A Bayesian Approach to Tackling Hard Computational Challenges

Eric Horvitz
Microsoft Research

Joint work with:

Y. Ruan, C. Gomes, H. Kautz, B. Selman, D. Chickering

MS Research   University of Washington   Cornell University

Looking Beyond Complexity Classes

NP-Hard doesn't mean intractable

CORE Project

Cornell MSR UW

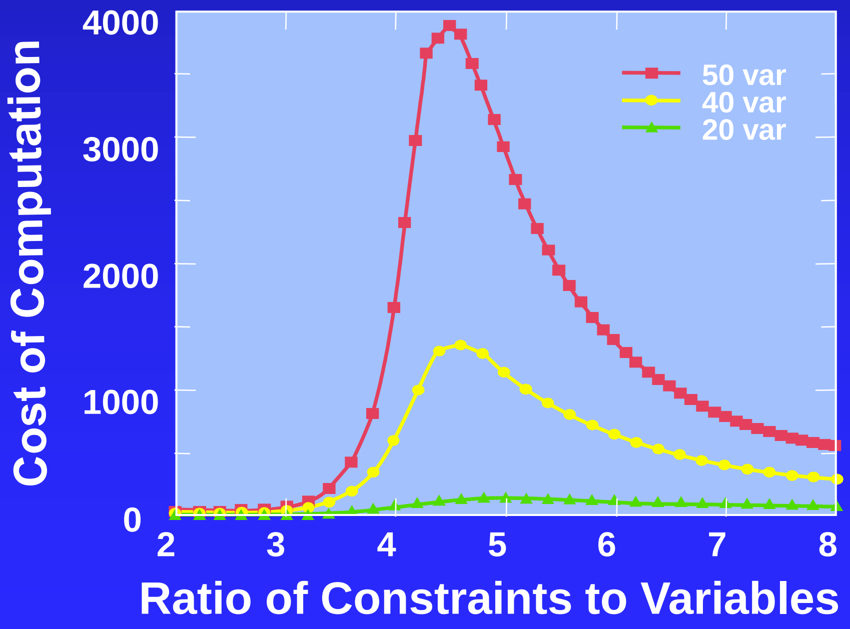
- In pursuit of principles, empirical methods for evaluating hardness of problem instances
- Learning models of execution time
- Policies for harnessing learned models
- Links to UAI work on flexible computation, decision-theoretic control

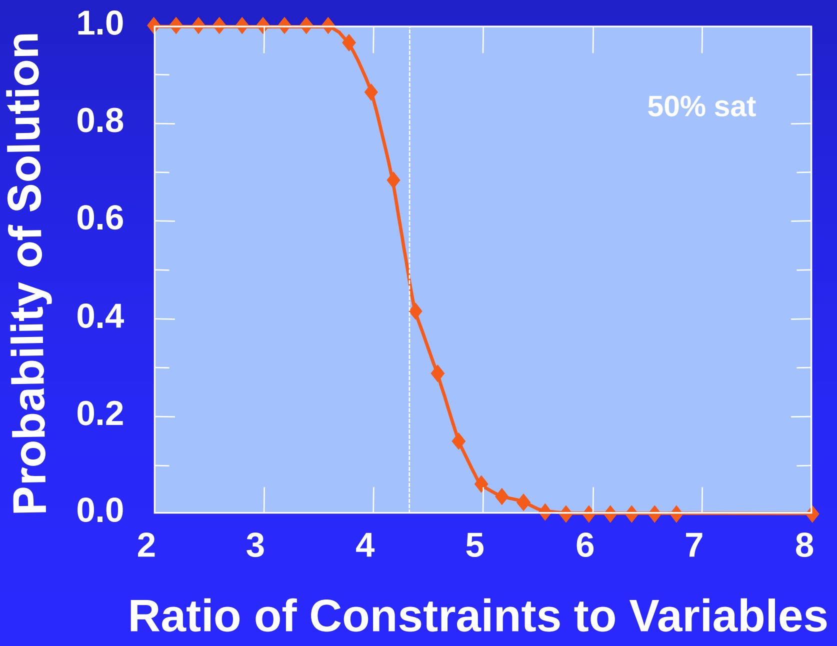
Focus

- NP-hard problems encoded as constraint satisfaction (CSP) and Boolean satisfiability (SAT)
- Randomized search: e.g., randomized Davis-Putnam Procedure
- Recent foci of attention
  - Phase transitions in difficulty
  - Heavy-tailed distributions on run time
  - Restart policies to avoid getting caught in tail

Phase Transitions in Computational Cost

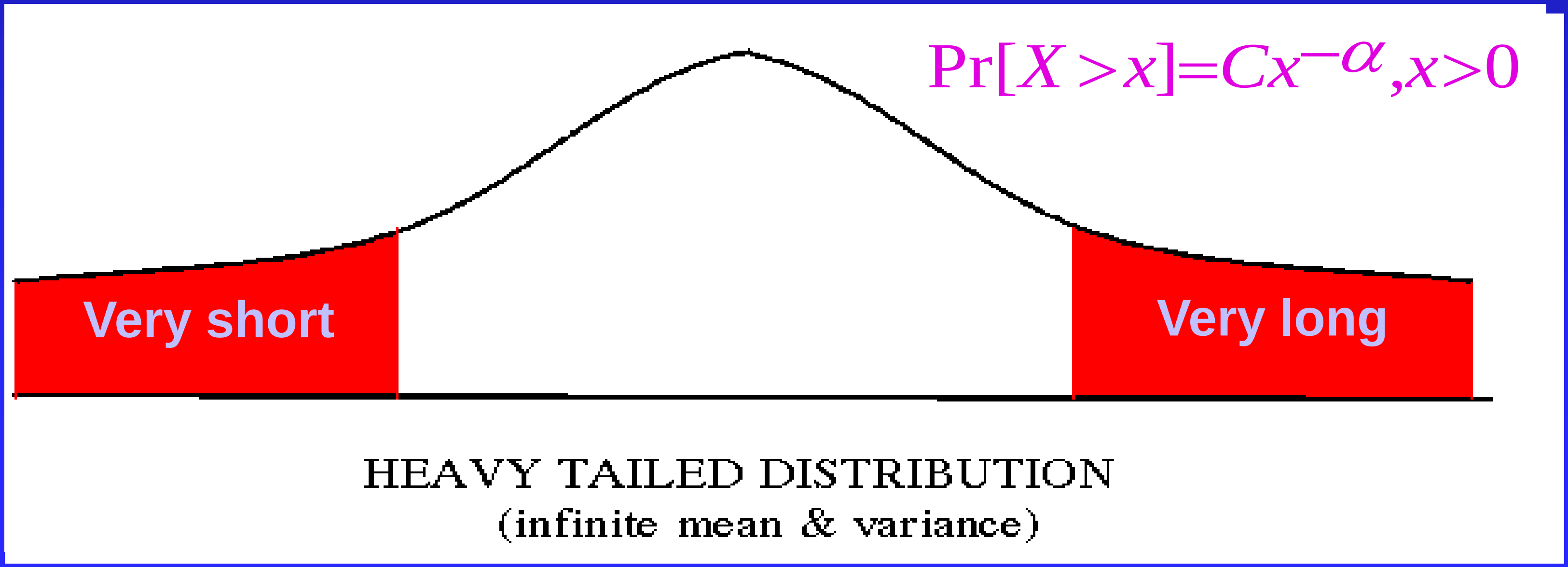
- Consider random 3SAT instances
- Critical parameter: ratio of the number of clauses to the number of variables.
- Hardest 3SAT problems at ratio = 4.25





