DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

PhD. Manuel Capel Tuñón
mcapel@ugr.es

PhD. Mario Rossainz López
mariorl@siu.buap.mx

Lines of Investigation:

- Theory of the Programming
- Parallel Programming
- Concurrent Programming
- Object-oriented programming

ETSII-DLSI – FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

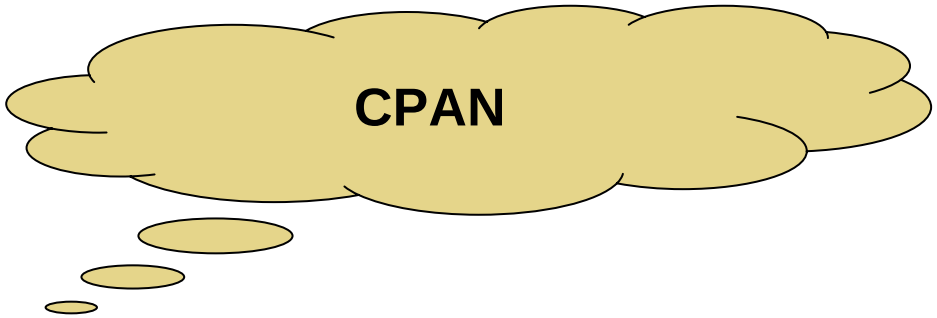
ETSII-DLSI - FCC-BUAP

DEFINITION OF CPAN:

- CPAN *(according to its Spanish acronym)* is the abbreviation of High Level Parallel Coimpositions.
- A CPAN is a parallel pattern of communication that is identified in terms of its components and their communication.
- The CPANs uses the user within the paradigm of Orientation to Objects like abstractions of high level.
- At the moment the CPANs that is had implemented represents the following structures of interconnection of processors:
  - Pipelines
  - Farms
  - Trees Binary with that define the technique of Divide & Conquer.

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

DEFINITION OF CPAN:



The basic idea is the one of implementing any type of parallel patterns of communication between the processes of an application or distributed/parallel algorithm as classes, following the paradigm from the Orientation to Objects. Starting from these classes, an object can be instanced and the execution of a method of the object in question you can carry out through a petition of service. A CPAN comes from the composition of a set of objects of three types: An Object Manager, the Objects Stage and an Object Collector

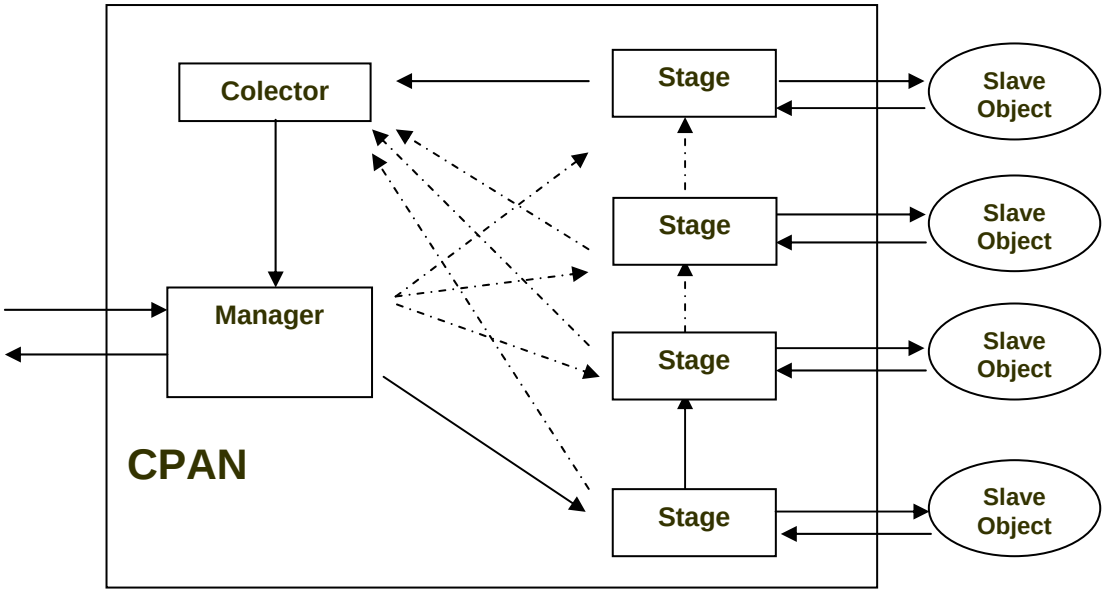
ETSII-DLSI - FCC-BUAP

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

DEFINITION OF CPAN:

Generic Structure of a CPAN (Composition of Objects)



DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

ETSII-DLSI - FCC-BUAP

DEFINITION OF CPAN:

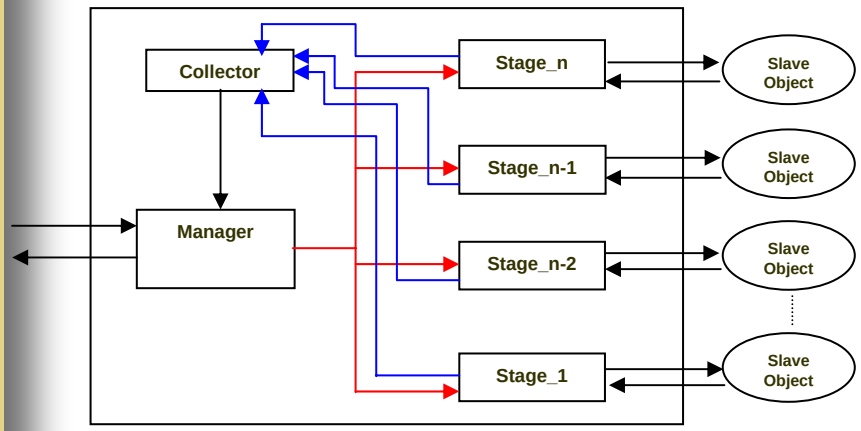
The characteristics of the objects Manager, Stage and Collector are the following ones:

- They are defined as parallel objects, that is to say, objects that have capacity of execution in themselves, structured by state, behavior and policies of planning in operations executed in parallel.
- The types of communication that use are: The synchronous way, the asynchronous way and the asynchronous future way.
- The synchronization restrictions that use are: MaxPar, MutEx and Sync

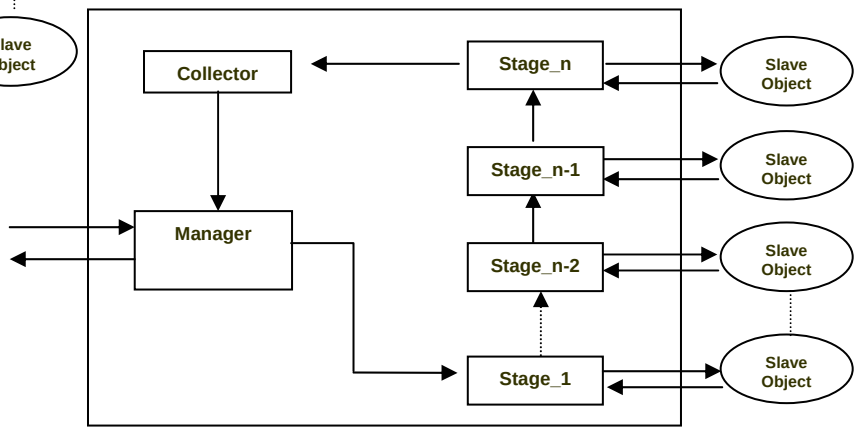
ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

DEFINITION OF CPAN:



The Cpan Farm

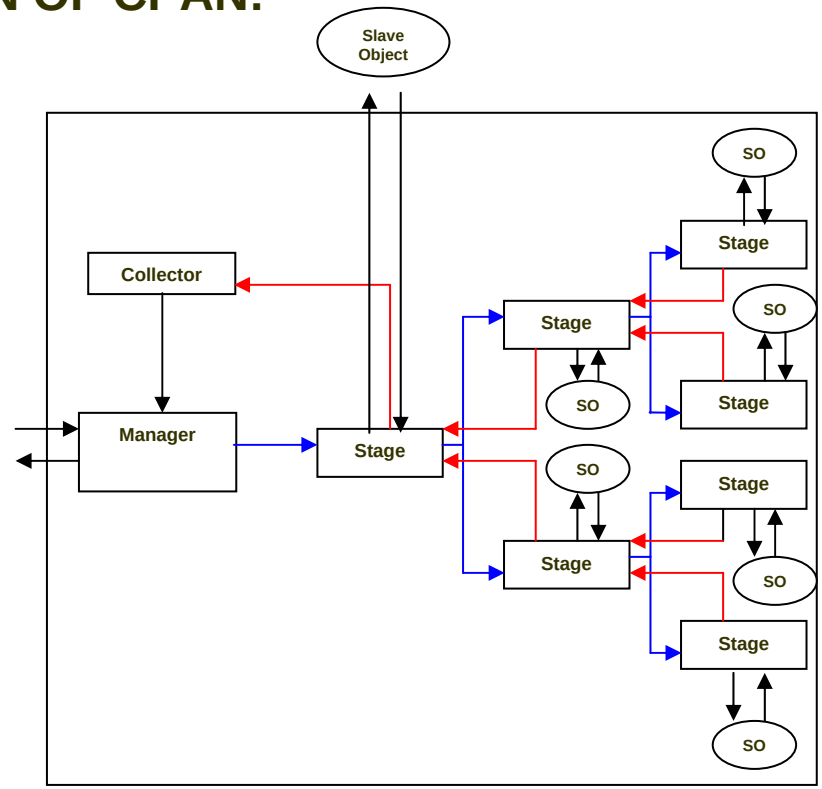


The Cpan PipeLine

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

DEFINITION OF CPAN:



The Cpan TreeDV

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

THE TECHNIQUE OF BRANCH & BOUND:

- It is a algoríthmic technique of design
- It is applied in the solution of optimization problems where the computacional complexity is great
- It makes a partial enumeration of the space of solutions of the problem, in tree form, where the root represents the initial problem or a solution that is far from the optimal one and obtained by means of a heuristic one.
- This technique follows a route of its tree of expansion by means of several strategies:
  - The strategy first in depth (LIFO)
  - The strategy first in width (FIFO)
  - The strategy of the first best (HEAP-LC)

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

THE TECHNIQUE OF BRANCH & BOUND:

1. **Stage of Selection:** A node of between the set of the nodes is extracted that have not been pruned and that have the possibility of being branching.
2. **Stage of Ramification:** Children of the selected node are constructed, forming therefore the tree of expansion (space of solutions)
3. **Stage of Pruning:** Some of the nodes created in the previous stage are eliminated, those whose partial cost is greater than the best calculated minimum level until that moment, for another node already explored.
4. **Stage of Conclusion:** One obtains the optimal solution of the problem or the algorithm finishes when the set of nodes with possibility of being explored is exhausted

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

THE TECHNIQUE OF BRANCH & BOUND:

1. **Contribution:** It produces the diminution of the space search and, therefore, the attenuation of the computacional complexity.

2. **Difficulty:** To find a good function of cost for the problem that is tried to solve ("good" in the sense that it guarantees the early pruning and that their calculation is not very expensive).

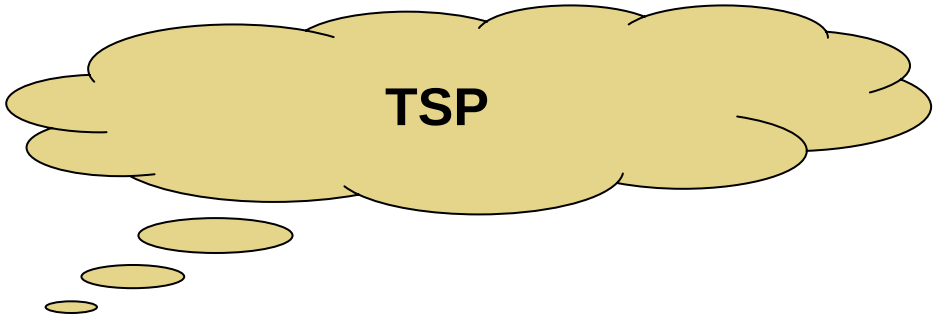
3. ***Classic problems (of optimization and search):***

   a) ***The Travelling Salesman Problem***
   b) The Problem of the knapsack
   c) The Problem of the n queens
   d) The Problem of the distribution of tasks

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

THE TRAVELLING SALESMAN PROBLEM (TSP):



The distances between a certain number of cities are known. An Travelling Salesman must, from one of them, to visit each city exactly once and to return to the departure point having crossed the smaller possible distance altogether.

