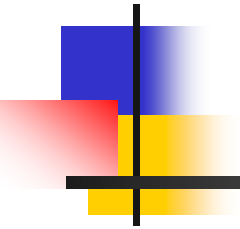


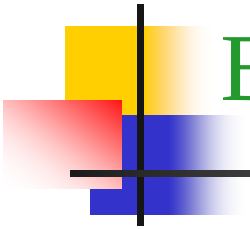
Mobile and Heterogeneous databases

Broadcasting

A.R. Hurson

Computer Science

Missouri Science & Technology



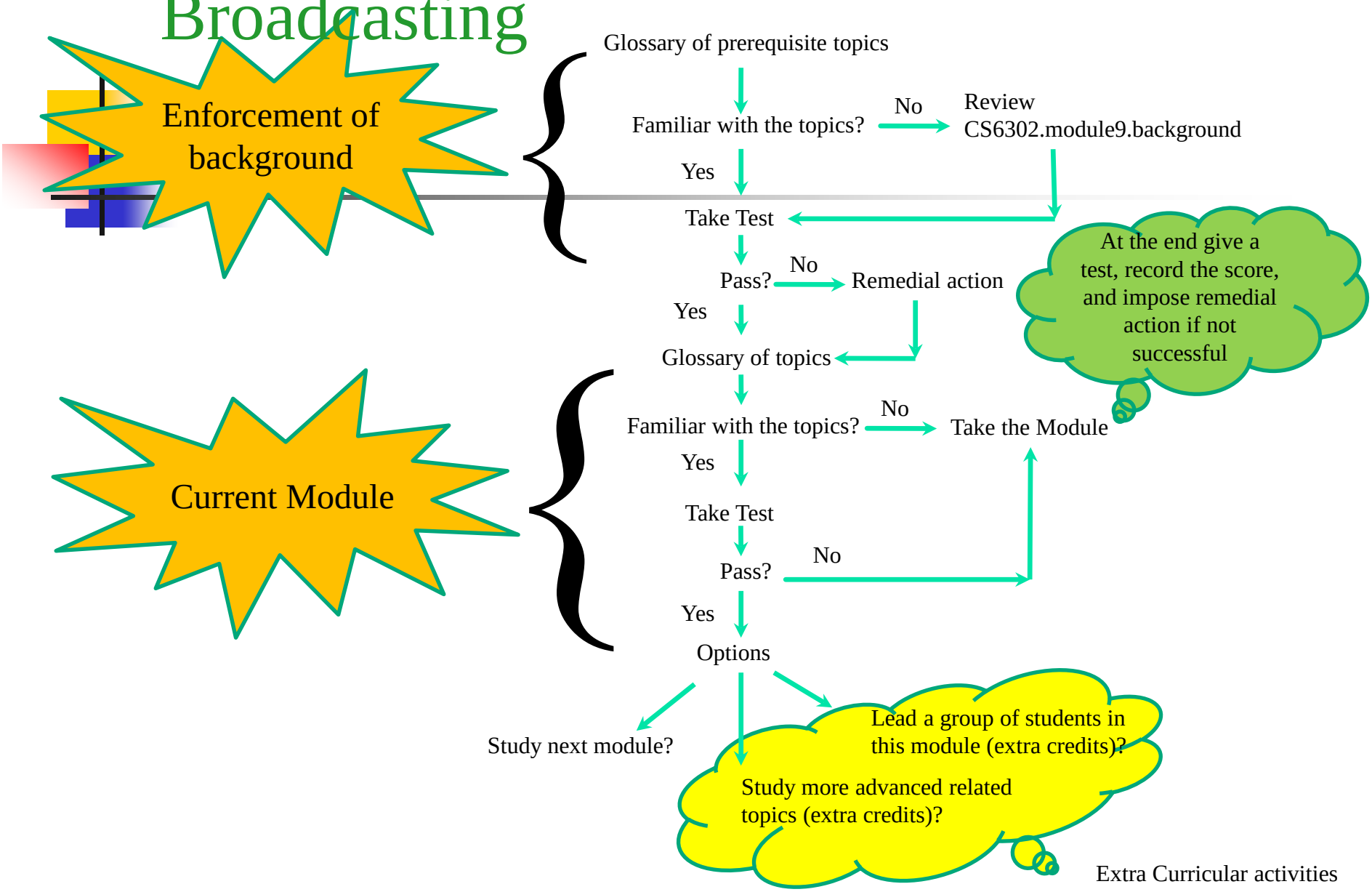
Broadcasting

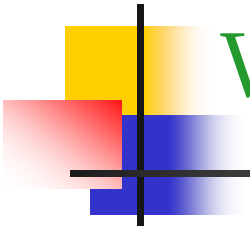
Note, this unit will be covered in nine lectures. In case you finish it earlier, then you have the following options:

1) Take the early test and start CS6302.module10
2) Study the supplement module (supplement CS6302.module9)
3) Act as a helper to help other students in studying CS6302.module9

Note, options 2 and 3 have extra credits as noted in course outline.

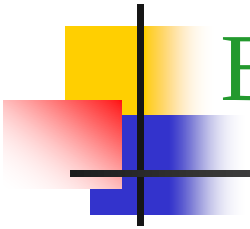
Broadcasting





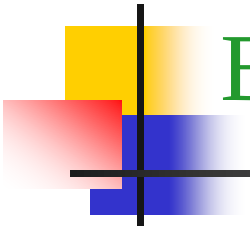
Wireless Communication & Mobility

- You are expected to be familiar with:
  - Mobile DataAccess Systems, if not, you need to study CS6302.module7.



Broadcasting

- Previous module introduced mobility and wireless communication in traditional multidatabase platform. As you recall, we distinguished three classes of data and three classes of services.
- This module will concentrate on “public data” and “broadcasting” as a means to make public data available to the users.
- Within the scope of broadcasting, we will talk about challenges and potential solutions.

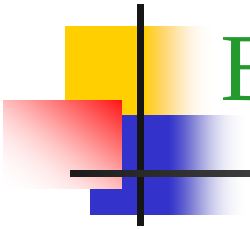


Broadcasting

- Many applications are directed towards public information that are characterized by the:
  - Massive number of users,
  - Similarity and simplicity in the requests, and
  - Fact that data is modified by a few (at the server level).

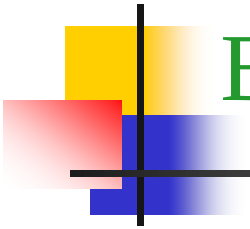
Broadcasting

- The reduced bandwidth attributed to the wireless environment places limitations on the rate and amount of communication.



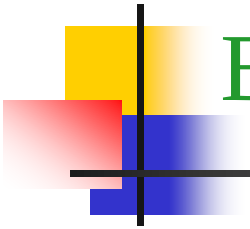
Broadcasting

- Broadcast-Based Service
  - In such an environment, broadcasting has been suggested as a possible solution for query processing.
    - In broadcasting, public data is broadcast over the air channel.
    - A mobile host unit, in search of data, could tune to the appropriate frequency upon which the data is being broadcast.



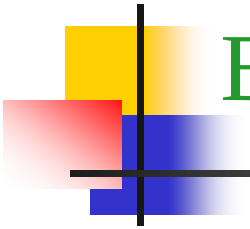
Broadcasting

- Broadcast-Based Service
  - Lower Bandwidth: Response time is independent of the number of users accessing a channel — it scales up as the number of users increases.
  - Limited Energy Resources: Less energy is consumed to receive (to pull) an information from the air channel than sending (to push) the information.
  - Limited Storage Medium: Air channel can be considered as a persistent storage medium.



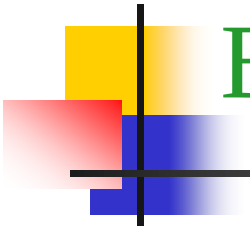
Broadcasting

- Broadcast-Based Service – Challenges
  - Power Consumption – Application of index and organization of objects on broadcast channel (s).
  - Network Latency – Application of parallel channels to reduce broadcast length:
    - Distribution of Objects over parallel channels,
    - Creation of Conflicts.
  - Broadcasting contents – Application specific domain.



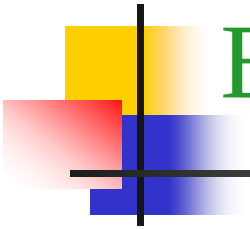
Broadcasting

- Broadcast-Based Service
  - Application of Index in Single and Parallel Broadcast Channel(s),
  - Data Organization on Single and Parallel Broadcast Channel(s),
  - Heuristic and Theoretical conflict resolution schemes,
  - Conflict resolution in the face of replication.



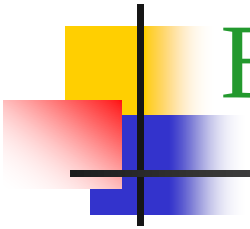
Broadcasting

- Retrieving information from the air channel has two requirements:
  - Minimizing the overall response time at the mobile unit, and
  - Minimizing the amount of power consumed in the retrieval process at the mobile unit.



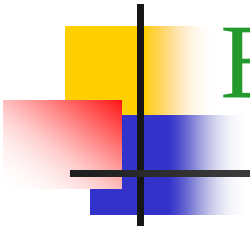
Broadcasting

- Indexing Objects on air Channel
  - An index is a mechanism that speeds up associative searching. It has been extensively used in traditional file and database systems.
  - An index can be formally defined as a function that takes a key value and provides an address referring to the location of the associated data. Its main advantage lies in the fact that it eliminates the need for an exhaustive search through the pages of the data on the storage medium.



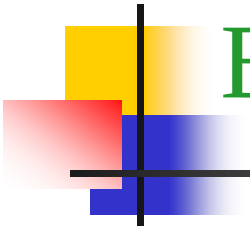
Broadcasting

- Indexing Objects on air Channel
  - B-trees, B+-trees, and hash tables are some of the most common data structures that have been used to implement indexes.
  - In databases, an index is used during the query optimization stage as part of query processing. Normally, a query engine generates multiple plans that can be used to access the required data.



Broadcasting

- Indexing Objects on air Channel
  - Similarly, within the scope of broadcasting, an index is an auxiliary information pointing to the location or possible availability of a data item on the broadcast and hence, allowing mobile unit to predict the arrival time of the data item requested.
  - The prediction of the arrival time enables the mobile unit to switch its operational mode into an energy-saving mode.



Broadcasting

- Indexing Objects on air Channel
  - In short, an indexing mechanism facilitates object retrieval from the air channel(s), reducing power consumption.
  - It should be noted that the advantages of indexing schemes come at the expense of computational overhead and increased length of the broadcast (response time).

Broadcasting

- Indexing Objects on air Channel

Advantages	Disadvantages
Provides auxiliary information that allows mobile users to predict arrival time of objects.	Longer Broadcast.
Enables utilization of different operational modes (active, nap, doze, etc).	Longer response time.
Reduces power consumption (less tune-in time).	Computational overhead due to complexity in retrieval, allocation, and maintenance of the indexes.

Broadcasting

- Indexing Objects on air Channel – Signature-Based Indexing
  - A signature is an abstraction of the information stored in a record or a file.
  - The basic idea behind the application and use of signatures in a broadcast channel is to add a control part to the contents of an information frame.

Broadcasting

- Indexing Objects on air Channel – Signature-Based Indexing
  - A hash function is applied to the contents of the information frame, generating a bit vector and superimposing it on the data frame.
  - A signature partially reflects the data content of a frame. In short, this technique creates a set of signatures for data frames on a broadcast and interleaves them with their associated data frame.
  - Different allocation schemes of signatures on a broadcast channel have been studied in the literature;
    - Single signature,
    - Integrated signature and
    - Multi-level signature.

Broadcasting

- Indexing Objects on air Channel – Signature-Based Indexing
  - In the single signature scheme, the signature frame is broadcast before the corresponding data frame.
  - In the integrated signature scheme, a signature is constructed for a group of consecutive frames called a frame group.
  - The multi-level signature scheme is a combination of the single and integrated signature methods in which the upper level signatures are integrated signatures and the lower level signatures are single signatures.

Broadcasting

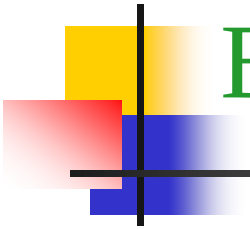
- Indexing Objects on air Channel – Signature-Based Indexing
  - During the retrieval process, a query is resolved by generating a signature based on the user's request.
  - The query signature is then compared against the signatures of the data frames in the broadcast.
  - A successful match indicates a possible hit. Consequently, the content of the corresponding information frame is checked against the query to verify that it corresponds to the user's demands.
  - If the data of the frame corresponds to the user's request, the data is recovered; otherwise, the corresponding information frame is ignored.

Broadcasting

- Indexing Objects on air Channel – Signature-Based Indexing
  - In general, the signature based scheme reduces the access time and the tune-in time when pulling information from the air channel.
  - By checking the signature, the mobile unit partially determines the validity of the content of an information frame – if the data frame does not correspond to the query, the mobile unit switches from active mode to doze mode and waits until the next signature is transmitted.

Broadcasting

- Indexing Objects on air Channel – Signature-Based Indexing
  - As part of the studies done on signature-based indexing schemes, the three aforementioned signature-based schemes have been analyzed with respect to each other using the access time and tune-in time as the performance metrics:
    - With fixed signature size, the multi-level scheme has the best tune-in time performance at the expense of a longer access time.
    - The integrated scheme, on the other hand, has the best average access time, but its tune-in time depends on the similarity among the information frames.
    - Finally, the simple scheme has an adequate access time and tune-in time.
    - In general, in comparison to broadcasting without using indexing, all three signature-based schemes improved tune-in time performance significantly with a reasonable access time overhead.

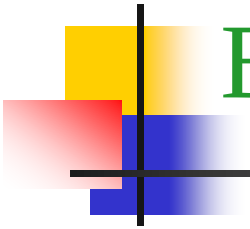


Broadcasting

- Indexing Objects on air Channel – Tree-Based Indexing
  - As noted before, an index is auxiliary information representing one or several data attributes pointing to the location of data collection (i.e., information frames) sharing the same common attribute value(s).
  - This auxiliary information is usually organized as a tree in which the lowest level of the tree points to the location of the information frames on the broadcast channel.
  - With this concept in mind, the frames on the broadcast are of two kinds:
    - Data frames and
    - Index frames.

Broadcasting

- Indexing Objects on air Channel – Tree-Based Indexing
  - A broadcast channel is a sequential medium and hence, to reduce the mobile unit active and tune-in time, and consequently to reduce the power consumption, the index frames are usually replicated and interleaved with the data frames. Otherwise, the request would have to wait for the beginning of the next broadcast cycle — an increase in the query response time.
  - One study investigated two index replication schemes:
    - Distributed indexing and
    - (1, m) indexing



Broadcasting

- Indexing Objects on air Channel – Tree-Based Indexing
  - In distributed indexing, the index is partitioned and interleaved in the broadcast cycle. Each part of the index in the broadcast is followed by its corresponding data frame(s).
  - In (1, m) indexing, the entire index is interleaved *m* times during the broadcast cycle — the whole index is broadcast before every 1/*m* fraction of the cycle.

Broadcasting

- Indexing Objects on air Channel – Tree-Based Indexing
  - A series of analysis for both methods were carried out and compared with the algorithms that provided the best case for tune-in time and response time.
  - It was concluded that, in general, the (1, m) indexing scheme reduces power consumption at the expense of an increase in the response time.
  - The distributed indexing scheme, relative to the (1, m) indexing scheme, increases the response time at a much lower rate at the expense of higher power consumption.

Broadcasting

- Indexing Objects on air Channel – Tree-Based Indexing
  - Previous work has shown that the tree-based indexing schemes are more suitable for the application domains where information is accessed from the broadcast channel randomly, and the signature-based indexing schemes are more suitable in retrieving sequentially structured data elements.
  - In addition, tree-based indexing schemes have shown superiority over the signature-based indexing schemes when user request is directed towards interrelated objects clustered on the broadcast channel(s). Furthermore, tree-based indexing schemes relative to signature-based indexing schemes are more suitable in reducing the overall power consumption.
  - A tree-based indexing provides global information regarding the physical location of the data frames on the broadcast channel. On the other hand, signature-based indexing schemes are more effective in retrieving data frames based on multiple attributes.

Broadcasting

- Indexing Objects on air Channel

Feature	Signature - based Indexing	Tree -based Indexing
Less power consumption		√
Longer length of broadcast	√	√
Computational overhead	√	√
Longer response time	√	√
Shorter tune-in time		√
Random data access		√
Sequentially structured data	√	
Clustered data retrieval		√
Multi-attribute retrieval	√	

Broadcasting

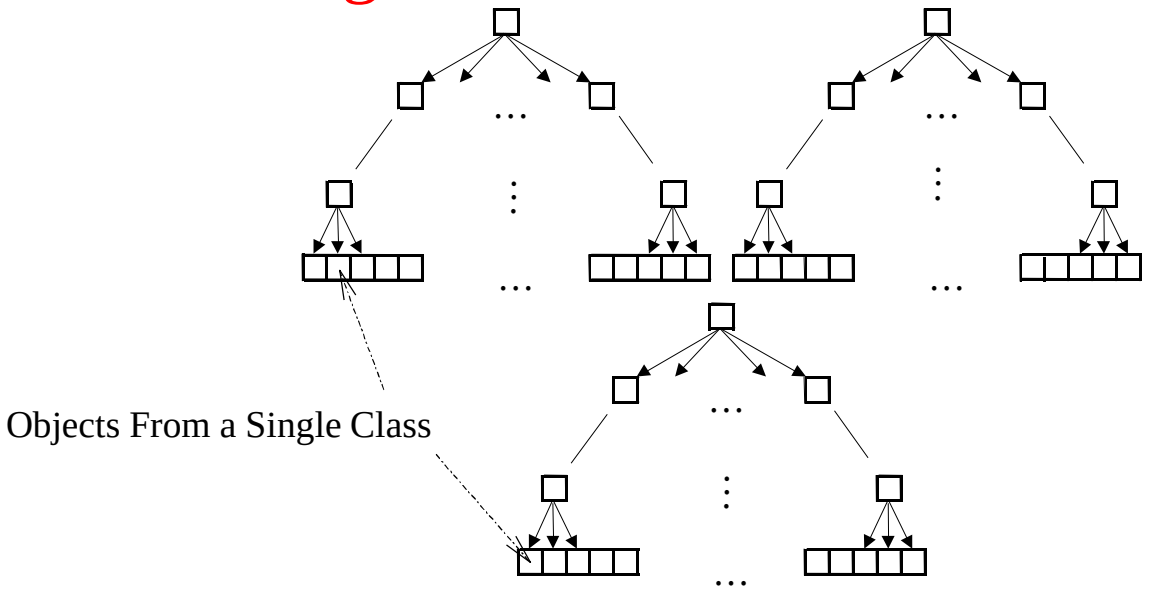
- Indexing Objects on air Channel – Object Oriented Indexing
  - In general, in the presence of an indexing mechanism one can adapt the following protocol for retrieving data objects from the air channel(s) in order to minimize the power consumption.
  - The protocol involves the following steps:
    - Initial probe: The client tunes into the broadcast channel to determine when the next index tree is broadcast.
    - Search: The client accesses the index and determines the offset for the requested objects.
    - Retrieve: The client tunes into the channel and downloads all the required data objects.

Broadcasting

- Indexing Objects on air Channel – Object Oriented Indexing
  - Object-oriented indexing is normally implemented via a multi-level tree.
  - We can classify the possible implementation techniques into two general schemes:
    - Single-class indexing and
    - Hierarchical indexing.
  - In the single-class scheme, multiple multi-level trees are constructed, each representing one class. In this case, the leaf nodes of each tree point to objects belonging only to the class indexed by that tree.
  - The hierarchical-based scheme constructs one multi-level tree representing an index for all classes. The same query has to only navigate the common tree.

Broadcasting

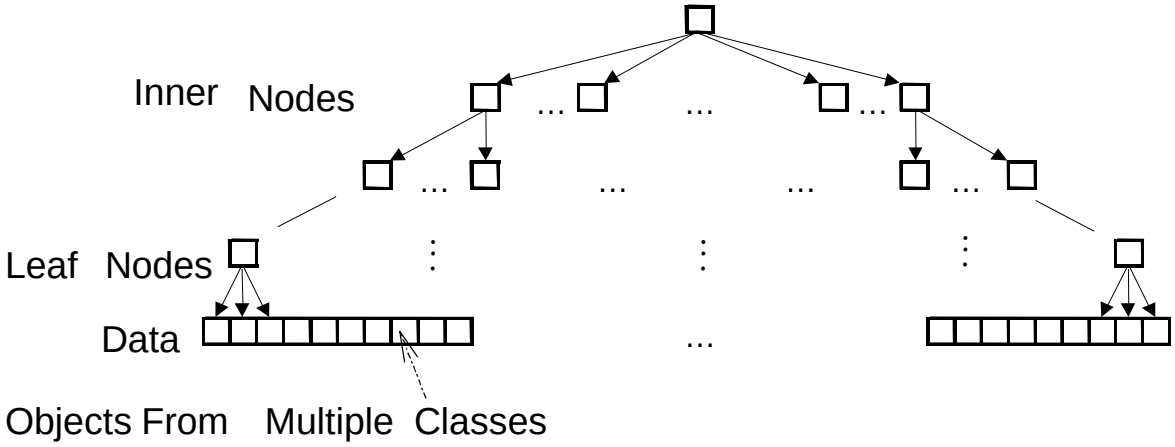
- Indexing Objects on air Channel – Object Oriented Indexing



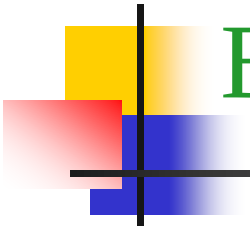
Single-Class-Based IndexTrees

Broadcasting

- Indexing Objects on air Channel – Object Oriented Indexing



Hierarchical-Based Index Trees

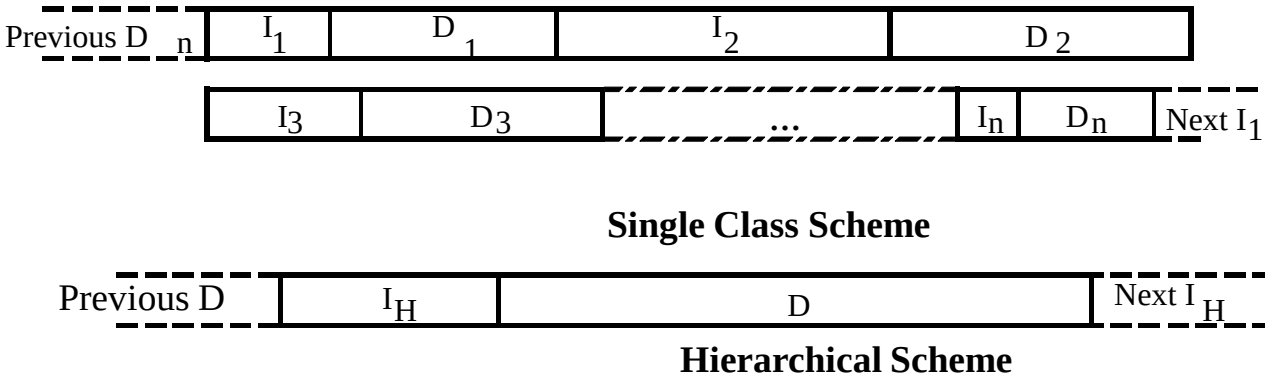


Broadcasting

- Indexing Objects on air Channel – Object Oriented Indexing
  - We assume an air-channel page as the storage granule on the air channel.
  - Due to the sequential nature of the air channel, the allocation of the nodes of a multi-level tree has to follow the navigational path used to traverse the tree, starting at the root. Therefore, an ordering scheme is used to sequentially map the nodes on the air channel. Similarly, data objects are allocated onto air channel pages following their index.
  - Note that, it is possible to interleave and distribute the index pages and associated data pages in a variety of methods.

Broadcasting

- Indexing Objects on air Channel – Object Oriented Indexing



I_i Index of single class i

D_i Data of single class i

I_H Hierarchical index

D Data of all classes

Broadcasting

- Indexing Objects on air Channel – Object Oriented Indexing (Hierarchical Method)
  - In this scheme, whether the domain of the query covers one class or all classes along the hierarchy, the same index structure has to be traversed.
  - Any request has to probe the channel first, read one page, and get an offset to the first page of the index.
  - The modules of the mobile unit can then go into doze mode. Once the index is reached the modules are brought back into active mode.
  - A number of index pages are read and offsets to the required objects are obtained. The offsets are followed and the required objects are retrieved. In the interim between the retrieval of objects, the modules are brought into doze mode.

Broadcasting

- Indexing Objects on air Channel – Object Oriented Indexing (Hierarchical Method)
  - Hierarchical Protocol

Probe onto channel and get offset to the next index	*active*
Reach the index	*doze*
Retrieve the required index pages	*active*
Reach the required data pages	*doze*
Retrieve required data pages	*active*

Broadcasting

- Indexing Objects on air Channel – Object Oriented Indexing (Single Class Method)
  - Single-Class Protocol

Probe onto channel and get offset to the next index	*active*
Reach the index	*doze*
Retrieve offsets to the indexes of required classes	*active*
for every required class	
Reach the index	*doze*
Retrieve the required index pages	*active*
Reach the required data	*doze*
Retrieve required data pages	*active*

Broadcasting

- Broadcast-Based Service – Simulation
  - A simulator was developed to evaluate the feasibility and effectiveness of the proposed algorithms.
  - The NASDAQ database with 4290 securities is used as the source data for the objects on the broadcast.
  - The simulator views the parallel air channels as a two-dimensional $N \times M$ array, where N and M represent the number of parallel air channels and the number of objects on a broadcast, respectively.
  - For each simulation run, user requests are generated, randomly, requesting K objects on the broadcast.
  - For every simulated configuration, the simulator is run 1000 times, and the average number of every estimated performance metric is calculated and reported.

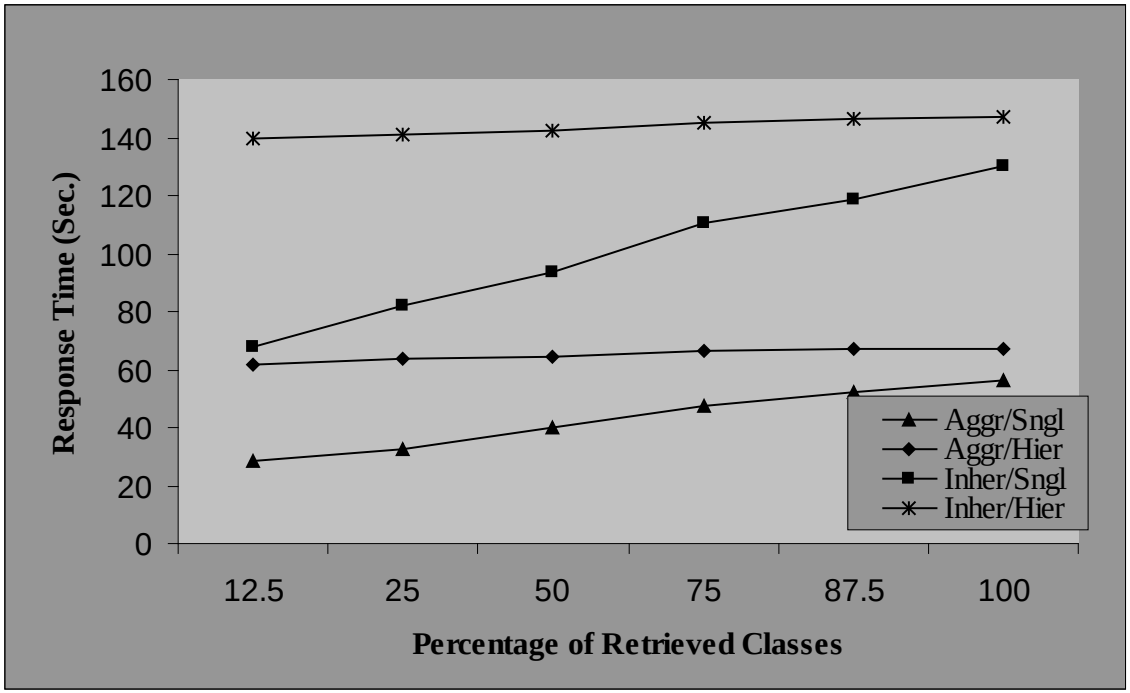
Broadcasting

- Indexing Objects on air Channel – Object Oriented Indexing
  - Response Time Degradation Relative to the No-Indexing Scheme

Aggregation/ Hierarchical	Aggregation/ Single	Inheritance/ Hierarchical	Inheritance/ Single
1.17	1.05	1.1	1.02

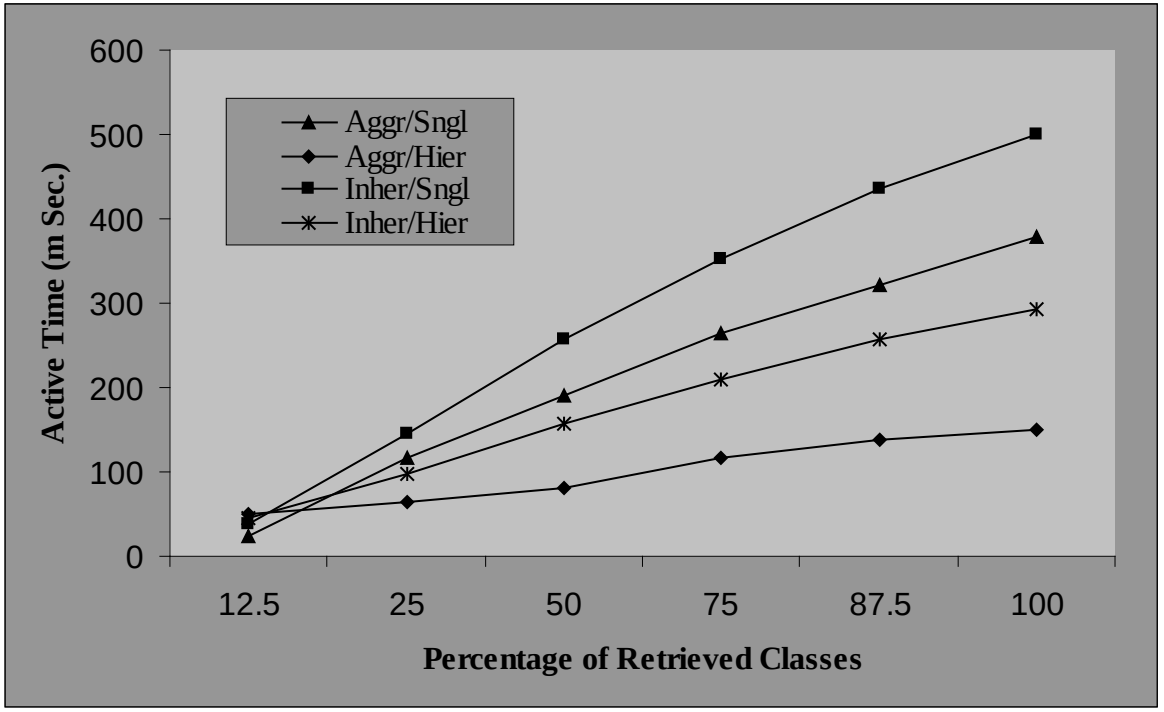
Broadcasting

- Indexing Objects on air Channel – Object Oriented Indexing
  - Response Time *vs.* Number of Retrieved Classes



Broadcasting

- Indexing Objects on air Channel – Object Oriented Indexing
  - Active Time vs. Number of the Retrieved Classes



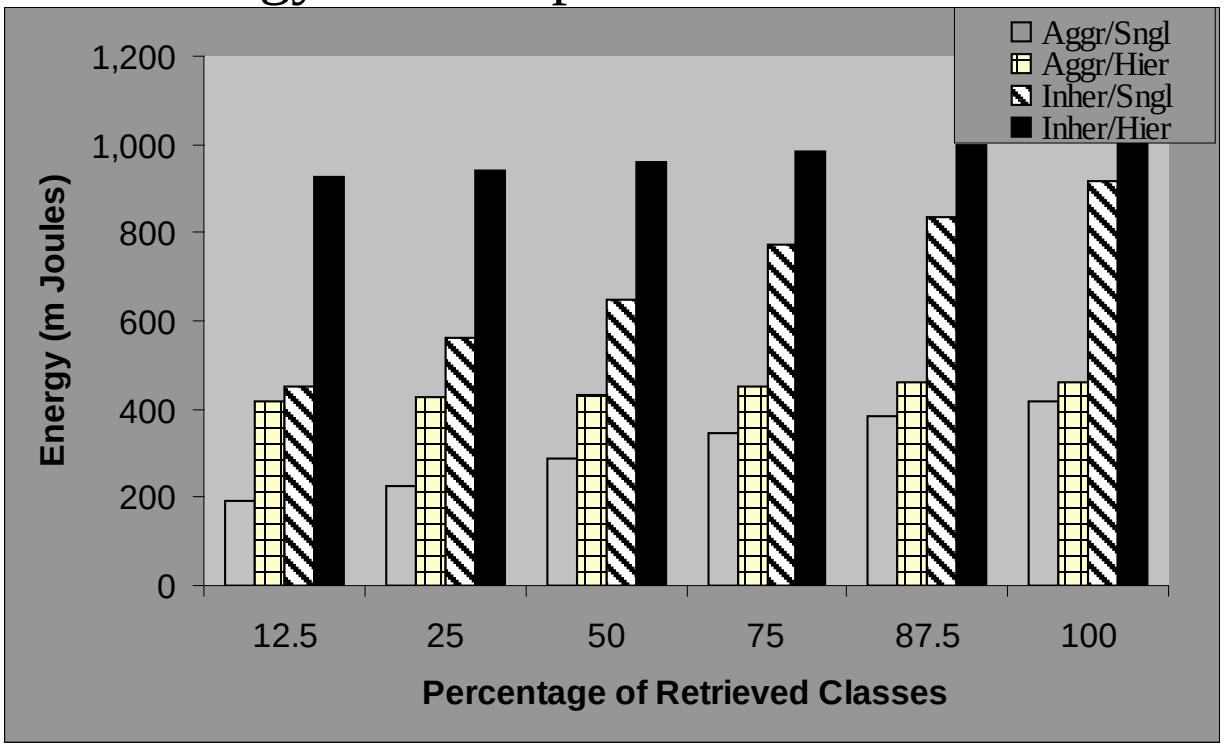
Broadcasting

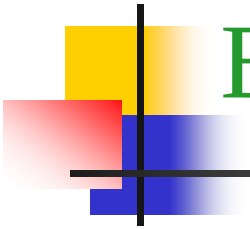
- Indexing Objects on air Channel – Object Oriented Indexing
  - Energy Improvement Relative to the No-Index Scheme

Aggregation/ Hierarchical	Aggregation/ Single	Inheritance/ Hierarchical	Inheritance/ Single
17.5	18.9	18.4	19

Broadcasting

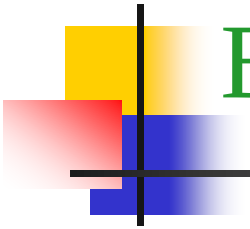
- Indexing Objects on air Channel – Object Oriented Indexing
  - Detailed Energy Consumption





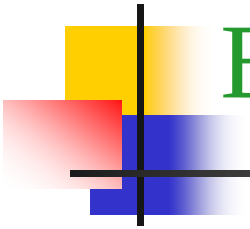
Broadcasting

- Object Organization on the Air Channel
  - Organizing data objects on the air channel as a means to reduce access latency.
  - An appropriate data placement algorithm should attempt to detect data locality and cluster related data close to one another.
  - The objects in an object-oriented paradigm are normally associated with one another through semantic links — inheritance, aggregation, or association.
  - An object-clustering algorithm takes advantage of such relationships and attempts to map a complex object into a linear sequence of objects along the semantic links.



