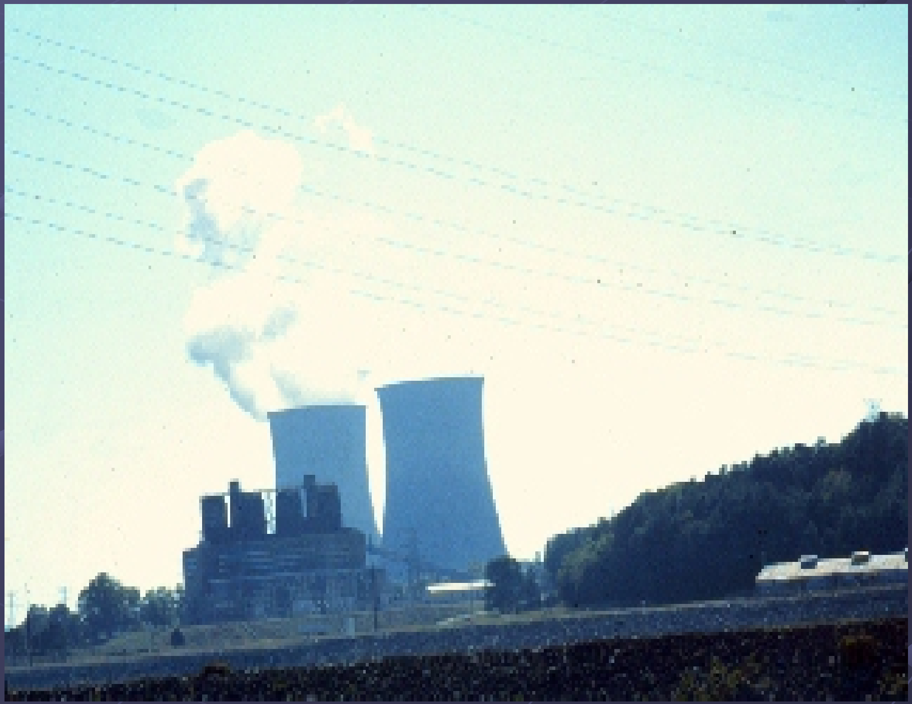
Network Stochastic Programming

J.D. Emmert

The Research Project

- Motivation
- Previous Research
- Network Stochastic Programming
    - Upper Bound
    - Lower Bound



Basic Optimization Approach

- Optimization based on 1 week model: current value plus future value
- obj:

  Max avoided cost + future value

- Existing Model: future value is input (from VPS)
- New Model: future value based on alternative hydrologic scenarios for future weeks