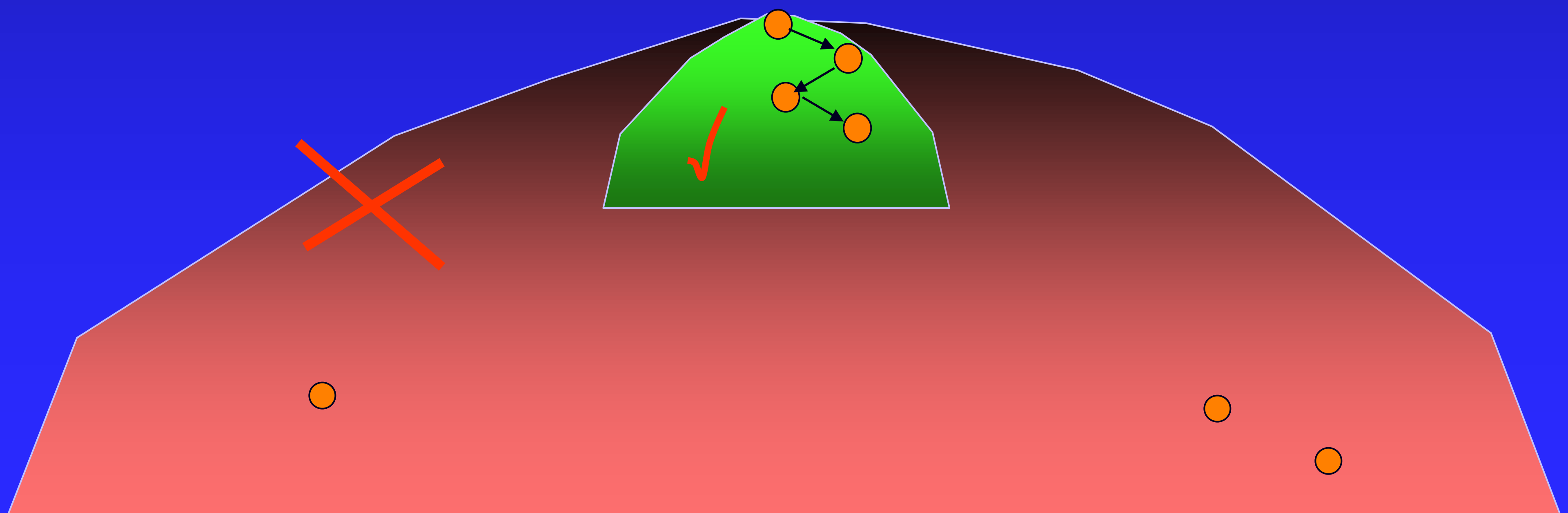
Great Variation in Execution Time

- Very short and very long runs for different randomized runs on *same* instances
- Highly sensitive to branching choices
- Heavy-tailed distribution (Pareto)



Intuition

- Search procedures that do not branch early on critical variables have *very long* run-times.
- Those that do, have short run-times
- Branch on right variables early



Application: Dynamic Restart Policies

- **To date: success with simple fixed policy:**
  Restart search if run-time is greater than x
- **Orders of magnitude speedup**