Broadcasting

- Object Organization on the Air Channel
  - Object clustering to minimize the access delay – Taking object connectivity (degree of dependence among objects) into account.
  - How to map a weighted graph (complex object) onto a one dimensional air channel to reduce access latency?
  - Cluster Objects Based on connectivity:
    - Strict Linearity – ApproximateLinearOrder
    - Varying Levels of Connectivity – PartiallyLinearOrder
  - Issues of concern:
    - Linearity
    - Replication



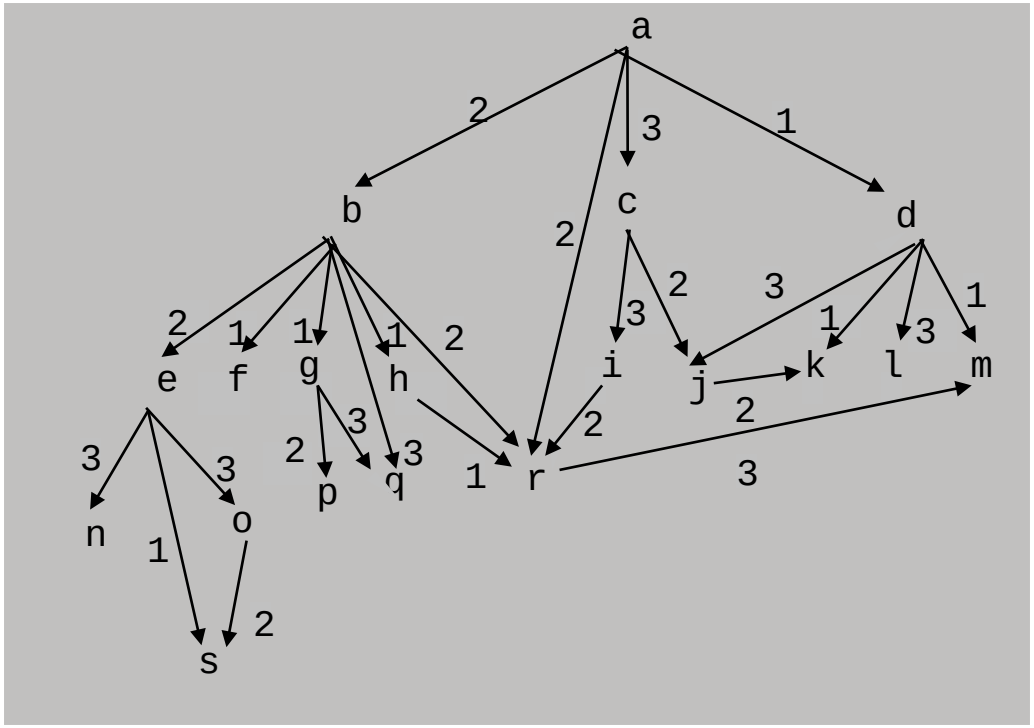
Broadcasting

- Object Organization on the Air Channel
  - A complex object can be expressed as a hierarchy or a directed acyclic graph (DAG) in which objects are represented as the nodes and links are the relationships among these objects.
  - Within the scope of disk storage medium, it has been shown that clustering objects based on their semantic connectivity can improve the response time by an order of magnitude.

Broadcasting

- Object Organization on the Air Channel – An Example

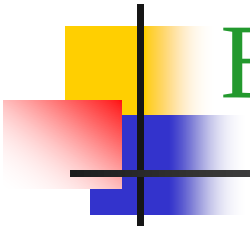
Clustering Method	Resulting Sequence
Depth First	abensofgpqhrmcijdkl
Breadth First	abrcdefgqhijklmnsop
Children- Depth First	abrcdefgqhmnsopijkl
Level Clustering	acibgqprmenosjdlkfh



Broadcasting

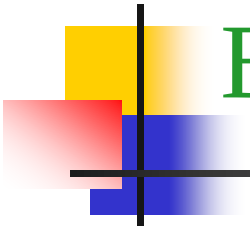
- Object Organization on the Air Channel – Different Clustering Methods

Clustering Method	Resulting Sequence
Depth First	abensofgpqhrmcijdkl
Breadth First	abrcdefgqhijklmnsop
Children- Depth First	abrcdefgqhmnsopijkl
Level Clustering	acibgqprmenosjdlkfh



Broadcasting

- Object Organization on the Air Channel
  - In spite of the differences between a disk and an air channel, from a database perspective, the goal in organizing information on the air channel is still the same as that of the disk — to reduce the response time for accessing a series of objects requested.

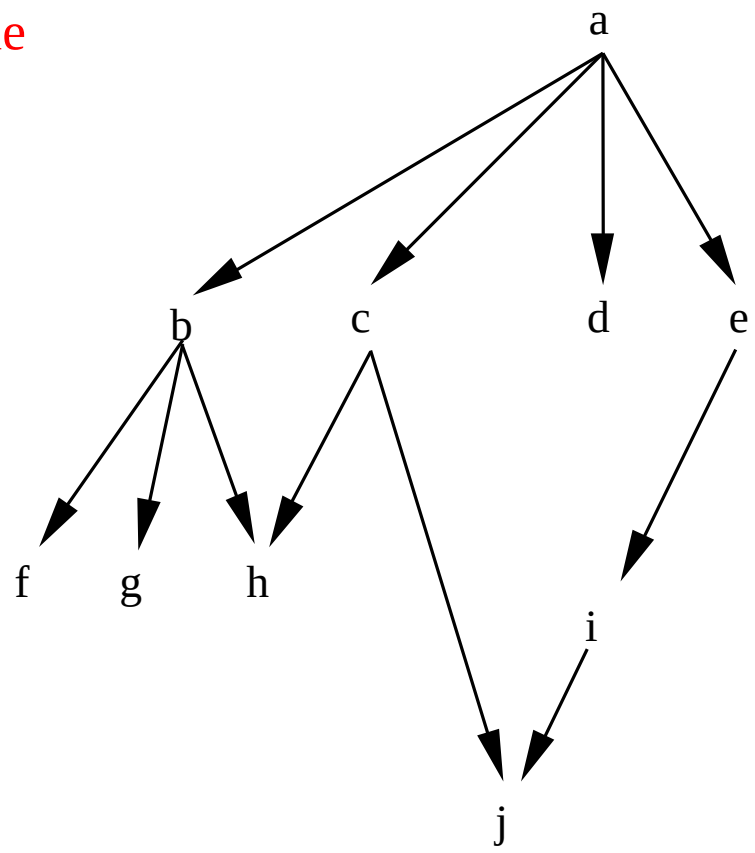


Broadcasting

- Object Organization on the Air Channel
  - In order to achieve this goal, the object organization on an air channel has to meet the following three criteria:
    - Linear ordering: The air channel is a one-dimensional sequential access structure. This fact requires that the object ordering be linear.
    - Minimum linear distance between related objects: multiple objects might be retrieved based on their connection patterns. Reducing the distance among these objects along the broadcast reduces the response time and power consumption.
    - More availability for popular objects: Generally, requests to data follow the 20/80 rule, providing more availability for popular objects can be achieved by simply replicating such objects .

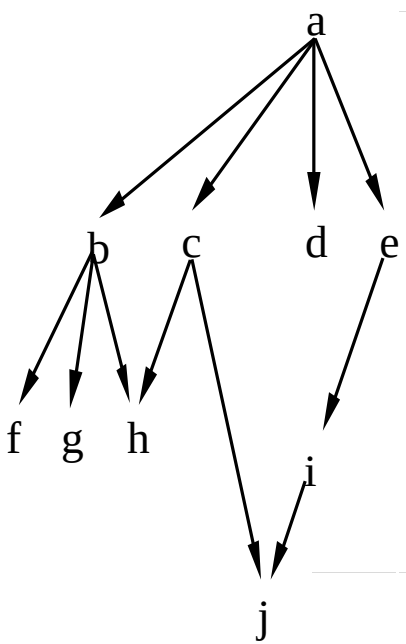
Broadcasting

- Object Organization on the Air Channel – An Example



Broadcasting

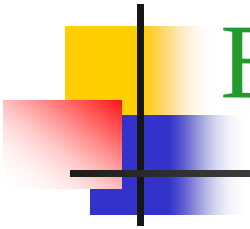
- Object Organization on the Air Channel – An Example



	Linear Sequence	Individual Costs											Total Cost
		ab	ac	ad	ae	bf	bg	bh	ch	cj	ei	ij	
1	abfgchdeij	1	4	6	7	1	2	4	1	5	1	1	33
2	abfgcheijd	1	4	9	6	1	2	4	1	4	1	1	34
3	abcdefghij	1	2	3	4	4	5	6	5	7	4	1	42
4	abgfeichjd	1	6	9	4	2	1	6	1	2	1	3	36
5	acdeijbhgf	6	1	2	3	3	2	1	6	4	1	1	30
6	adeicjbhgf	6	4	1	2	3	2	1	3	1	1	2	26
7	adecbihgfj	4	3	1	2	4	3	2	3	6	3	4	35
8	adecbhgfij	4	3	1	2	3	2	1	2	6	6	1	31
9	adecijbhgf	6	3	1	2	3	2	1	4	2	2	1	27
10	adbfgcheij	2	5	1	7	1	2	4	1	4	1	1	29
11	adceijbhgf	6	2	1	3	3	2	1	5	3	1	1	28
12	aeidcjbhgf	6	4	3	1	3	2	1	3	1	1	3	28
13	aedcbihgfj	4	3	2	1	4	3	2	3	6	4	4	36
14	aedcijbhgf	6	3	2	1	3	2	1	4	2	3	1	28

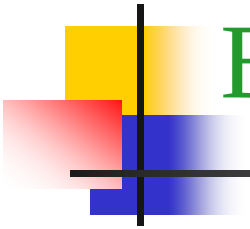
Broadcasting

- Object Organization on the Air Channel – An Example
  - Refer to the previous table, the **abfgchdeij** sequence objects *a* and *d* are separated by the sequence *bfgch* and thus have a cost of 6.
  - The right-most column represents the total cost associated with each individual linear sequence.
  - An optimal sequence is the linear sequence with the minimum total sum.
  - In a query where multiple related objects are retrieved, a minimum average linear distance translates into smaller average response time. In this example, the optimum linear sequence achieves a total sum of 26.



Broadcasting

- Object Organization on the Air Channel
  - One method in obtaining an optimal linear sequence is to enumerate and calculate all possible linear sequences with their associated costs and then choose the sequence with the minimum cost.
  - Naturally, such a solution, though simple, is computationally impractical.
  - In practice heuristic rules are used to generate a linear sequence with a reasonable cost.



Broadcasting

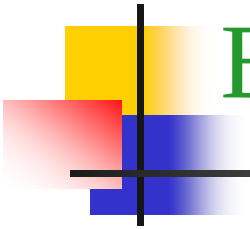
- Object Organization on the Air Channel – Strict Linearity – ApproximateLinearOrder
  - Independent Node: An independent node is a node that has either one or no parent.
  - In example on slide 52, node *e* is an independent node whereas node *h* is not.
  - A graph containing only independent nodes makes up a forest.

Broadcasting

- Object Organization on the Air Channel – Strict Linearity – ApproximateLinearOrder
  - Prioritize node based on their number of descendants in ascending order — the node with the least number of descendants is given a higher priority.
  - Once a node is selected, all of its descendants should be visited and placed on the sequence in a depth first manner, without any interruption from breadth siblings.
  - If a node has a non-independent child, with all of its parents already visited, the non-independent child should be inserted in the linear sequence before any independent child.

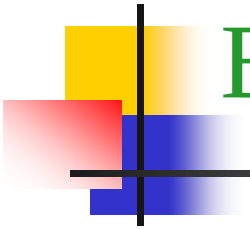
Broadcasting

- Object Organization on the Air Channel – Strict Linearity – ApproximateLinearOrder
  - ApproximateLinearOrder Algorithm
    - Traverse DAG using DFS traversal and as each node is traversed
    - Append the traversed node *N* to the sequence
    - Remove *N* from {nodes to be traversed}
    - if {non-independent children of *N* having all their parents in the sequence} $\neq \varnothing$
    - Set $\leftarrow$ {non-independent children of *N* having all their parents in the sequence}
    - else
    - if {independent children of *N*} $\neq \varnothing$
    - Set $\leftarrow$ {independent children of *N*}
    - NextNode $\leftarrow$ node $\in$ Set | node has least # of descendants among the nodes in Set.



Broadcasting

- Object Organization on the Air Channel – Strict Linearity – ApproximateLinearOrder
  - The algorithm assumes a greedy strategy and starts by selecting a node with an in-degree of zero and out-degree of at least one.
  - Applying ApproximateLinearOrder algorithm to the graph of slide 52 generates either the fifth or eleventh sequence (slide 53). This is dependent on whether *c* or *d* was chosen first as the child with the least number of independent-children.



Broadcasting

- Object Organization on the Air Channel – Varying Levels of Connectivity – PartiallyLinearOrder
  - In a complex object, objects are connected through semantic links with different degrees of connectivity. The frequency of accesses of objects in an object-oriented database reveals that some patterns are more frequently traversed than others.
  - The algorithm PartiallyLinearOrder assumes a weighted DAG as its input and produces a linear sequence.

Broadcasting

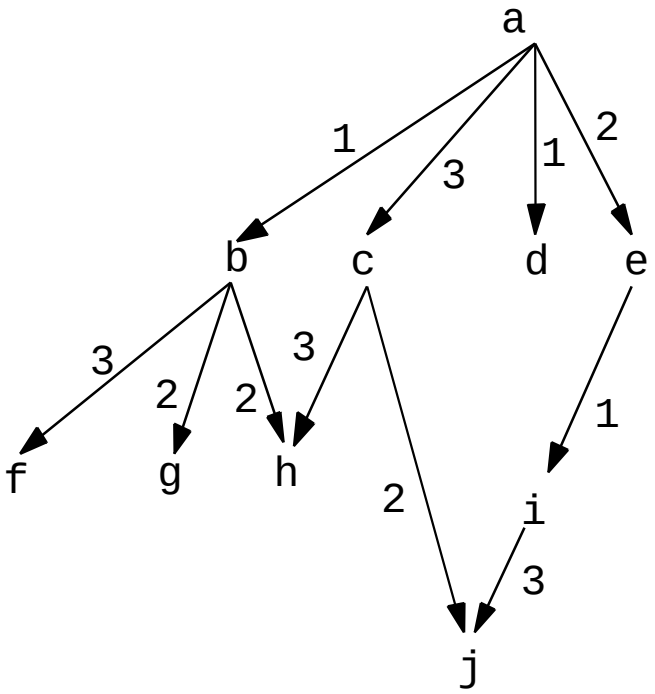
- Object Organization on the Air Channel – Varying Levels of Connectivity – PartiallyLinearOrder
  - The algorithm combines the nodes (single_node) of the graph into multi_nodes in a descending order of their connectivity.
  - The insertion of single_nodes within a multi_node respects the linear order at the granularity level of the single_nodes.
  - The multi_nodes are merged (with multi_nodes or single_nodes) at the multi_node granularity, recursively, without interfering with internal ordering sequences of a multi_node.

Broadcasting

- Object Organization on the Air Channel – Varying Levels of Connectivity – PartiallyLinearOrder
  - PariallyLinearOrder Algorithm
    - for every weight w_s in descending order
    - for every two nodes N_1 & N_2 connected by w_s
    - merge N_i & N_j into one multi_node
    - for every multi_node *MN*
    - $w_m = w_s - 1$
    - for every weight w_m in descending order
    - while ∃ adjacent_node *AN* connected to *MN*
    - if ∃ an edge in both directions between *MN* & *AN*
    - compute *WeightedLinearDistanceMN_AN* & *WeightedLinearDistanceAN_MN*
    - merge *MN* & *AN* into one multi_node, based on the appropriate direction

Broadcasting

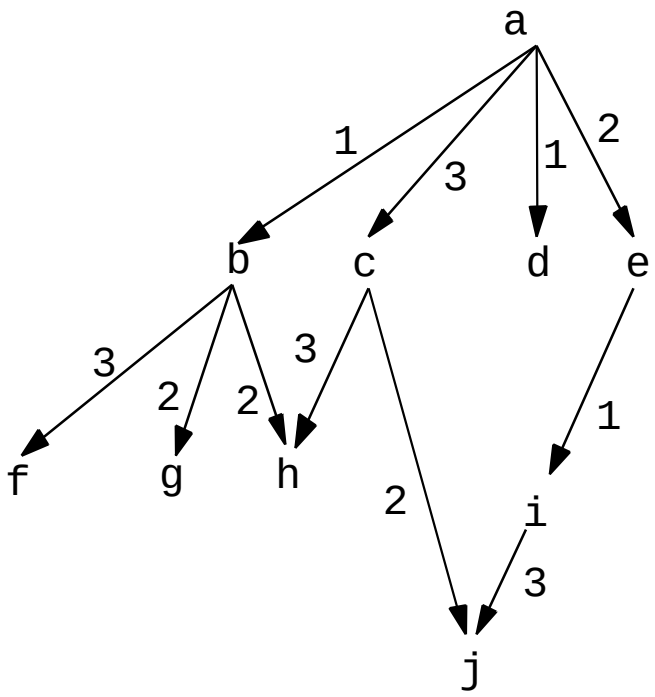
- Object Organization on the Air Channel – Varying Levels of Connectivity – PartiallyLinearOrder



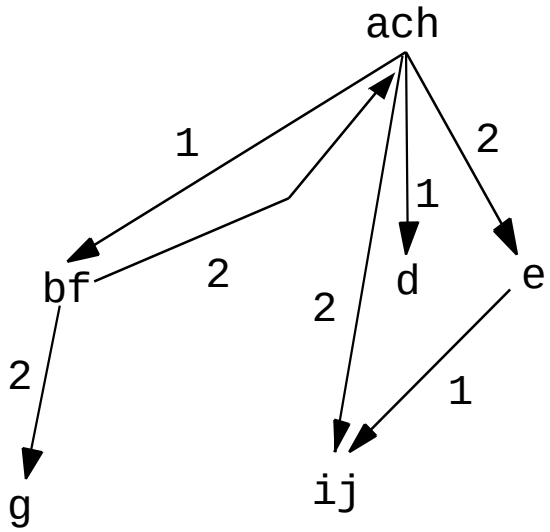
Original Graph

Broadcasting

- Object Organization on the Air Channel – Varying Levels of Connectivity – PartiallyLinearOrder



Original Graph



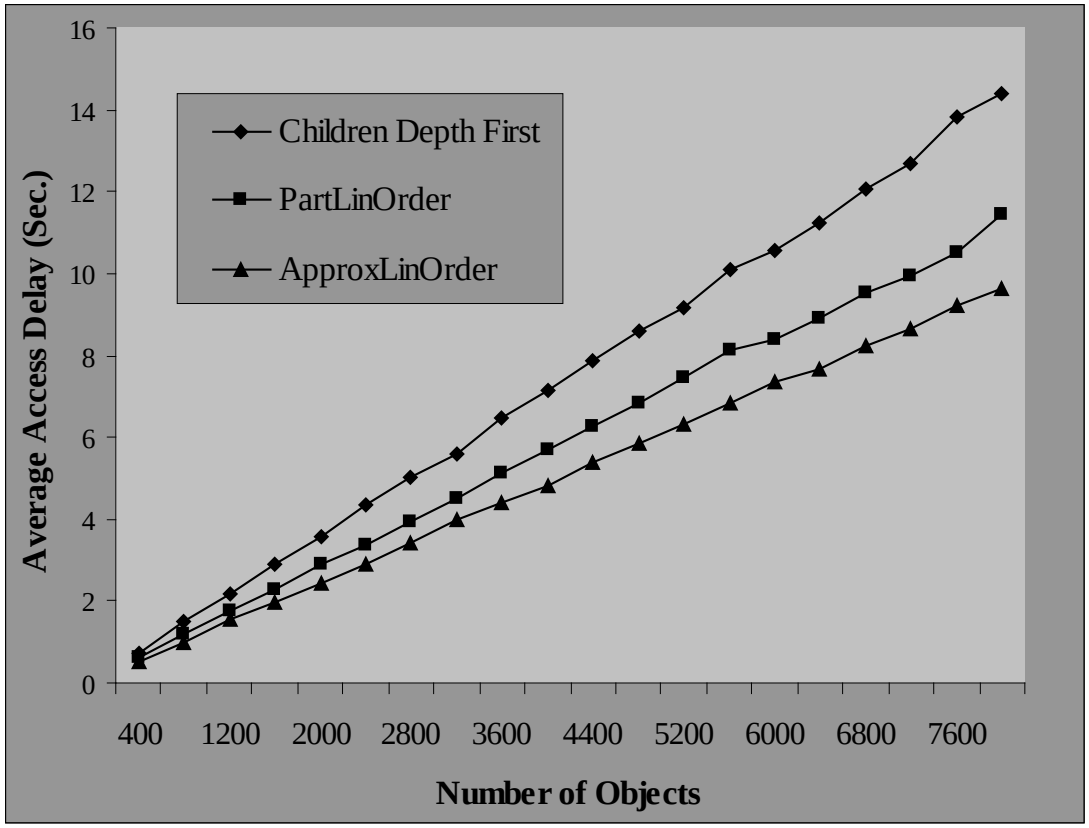
First and Second iterations

b f g a c h e i j d

Third iteration

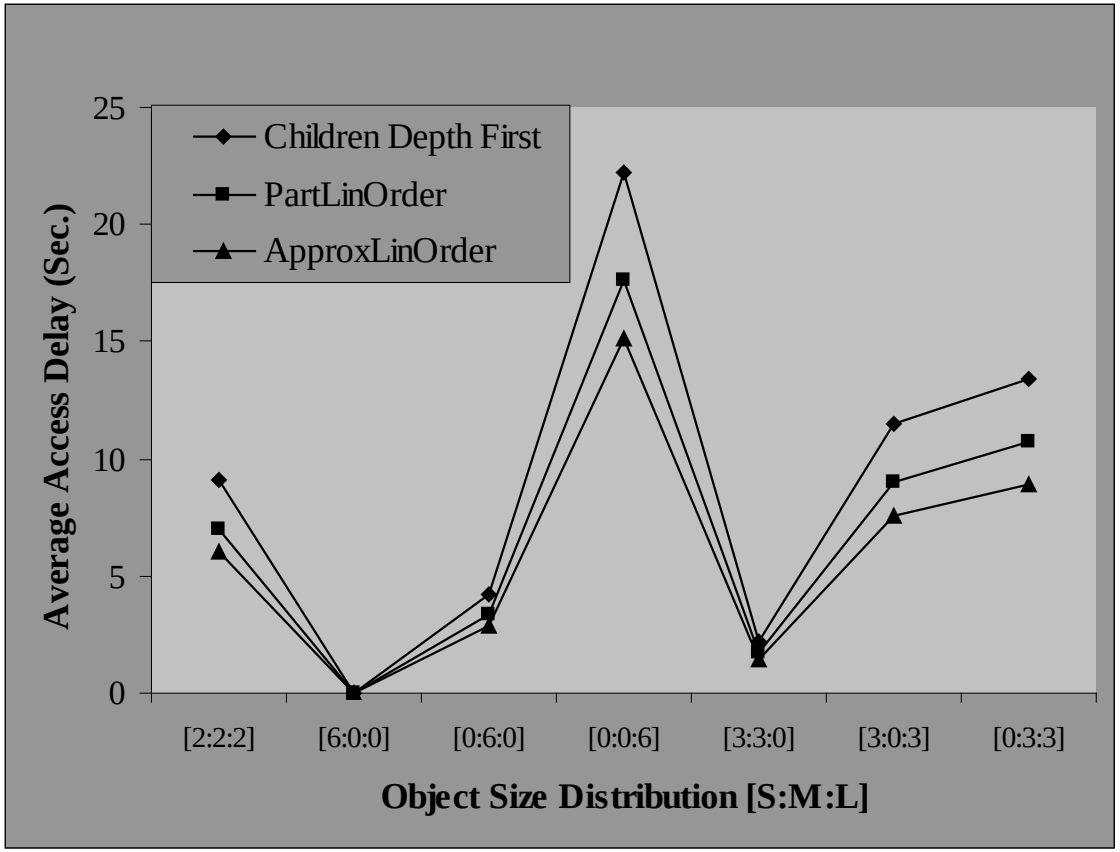
Broadcasting

- Object Organization on the Air Channel



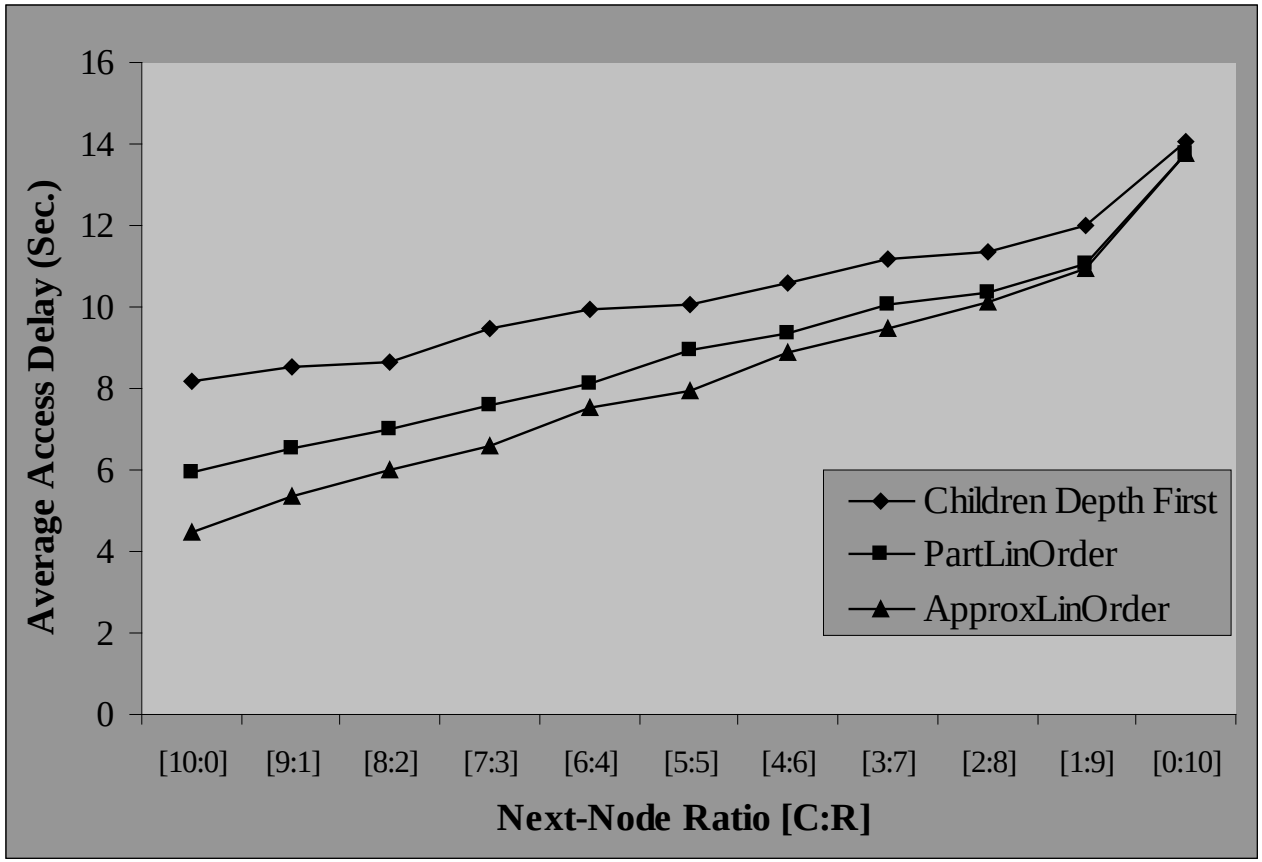
Broadcasting

- Object Organization on the Air Channel



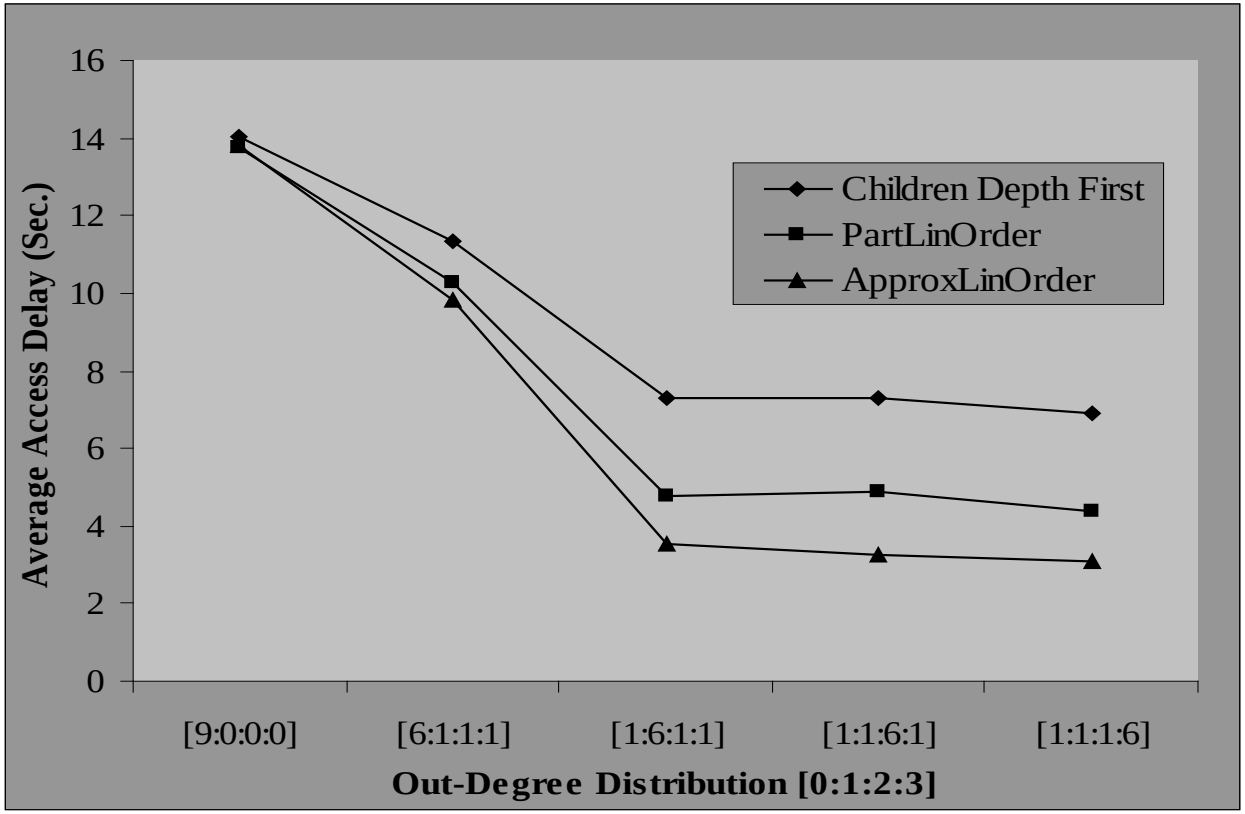
Broadcasting

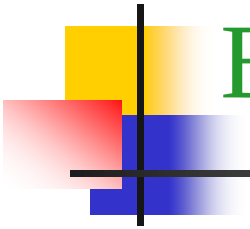
- Object Organization on the Air Channel



Broadcasting

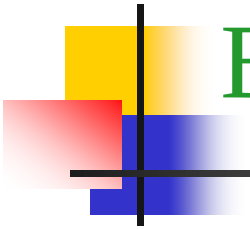
- Object Organization on the Air Channel





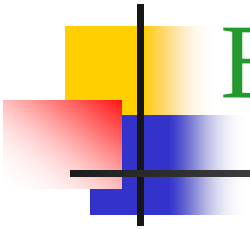
Broadcasting

- Object Organization on Parallel Channels
  - The broadcast length is a factor that affects the average response time in retrieving objects from the air channel.
  - The broadcast length can be reduced if objects are broadcast along the parallel air channels.
  - Reduced average response time could also translate into lower power consumption.