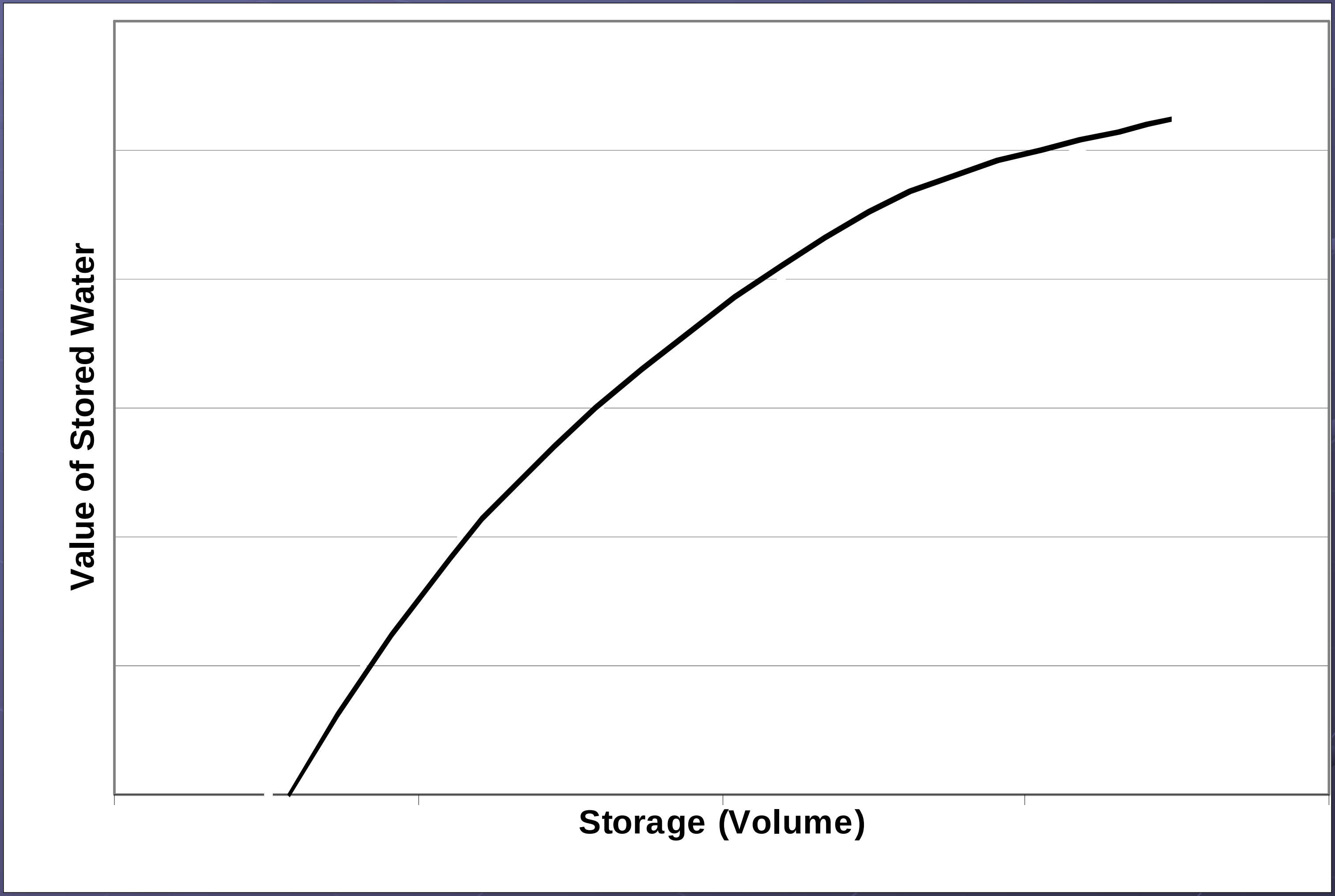
Value of Project Storage

- One curve produced per reservoir
- No interaction between reservoirs is considered
- System wide operation so clearly there is reservoir interaction

  - ex: One very high reservoir, rest very low

    Operation decisions very different

Value of water vs. Storage
(exaggerated curve, not to scale)



Stochastic Nature of Inflow

- Because the future value of water depends on uncertain future hydrologic inflow, the model is by nature stochastic
- Optimizing reservoir scheduling with regard to uncertainty in inflows is a ***Stochastic Optimization*** problem