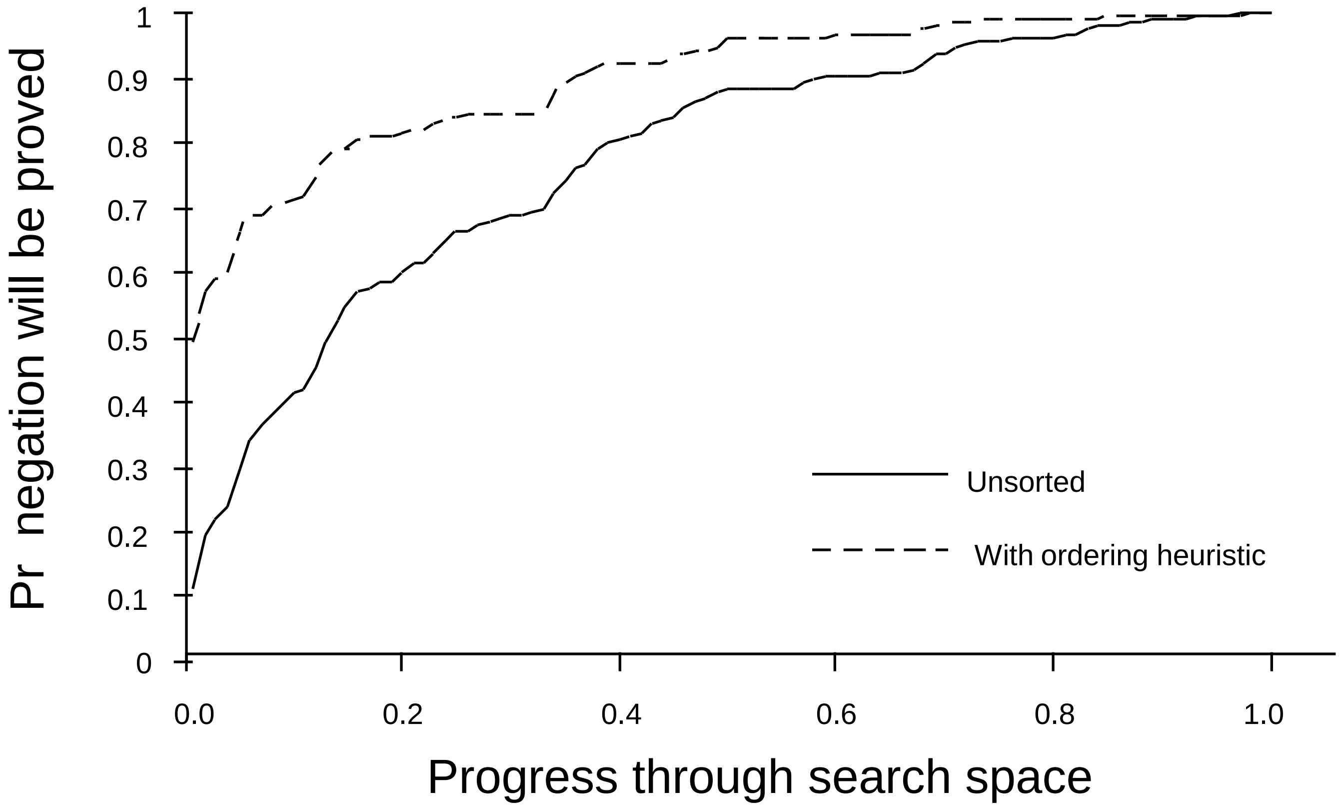
Gomes, et al. 1999

Beyond simple fixed policies: probabilistic analysis for dynamic restarts.

Opportunity: Learning Models that can Infer Beliefs about Run Time

- Identify discriminating features
- Learn predictive models
- Real-time inference about expected run-time
- Applications
  - Insights about hardness
  - Understanding of solver behavior
  - Refinement of procedures, heuristics
  - Dynamic restart policies

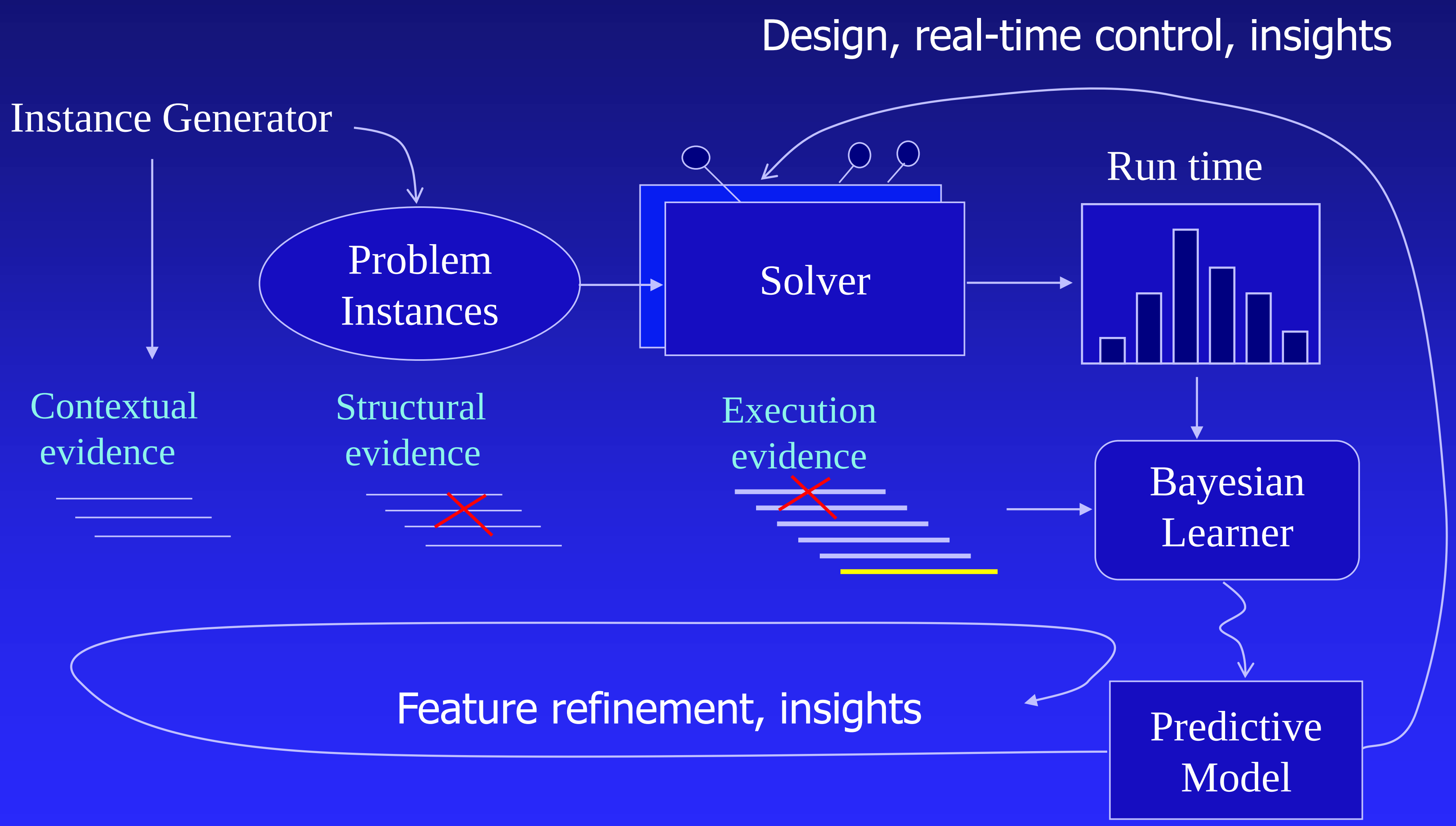
Some Prior Work at UAI



$$p(w \mid S, \xi) = \frac{p(S \mid w, \xi)\, p(w \mid \xi)}{p(S \mid w, \xi)\, p(w \mid \xi) + p(S \mid \neg w, \xi)\, p(\neg w \mid \xi)}$$

