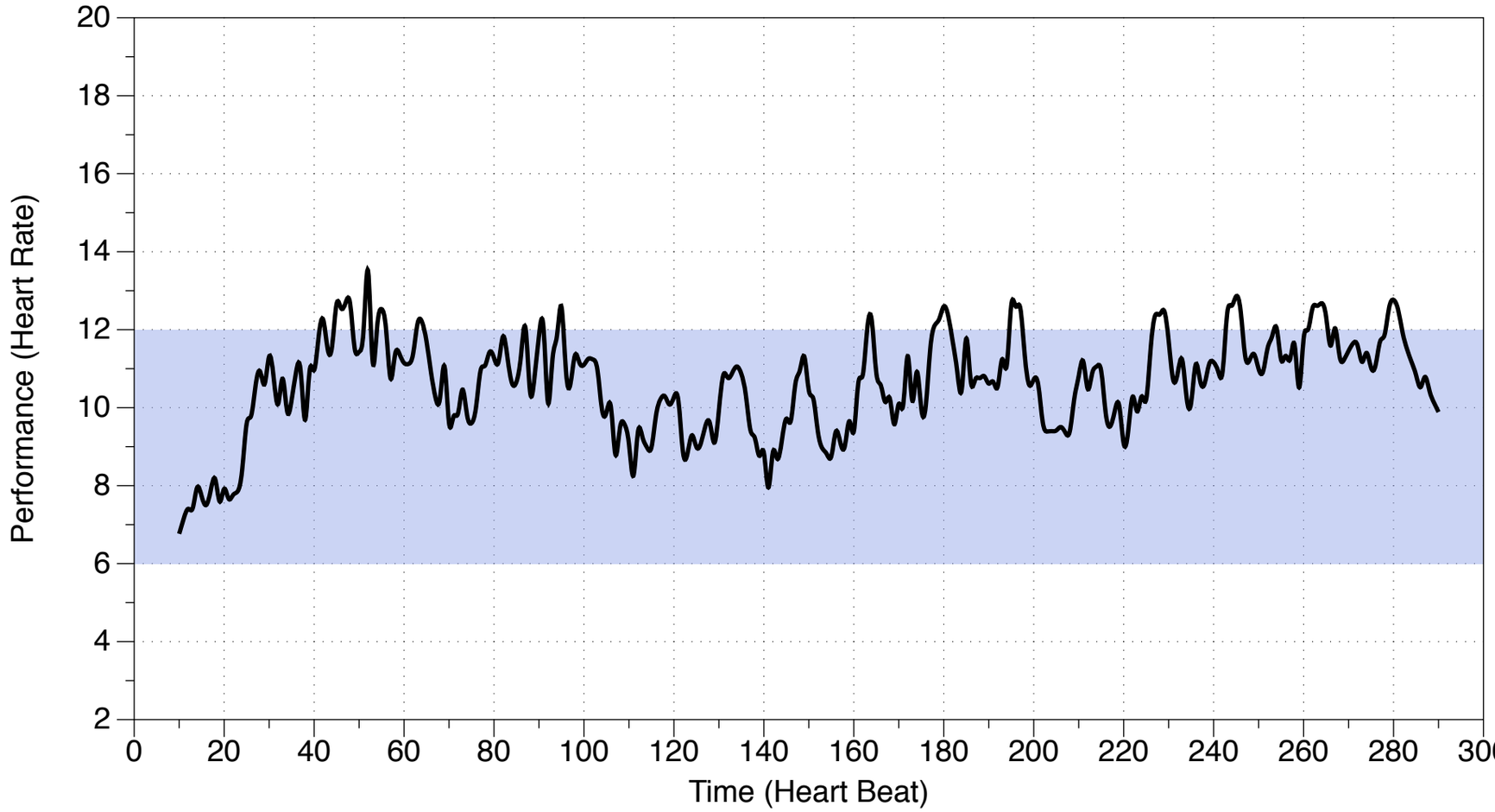


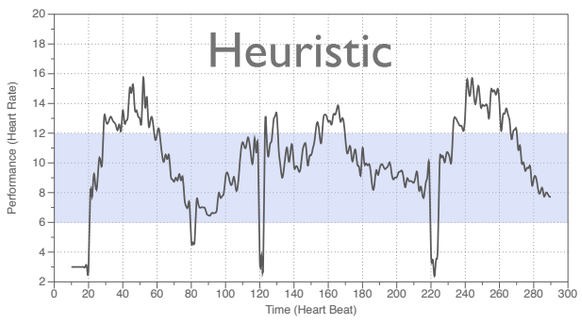


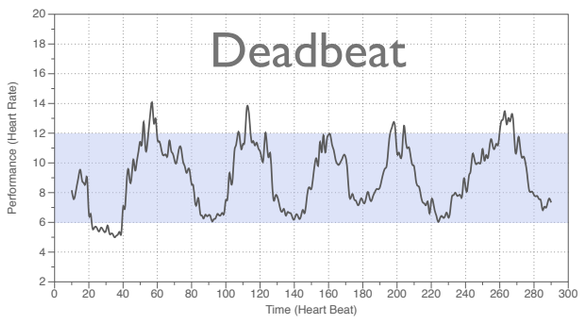


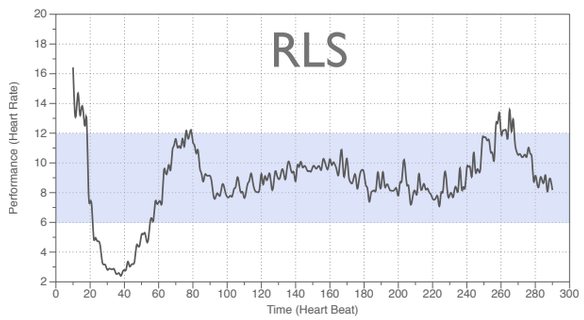
Experimental results.

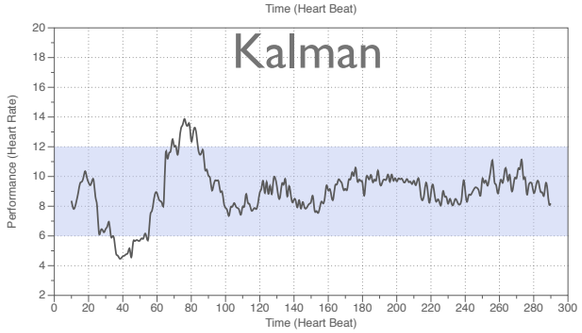
Swaptions - Reinforcement Learning