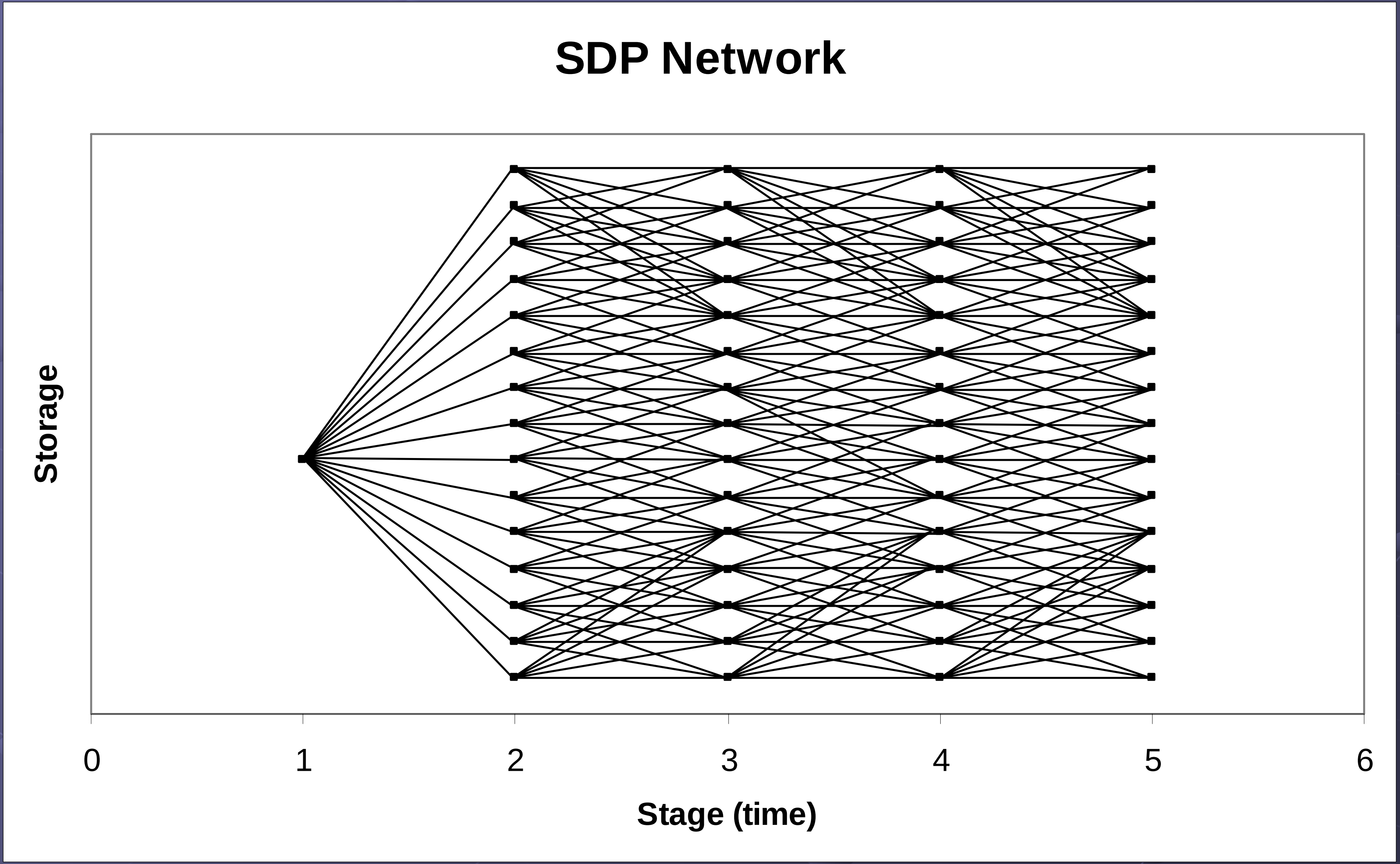
Previous Research

- Two Approaches have historically dominated this classification:

  1. Stochastic Dynamic Programming (SDP)

  2. Stochastic Programming with Recourse (SPR)

SDP (cont.)



SDP (cont.)

- Suppose X = 20 intervals of discretization

1 reservoir $\Rightarrow 20^2$ = 400 states
2 reservoirs $\Rightarrow 20^4$ = 160 thousand states
3 reservoirs $\Rightarrow 20^6$ = 64 million states
4 reservoirs $\Rightarrow 20^8$ = 25 billion states
5 reservoirs $\Rightarrow 20^{10}$ = 10 trillion states