E. Horvitz & A. Klein, UAI 93

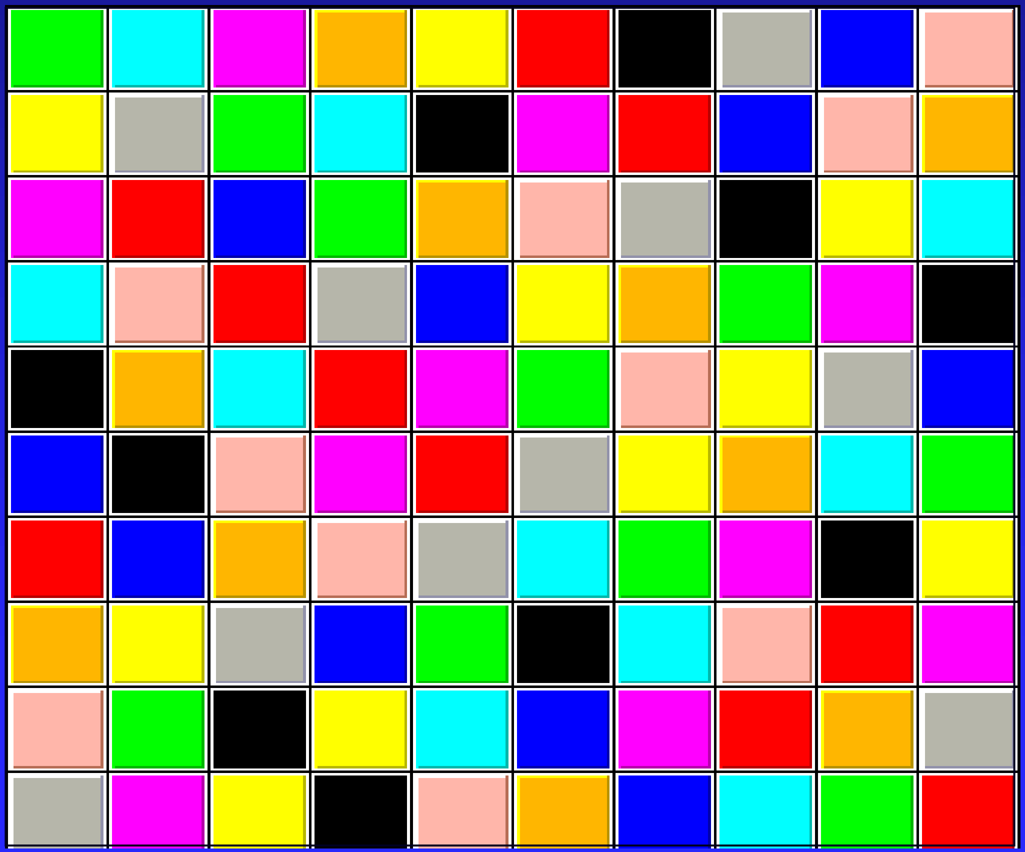
Big Picture



Design, real-time control, insights
Instance Generator
Problem Instances
Solver
Run time
Contextual evidence
Structural evidence
Execution evidence
Bayesian Learner
Feature refinement, insights
Predictive Model

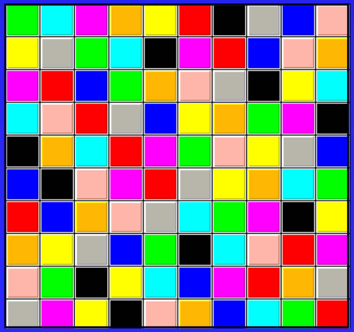
Experimental Domain

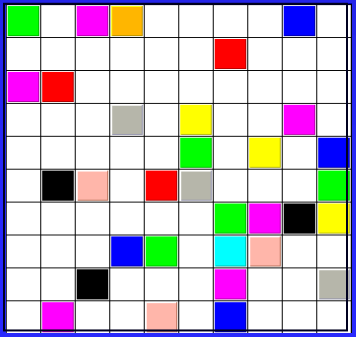
- **Quasigroup (Latin Square)**
  - n x n square filled with n colors such that there is no repeated color in any row or column



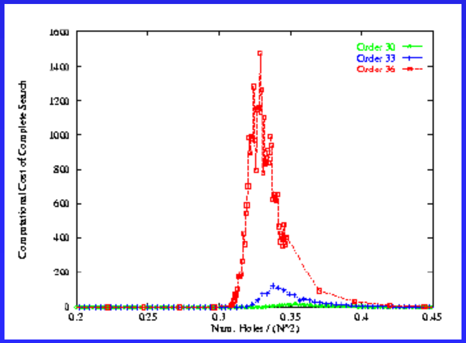
In Pursuit of Hard Instances

- **Quasigroup Completion Problem (QCP)**
  - Transform partial quasigroup (n x n Latin square) to quasigroup of same order (NP complete)
  - Random instances: peak in hardness,
  - f (# uncolored / # cells) ~ 0.4
- **Quasigroup with holes (QWH)** **Achlioptas, et al. AAAI 00.**
  - *Satisfiable instances only*
  - Balancing holes in rows, columns increases hardness
  - % holes, transition of backbone, region of hardness





32% holes



% holes

Exploration: Problem Solvers

- **Randomized SAT solver**
  - Satz-Rand (Gomes, et al.), a randomized version of Satz (Li & Anbulagan)
  - Davis-Putnam (DP) with 1-step lookahead; heuristic variable selection: convert max # of ternary clauses to binary clauses
  - Randomization with noise parameter for increasing variable choices
- **Randomized CSP solver**
  - Specialized CSP solver for QCP
  - ILOG constraint programming library
  - Variable choice, variant of Brelaz heuristic