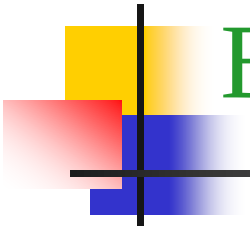
Broadcasting

- Object Organization on Parallel Channels
  - Formally we are intended to assign the objects from a weighted DAG onto multiple channels, while:
    - Preserving dependency implied by the edges,
    - Minimizing the overall broadcast time (load balancing) and
    - Clustering related objects close to one another (improving the response time).



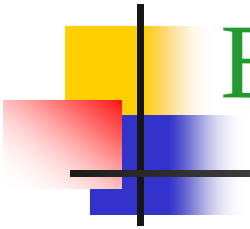
Broadcasting

- Object Organization on Parallel Channels
  - Assuming that all channels have the same data rate, one can draw many analogies between object organization on parallel channels and static task scheduling in a homogeneous multiprocessor environment:
  - Tasks are represented as a directed graph D ≡ (N, A), where nodes (N) and directed edges (A) representing processes and dependence among the processes, respectively.
  - An optimal solution to such a problem is proven to be NP hard and hence, in many solutions, heuristics have been developed to generate a “suitable solution”.



Broadcasting

- Object Organization on Parallel Channels
  - We are intended to assign the objects from a weighted DAG onto multiple channels, while:
    - Preserving dependency implied by the edges,
    - Minimizing the overall broadcast length (load balancing) and
    - Clustering related objects close to one another (improving the response time).
  - Largest Object First (LOF)
  - Clustering Critical-Path (CCP)

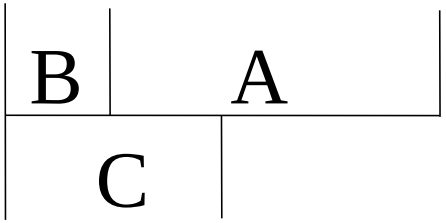


Broadcasting

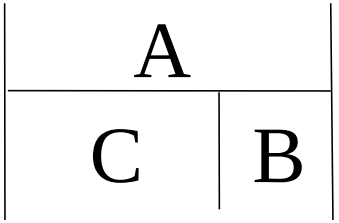
- Object Organization on Parallel Channels – Largest Object First
  - This algorithm relies on a simple and localized heuristic by giving priority to larger objects.
  - Consider a 2-channel environment and three objects A, B, and C with the following relationships among the sizes of the objects A > B > C.

Broadcasting

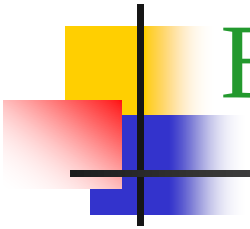
- Object Organization on Parallel Channels – Largest Object First



Random allocation

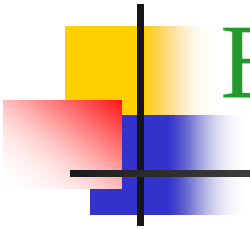


Allocation based on the Largest Object First Heuristic



Broadcasting

- Object Organization on Parallel Channels – Largest Object First
  - The algorithm follows the following procedure; Recursively, a "proper" node with in-degree of zero is chosen (initially the root) and assigned to a "proper" channel.
  - A “proper” channel is the one with the smallest overall size.
  - A “proper” node is the largest node with in-degree of zero.



Broadcasting

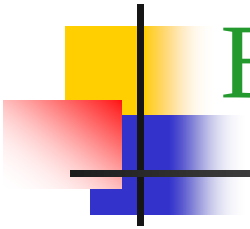
- Object Organization on Parallel Channels – Largest Object First
  - Largest Object First Algorithm

    Repeat until all nodes are assigned

    Assign a free node with the largest weight whose parents are fully allocated to the least-loaded channel

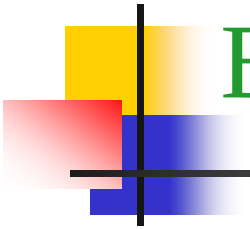
    Remove all out-edges of assigned node from the dag

    Insert resulting free nodes into the list of free nodes.



Broadcasting

- Object Organization on Parallel Channels – Clustering Critical-Path
  - AssignWeights(*DAG*) Algorithm
    - for every node *i*
      - Calculate W_i
  - Weight of each node is calculated based on the:
    - Size of the node,
    - Maximum weight of its children,
    - Number of children with in-degree 1 and in-degree > 1, and
    - Degree of connectivity among objects.

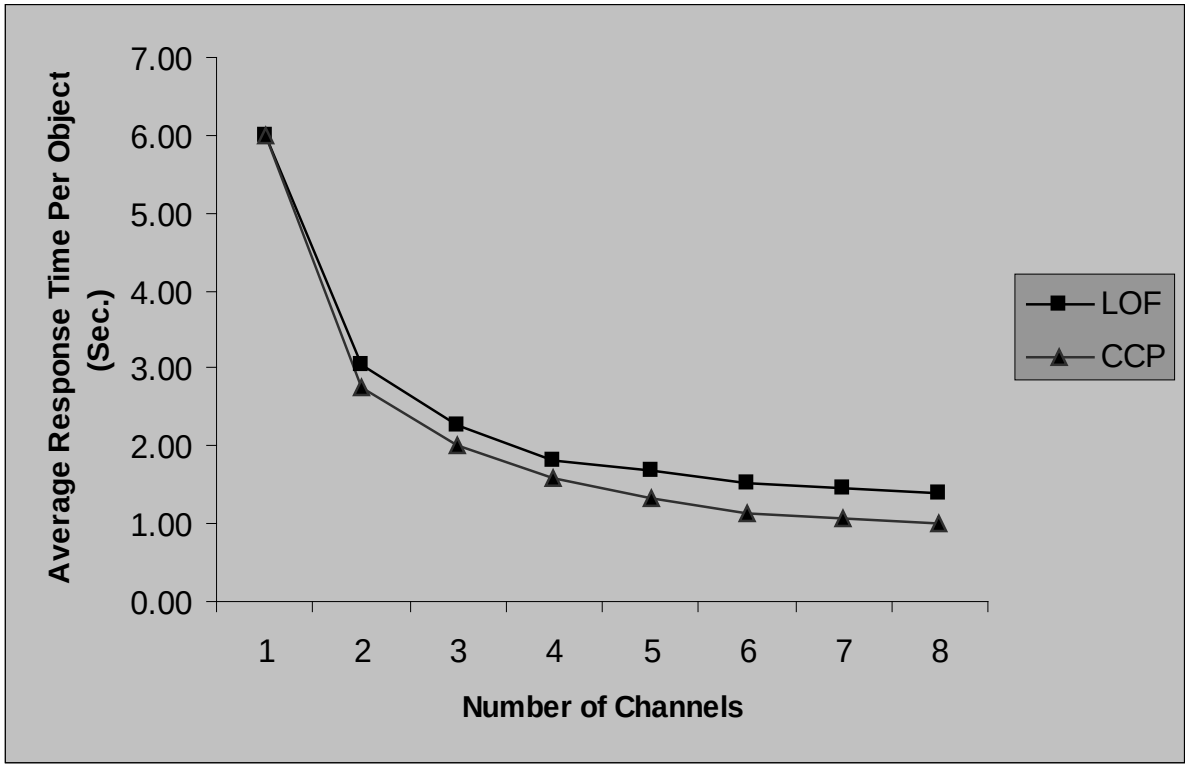


Broadcasting

- Object Organization on Parallel Channels – Clustering Critical-Path
  - Critical Path Algorithm
    - AssignWeights(*DAG*)
    - repeat until all the nodes have been processed
      - Select the free node *N* with the largest weight
      - if all parents of *N* are fully allocated on the channels
        - place it on the currently least-loaded channel
      - else
        - fill up the least-loaded channel(s) with nulls up to the end of the last allocated parent of *N* then place *N* on it.

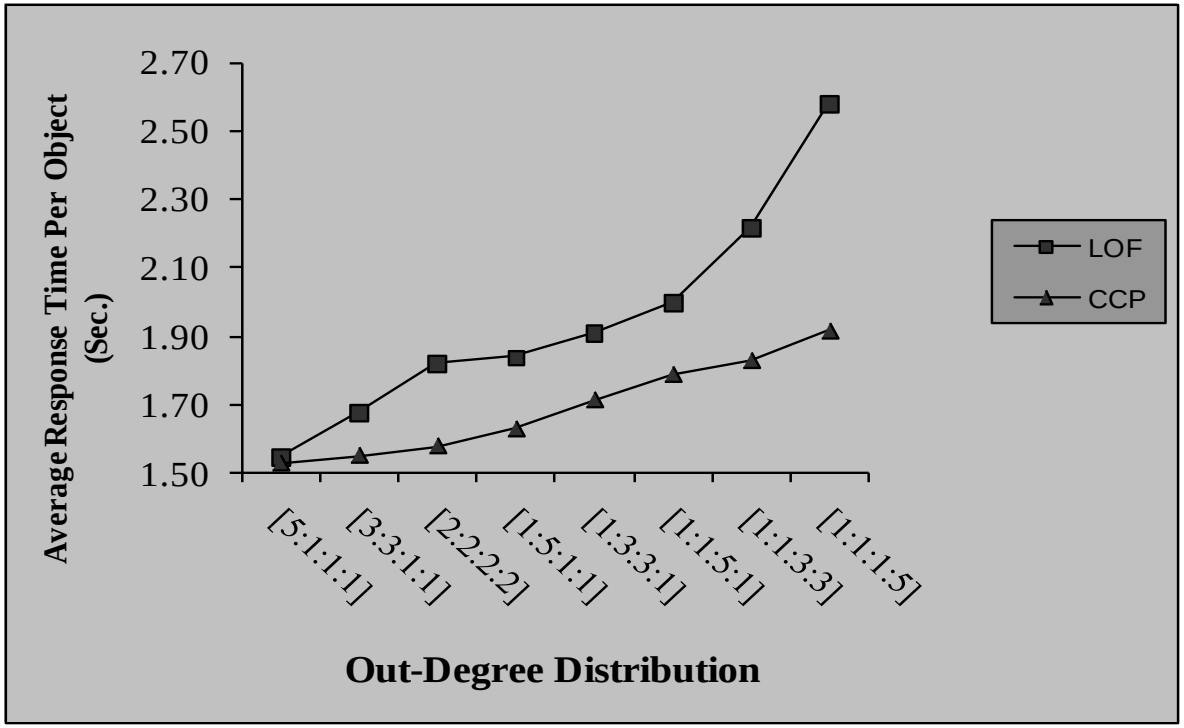
Broadcasting

- Object Organization on Parallel Channels



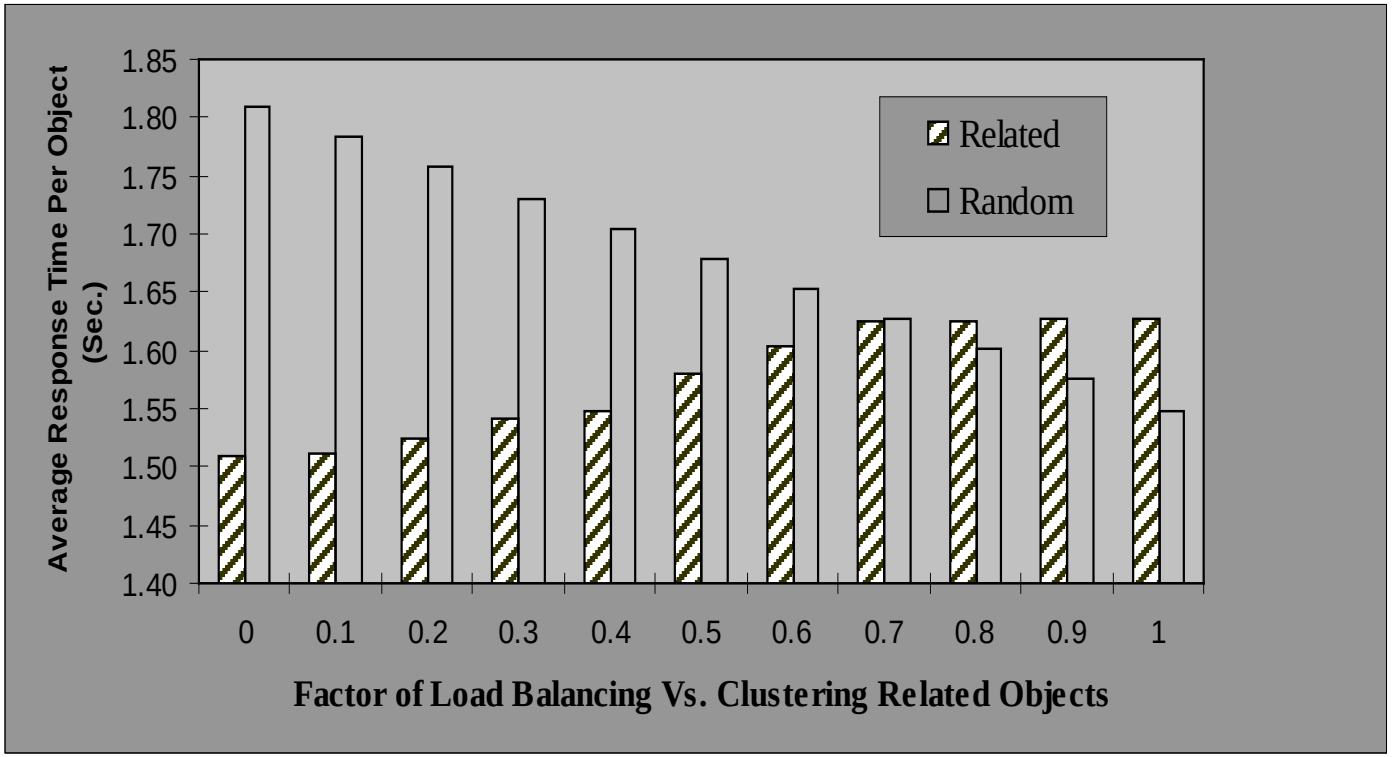
Broadcasting

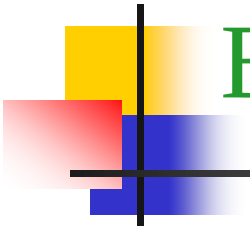
- Object Organization on Parallel Channels



Broadcasting

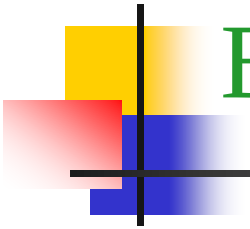
- Object Organization on Parallel Channels





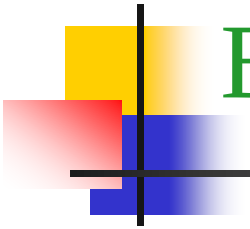
Broadcasting

- Object Organization on Parallel Channels
  - Object organization on parallel channels brings the issue of access conflicts among requested objects distributed along different channels. The access conflict is due to two factors:
    - The receiver at the mobile host can only tune into one channel at any given time, and
    - The time delay to switch from one channel to another.



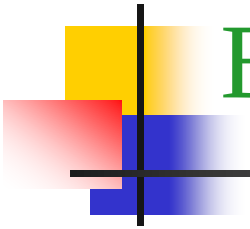
Broadcasting

- Object Organization on Parallel Channels
  - Access conflicts require the receiver to wait until the next broadcast cycle (s) to retrieve the requested information.
  - Naturally, multiple passes over the broadcast channels will have a significant adverse impact on the response time and power consumption.



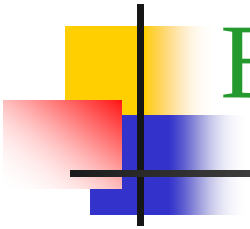
Broadcasting

- Object Organization on Parallel Channels – Conflict
  - A K-object request is an application request intended to retrieve *K* objects from a broadcast.
  - Assume that each channel has the same number of pages (frames) of equal length and, without loss of generality; each object is residing on only a single page.
  - A parallel broadcast channels can be modeled as an *N* x *M* grid, where *N* is the number of pages per broadcast, and *M* is the number of channels.
  - In this grid, *K* objects $0 < K \leq MN$ are randomly distributed throughout the *MN* positions of the grid.



Broadcasting

- Object Organization on Parallel Channels – Conflict
  - The mobile host can switch channels, but it takes time to do this.
  - Based on the common page size and the network speed, the time required to switch from one channel to another is equivalent to the time it takes for one page to pass in the broadcast.
  - Thus, it is impossible for the mobile unit to retrieve both the i*th* page on channel *A* and (i+1)*th* page on channel *B* (where A ≠ B).

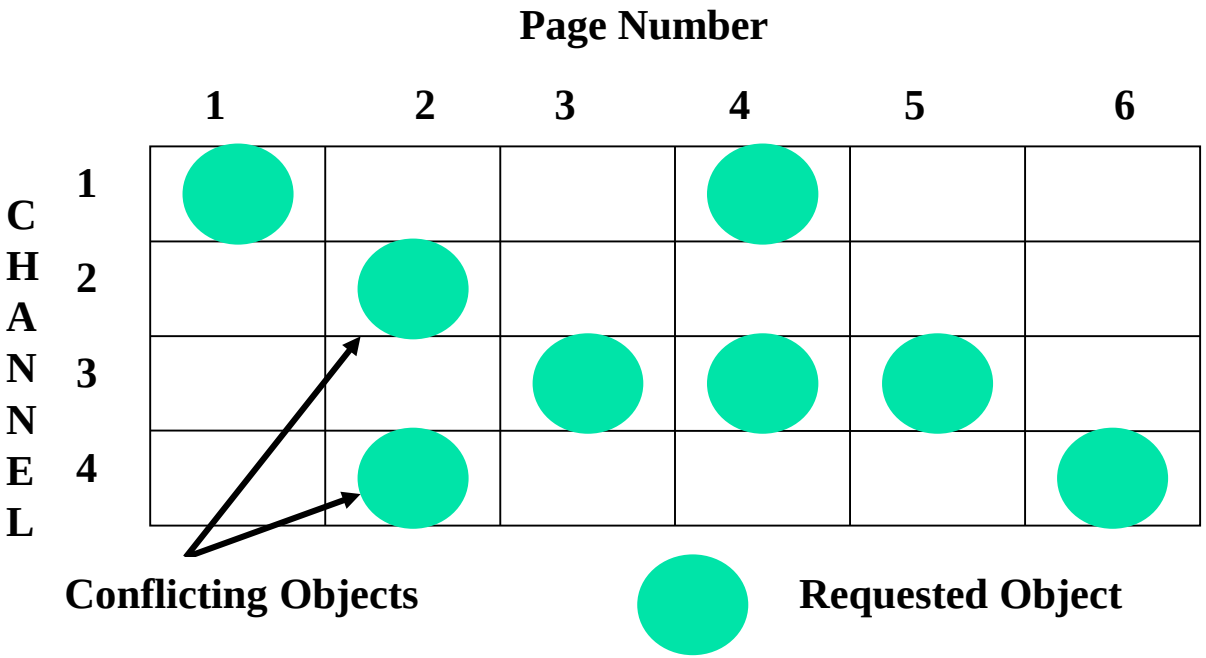


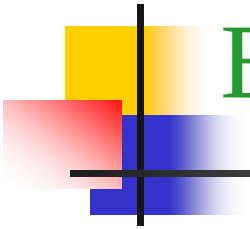
Broadcasting

- Object Organization on Parallel Channels – Conflict
  - Two objects are defined to be in conflict if it is impossible to retrieve both objects on the same broadcast cycle.

Broadcasting

- Object Organization on Parallel Channels – Conflict



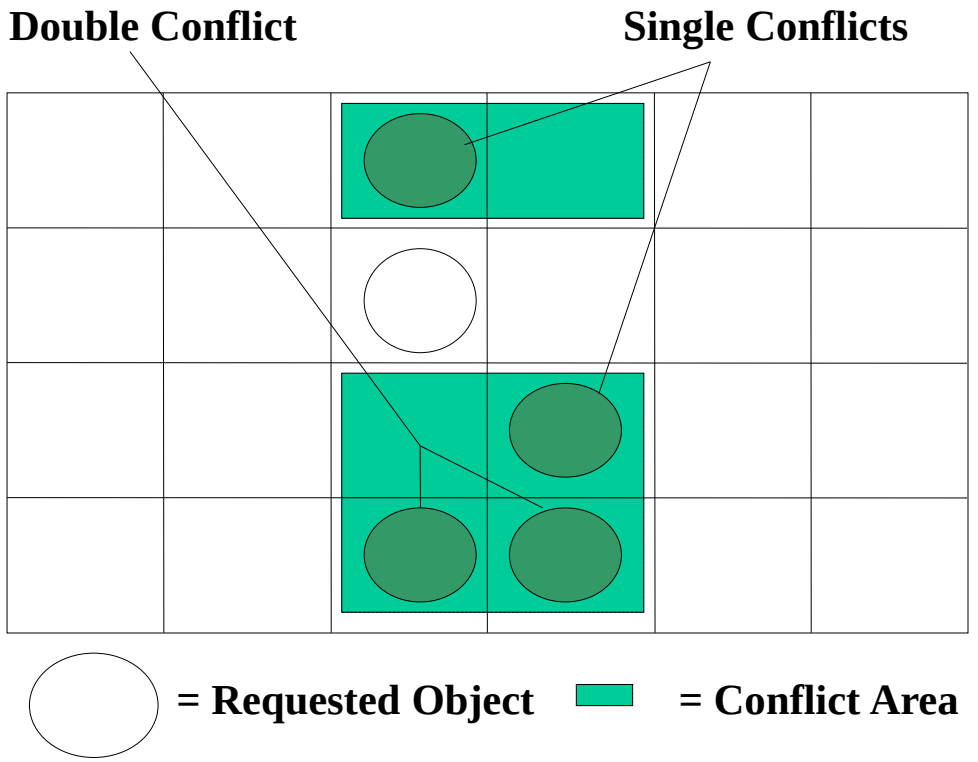


Broadcasting

- Object Organization on Parallel Channels – Conflict
  - For any particular object, all objects in the same or succeeding page (column) and on a different row (channel) will be in conflict.
  - Thus, for any specific page (object) in the grid, there are $2M - 2$ conflicting pages (objects) in the broadcast.
  - These positions are known as the conflict region.

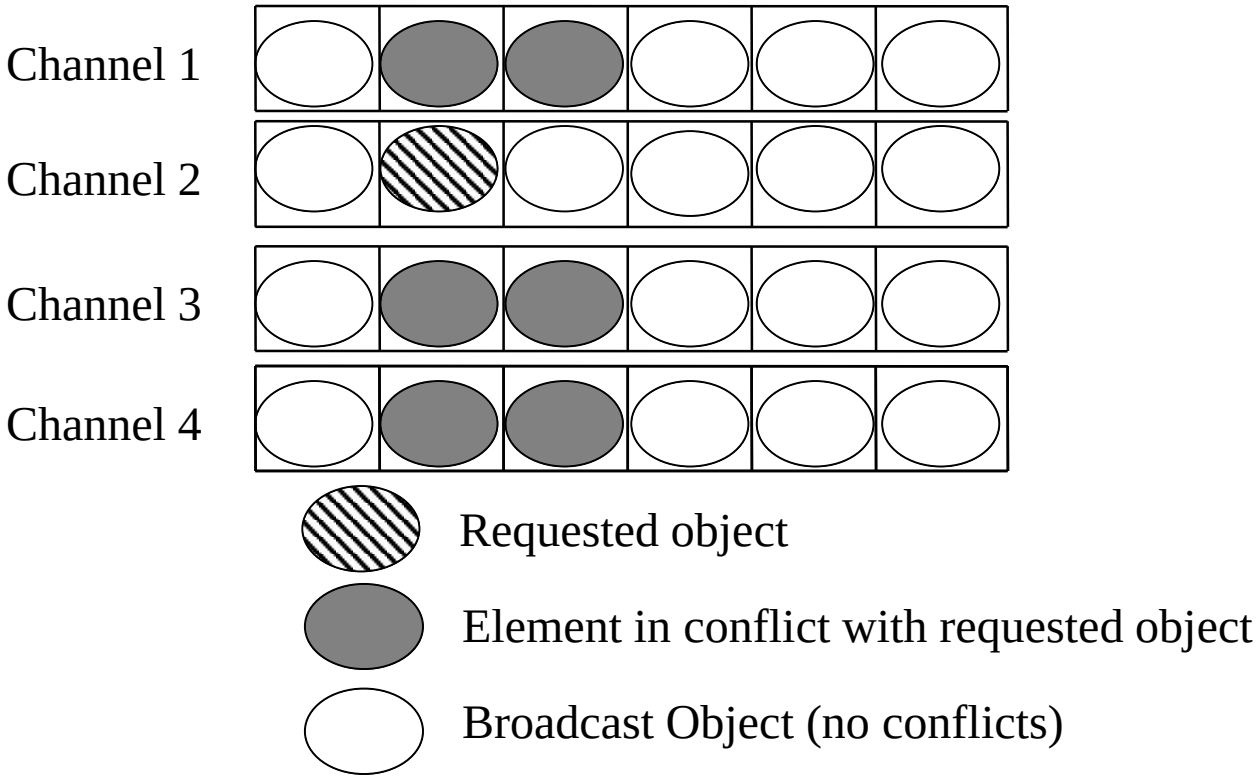
Broadcasting

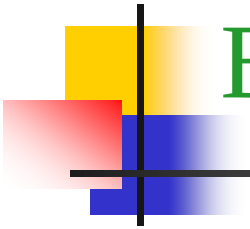
- Object Organization on Parallel Channels – Conflict



Broadcasting

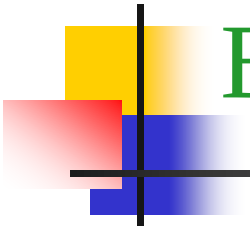
- Object Organization on Parallel Channels – Conflict





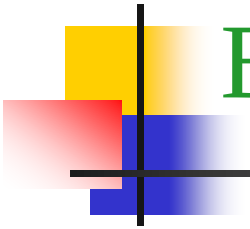
Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - As noted before, conflicts affect the access latency, hence the response time, and power consumption.
  - To reduce the impact of conflicts on the access time and power consumption, retrieval procedures should be enhanced by a scheduling protocol that determines object retrieval sequence during each broadcast cycle.



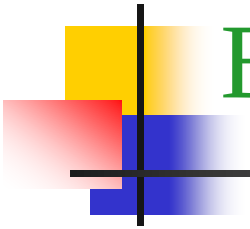
Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - The scheduling protocol is based on the following three prioritized conditions:
    - Eliminate the number of conflicts.
    - Retrieve the maximum number of objects.
    - Minimize the number of channel switches.
  - The scheme determines the order of retrieval utilizing a forest — an access forest.
  - An access forest is a collection of trees (access trees), where each access tree represents a collection of access patterns during a broadcast cycle. Naturally, the structure of the access forest, i.e., the number of trees and the number of children that any parent can have, is a function of the number of broadcast channels.



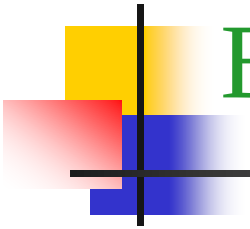
Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - An access tree is composed of two elements:
    - Nodes and
    - Arcs.
  - Each access tree in the access forest has a different node as a root — the root is the first accessible requested object on a broadcast channel.



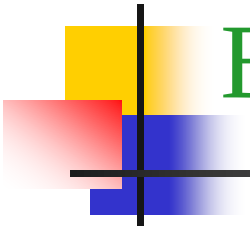
Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - A node represents a requested object. The nodes are labeled to indicate its conflict status:
    - Mnemonically, "C_1" represents if the object is in conflict with another object(s) in the broadcast; and "C_0" indicates the lack of conflict.
  - The arcs of the trees are weighted arcs. A weight denotes whether or not channel switching is required to retrieve the next scheduled object in the access pattern.
  - A branch in a tree represents a possible access pattern of objects during a broadcast cycle with no conflicts.
  - Starting from the root, the total number of branches in the tree represents all possible access patterns during a broadcast cycle.



Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - Generation of the access forest
  - For each broadcast cycle determine best access patterns based on the following priority list:
    - Eliminate the number of conflicts,
    - Retrieve the maximum number of objects, and
    - Minimize the number of channel switches.

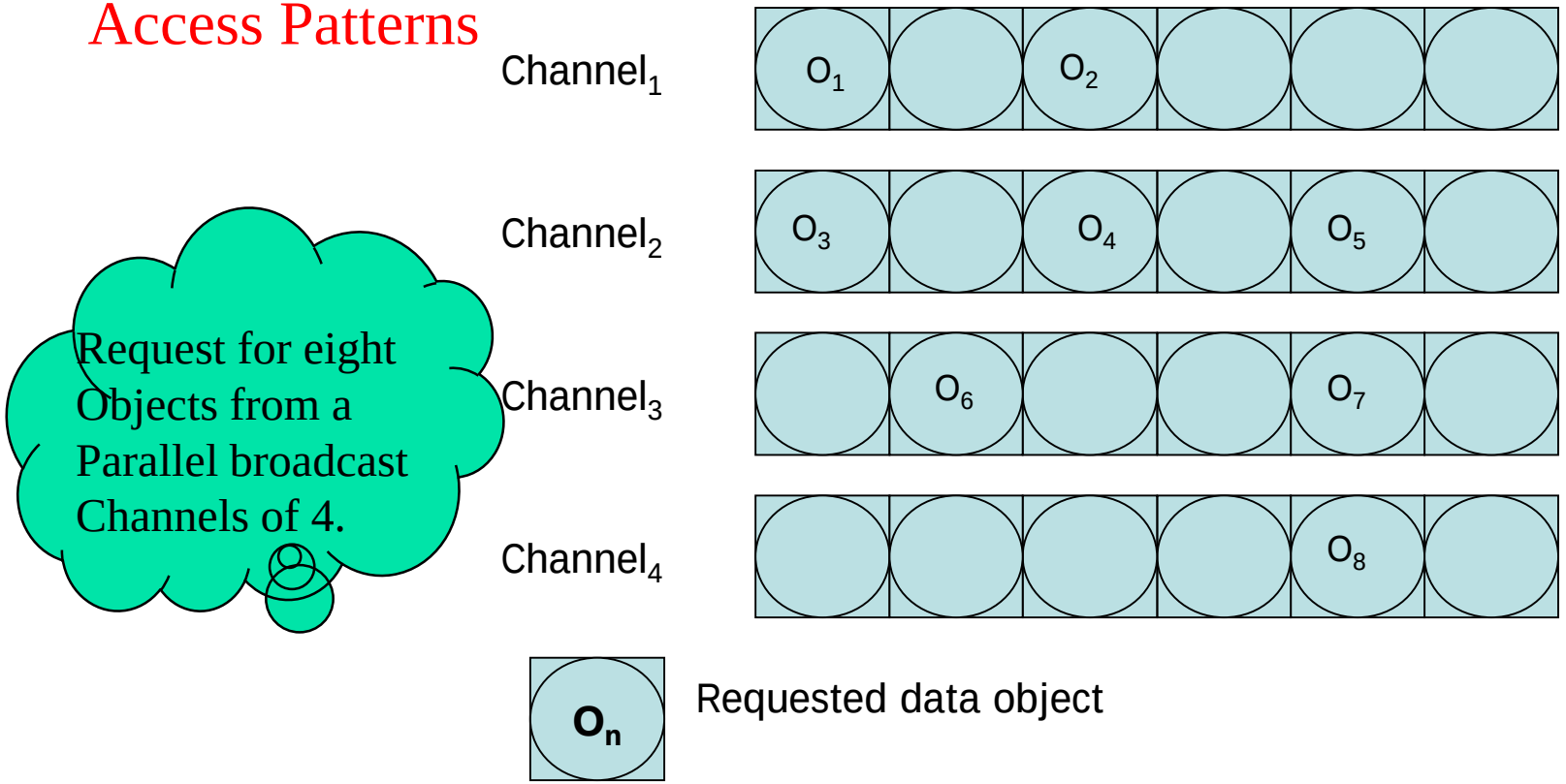


Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - Search: Based on the user's query, this step determines the offset and the channel number of the requested objects on the broadcast channels.