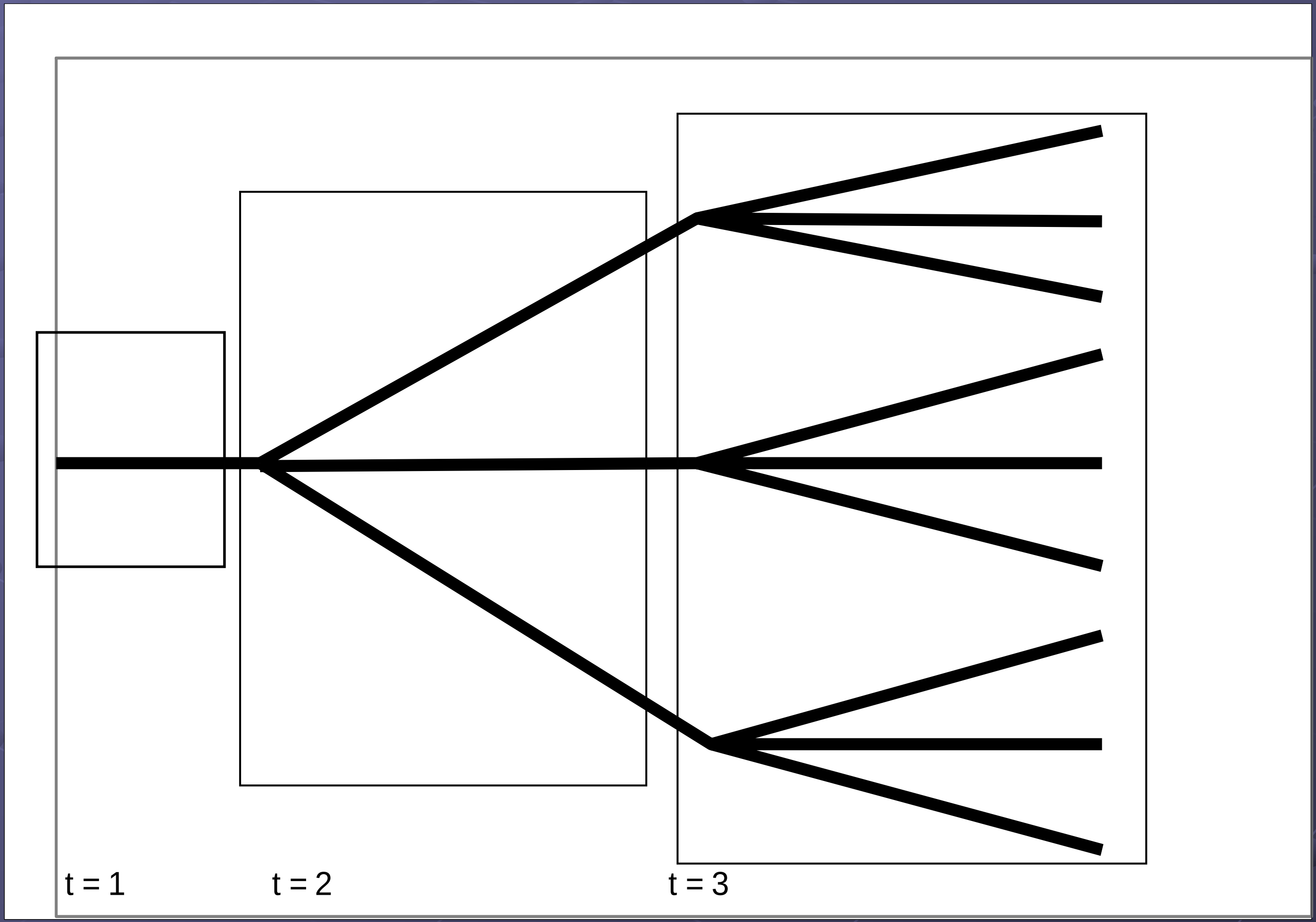
- Computationally infeasible for even small reservoir systems

Alternative: Stochastic Programming with Recourse (SPR)

- Modeling under uncertainty
- Maximizing over a number of scenarios
- Maximizing current period plus expected value of future periods.
- Avoids the curse of dimensionality by constructing an approximation of the future value function from shadow price information gathered from the linear program **(CUTS)**

a.k.a. *Bender's Decomposition*

SPR tree within framework of this research



SPR: Benders Decomposition Approach

- Algorithm decomposes into a collection of subproblems by stage and scenario
- Each subproblem requires initial reservoir storages
- After solving subproblem:
  - pass ending storage values to the next scenario
  - pass “cuts” to previous scenario
- Solved iteratively by “tree traversing strategies” until a first stage solution is converged upon

Subproblem Objective Function sub (ω_t)

$$Max \quad c_t x_t + \sum_{\omega \in scenarios} P_{\omega,t+1} f_{\omega,t+1}$$

$$A_t x_t = b_t(\omega_t) + B_t x_{t-1}$$

$$x_t \geq 0$$

$$f_{\omega,t+1} \leq Obj'_{\omega,t+1} + \sum_r \pi'_{r,\omega}(S_r - S'_r) \forall \omega_{t+1} | \omega_t$$