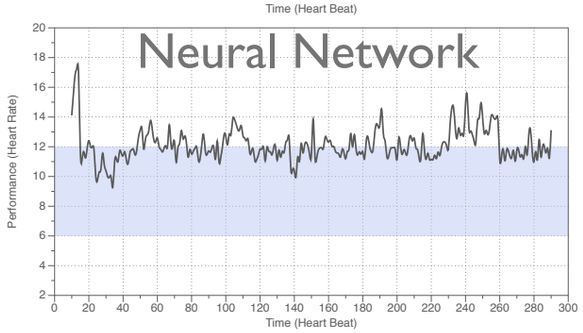


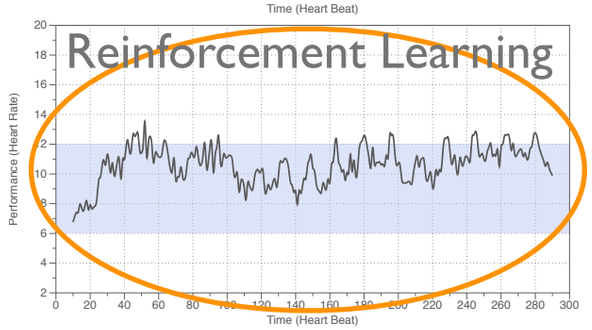












Experimental results.

Conclusions

- No universal best solution.

In general the control-theoretical guarantees are enforced.

- Need to explore different paths.

Questions?