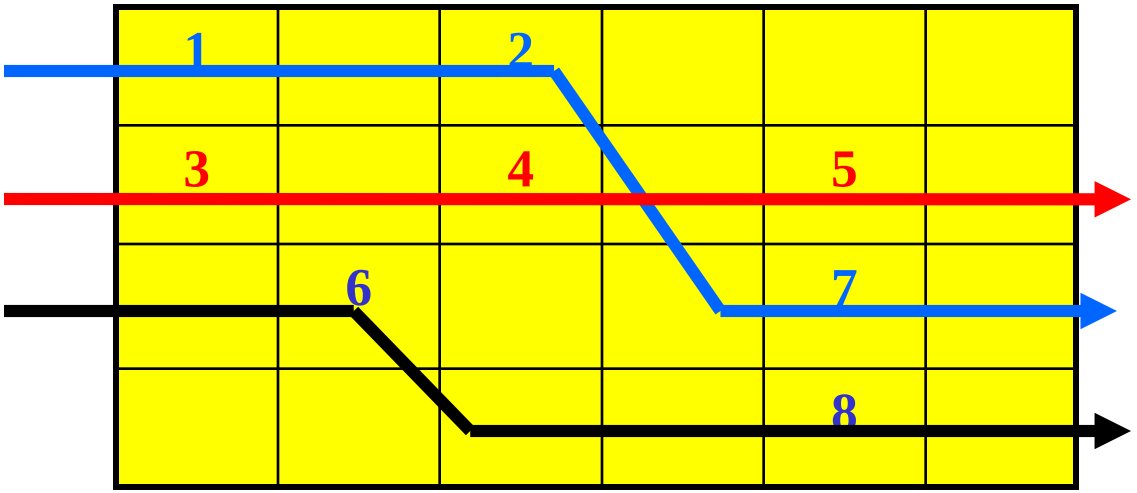
Broadcasting

- Object Organization on Parallel Channels – Access Patterns



Broadcasting

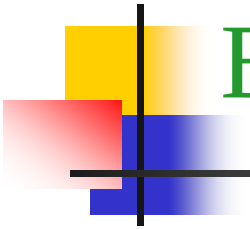
- Object Organization on Parallel Channels – Access Patterns



8 # of requested objects

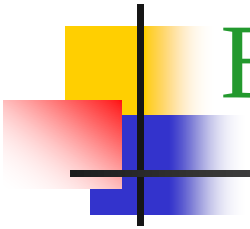
4 # of parallel channels

6 the length of a broadcast cycle (# of objects)



Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - Generation of access patterns
    - The Traveling Salesman
    - Next Object
    - Row Scan
    - The proposed Heuristic approach



Broadcasting

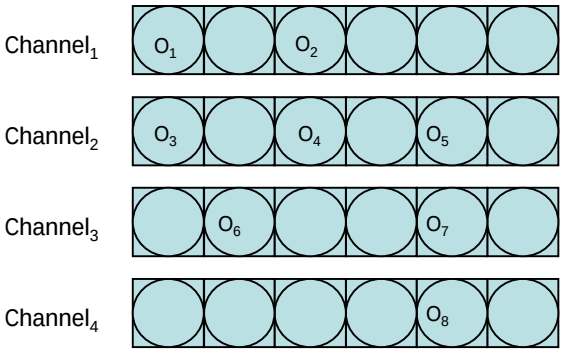
- Object Organization on Parallel Channels – Access Patterns
  - Generation of the access forest: For each broadcast channel search for the requested object with the smallest offset (these objects represent the roots of access tree). For the example, the objects with the smallest offsets are O_1, O_3, O_6 and O_8. Note that the number of access trees is upper bounded by the number of broadcast channels.
  - Root assignment: For each channel with at least one object requested, generate a tree with root node as determined in previous step. The roots are temporarily tagged as "C_0".

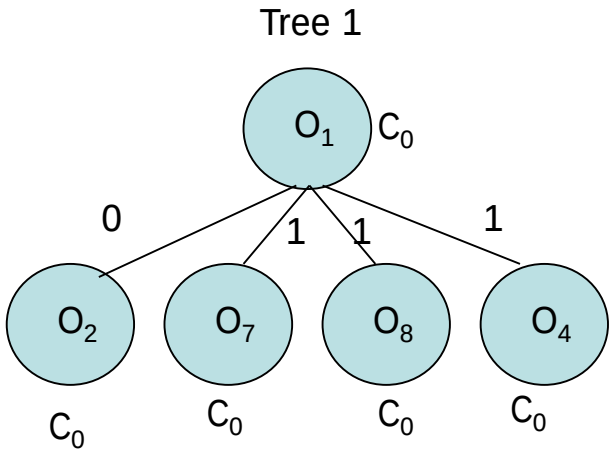
Broadcasting

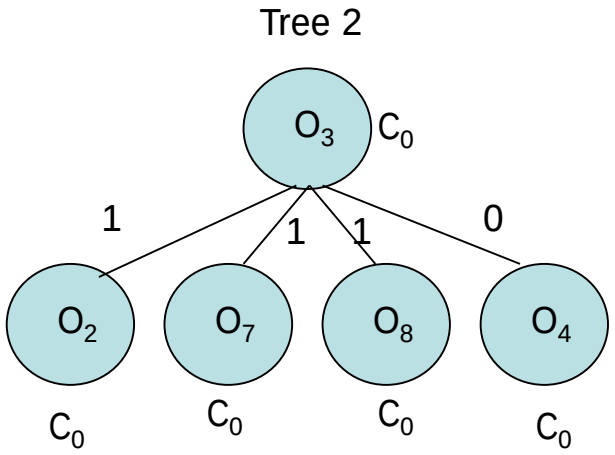
- Object Organization on Parallel Channels – Access Patterns
  - Child assignment: Once the roots are determined, it is necessary to select the child or children of each rooted access tree.
  - For each root, and relative to its position on the air channel, the algorithm determines the closest non conflicting objects on each channel. With respect to an object $O_{i,x}$ at location X on air channel i $(1 \leq i \leq n)$ the closest non conflicting object is either the object $O_{i,x+1}$ or the object $O_{j,x+2}$ $j \neq i$. If the child is in the same broadcast channel as the root, the arc is weighted as "0"; otherwise it is weighted as "1". Each added node is temporarily tagged as "C_0".

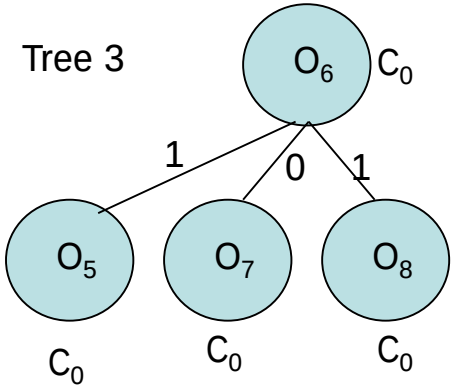
Broadcasting

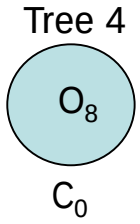
- Object Organization on Parallel Channels – Access Patterns



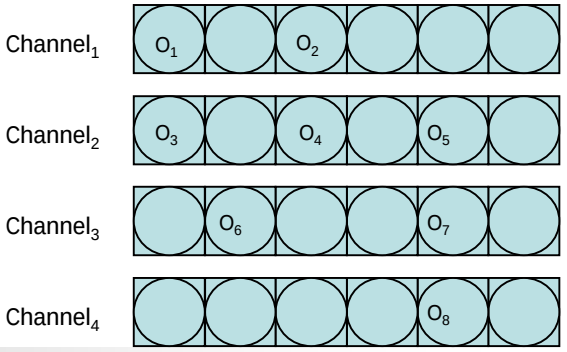




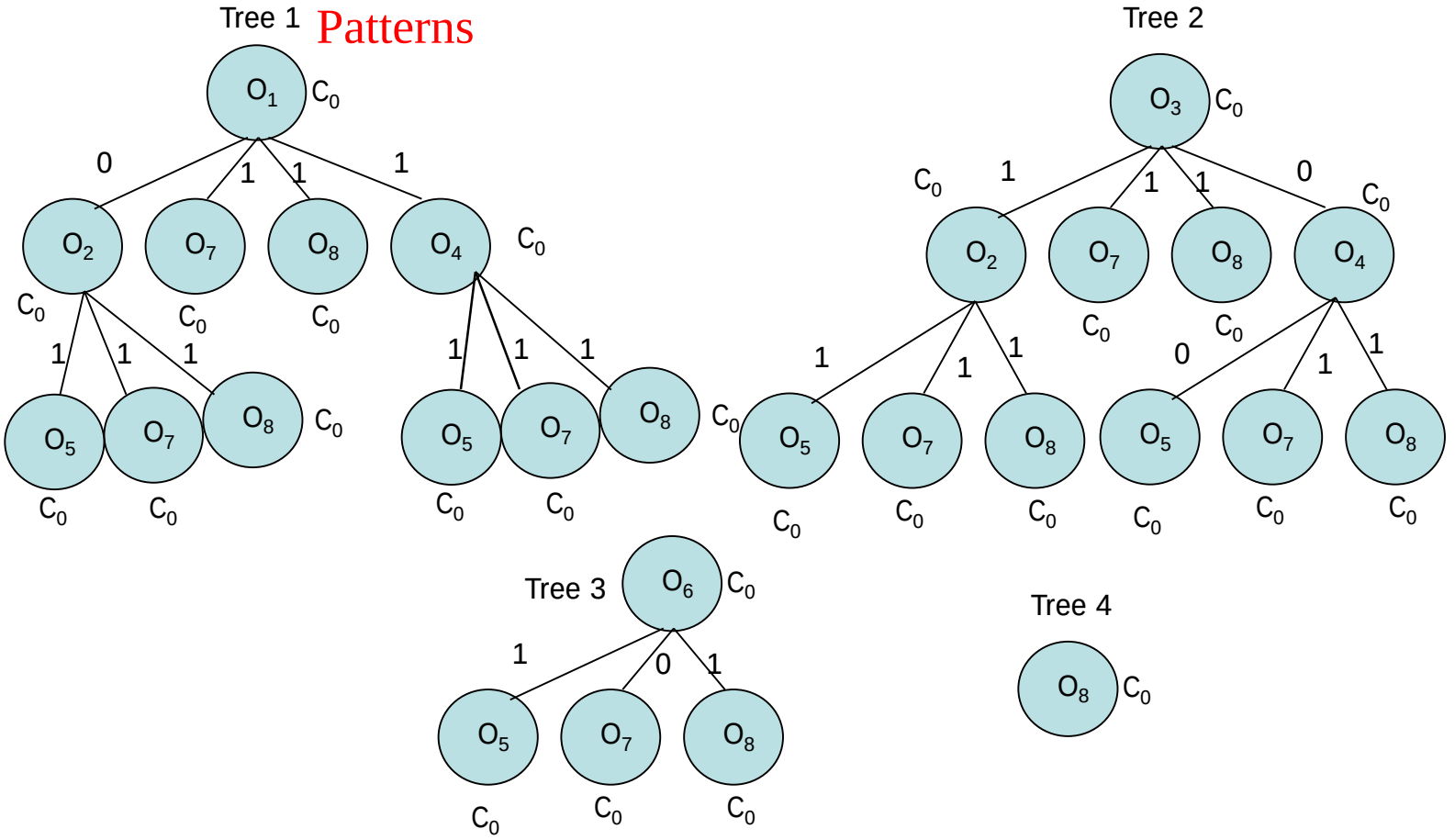


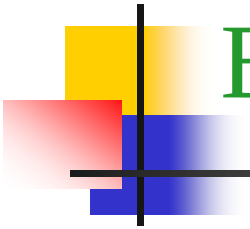


Broadcasting



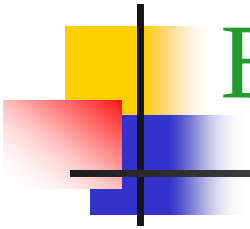
- Object Organization on Parallel Channels – Access Patterns





Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - Root label update: Once the whole set of requested objects is analyzed and the access forest is generated, the conflict labels of the nodes of each tree are updated. This process starts with the root of each tree. If a root is in conflict with any other root(s) a label of "C_1" is assigned to all the roots involved in the conflict, otherwise the preset value of "C_0" is maintained.



Broadcasting

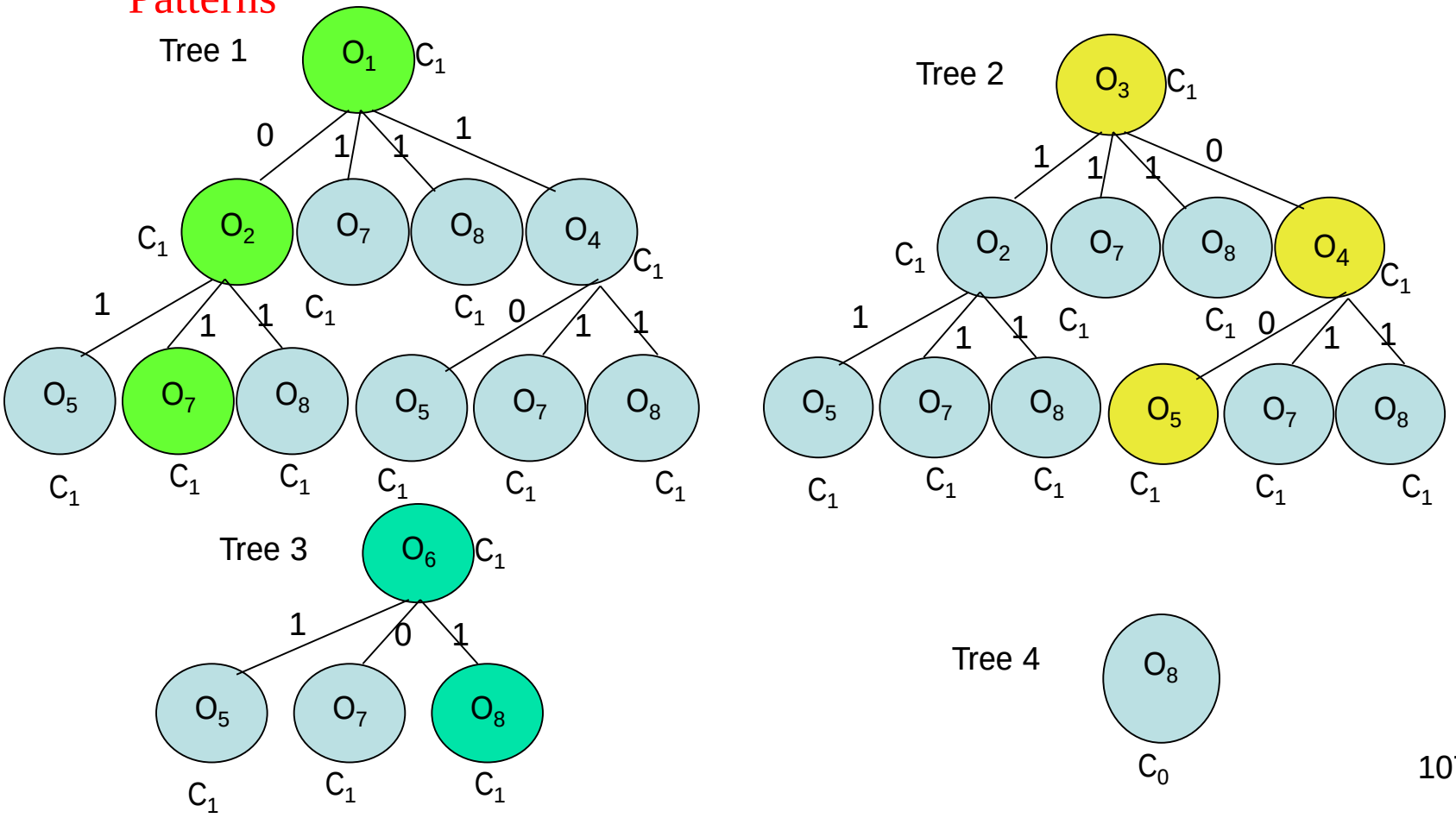
- Object Organization on Parallel Channels – Access Patterns
  - Node label update: Previous step will be applied to the nodes in the same level of each access tree in the access forest.
  - As in the previous step a value of "C_1" is assigned to the nodes in conflict.

Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - Sequence selection: The generation of the access forest then allows the selection of the suitable access patterns in an attempt to reduce the network latency and power consumption. A suitable access pattern is equivalent to the selection of a tree branch that:
    - Has the most conflicts with other branches,
    - Allows more objects to be pulled off the air channels, and
    - Requires the least number of channel switches.

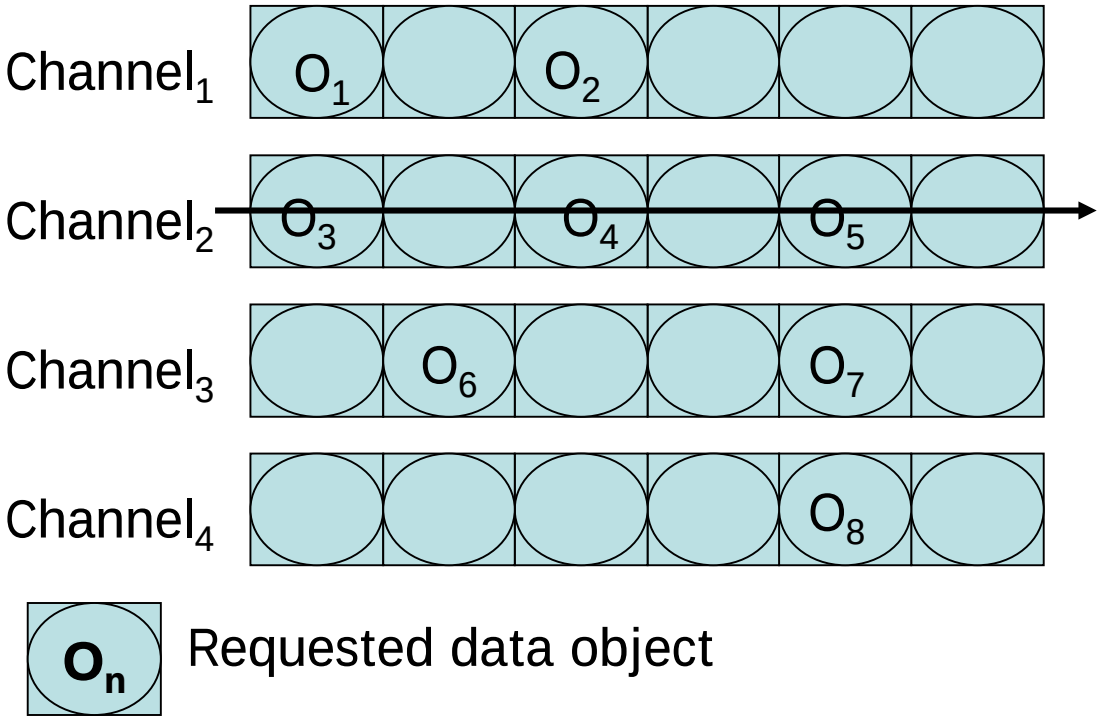
Broadcasting

- Object Organization on Parallel Channels – Access Patterns



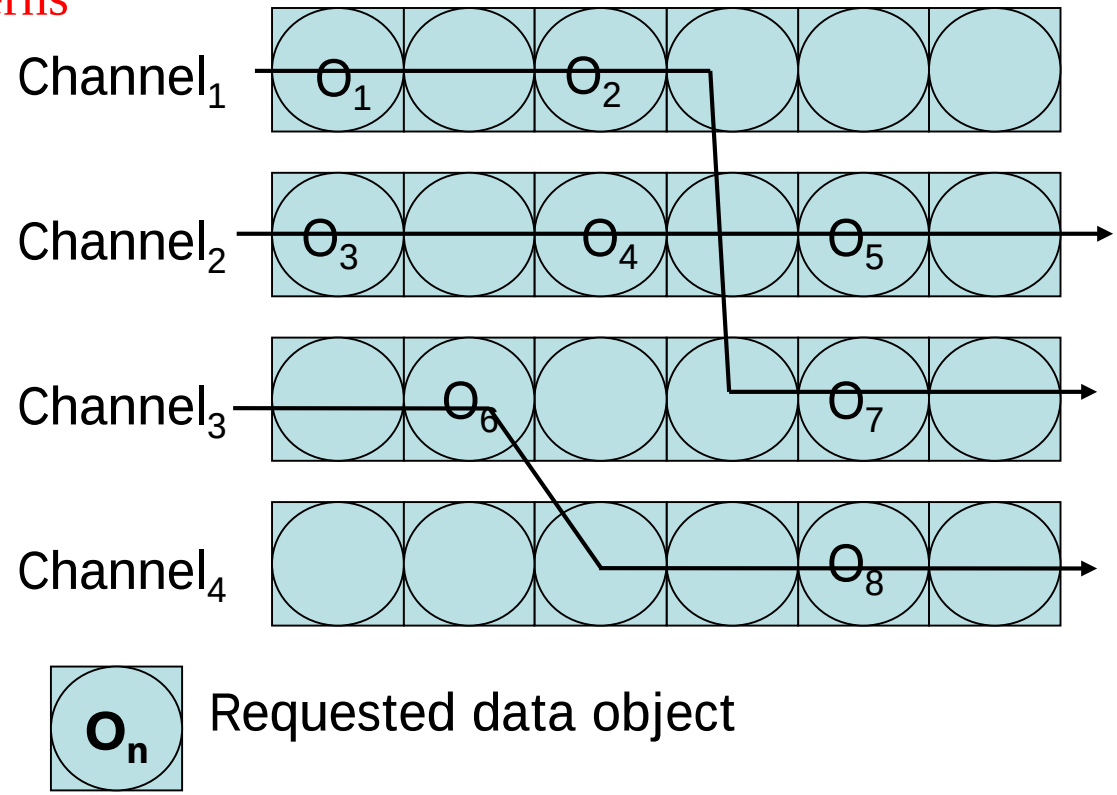
Broadcasting

- Object Organization on Parallel Channels – Access Patterns



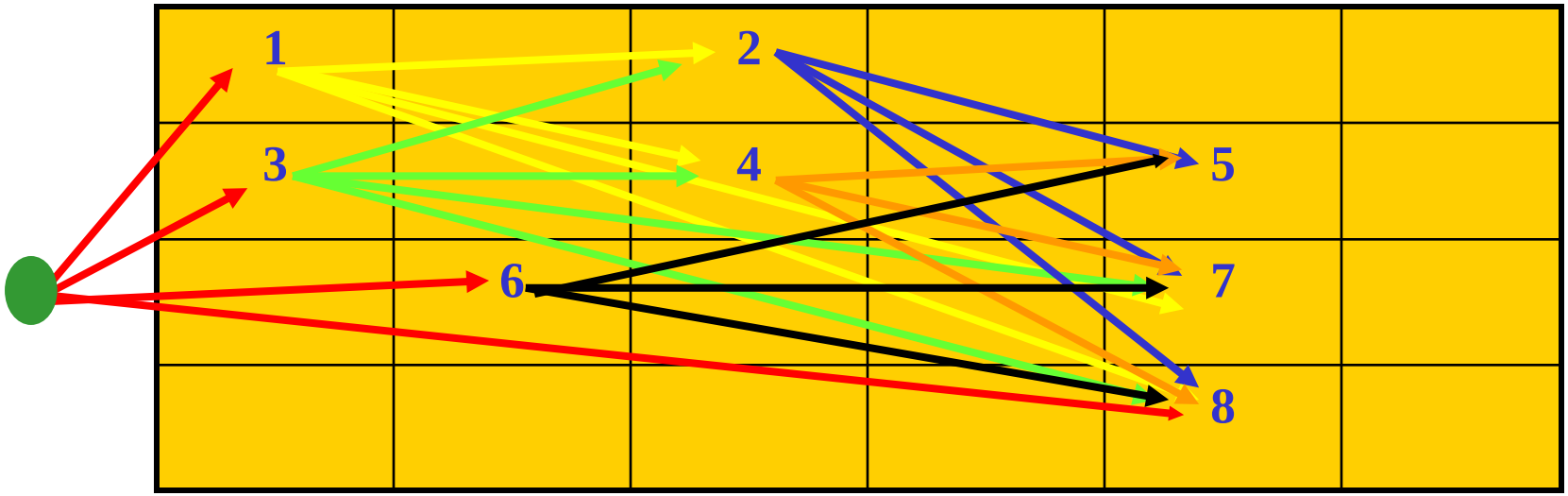
Broadcasting

- Object Organization on Parallel Channels – Access Patterns



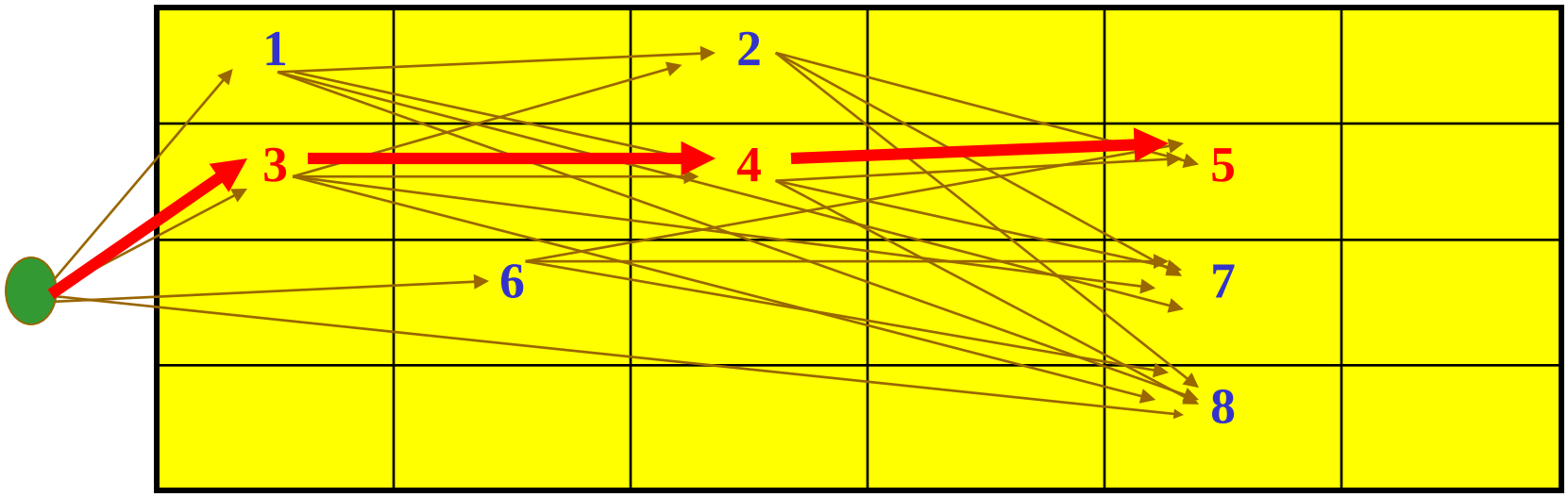
Broadcasting

- Object Organization on Parallel Channels – Access Patterns



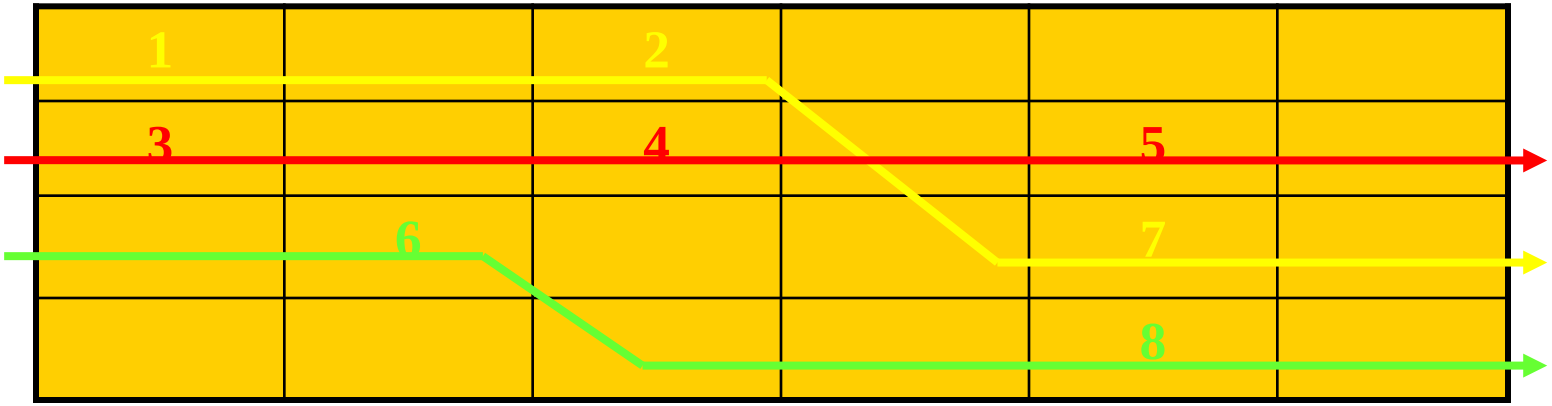
Broadcasting

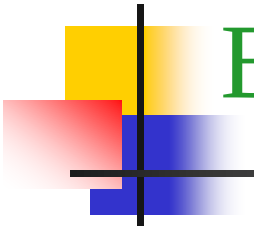
- Object Organization on Parallel Channels – Access Patterns



Broadcasting

- Object Organization on Parallel Channels – Access Patterns





Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - A simulator was developed to evaluate the feasibility and effectiveness of the proposed algorithm.
  - Simulation results showed that:
    - The proposed algorithm reduces both the number of passes and the response time compared to the Row Scan algorithm.
    - The energy consumption was also reduced, but only when the number of objects retrieved was approximately 15 or less.