where

c_t = a vector of first period objective function coefficients

x_t = a vector of first period decision variables

ω_t = scenario index

P_ω = Probability of particular scenario

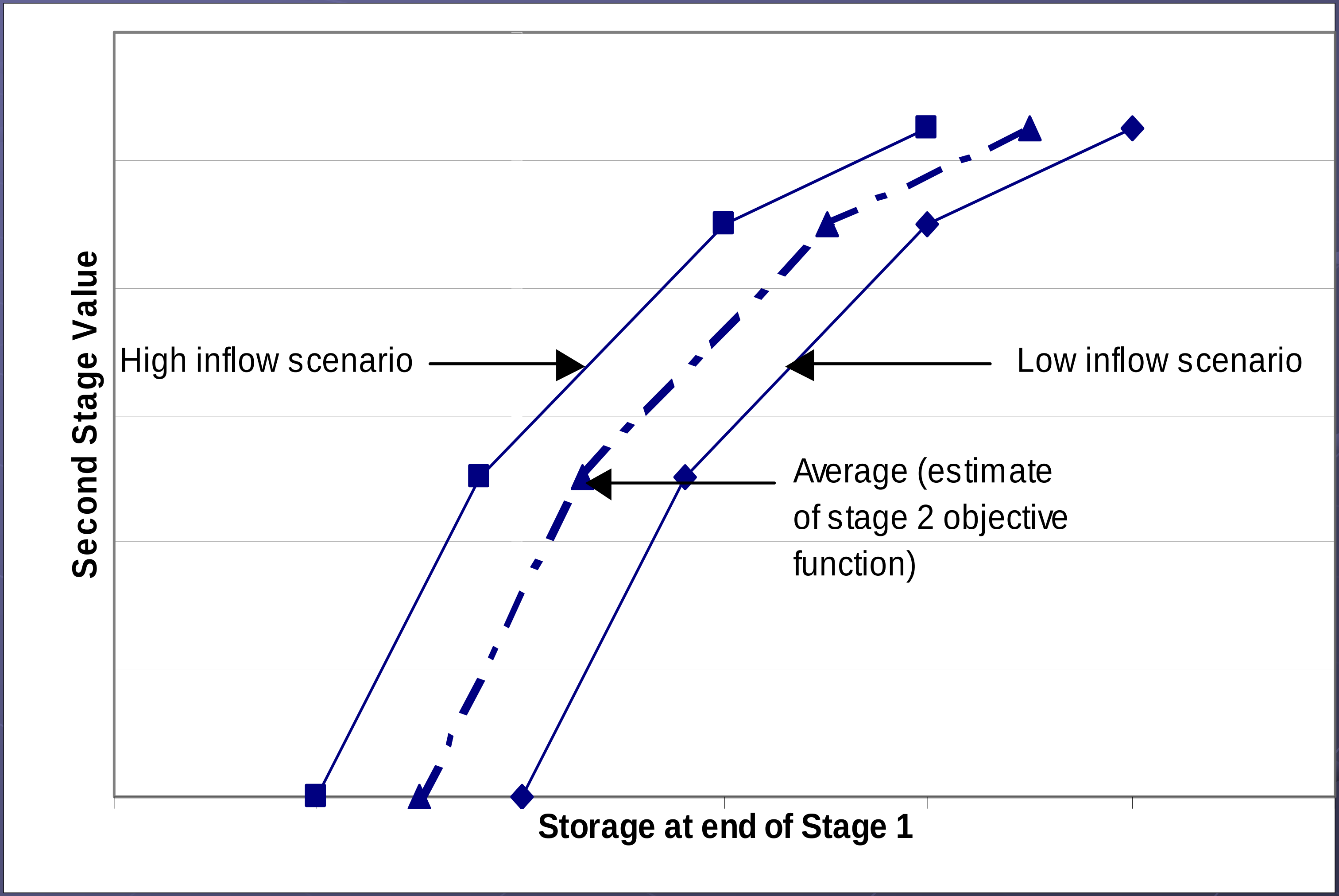
f_ω = future value under scenario ω

π = dual price from stage t+1 solution

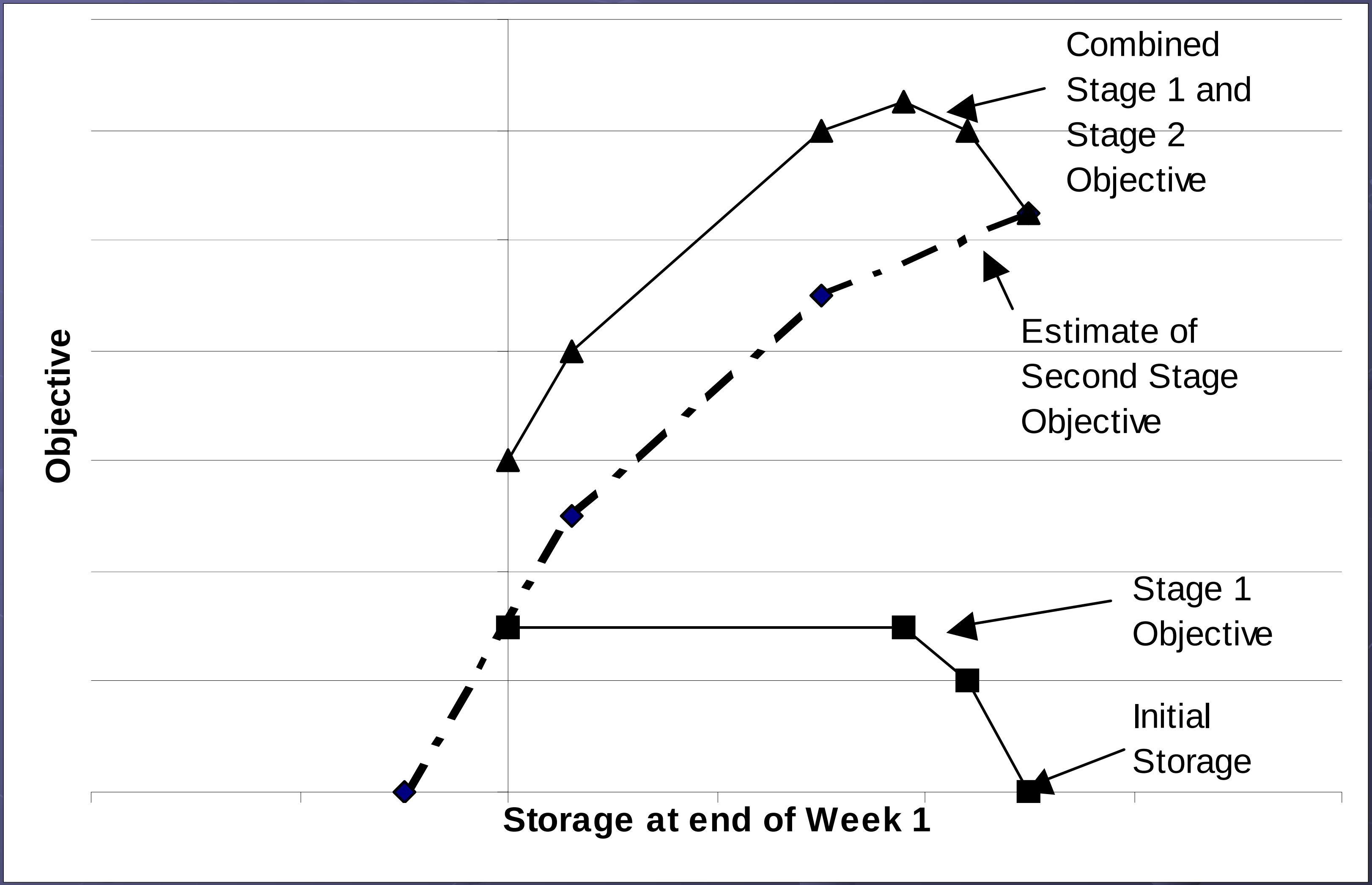
S_r = Storage at end of current stage (variable)

S_r' = Storage at beginning of next stage that produced this cut

Estimate of stage 2 objective function



Maximize combined stage 1 and stage 2 objective function



SPR Challenge

- Size of scenario tree grows exponentially as number of stages increases
- i.e. Problem with three scenarios per stage