Formulation of Learning Problem

- **Different formulations of evidential problem**
  - Examine time taken so far
  - Consider a burst of evidence over initial *observation horizon*
  - Observation horizon + time expended so far
  - General observation policies

Formulation of Learning Problem

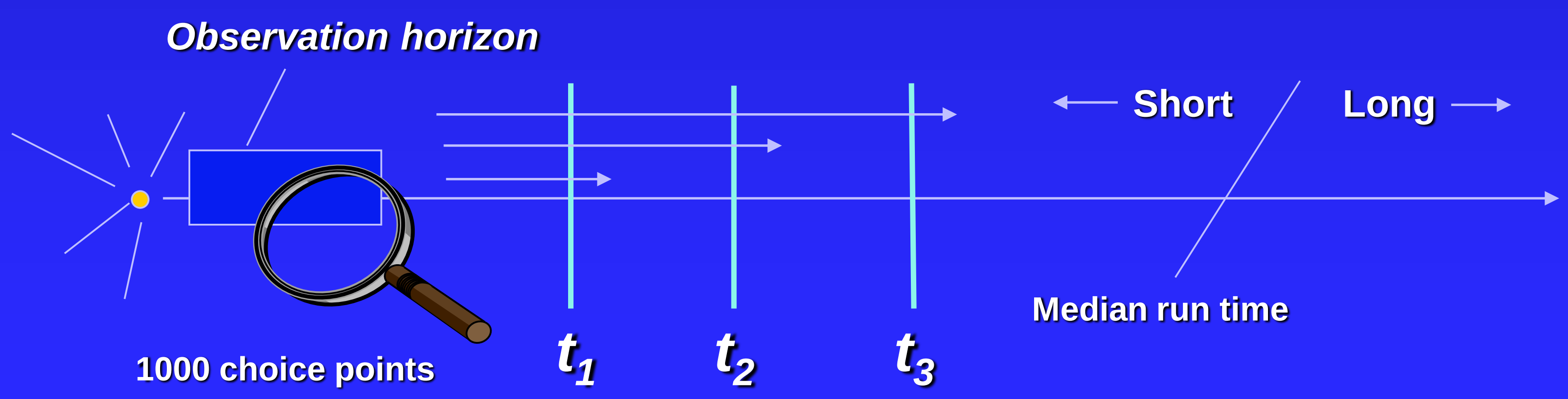
- Different formulations of evidential problem
  - Examine time taken so far
  - Consider a burst of evidence over initial *observation horizon*
  - Observation horizon + time expended so far
  - General observation policies

Observation horizon

Formulation of Learning Problem

- Different formulations of evidential problem
  - Examine time taken so far
  - Consider a burst of evidence over initial *observation horizon*
  - Observation horizon + time expended so far
  - General observation policies

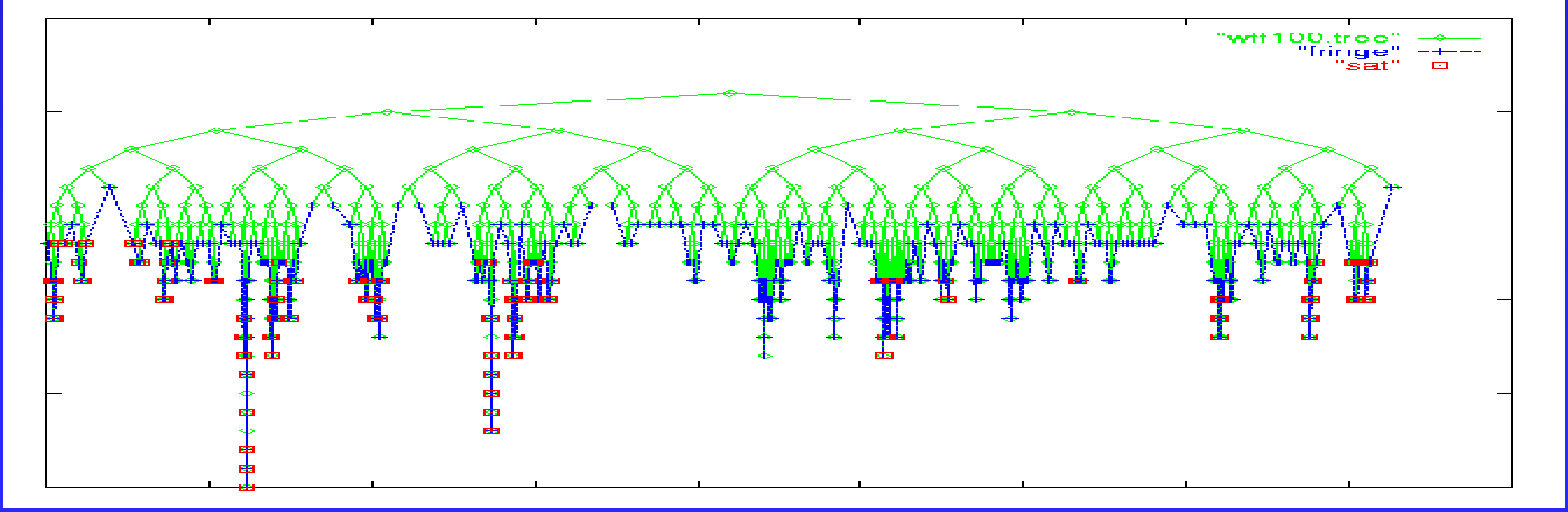
<u>Observation horizon + Time expended</u>



Formulation of Execution Observations

- **Feature classes**
  - Base-level indicators
  - Dynamics: First, second derivatives of values
  - Higher-level statistics over horizon
    - Initial, final, average, min, max values, # sign changes

  e.g., SAT solver:
    - # binary clauses, Avg 1st deriv., Avg 2nd deriv., etc.