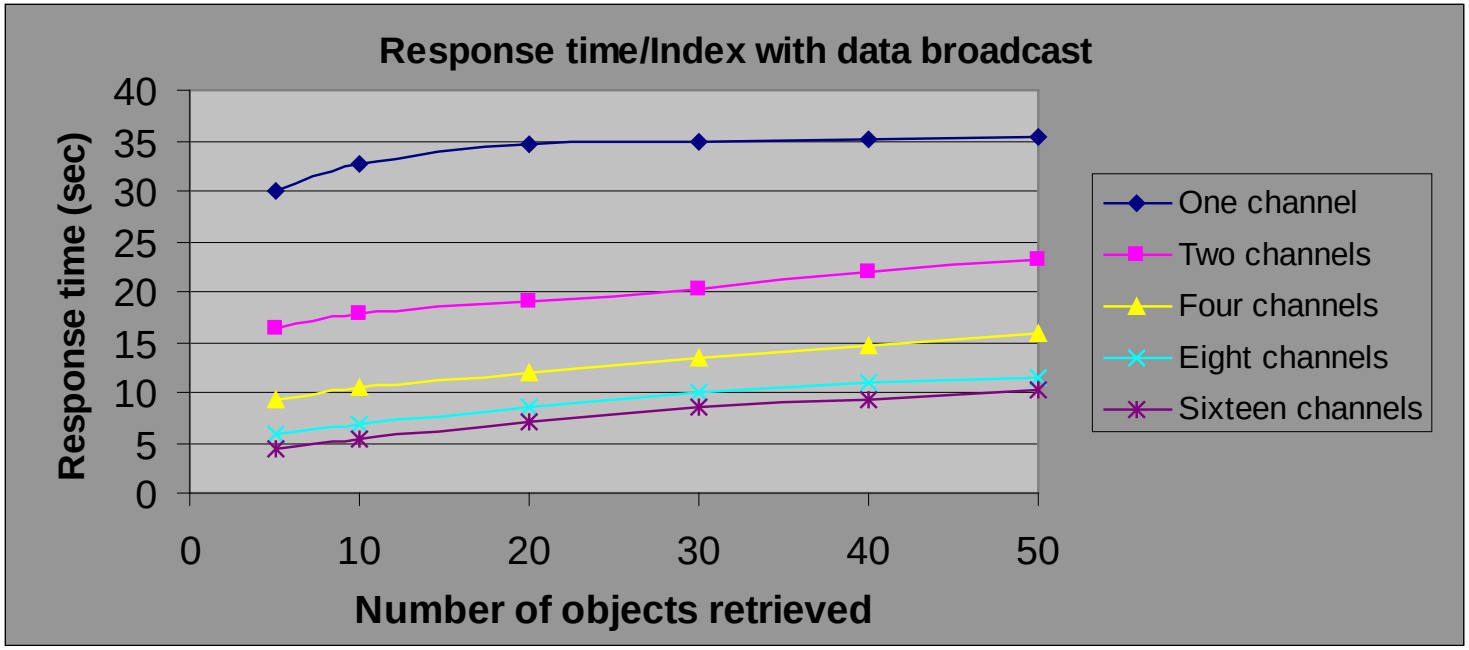
Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - In an environment composed of 16 channels when requesting 10 objects, the proposed algorithm performed the retrieval with:
    - 72% fewer passes,
    - 41% reduced response time, and
    - 3% less energy

  than that of the Row Scan algorithm

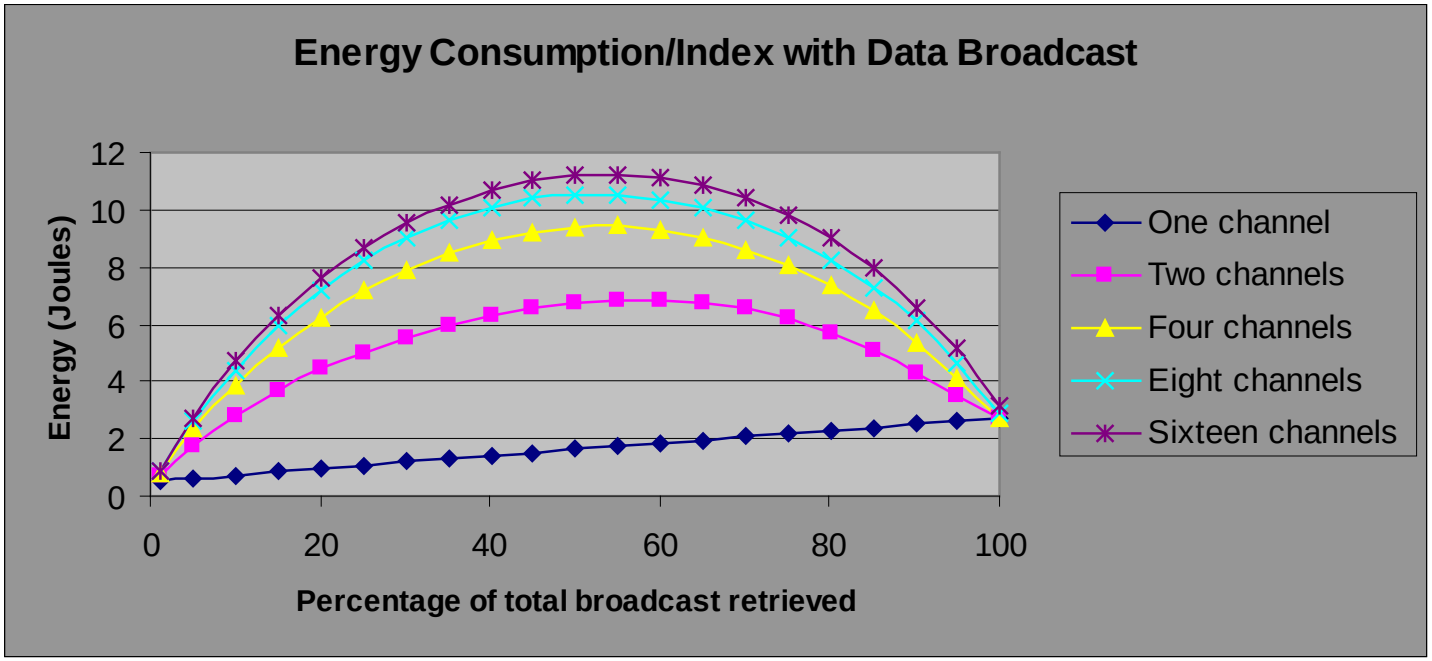
Broadcasting

- Object Organization on Parallel Channels – Access Patterns



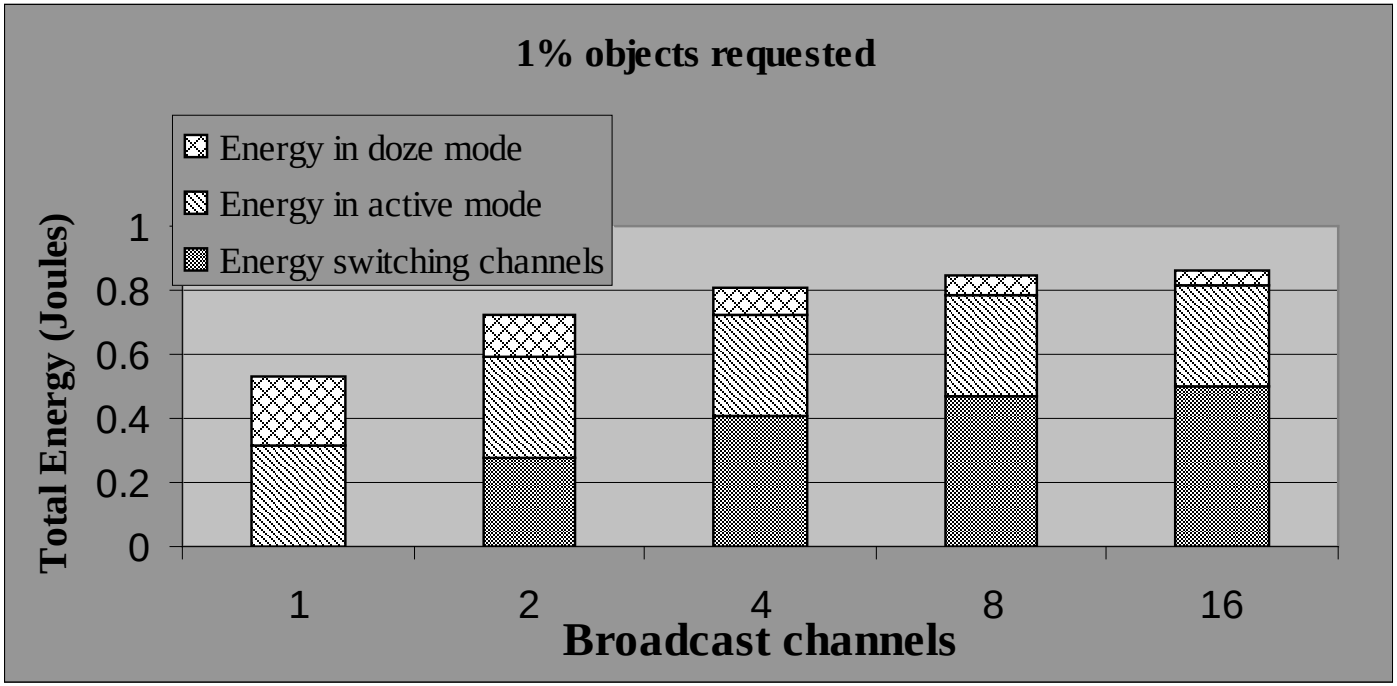
Broadcasting

- Object Organization on Parallel Channels – Access Patterns



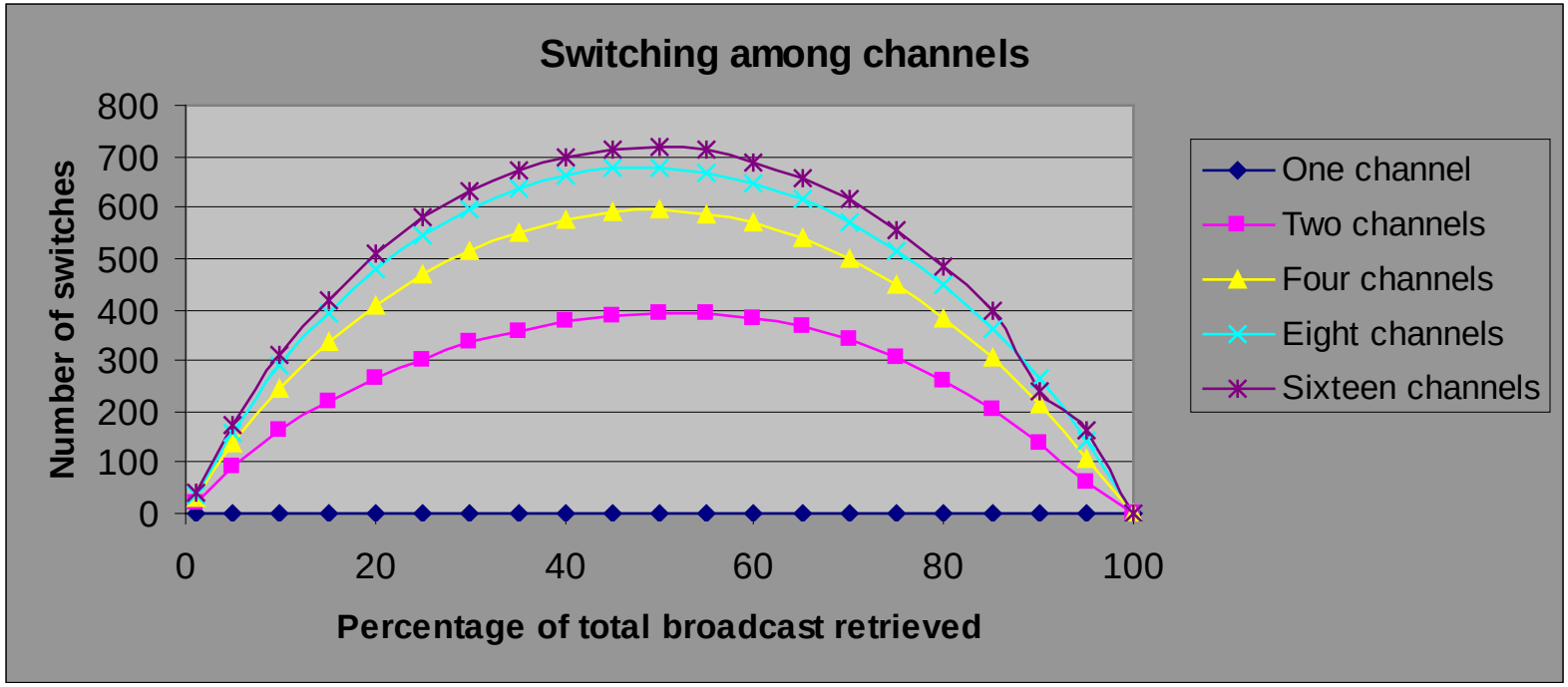
Broadcasting

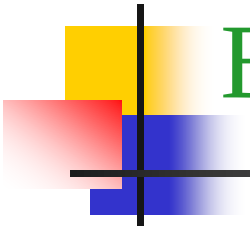
- Object Organization on Parallel Channels – Access Patterns



Broadcasting

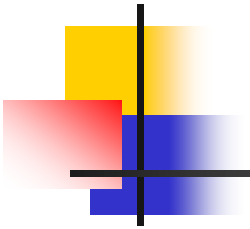
- Object Organization on Parallel Channels – Access Patterns





Broadcasting

- Object Organization on Parallel Channels – A New Approach
  - Artificial intelligence techniques (Least-cost path), in a top down fashion, is used to generate optimal access patterns for the requested data based on the following priority list of heuristics:
    - Eliminate the number of conflicts,
    - Minimize the number of channel switches, and
    - Retrieve the maximum number of objects.



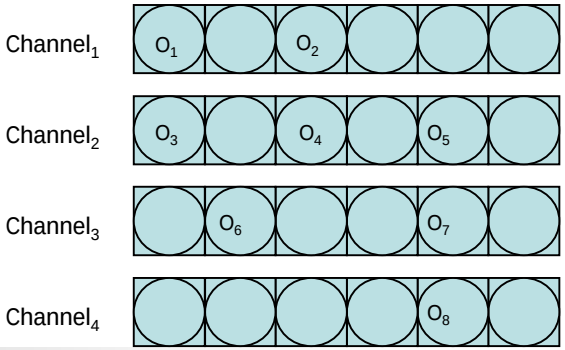
Broadcasting

- Object Organization on Parallel Channels – A New Approach
  - The temporal and spatial information provided by the index is used to plot an access pattern for the requested objects.
  - The index is used to generate a top-down best-fit weighted tree (Least-cost path) consisting of nodes as data objects and arcs connecting them:
    - At each level a cost is associated to each node, and
    - Only nodes that provide a potential least-cost path are expanded.

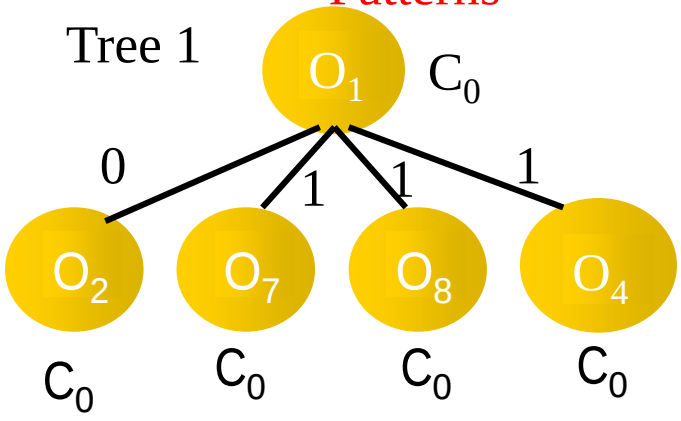
Broadcasting

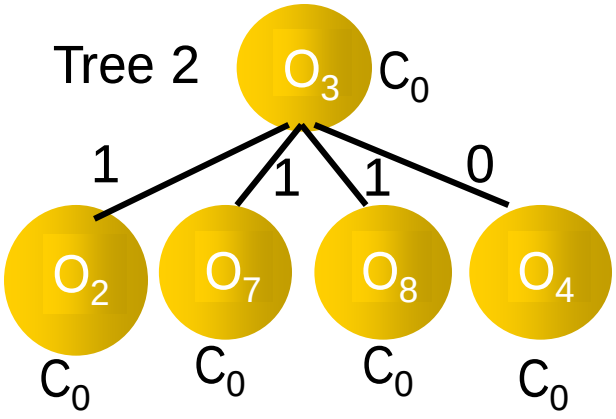
- Object Organization on Parallel Channels – A New Approach
  - Probe the index - To determine the channel number and offset of each requested item
  - Generate Access Forest - For each channel, find the first object to be broadcast
  - Root Node Assignment
  - Children Assignment - For each node, find the closest non-conflicting objects that may be accessed and label the arcs as before
    - Node Label Update - After the generation of the children, they are examined and assessed for conflicts
    - Cost Evaluation - The children are examined to determine the least-cost path of the tree
  - Expansion - The node with the lowest switching cost
  - Repeat Expansion
  - Compare - If more than one least-cost paths exist for each tree, the following heuristics are used to determine the most ideal path:
    - Choose a path that eliminates the most number of conflicts
    - Choose a path with the least channel switches
    - Choose a path that retrieves the most number of objects

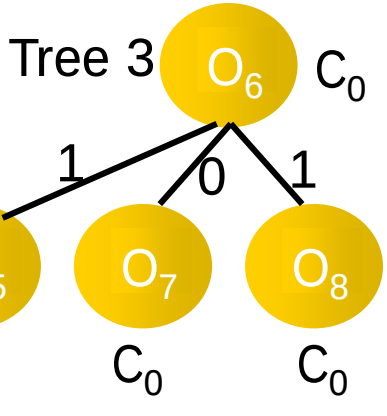
Broadcasting



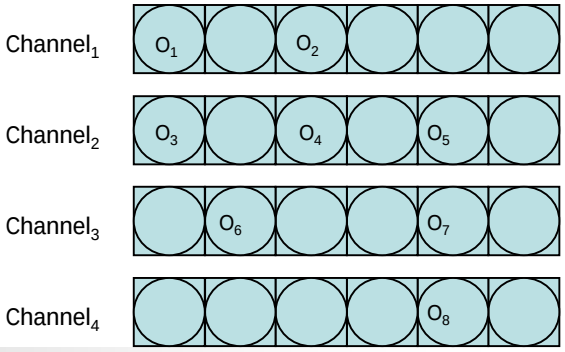
- Object Organization on Parallel Channels – Access Patterns



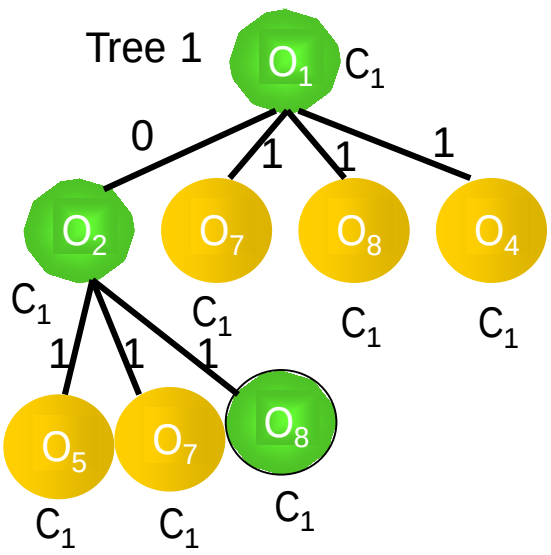


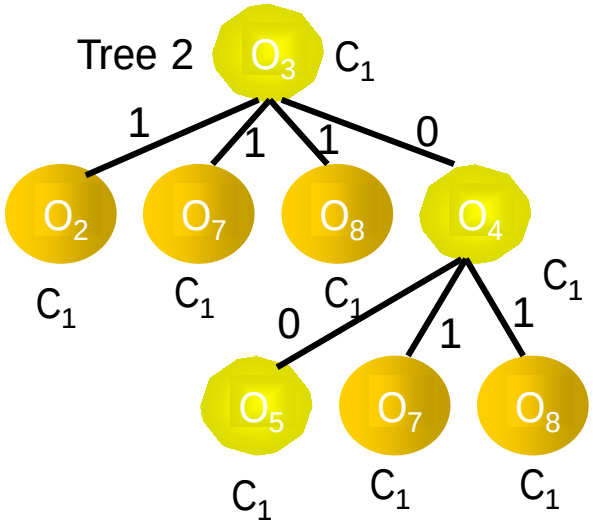


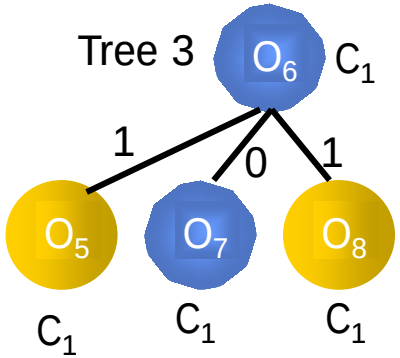
Broadcasting

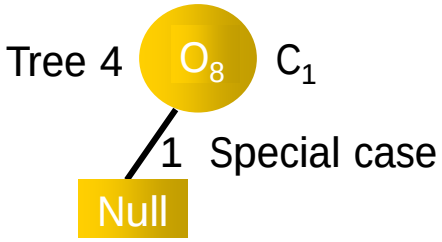


- Object Organization on Parallel Channels – Access Patterns



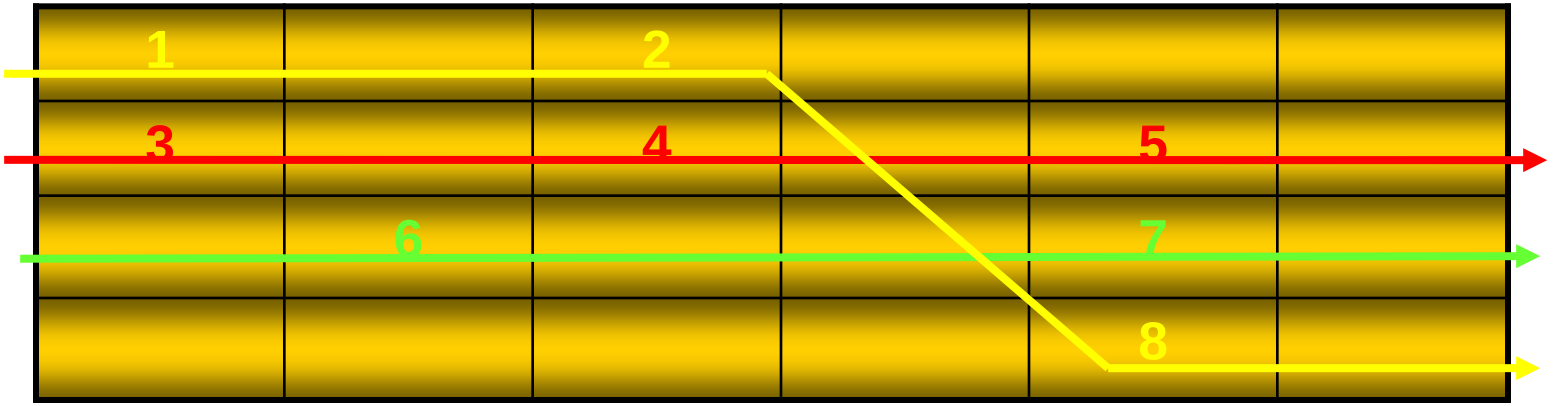






Broadcasting

- Object Organization on Parallel Channels – Access Patterns

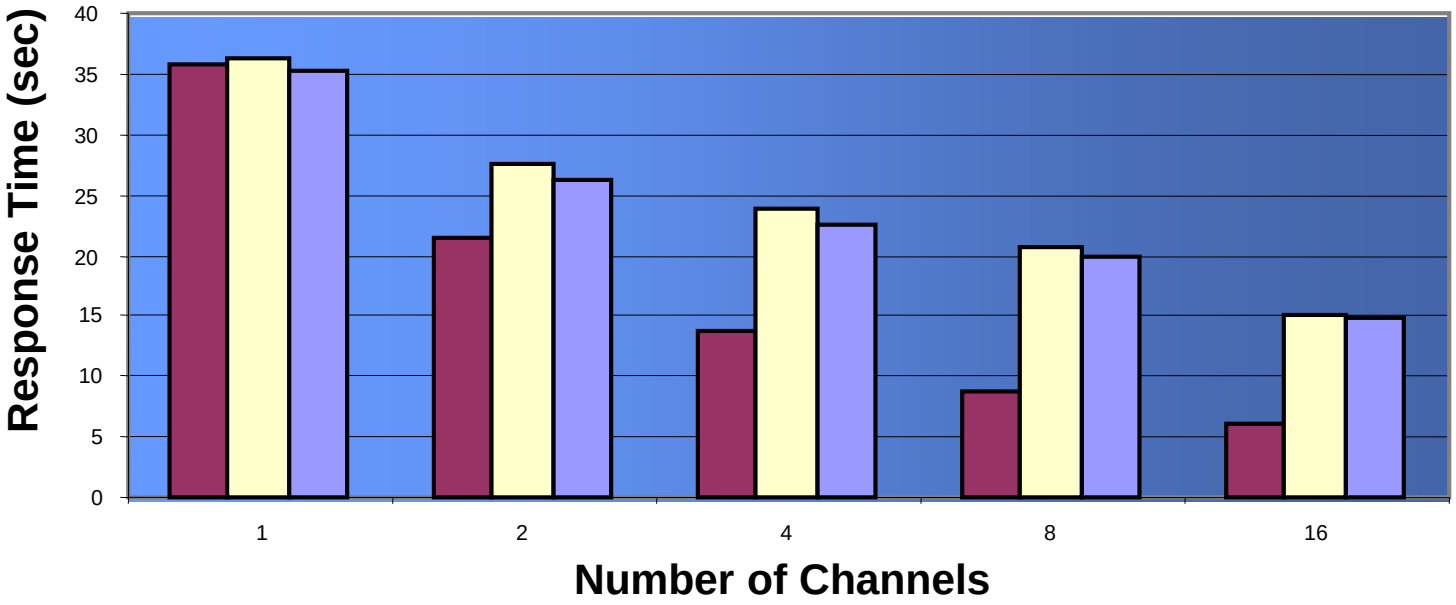


Broadcasting

- A simulator was developed to evaluate the feasibility and effectiveness of the proposed algorithms.
- The NASDAQ database with 4290 securities was used as the source data for the objects on the broadcast.
- The simulator views the parallel air channels as a two-dimensional $N \times M$ array, where N and M represent the number of parallel air channels and the number of objects on a broadcast, respectively.
- User requests are generated, randomly, requesting K objects on the broadcast.
- For each simulated configuration, the simulator is run 1000 times, and the average number of every estimated performance metric is calculated and reported.

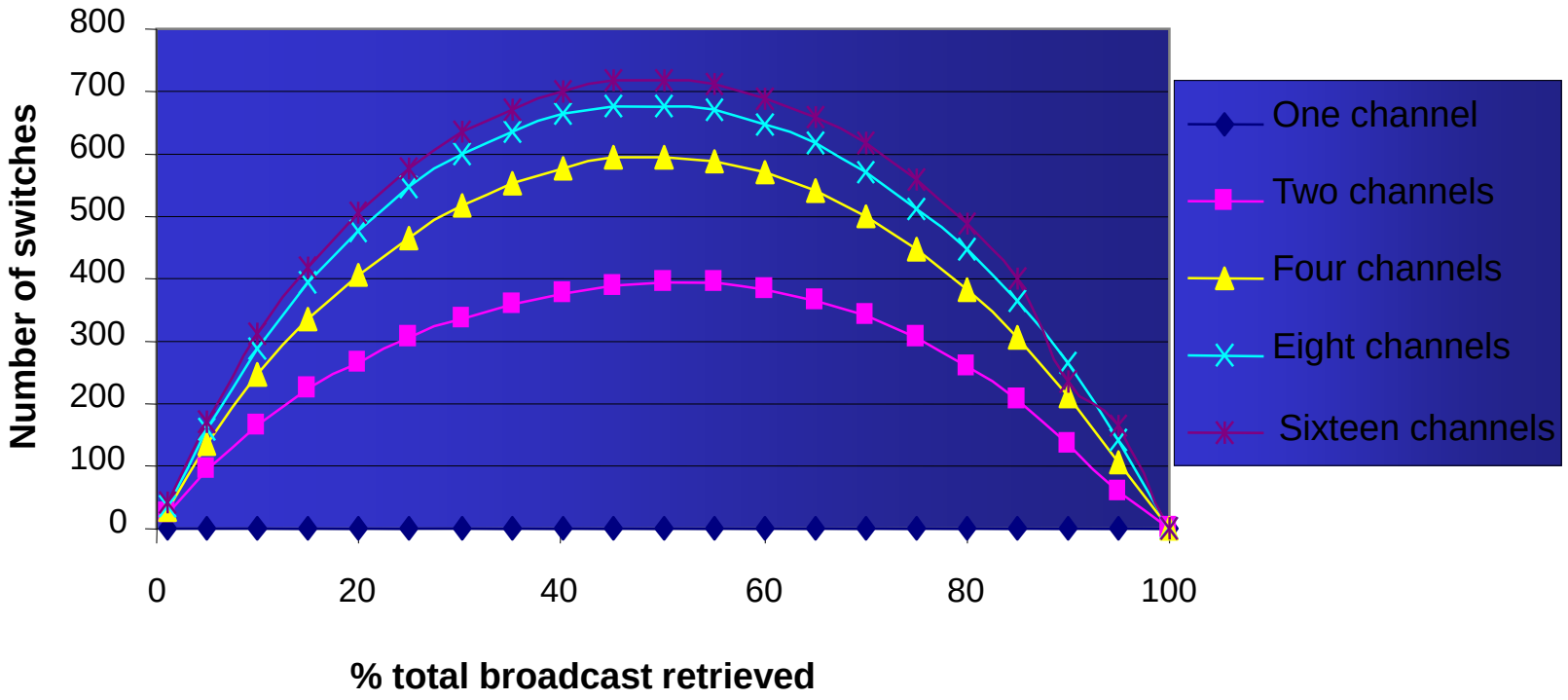
Broadcasting

- Response Time



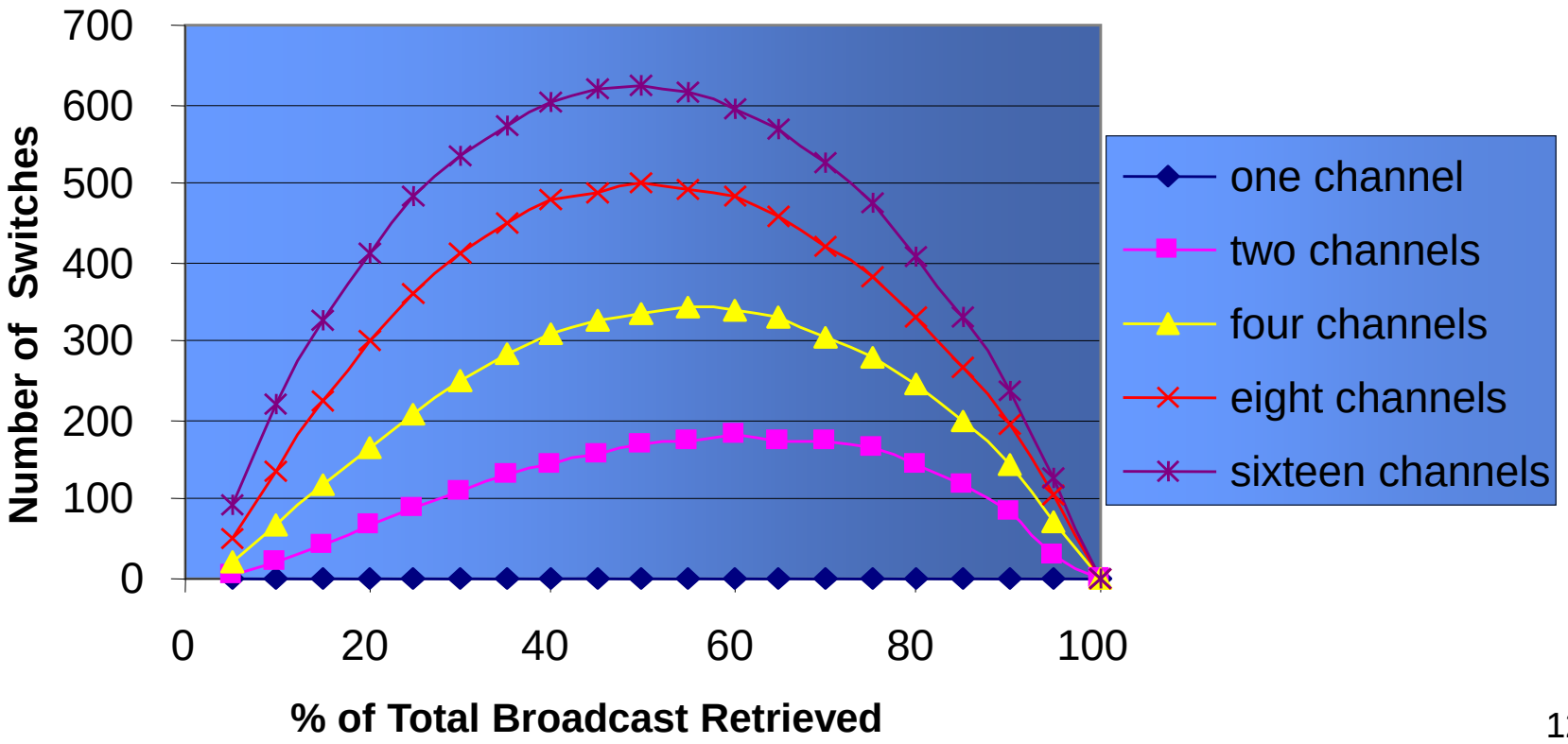
Broadcasting

- Channel Switching Frequency (1st Heuristic solution)



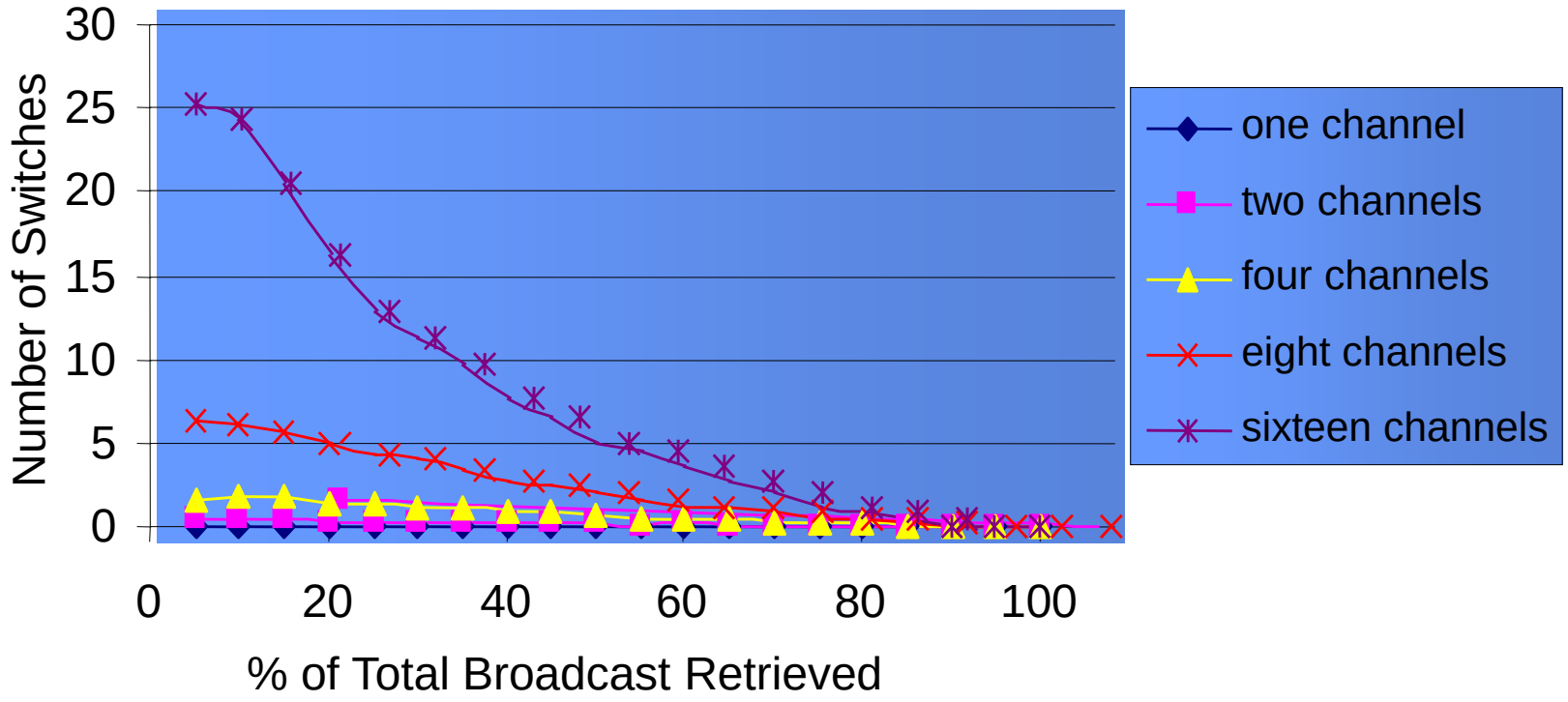
Broadcasting

- Channel Switching Frequency (1st Heuristic solution revised heuristic order)