2nd Stage:3 scenarios

3rd Stage:9 scenarios

4th Stage:27 scenarios

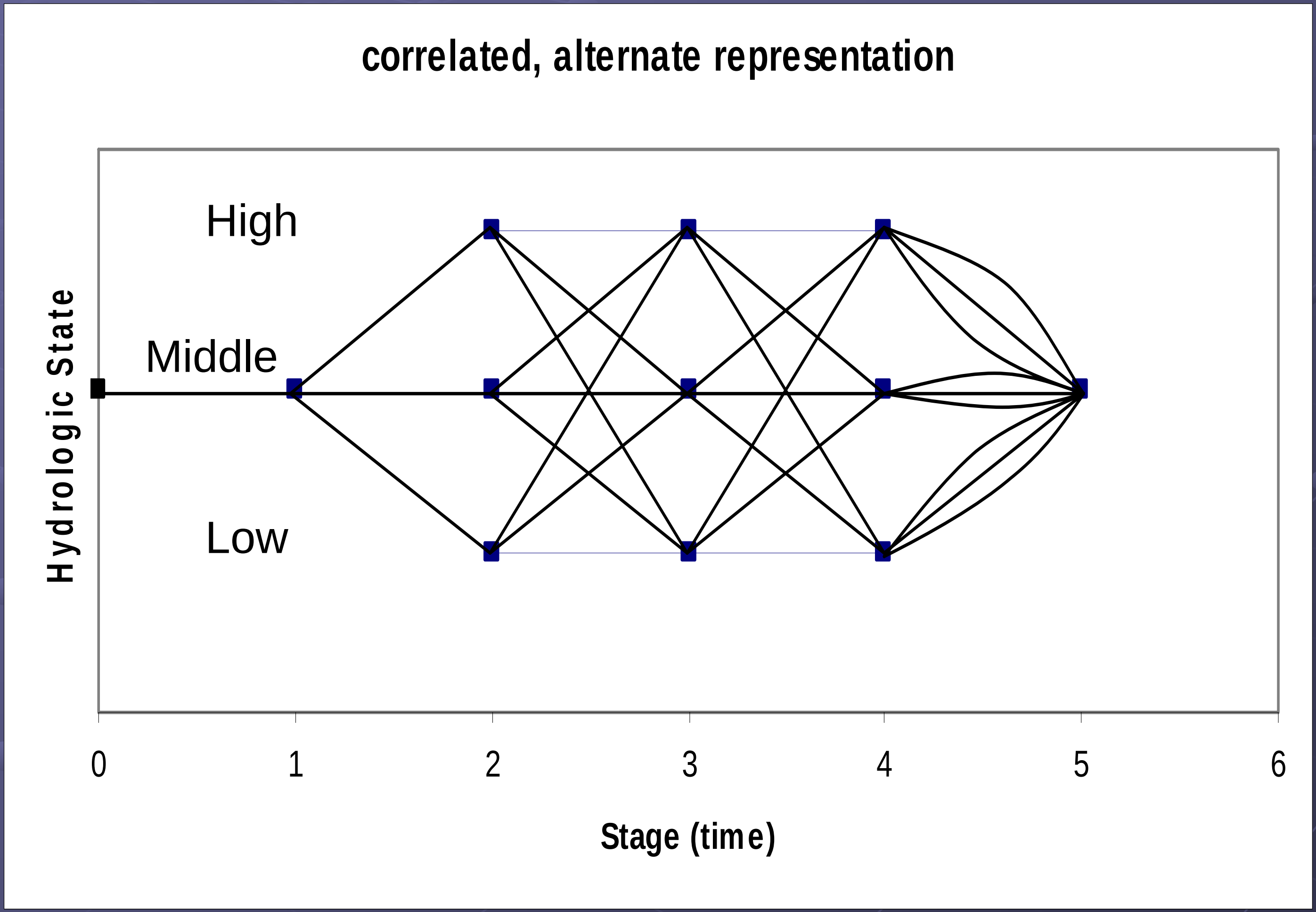
5th Stage:81 scenarios

Network Stochastic Programming – Main Idea

Alternative Representation of SPR tree

- Use hydrologic state to reduce trees to network of states
  - Future value at a state is a function of storage but otherwise independent of path: increased cut sharing
- Intentionally simple definition of state (not focus of this research)
  - Function of historical flows for each week
  - Historical flow mapped mapped into exactly one state for each week
  - Transitional arc between each state
- TVA uses a state definition concept for forecasting