Formulation of Execution Observations

- **CSP: 18 basic features for each choice point, summarized by 135 variables**
  - **Generic**

    e.g.,

    - **# backtracks**
    - **depth of search tree**
    - **avg. domain size of unbound CSP variables**
  - **Special**

    e.g.,

    - **variance in distrib. of unbound CSP vars across columns, rows**

- **Satz: 25 basic features, summarized by 127 variables**
  - **# Boolean variables set positively**
  - **Problem size (# unbound variables)**
  - **Size of search tree**
  - **Effectiveness of unit propagation and lookahead**
  - **Total # of truth assignments (models) ruled out**
  - **Degree interaction (shared variables) between binary clauses, λ**

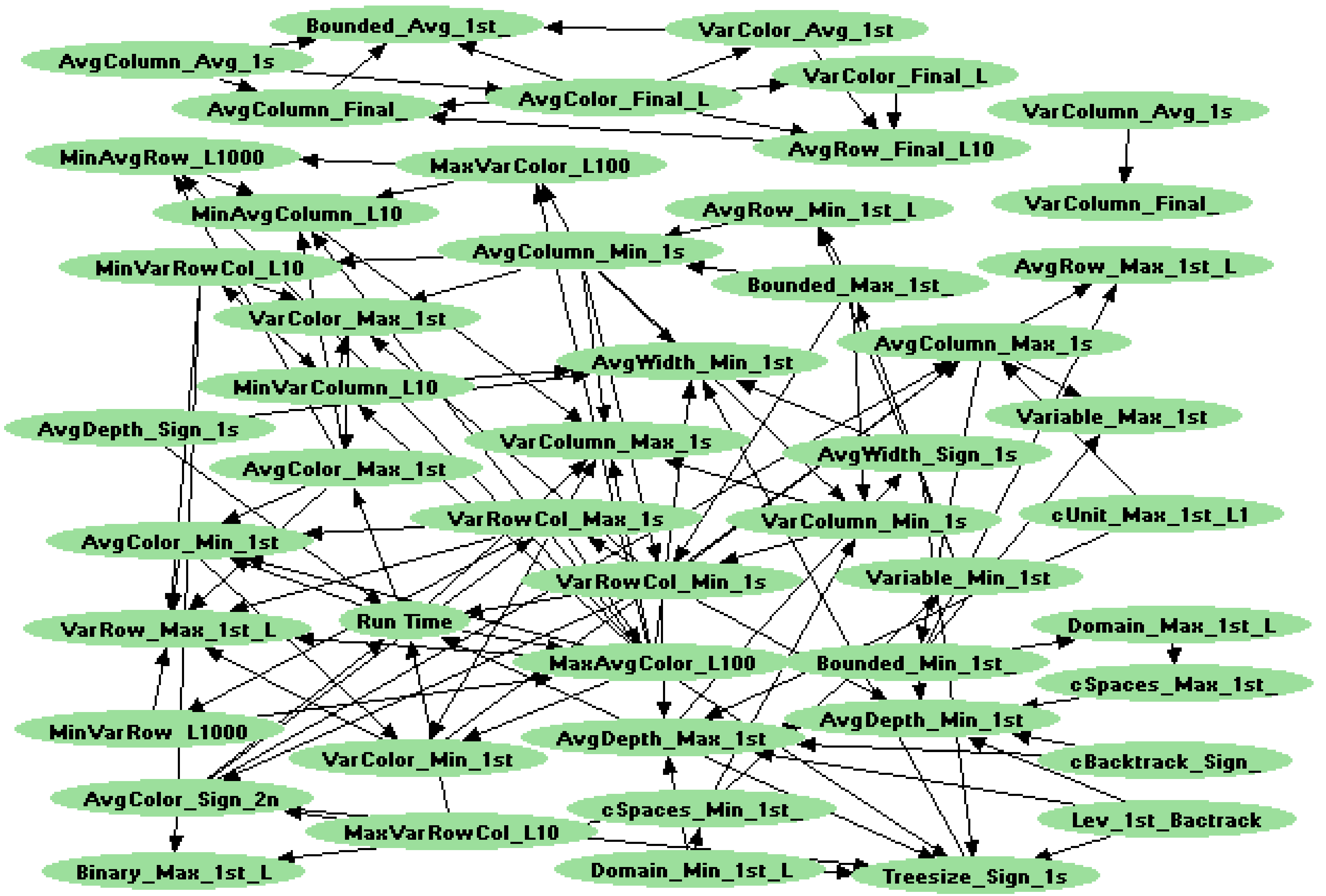
Problem Classes

- Motivated by different formulations of "solving a problem"
- Single Instance Problem
  - Solve a specific instance as quickly as possible
  - *Training and testing on same instance*
- Multiple Instance
  - Draw from a distribution of instances
  - Solve any instance as soon as possible
  - *Training and testing on multiple instances in same class*