Broadcasting

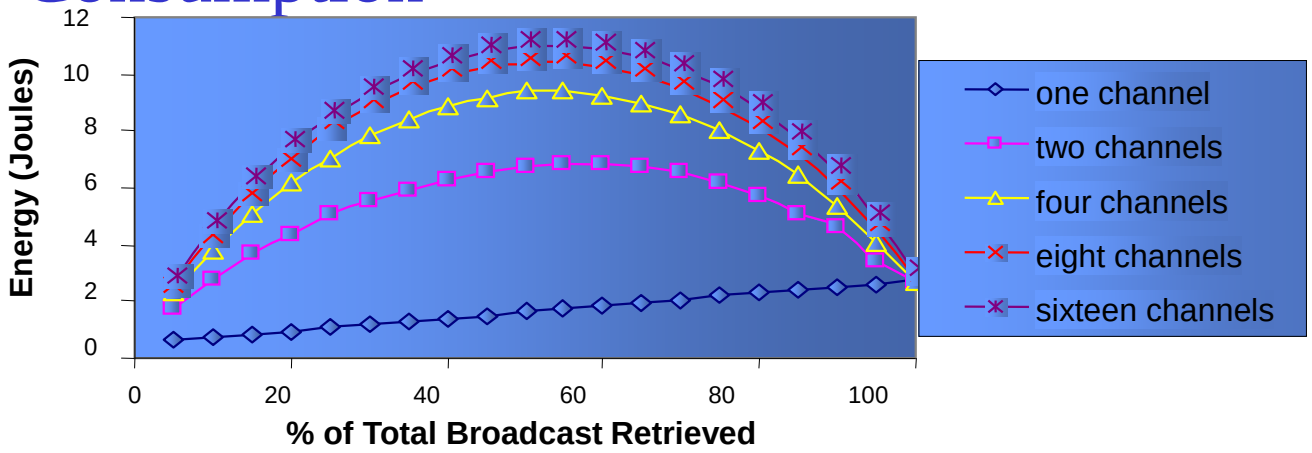
- Channel Switching Frequency (2nd Heuristic solution)



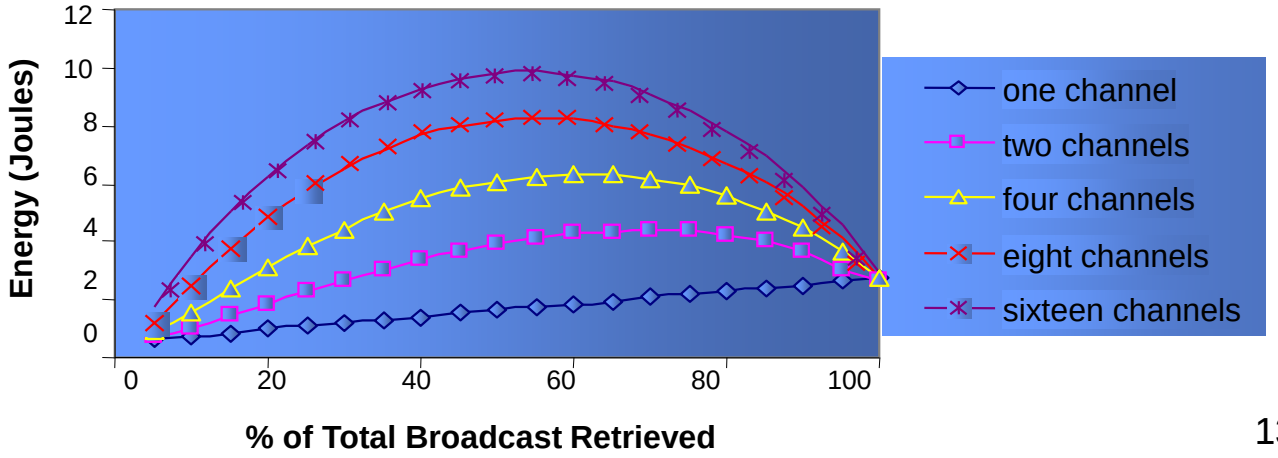
Broadcasting

- Energy Consumption

1st Heuristic

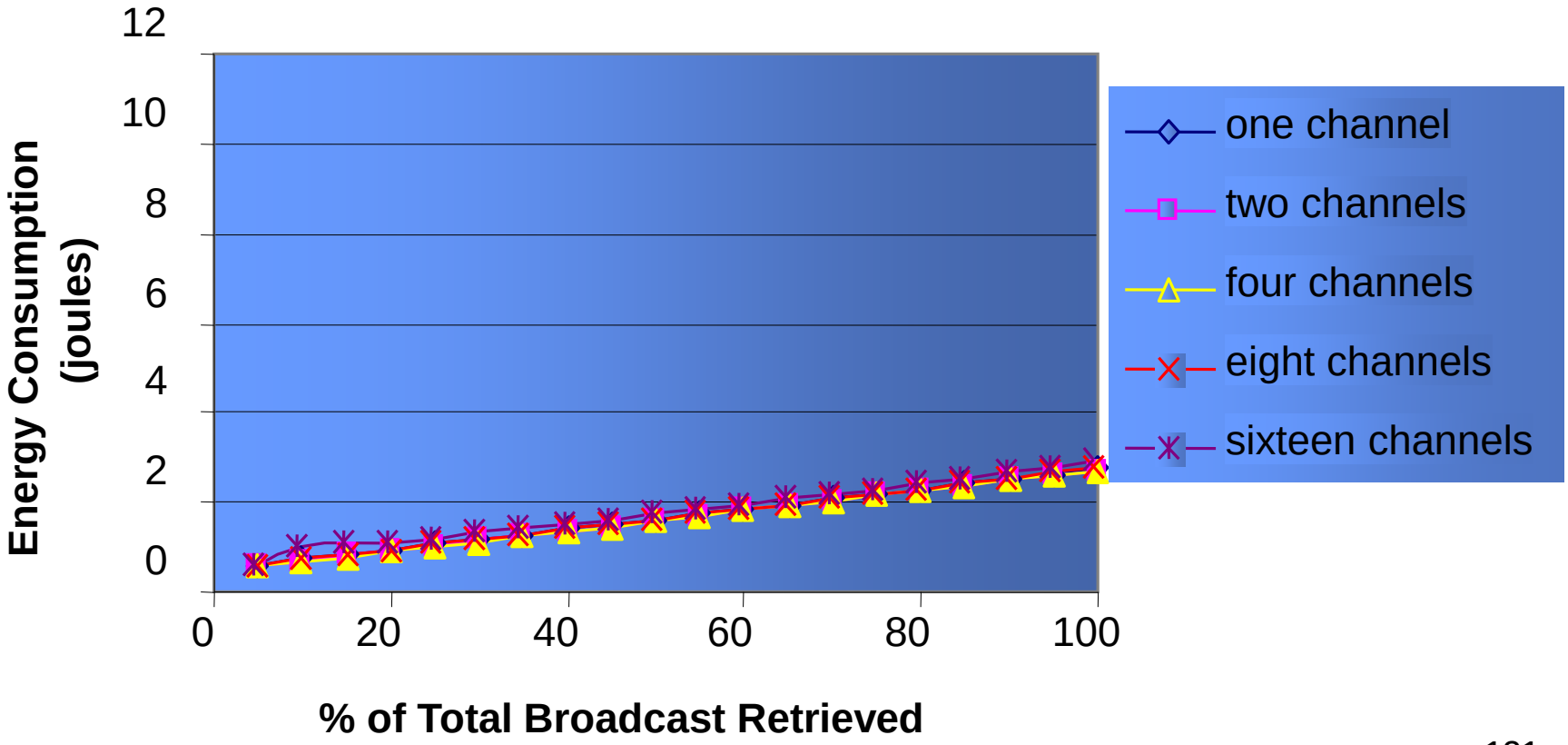


1st Heuristic revised priority



Broadcasting

- Energy Consumption (2nd Heuristic Solution)

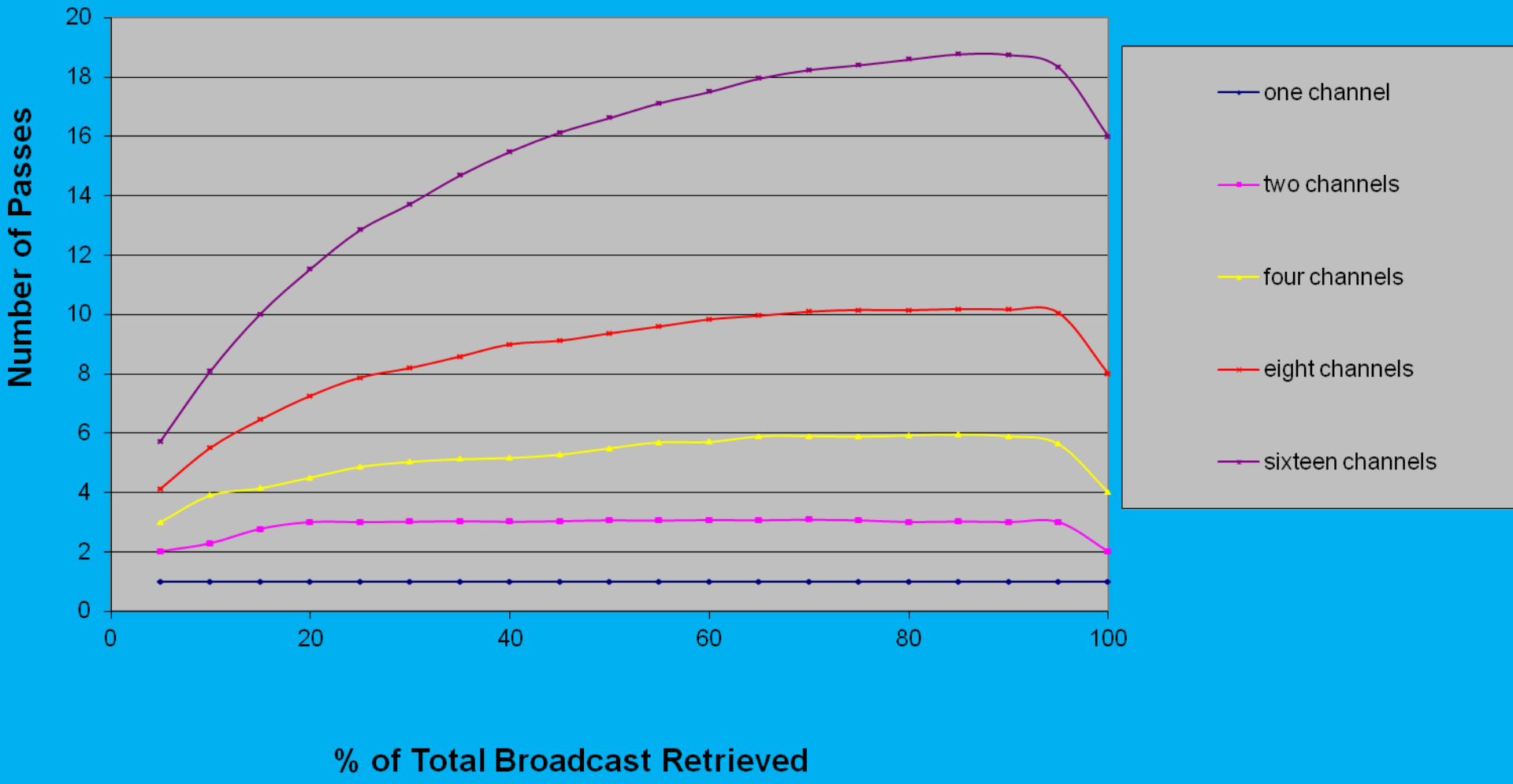


Broadcasting

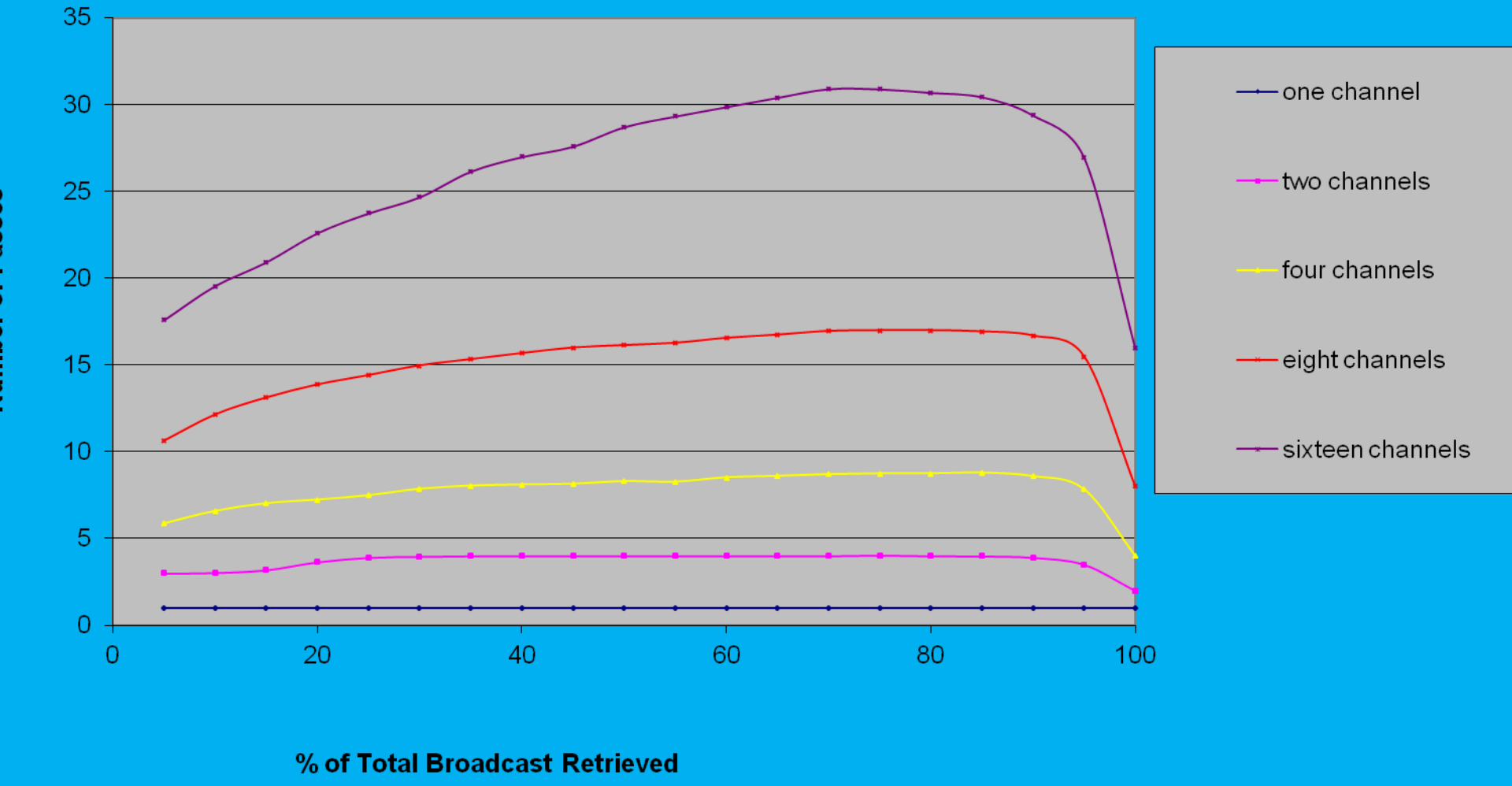
- Switching Frequency relative to 1st Heuristic Solution

Number of channels	% reduction using revised heuristics	% reduction using 2nd Heuristic Solution
Two	54%	99.87%
Four	43%	99.70%
Eight	26%	99.07%
Sixteen	13%	96.48%

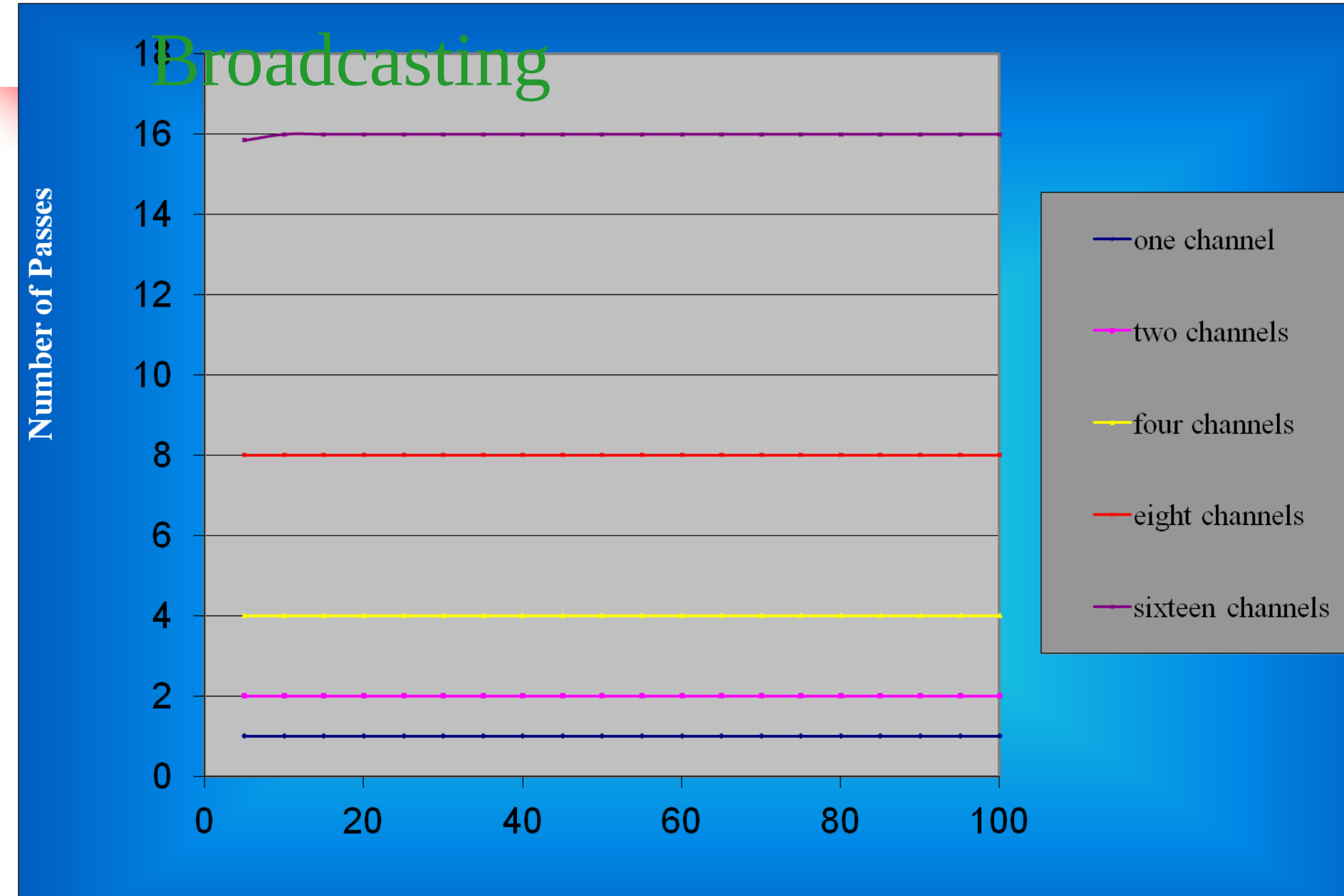
Broadcasting



Broadcasting



Broadcasting

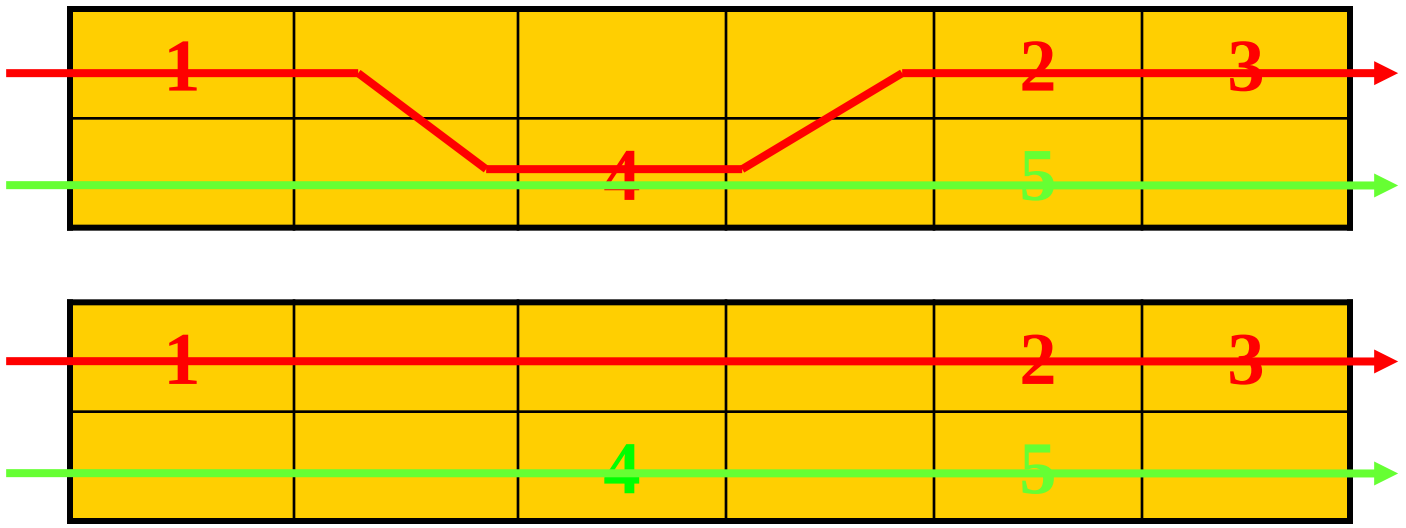


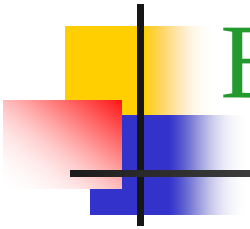
Broadcasting

- Object Organization on Parallel Channels – Access Patterns
  - It is a heuristic method and hence, in some instances may not find the least number of passes.

Broadcasting

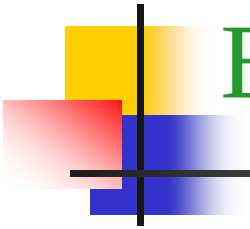
- Object Organization on Parallel Channels – Access Patterns
  - It is a heuristic method and hence, in some instances may not find the least number of channel switches.





Broadcasting

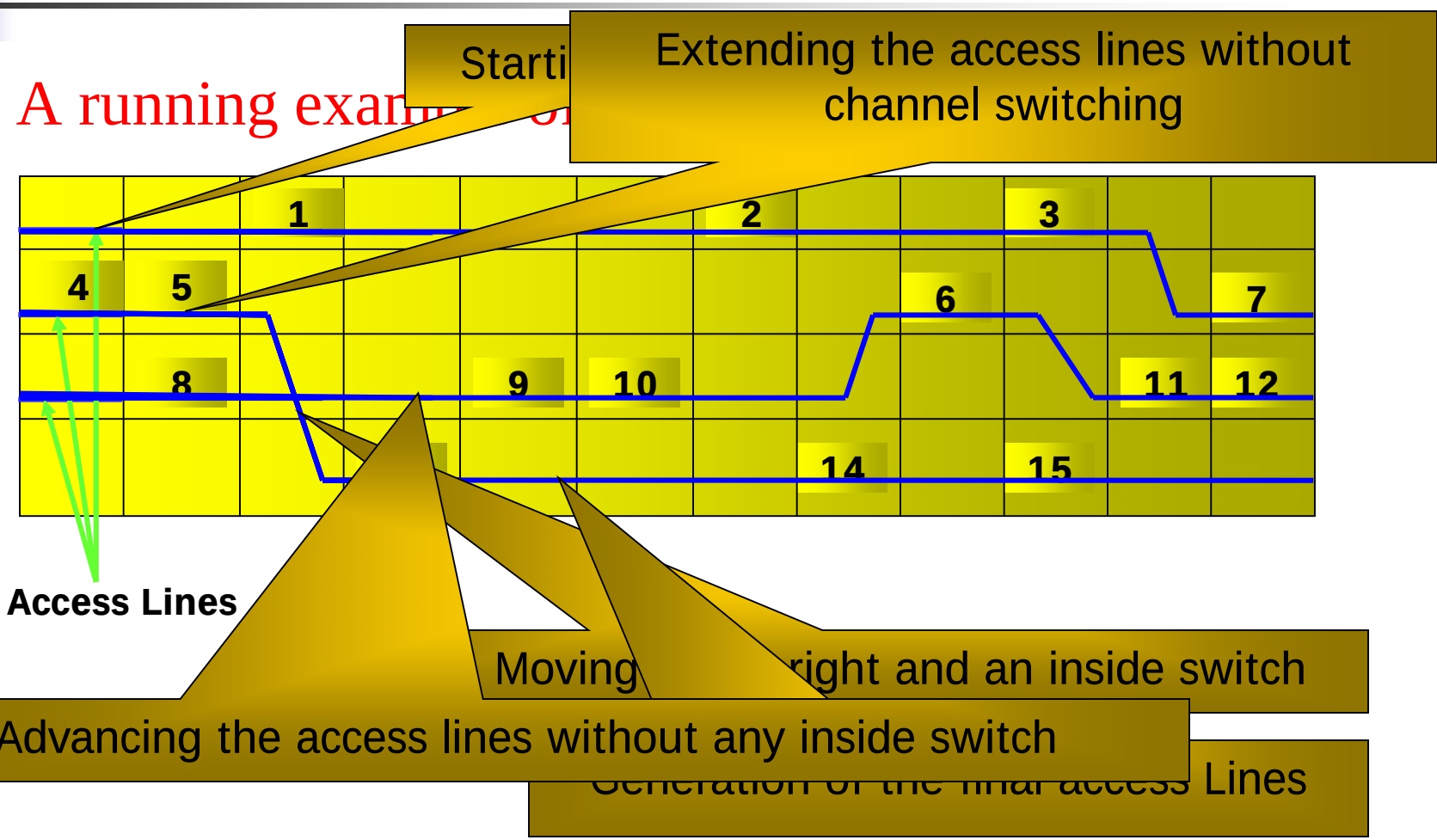
- Object Organization on Parallel Channels – Parallel Object Scan (POS)
  - POS is a scheduling algorithm that always generates the minimum number of passes to scan all requested objects on a broadcast. Starting from the left-most column, the algorithm:
    - Simultaneously (in parallel) constructs *access* lines, column by column.
    - Attempts to visit all the requested objects in the next column with the fewest switches from the previous column.
    - Observes which cell with a line has a less chance to have a requested object in the future on the same channel and uses this line to make a switch when necessary. This strategy can minimize the number of switches.

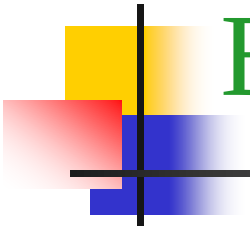


Broadcasting

- Object Organization on Parallel Channels – Parallel Object Scan (POS)
  - Determine # of access Lines
  - Initialization - Determine starting rows (Rows with leftmost requested items)
  - Compute access lines in parallel, working left to right. For each column, determine whether each line should:
    - Continue in same row because there is an item to read in the next column
    - Switch to another row because no line is reading that row and there is an item to be read in that row in the next column
    - Continue in the same row because there is nothing else to do

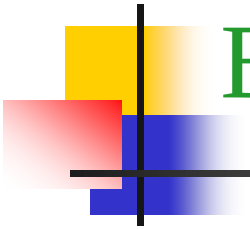
Broadcasting





Broadcasting

- Object Organization on Parallel Channels – Serial Empty Scan (SES)
  - The POS algorithm constructs *k* access paths that cover all the requested data items.
  - Serial Empty Scan (SES) implicitly constructs the paths examining the empty (non-requested) blocks instead of the requested data items.



Broadcasting

- Object Organization on Parallel Channels – Serial Empty Scan (SES)
  - It examines empty blocks instead of requested objects.
  - The basic idea behind the algorithm is as follows:
    - We construct paths that scan only empty blocks (empty paths).
    - As we do this we also compress the requested objects into logical channels.
    - The resulting “logical” channels represent the sequence of requested objects that an access line reads. The action of compressing (copying objects from one channel to another) simulates a switch during a scan.

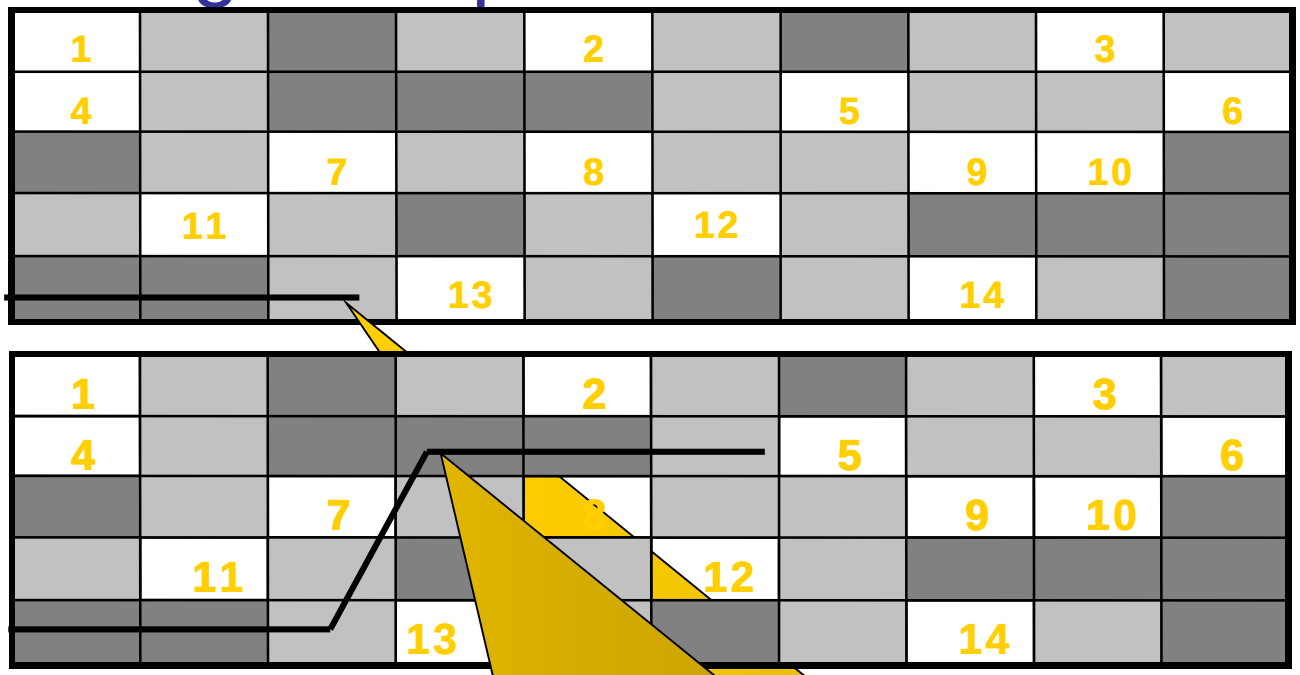
Broadcasting

- Object Organization on Parallel Channels – Serial Empty Scan (SES)
  - Working left-to-right we construct one empty row:
    - Find the longest empty initial block
    - Find the longest empty block overlapping with this one
    - Move objects to the left in overlapping block onto initial block (overlapping block is now longest initial block)
    - Continue this process until an entire empty row is constructed
    - Eliminate this row and repeat the process until MAX-CUT rows are left.

Results in logical rows describe the actual access lines.