Correlated multistage alternate representation for stochastic model



Iteration

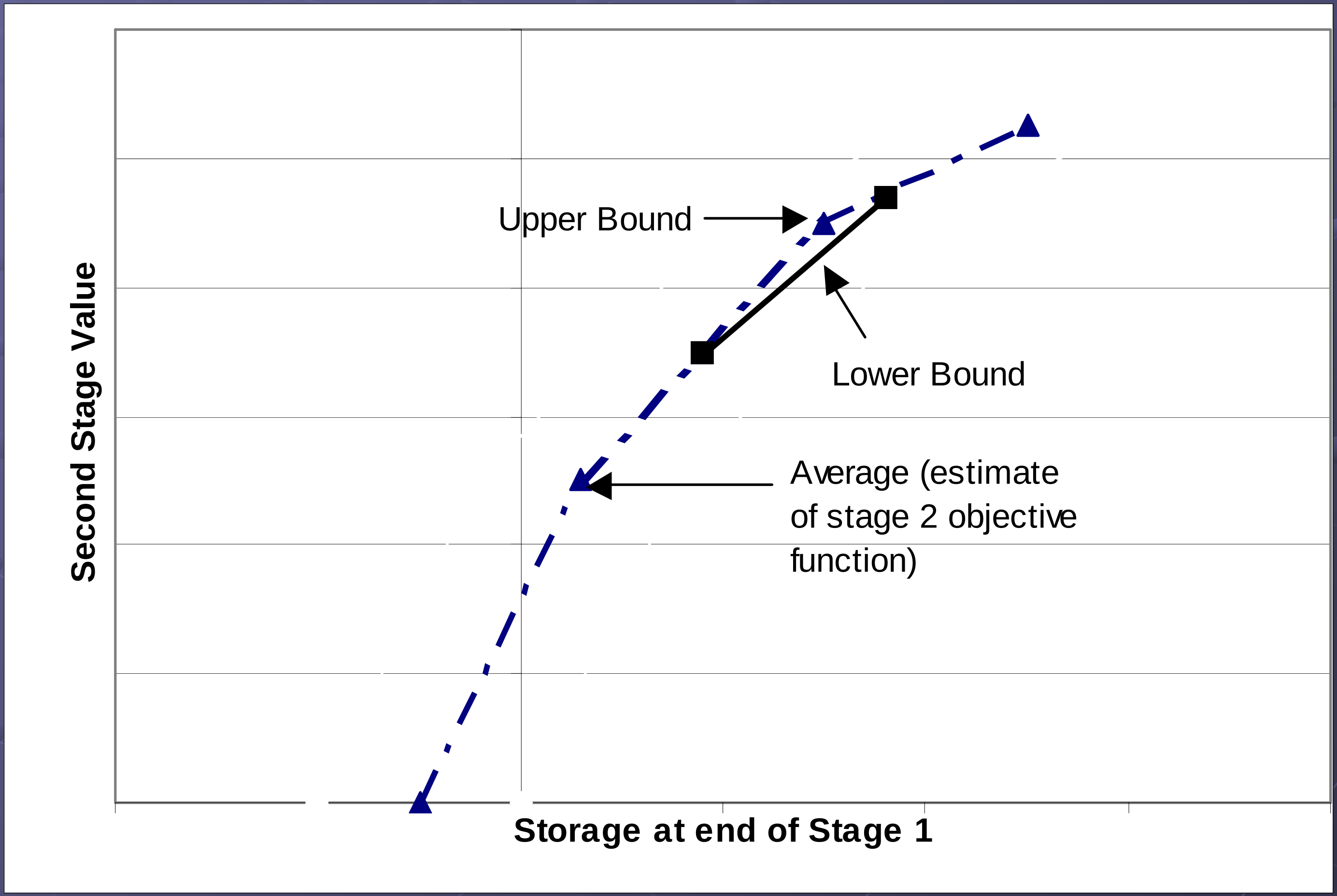
- Determined initial storages on first forward pass
- Solve last stage and pass cuts to the preceding stage
- Continue procedure backward and forward, adding cuts after each problem solved
- End when solution converges

Convergence

- Convergence requires and upper bound (UB), lower bound (LB), and Gap Tolerance.
- Program terminates when bounds are within the gap tolerance:

UB – LB <= Gap Tolerance

Visual Representation of lower bound



Status

- **In testing now with TVA data**
  - **Accomplished**:
    - 4-week Model
    - Generate Cuts for single reservoir
    - Single Objective
  - **To Do**:
    - Longer Model (6 – 8 weeks)
    - Cuts for all storage reservoirs (UB)
    - Compute a lower bound
    - Multiple Objective
  - **Future Implementation**
    - Proof of Concept
    - No GUI representation
    - Not Efficient

Summary

- NSP provides a Stochastic Optimization Solution
- Avoids the problems historically associated with SDP and SPR
- With appropriate enhancement could be a very valuable RiverWare tool