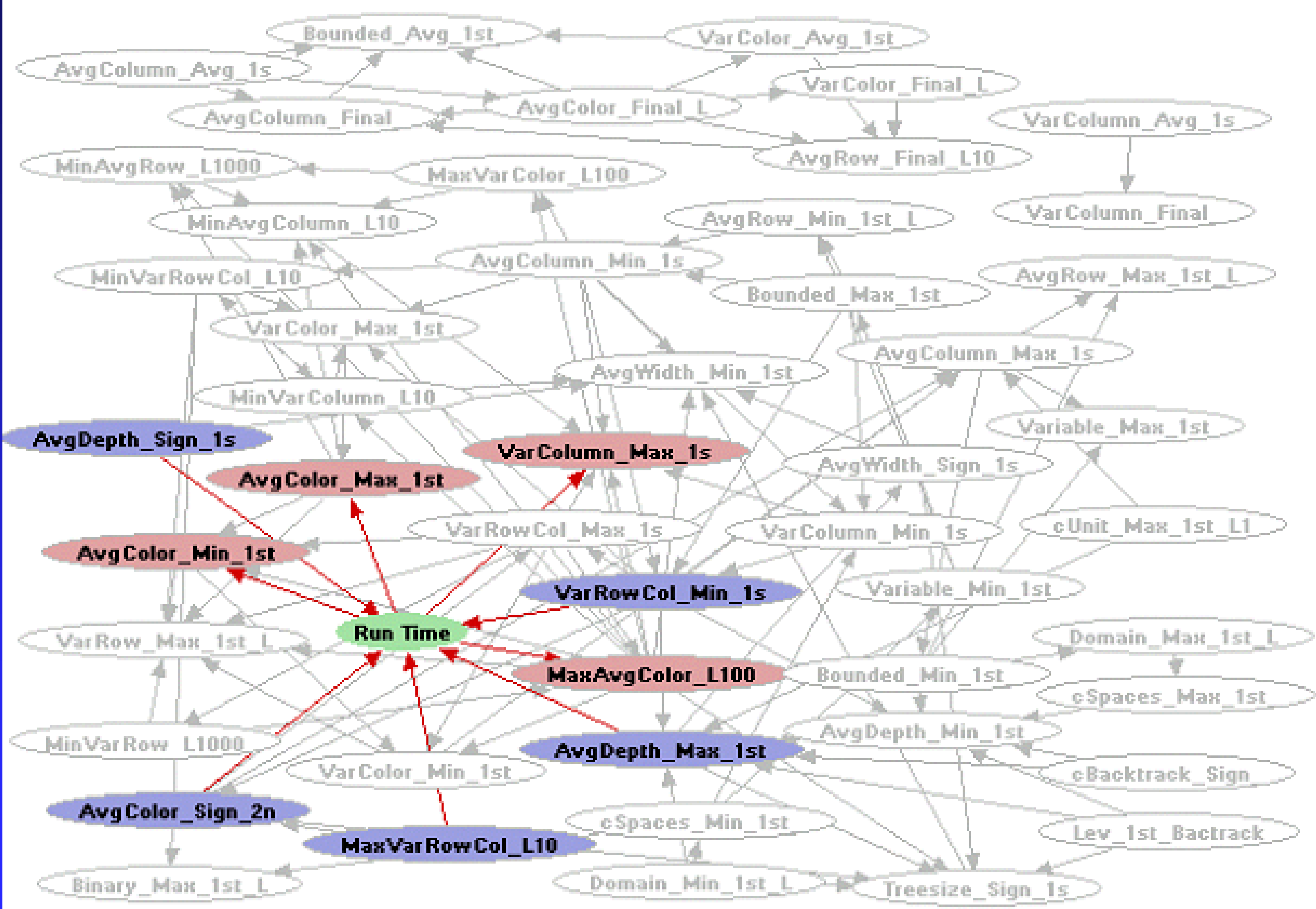
Several days of computing for each dataset!

Sample Results: CSP-QWH-Single

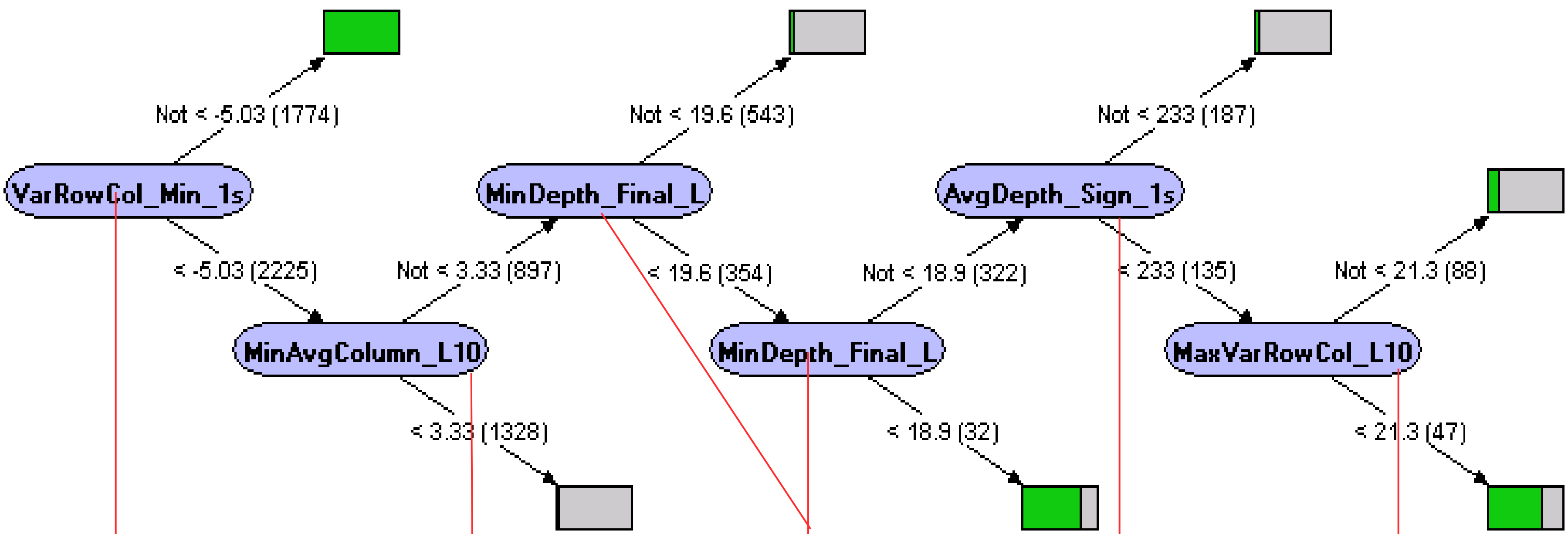
- **QWH order 34, 380 unassigned**
- **Observation horizon without time**
- **Training: Solve 4000 times with random seed**
- **Test: Solve 1000 times**
- **Learning: Bayesian network model**
  - MS Research tool, WinMine
  - Structure search with Bayesian score where conditional distributions are decision trees (Chickering, Heckerman, Meek 1997)
- **Model evaluation:**
  - 96% accurate at classifying run time vs. 49% with marginal model (at chance)