Broadcasting

A running example

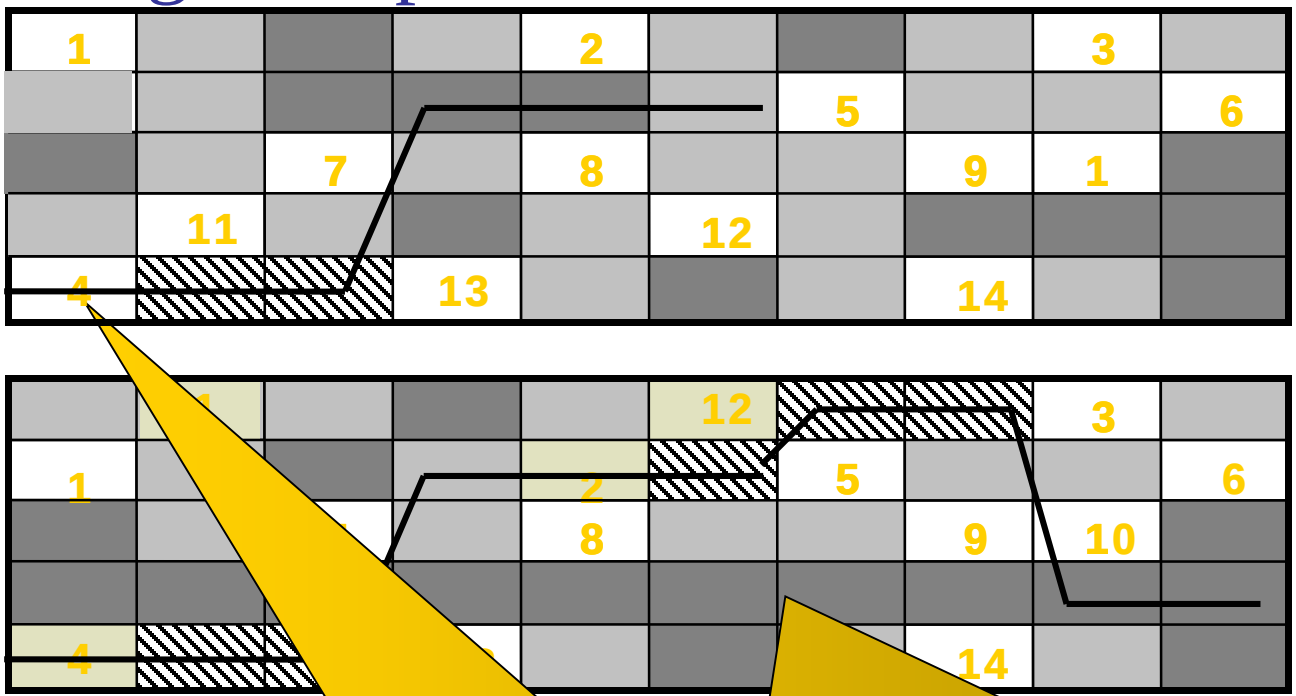


Find an overlapped empty block and switch

largest leftmost obstacle

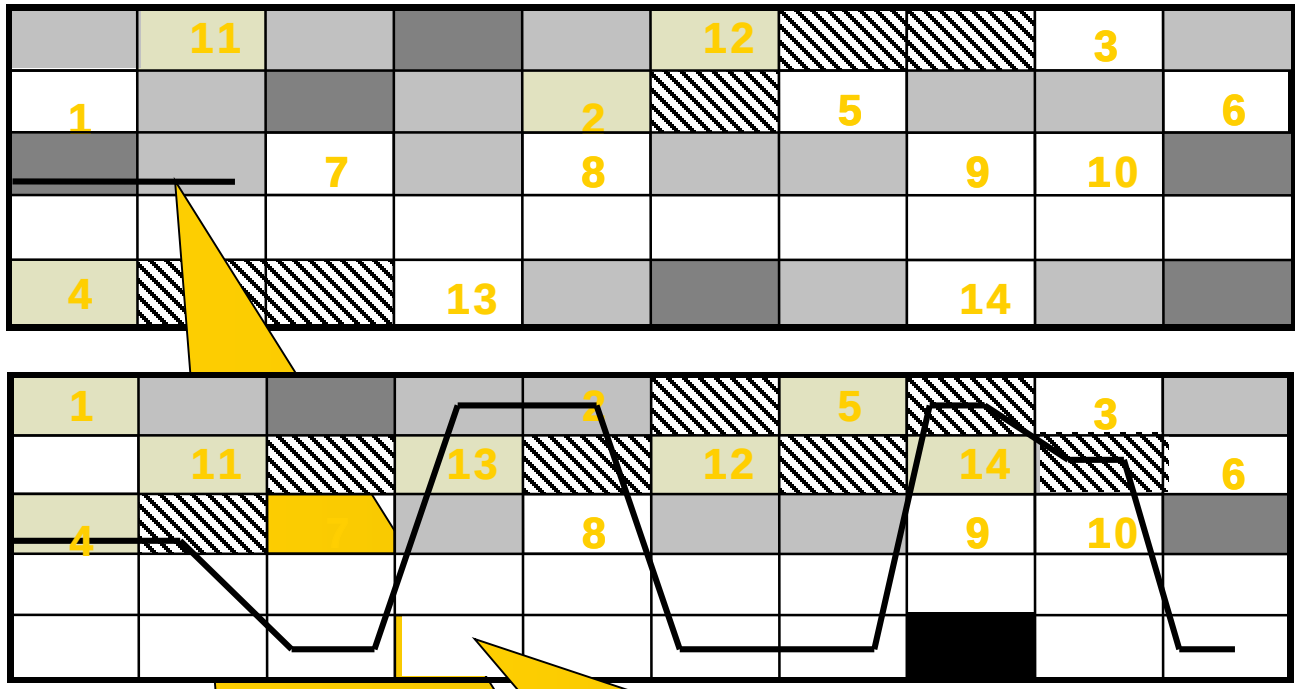
Broadcasting

A running example



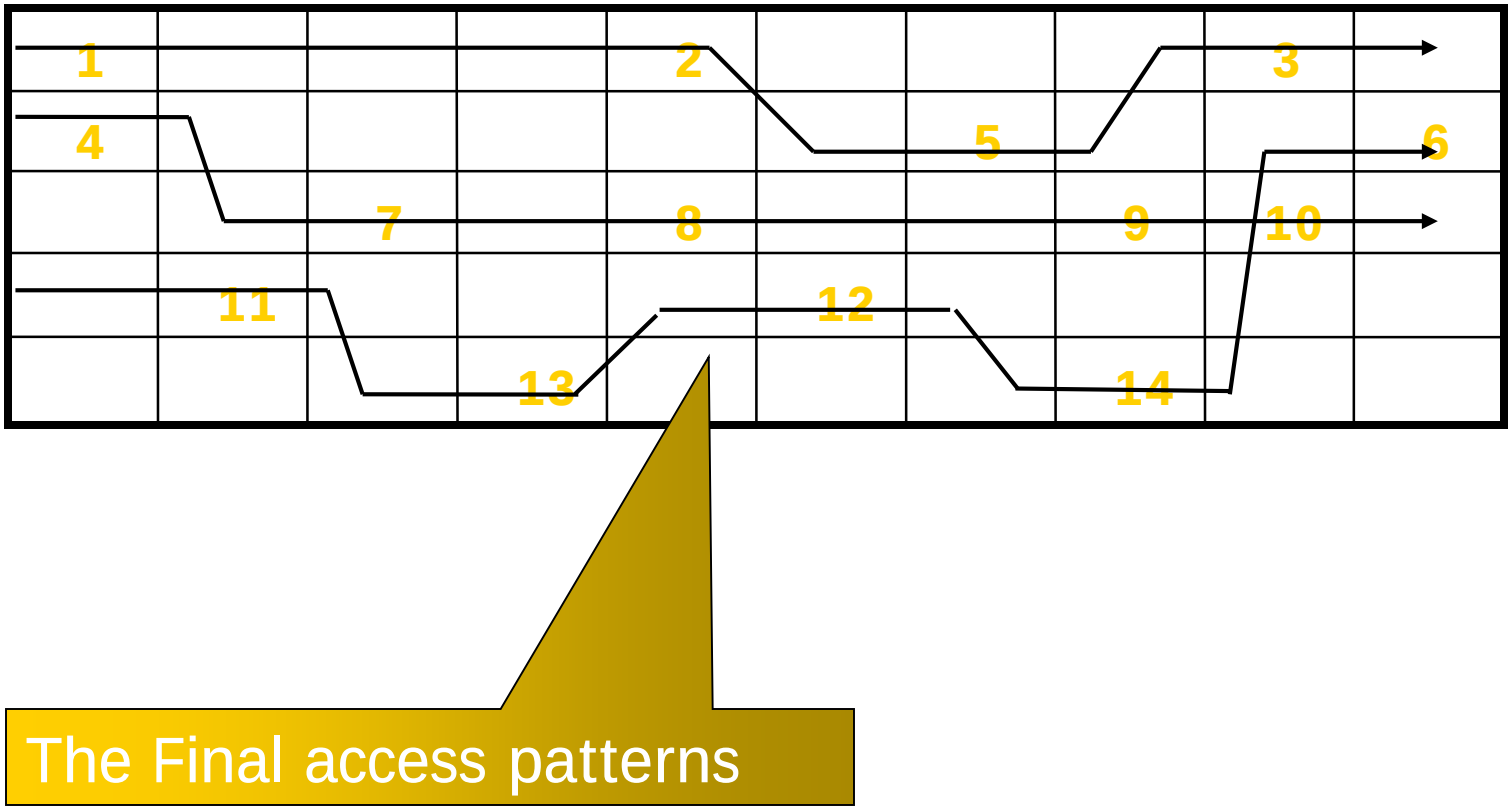
Broadcasting

A running example



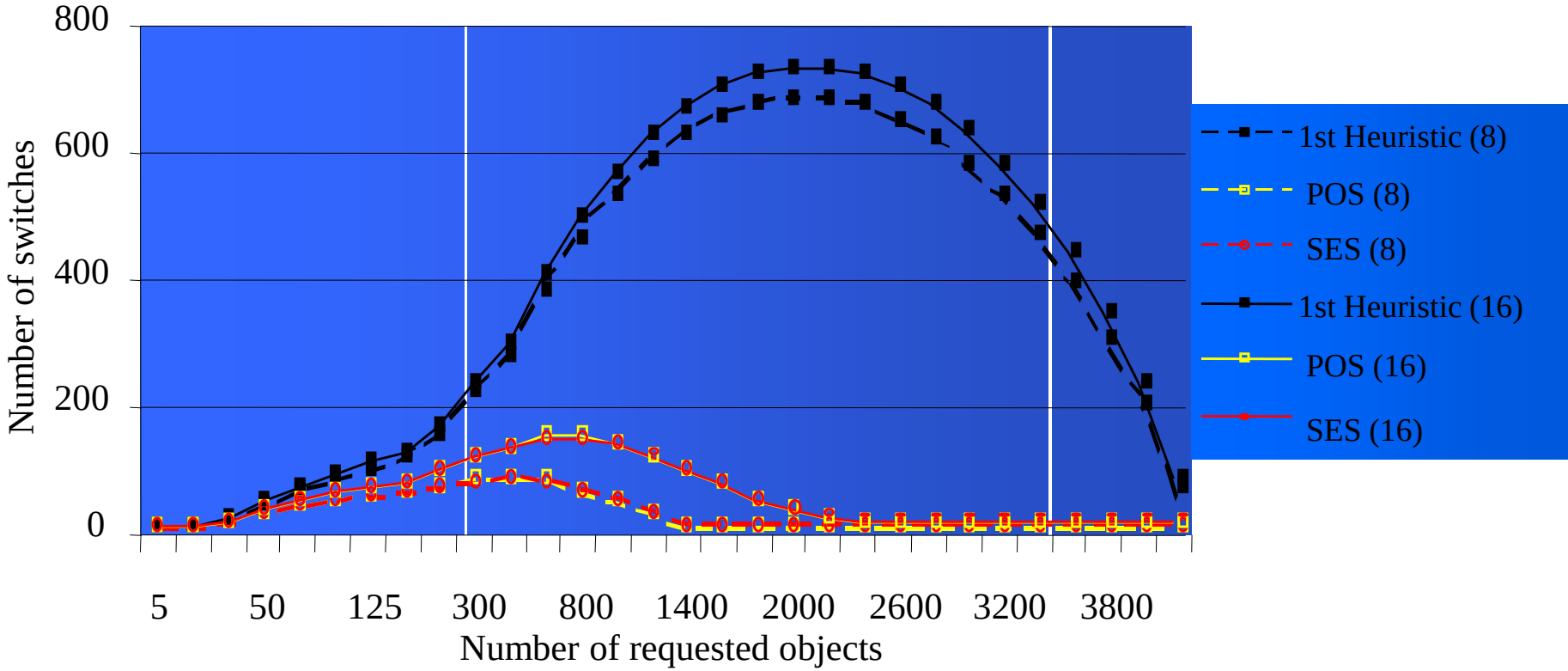
Broadcasting

A running example



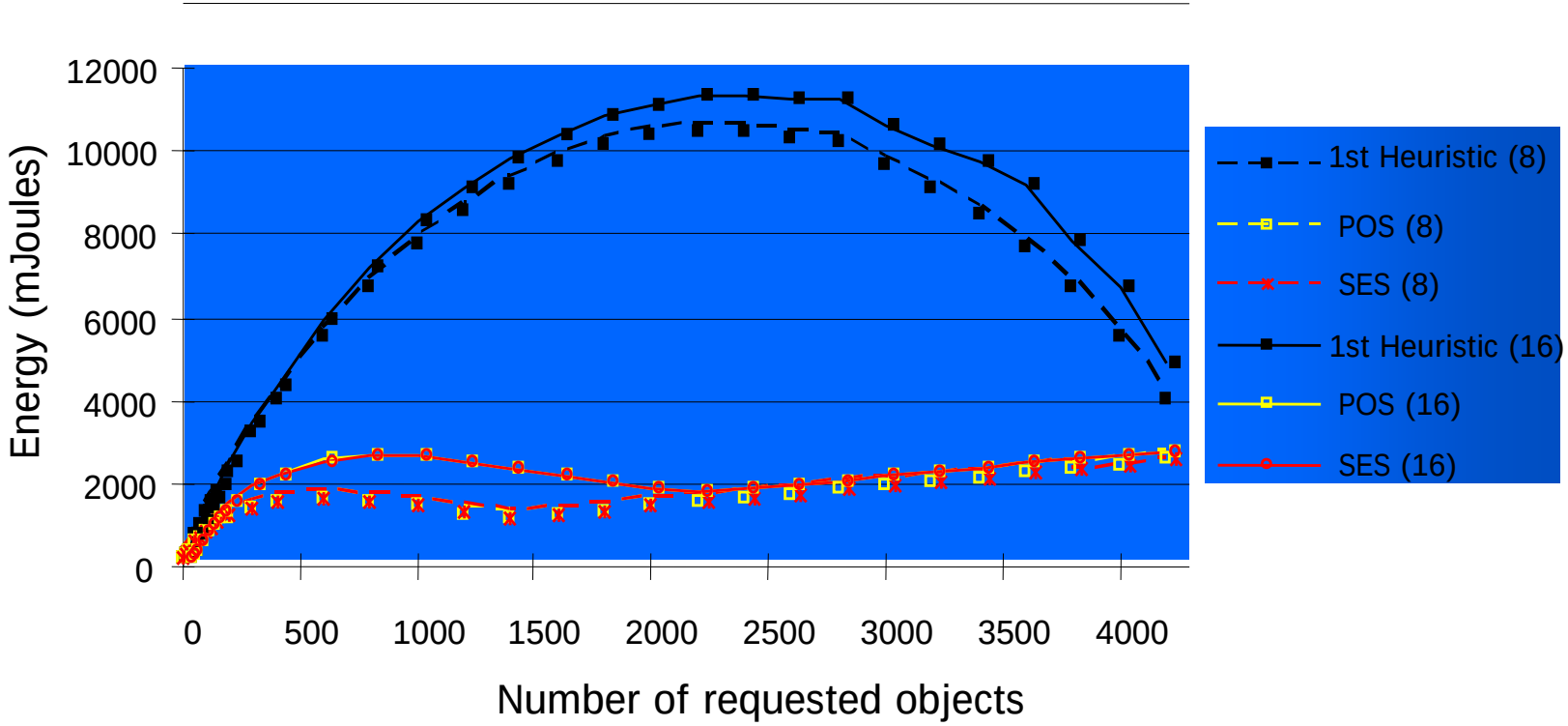
Broadcasting

- Channel Switching



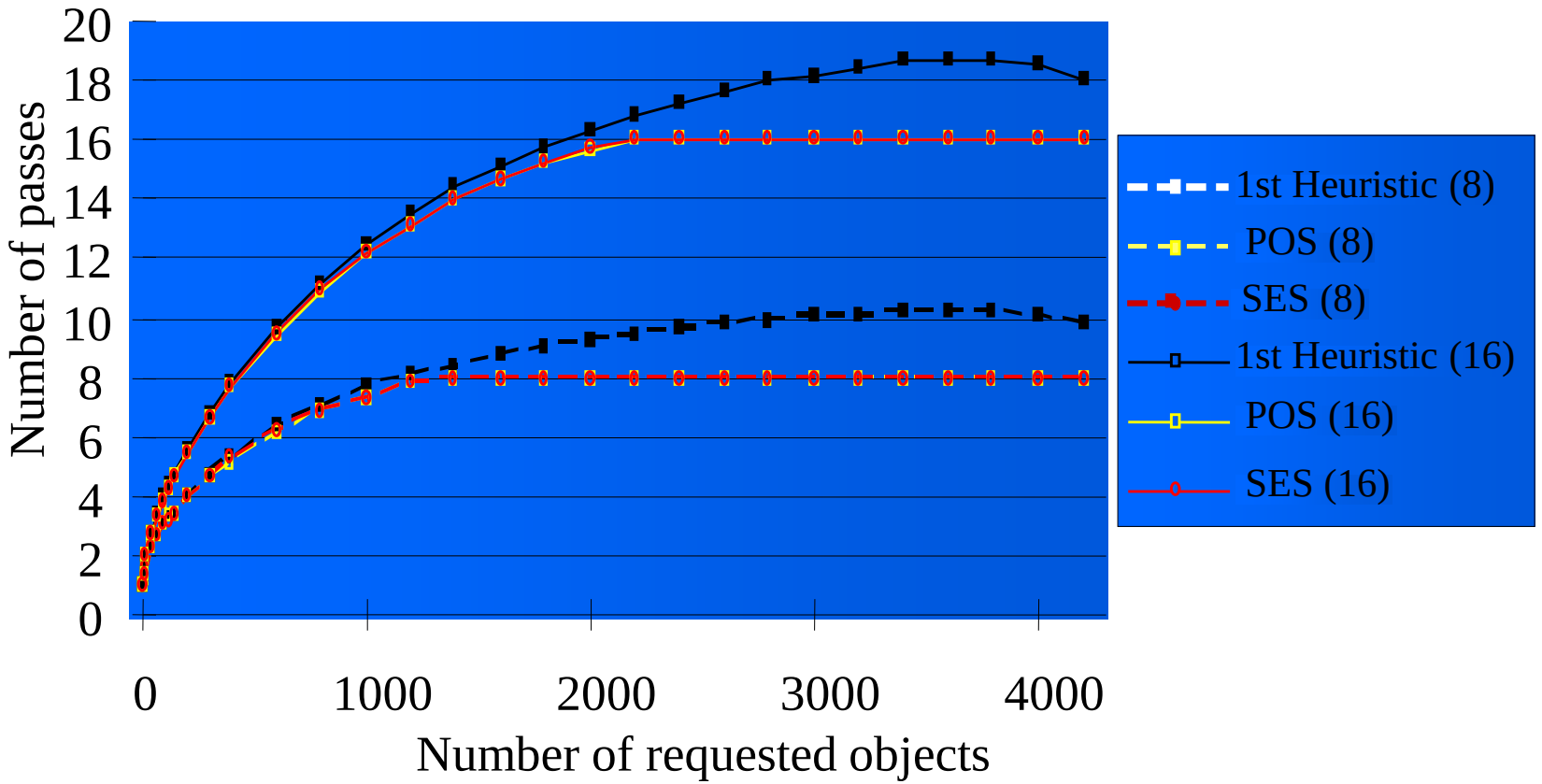
Broadcasting

- Energy Consumption

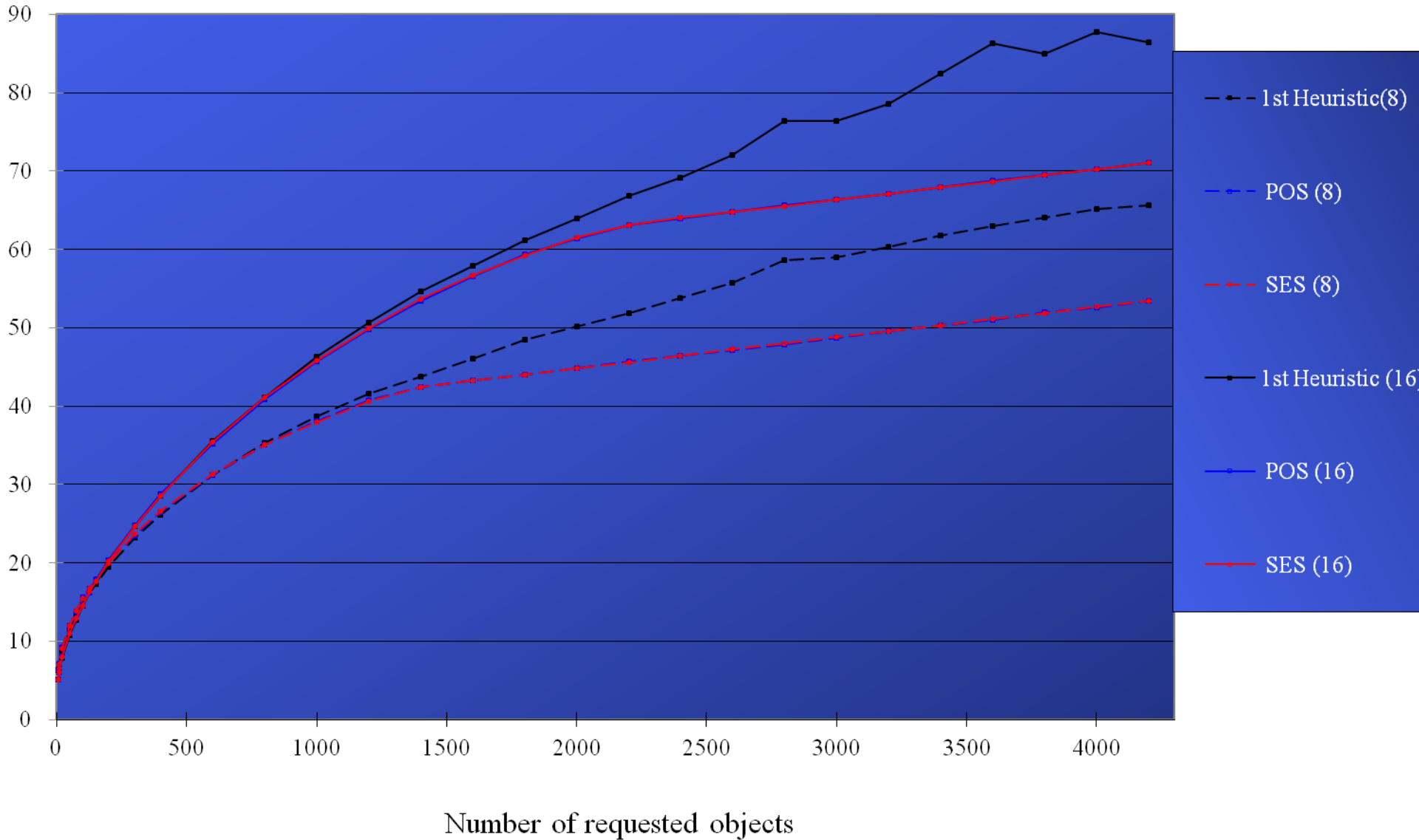


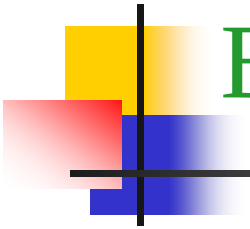
Broadcasting

- Number of Passes



Broadcasting



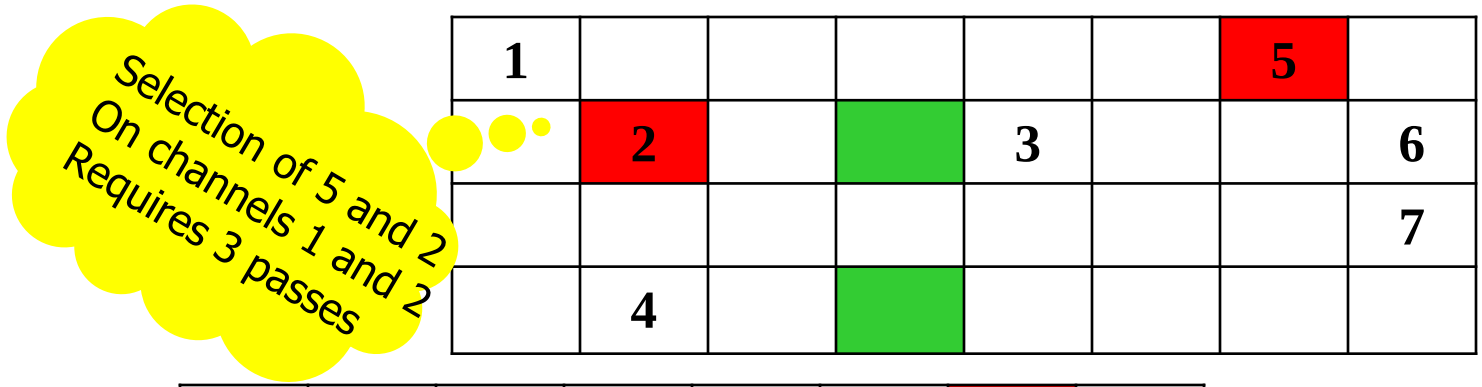


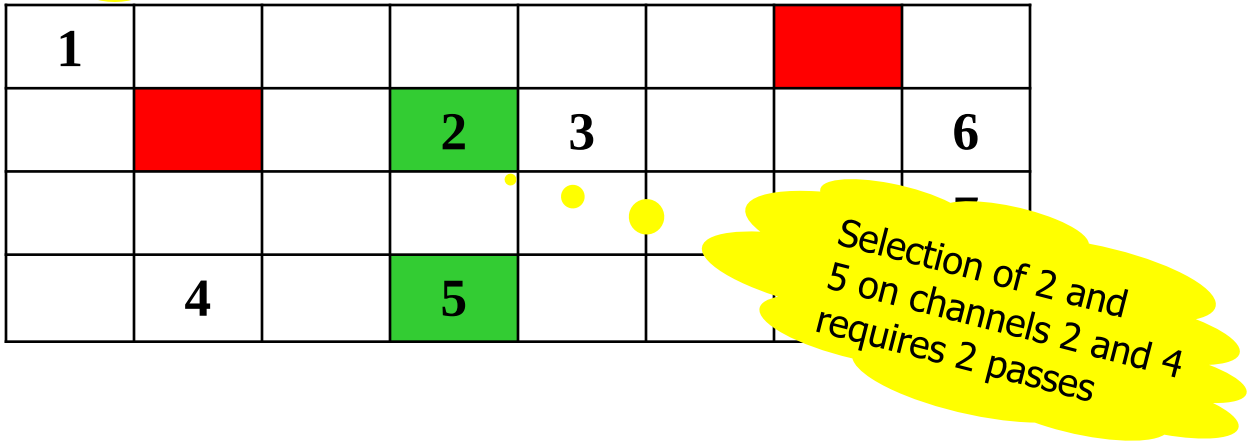
Broadcasting

- Object Organization on Parallel Channels – Replication
  - To further improve the response time, availability, and power consumption, frequently accessed objects can be replicated on the broadcast channels.
  - At the presence of replica, the access schedules must be extended to select the replicas that reduce the number of passes and the number of channel switches.

Broadcasting

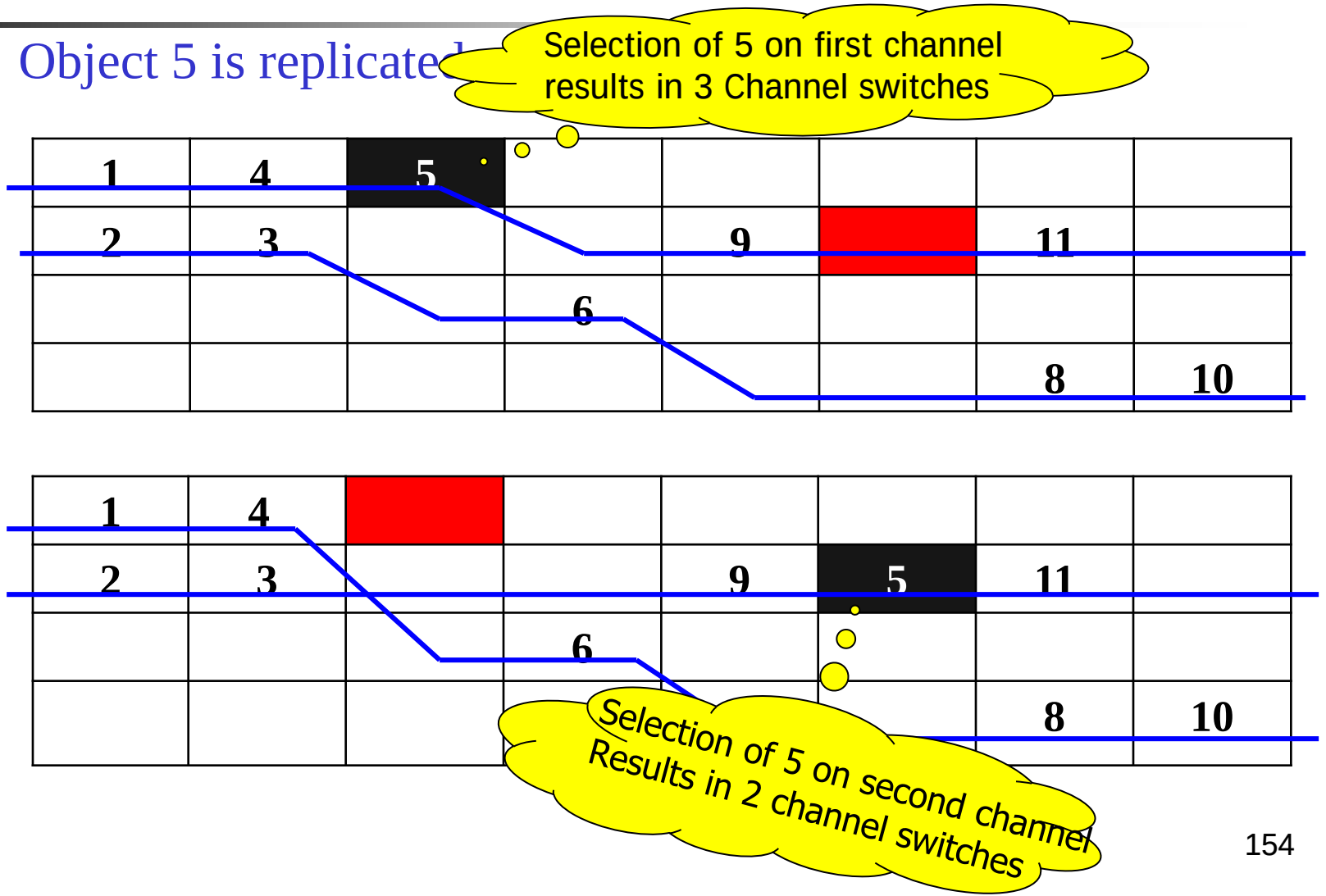
- Object Organization on Parallel Channels – Replication
  - Objects 2 and 5 are replicated

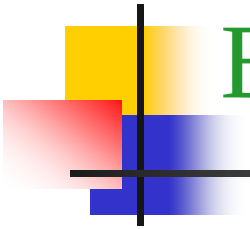




Broadcasting

- Object 5 is replicated





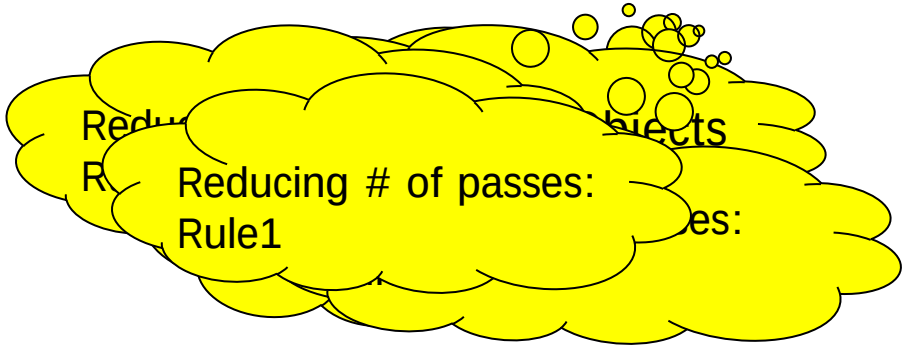
Broadcasting

- Eliminate the replica that:
  - Reduces number of passes
    - Rule1: Reduce number of passes at those critical columns that have only one removable object (pair),
    - Rule2: Apply least cut Rule
  - Reduces number of channel switches
    - Rule1: Largest Empty Block
    - Rule2: Rightmost Block
- Branch and Bound
- Select-First
- Select Random

Broadcasting

A running example

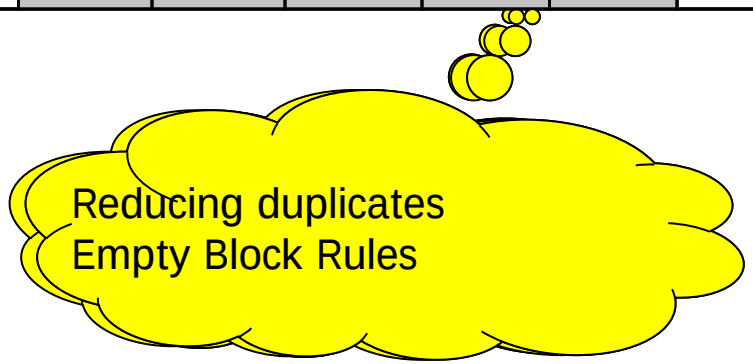
							1				
6	12	9			7	5	4				
		10*			2*						11
8										10	9

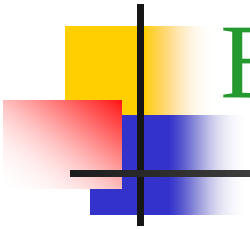


Broadcasting

A running example

		1*				2*			3	13*	
4	5		1*			2*		6*			7
				9	10			6*		11	6*
	15		13*				14				



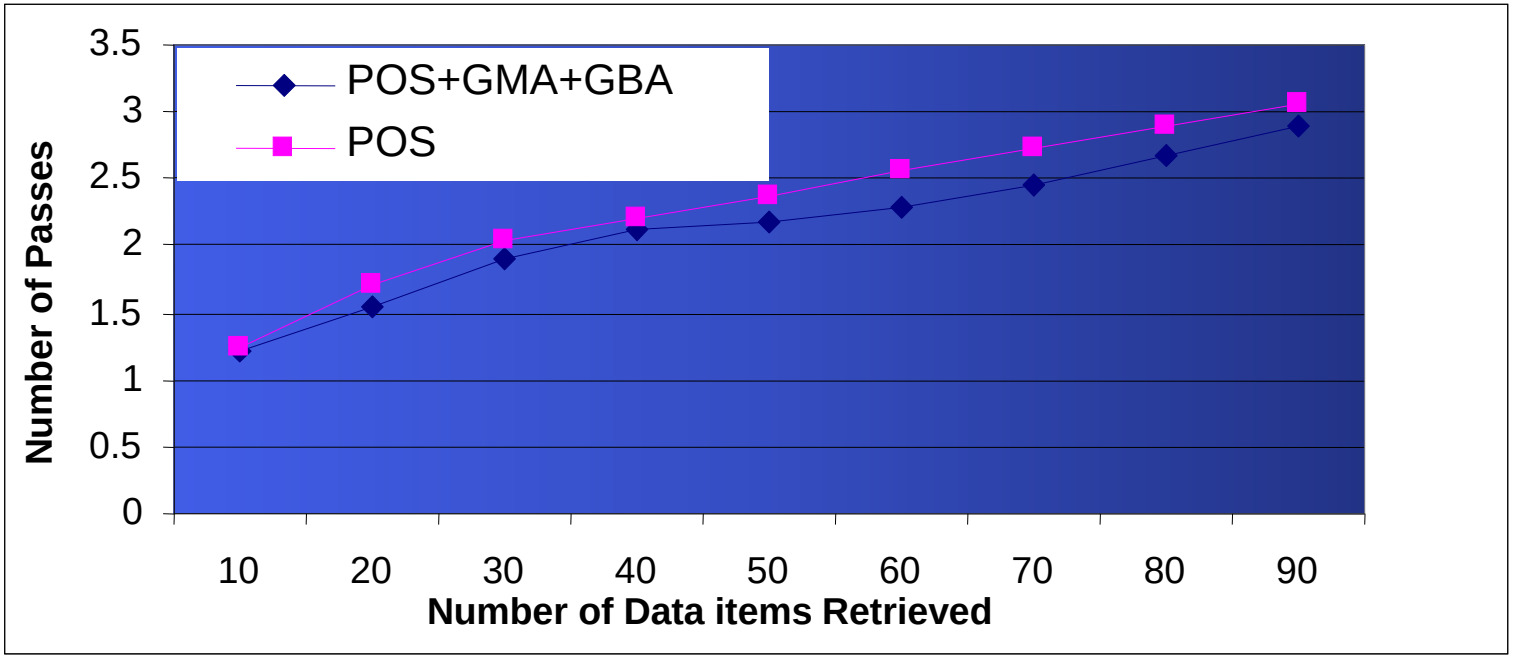


Broadcasting

- The scope of our simulator (Parallel Object Scan) was extended to evaluate the proposed algorithms, for different configurations, based on performance metrics such as the number of passes, number of switched, and the response time.

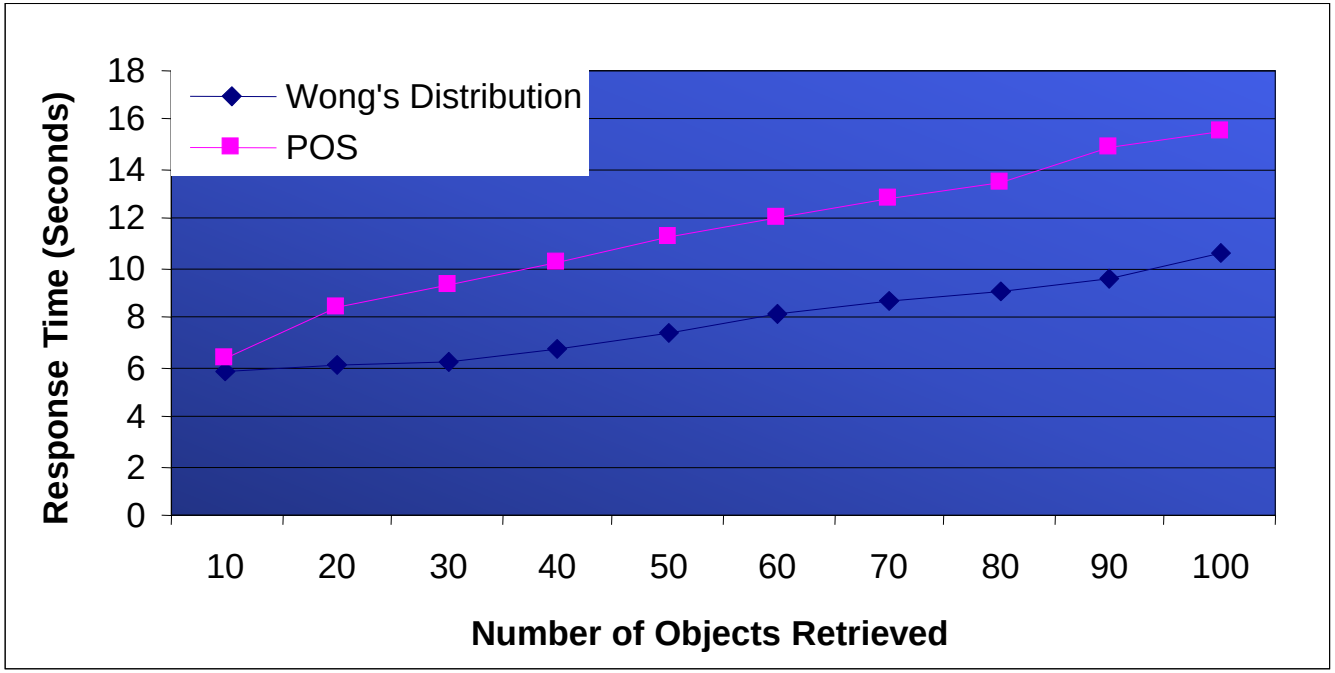
Broadcasting

- Number of passes



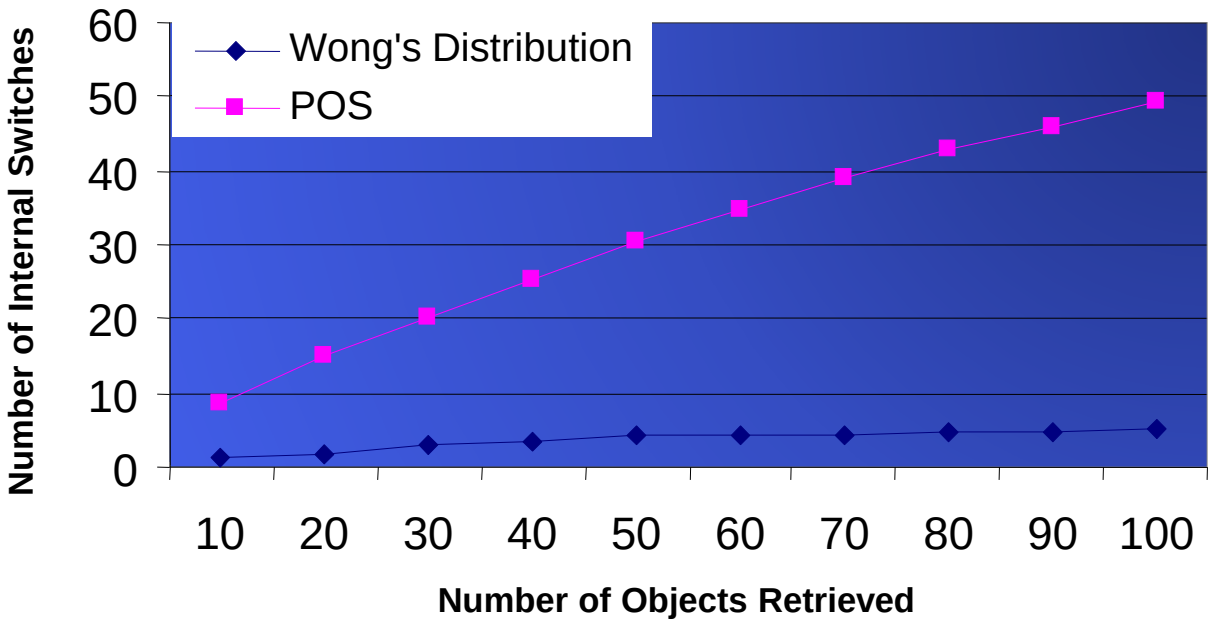
Broadcasting

- Response Time



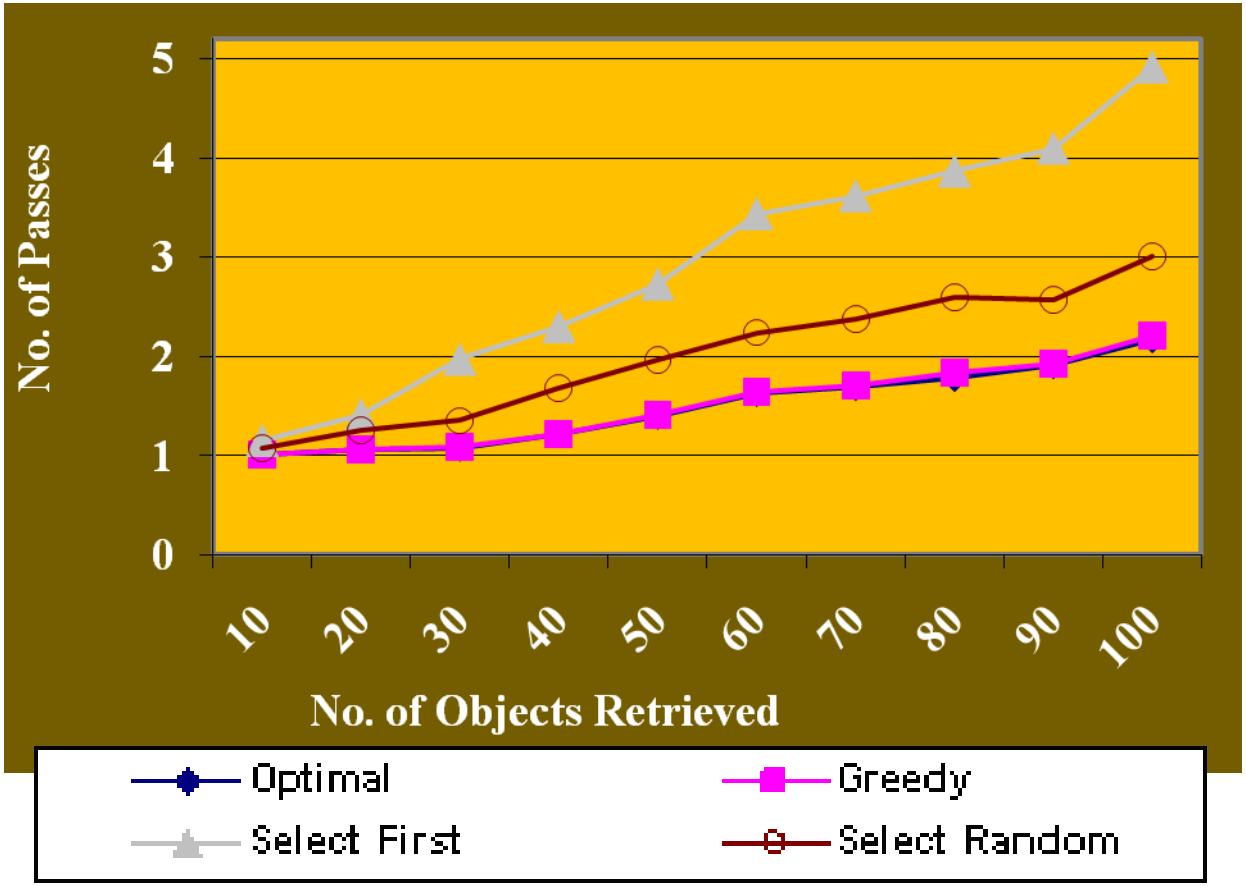
Broadcasting

- Number of Switches



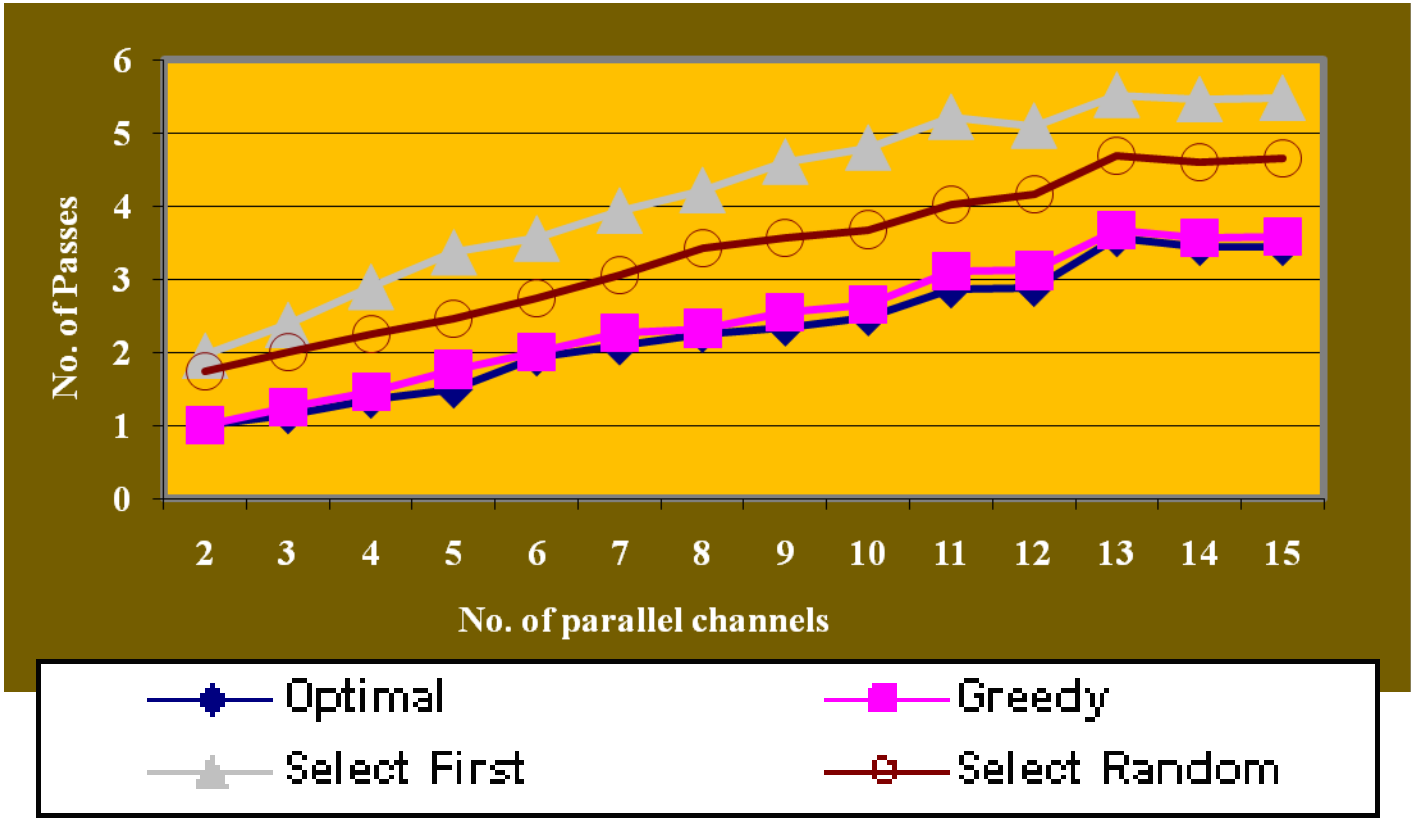
Broadcasting

- Number of Passes *vs.* number of Objects



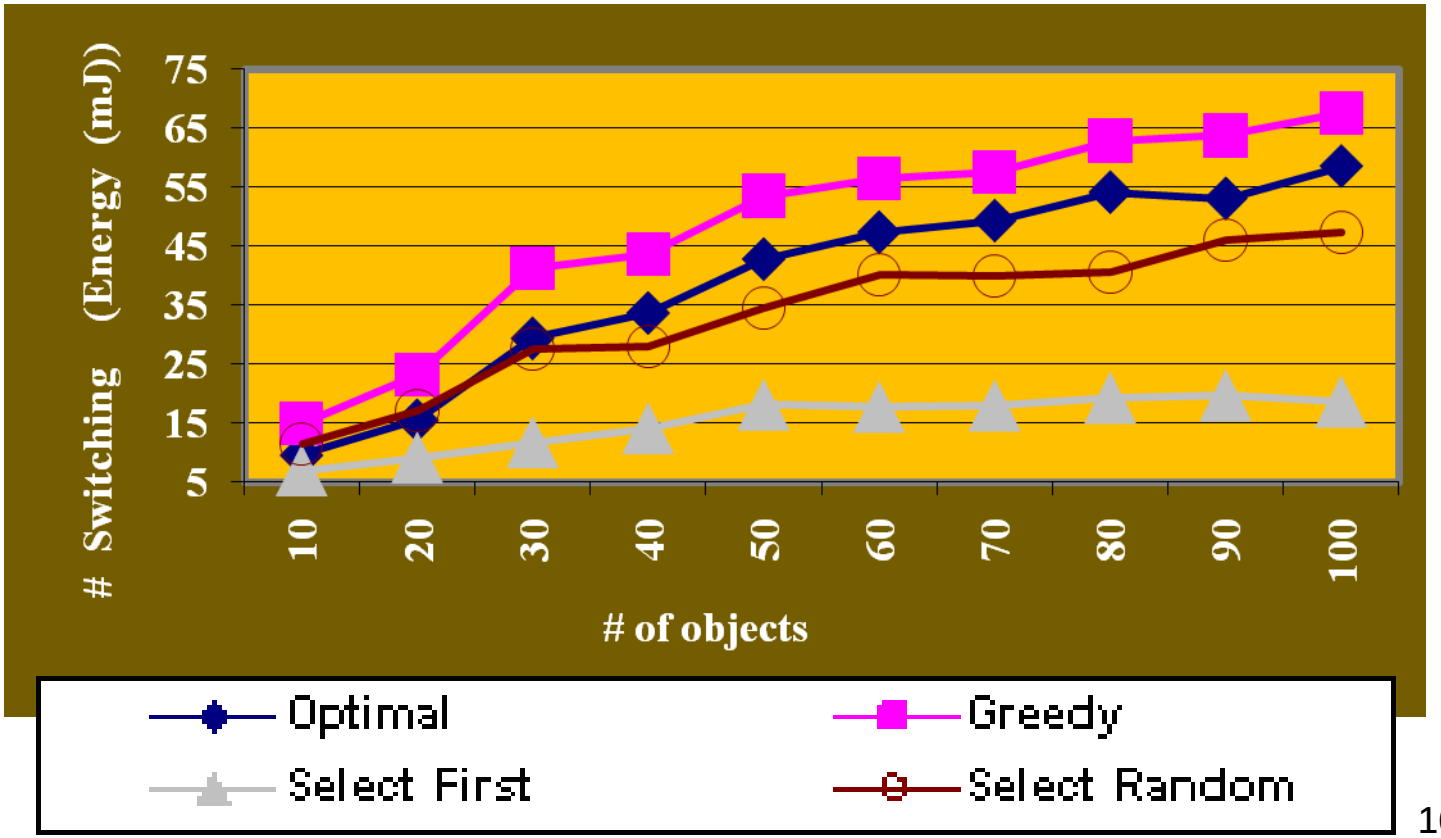
Broadcasting

- Number of Passes *vs.* number of Channels



Broadcasting

- Number of Switches *vs.* Number of Objects

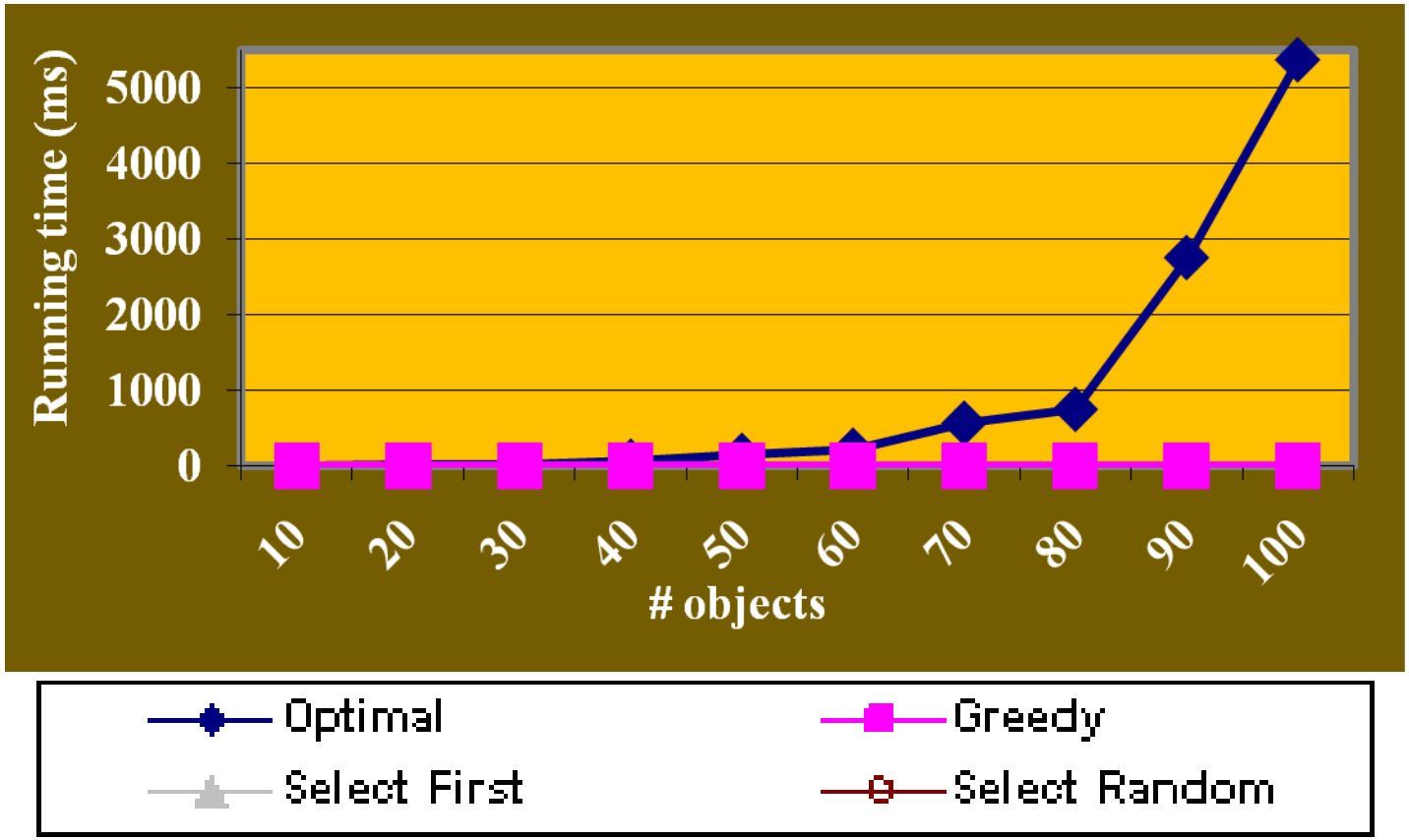


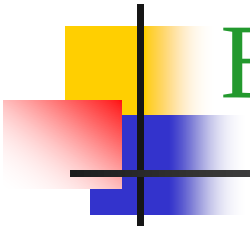
Broadcasting

- Response Time *vs.* Number of Objects

Broadcasting

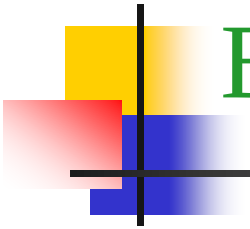
- Run Time of the algorithms





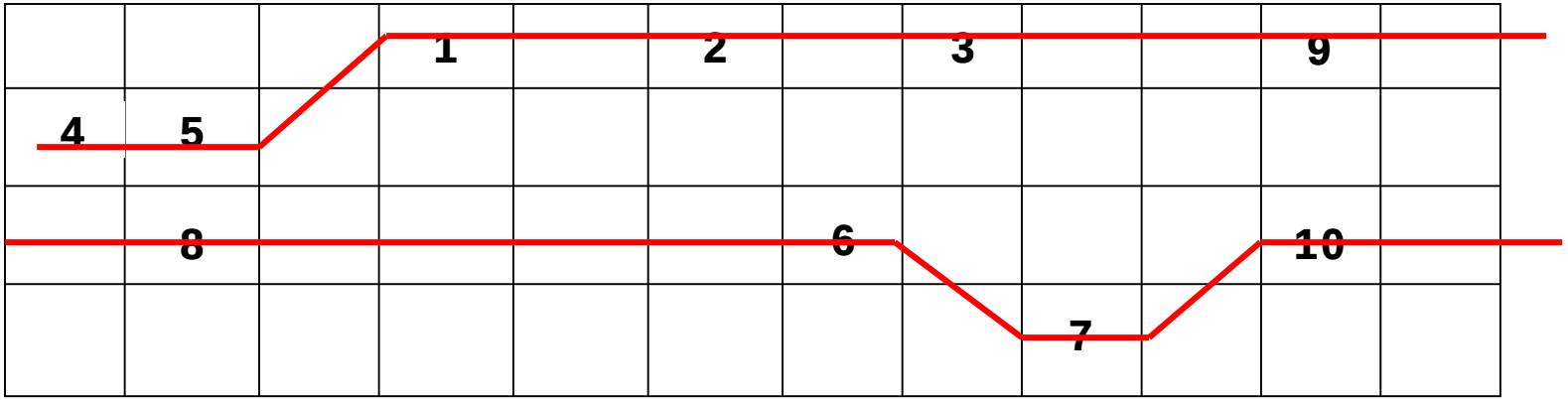
Broadcasting

- Future Directions
  - Reverse the Order of Heuristics
  - Broadcast Contents

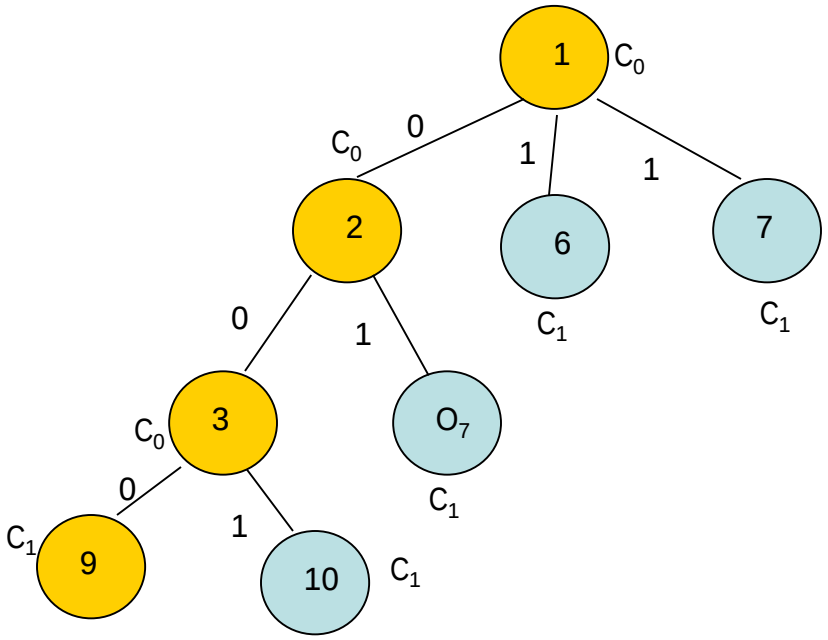


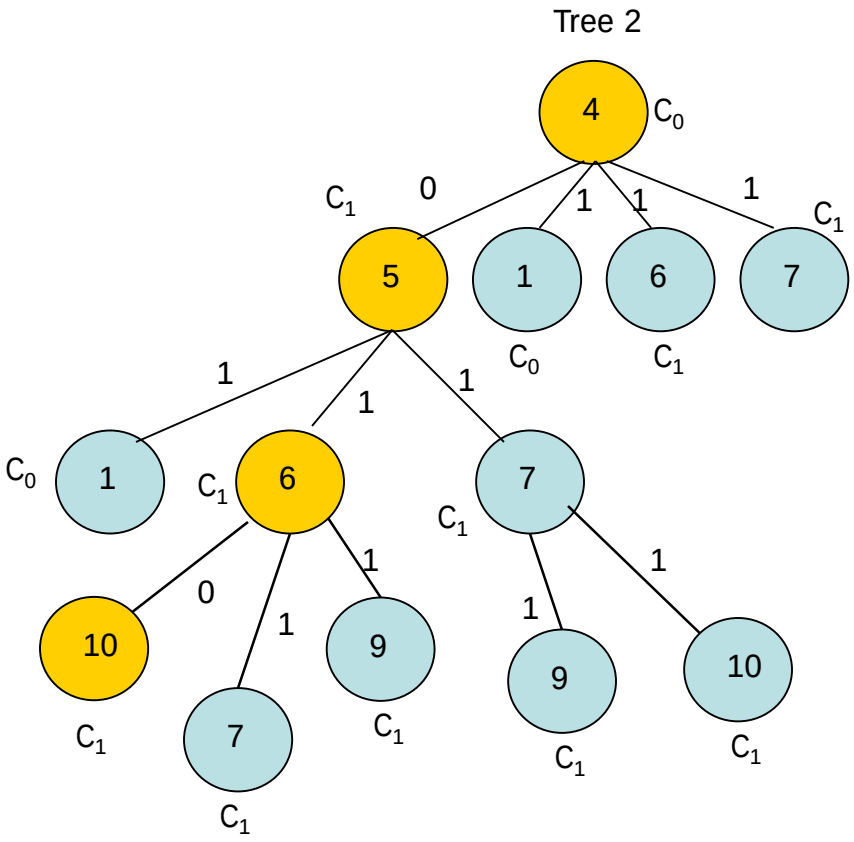
Broadcasting

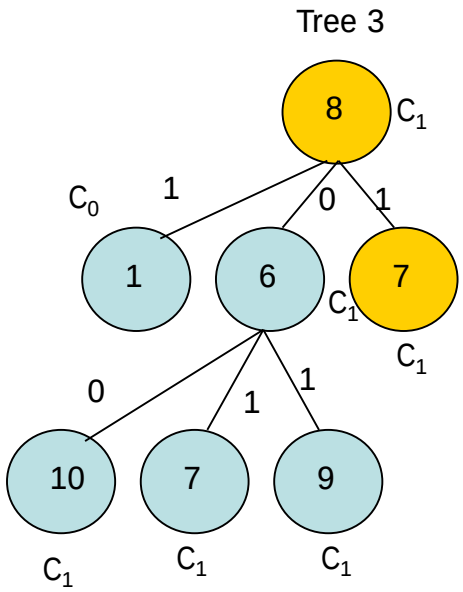
- Broadcasting Contents:
  - Push Based Schemes
    - Flat Broadcast
    - Broadcast Disks
    - Hierarchical Broadcast
    - Square-Root Based Allocation
  - Hybrid-Based Schemes
    - Adaptive Data Broadcast
    - Adaptive Push-Pull
    - Dynamic Leveling

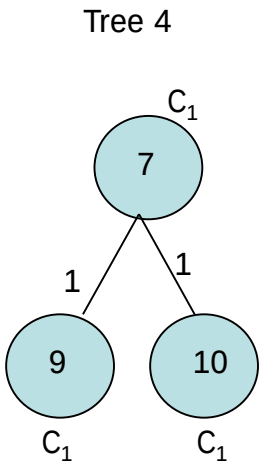

1
2
3
9
4
5
8
6
10
7

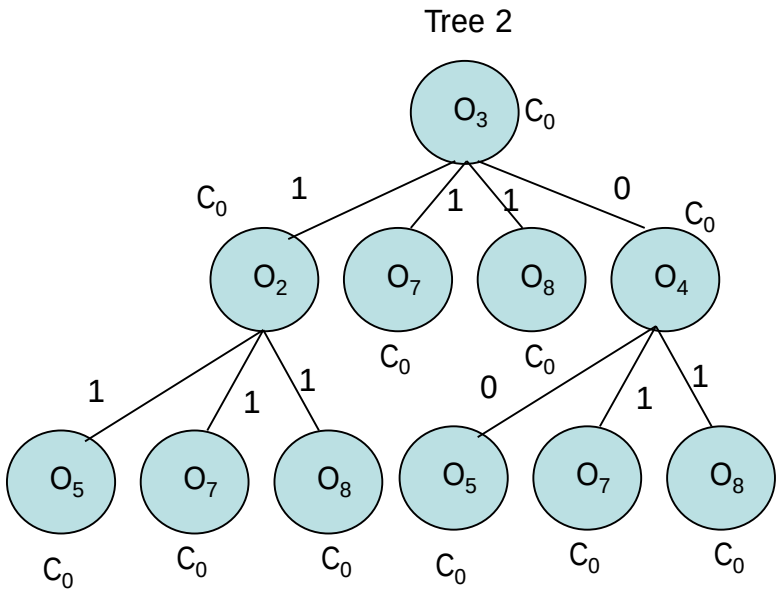
Tree 1




Tree 2
4
C_0
C_1
0
1
1
1
C_1
5
1
6
7
C_0
C_1
1
1
1
C_0
1
C_1
6
7
C_1
1
1
0
1
1
10
7
9
9
10
C_1
C_1
C_1
C_1
C_1


Tree 3
8
C_1
C_0
1
0
1
1
6
7
C_1
C_1
0
1
1
10
7
9
C_1
C_1
C_1


Tree 4
C_1
7
1
1
9
10
C_1
C_1


Tree 2
O3
C0
C0
1
1
1
0
C0
O2
O7
O8
O4
C0
C0
1
1
1
0
1
1
O5
O7
O8
O5
O7
O8
C0
C0
C0
C0
C0
C0