Bounded_Avg_1st_
VarColor_Avg_1st
AvgColumn_Avg_1s
VarColor_Final_L
AvgColor_Final_L
AvgColumn_Final_
VarColumn_Avg_1s
AvgRow_Final_L10
MinAvgRow_L1000
MaxVarColor_L100
MinAvgColumn_L10
AvgRow_Min_1st_L
VarColumn_Final_
AvgColumn_Min_1s
MinVarRowCol_L10
Bounded_Max_1st_
AvgRow_Max_1st_L
VarColor_Max_1st
AvgColumn_Max_1s
AvgWidth_Min_1st
MinVarColumn_L10
Variable_Max_1st
AvgDepth_Sign_1s
VarColumn_Max_1s
AvgWidth_Sign_1s
AvgColor_Max_1st
VarRowCol_Max_1s
VarColumn_Min_1s
cUnit_Max_1st_L1
AvgColor_Min_1st
VarRowCol_Min_1s
Variable_Min_1st
Run Time
VarRow_Max_1st_L
Domain_Max_1st_L
MaxAvgColor_L100
Bounded_Min_1st_
cSpaces_Max_1st_
MinVarRow_L1000
AvgDepth_Min_1st
AvgDepth_Max_1st
VarColor_Min_1st
cBacktrack_Sign_
AvgColor_Sign_2n
cSpaces_Min_1st_
Lev_1st_Bactrack
MaxVarRowCol_L10
Binary_Max_1st_L
Domain_Min_1st_L
Treesize_Sign_1s


Bounded_Avg_1st
VarColor_Avg_1st
AvgColumn_Avg_1s
VarColor_Final_L
AvgColumn_Final
AvgColor_Final_L
VarColumn_Avg_1s
AvgRow_Final_L10
MinAvgRow_L1000
MaxVarColor_L100
VarColumn_Final
AvgRow_Min_1st_L
MinAvgColumn_L10
AvgColumn_Min_1s
MinVarRowCol_L10
Bounded_Max_1st
AvgRow_Max_1st_L
VarColor_Max_1st
AvgColumn_Max_1s
AvgWidth_Min_1st
MinVarColumn_L10
Variable_Max_1st
AvgDepth_Sign_1s
VarColumn_Max_1s
AvgColor_Max_1st
AvgWidth_Sign_1s
VarRowCol_Max_1s
VarColumn_Min_1s
cUnit_Max_1st_L1
AvgColor_Min_1st
VarRowCol_Min_1s
Variable_Min_1st
Run Time
VarRow_Max_1st_L
Domain_Max_1st_L
MaxAvgColor_L100
Bounded_Min_1st
cSpaces_Max_1st
AvgDepth_Min_1st
MinVarRow_L1000
AvgDepth_Max_1st
VarColor_Min_1st
cBacktrack_Sign
AvgColor_Sign_2n
cSpaces_Min_1st
MaxVarRowCol_L10
Lev_1st_Bactrack
Binary_Max_1st_L
Domain_Min_1st_L
Treesize_Sign_1s

Learned Decision Tree



Min of 1st derivative of variance in number of uncolored cells across columns and rows.

Min number of uncolored cells averaged across columns.

Min depth of all search leaves of the search tree.

Change of sign of the change of avg depth of node in search tree.

Max in variance in number of uncolored cells.

Consistent Boost with Modeling

- **10 additional instances: CSP single**

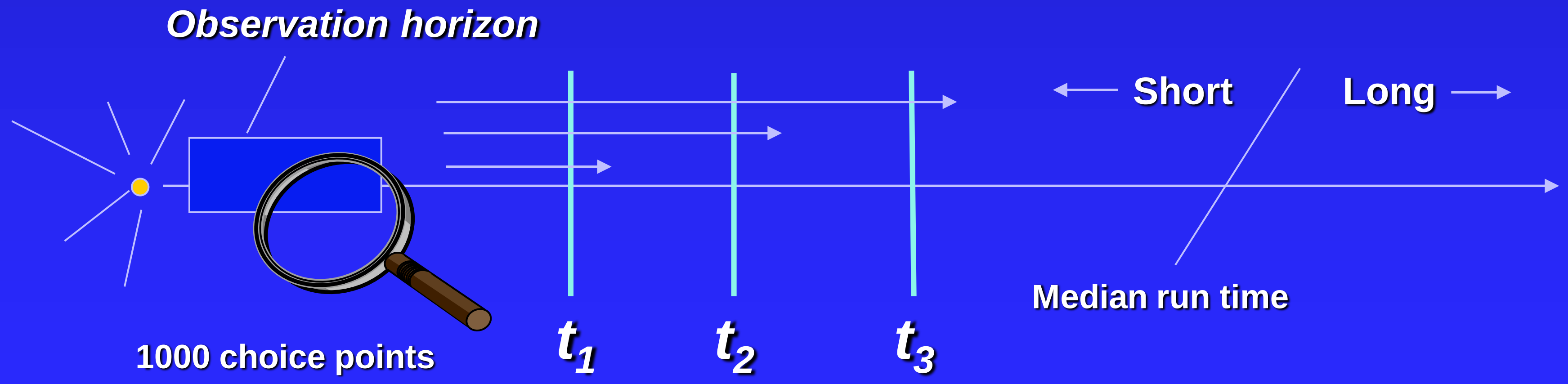
Instance	Accuracy	Marginal model
1004	0.75	0.46
107	0.68	0.5
108	0.92	0.51
121	0.98	0.49
138	0.79	0.52
146	0.88	0.47
160	0.66	0.52
161	0.87	0.48
169	0.78	0.48
28	0.81	0.54
Mean	0.81	0.50
Sd	0.10	0.03

Boosts with Inclusion of Time Expended

e.g., CSP Single QWH, Instance: 138

- Single atemporal model—accuracy: 0.79
- Models for different amount of effort expended
  - Short: .78
  - Medium: .80
  - Long: .85

Observation horizon + Time Expended



Directions with Prediction

- Continuous variables
- Dynamic observation policies
- Generalization of learned models
- Better understanding of basis for power of features
- Insights about problem solving, problem solvers

Application: Dynamic Restart Policies

- **Myopic and richer analyses**
  - Myopic: Is the total expected run time of the restart apriori less expensive than time remaining on current result?
  - More global considerations
- **Comparative analyses**
  - Luby, et al.: "Universal policy:" within log factor of optimal in *distribution free* case.
  - We can do better with a probability distribution
- **Ongoing collaboration among MSR, UW, Cornell on learning and policy**