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

THE TRAVELLING SALESMAN PROBLEM (TSP):

1. *An algorithm of polinomial time is not known to solve the problem.*

2. *The TSP is classified like a NP-Complet problem*

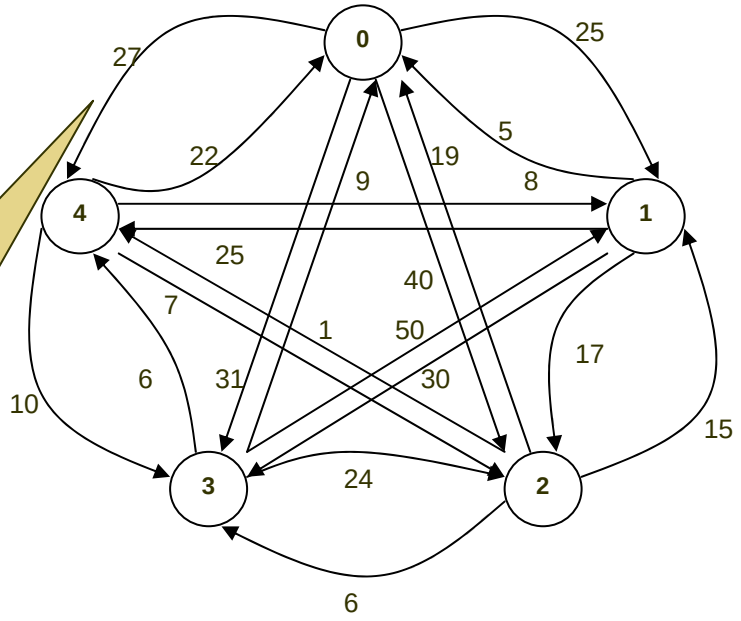
3. *The TSP can be solved using technical varied of the design of algorithms:*

    a) *Dinamic Programming*
    b) *Genetic Algorithms*
    c) *Eager Algorithms*
    d) *Return back*
    e) *Branch & Bound*

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

THE TRAVELLING SALESMAN PROBLEM (TSP):

Formally, the problem can be enunciated as follows: given a connected and weighted graph g and given one of its vertices v0 as the initial one, we must find the Hamiltonian cycle of minimum cost that begins and finishes in v0



DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

PARALELIZATION OF BRANCH & BOUND LIKE A CPAN:

Schemes for the Paralelización of the algorithms of Branch & Bound:

- Schemes of Shared Memory

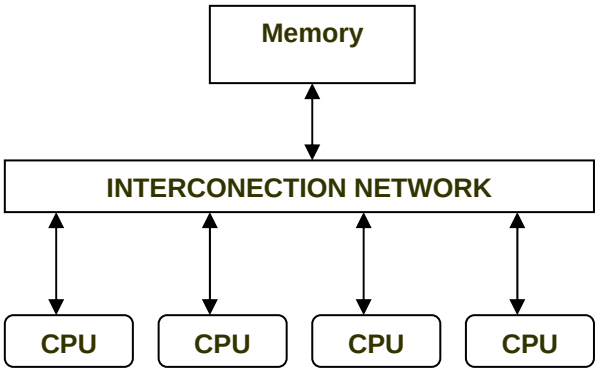


Fig. I.5. (a) Scheme NUMA

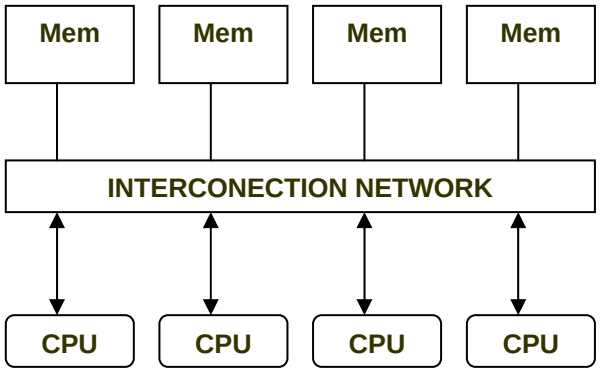


Fig. I.5. (b) Scheme UMA

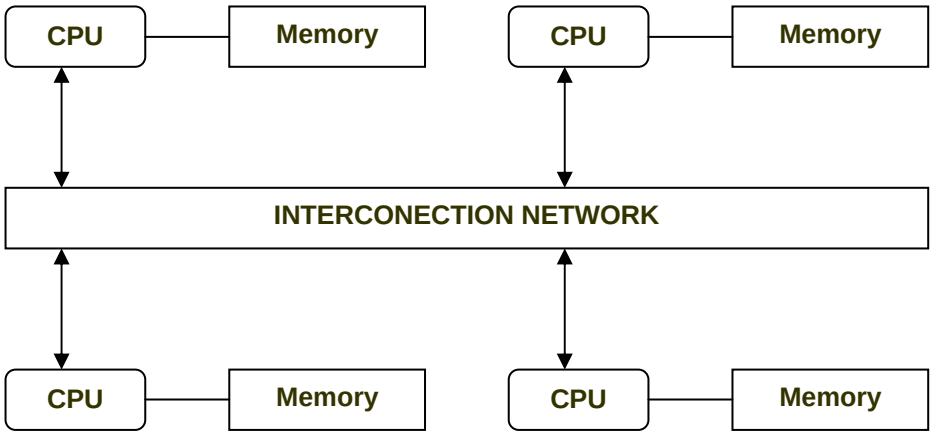
ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

PARALELIZATION OF BRANCH & BOUND LIKE A CPAN:

Schemes for the Paralelización of the algorithms of Branch & Bound:

- Schemes of Distributed Memory

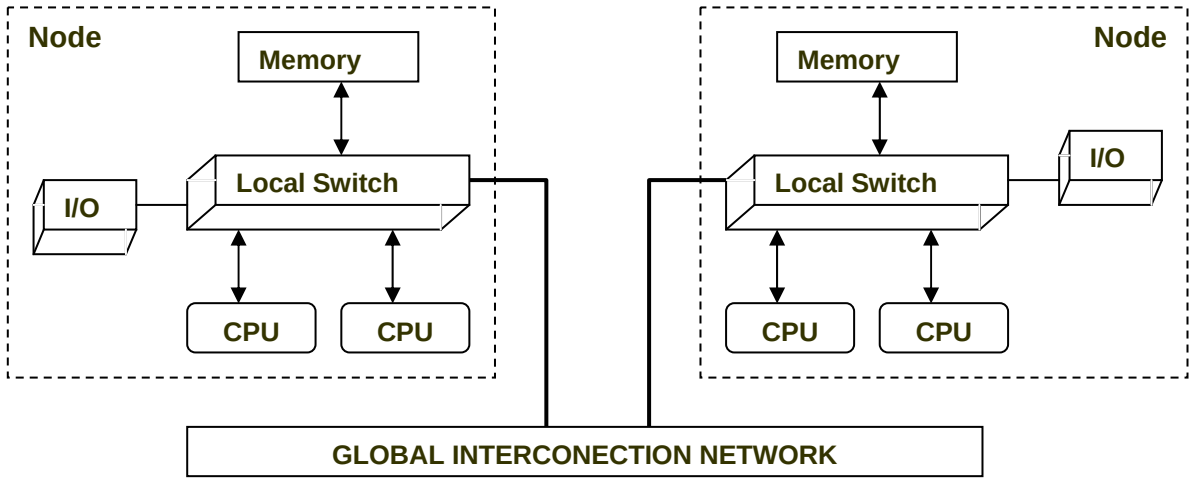


DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

PARALELIZATION OF BRANCH & BOUND LIKE A CPAN:

Schemes for the Paralelización of the algorithms of Branch & Bound:

- Schemes of Shared Memory Distributed



DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

PARALELIZATION OF BRANCH & BOUND LIKE A CPAN:

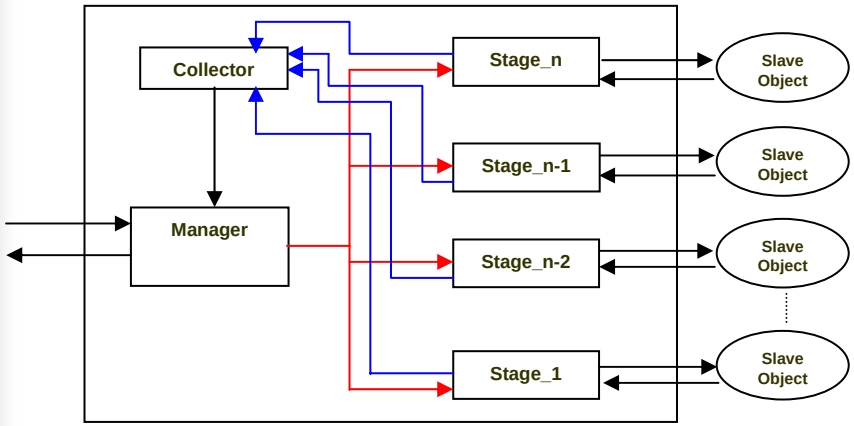
IMPLEMENTATION:

- The used scheme of paralelización is the Shared memory based one Distributed
- The strategy of the first best one (HEAP-LC) for the route of the expansion tree was used
- The idea is to separate the ramification (that is to say, the division and generation of new subproblems) of the pruning (this is, the activity of comparison of the value-levels of function LC, with the best level obtained until that moment)
- These two structures have been implemented using the Cpan Farm already mentioned, so that the ramification and the distribution of the work to the processes this parallel pattern is carried out using.
- The corresponding tree of expansion is created using objects Cpan Farms in a scheme totally connected between the processes

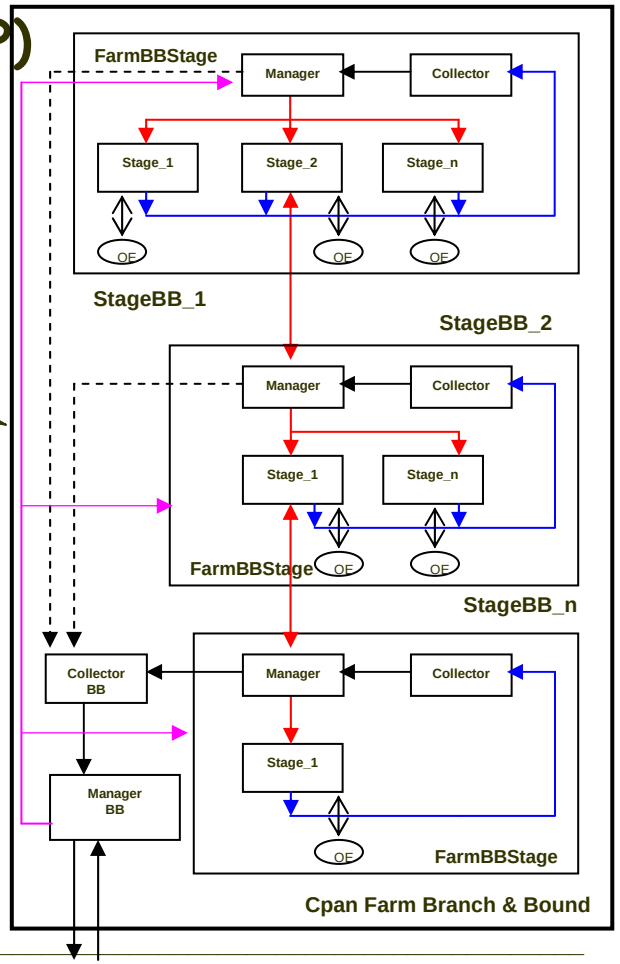
DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

ETSII-DLSI - FCC-BUAP

The Cpan Branch & Bound is composed of a set of *Cpans Farm* that represent worker processes and a controller, therefore, forming a new type of Farm, the *Farm Branch & Bound or FarmBB*



The Cpan Farm

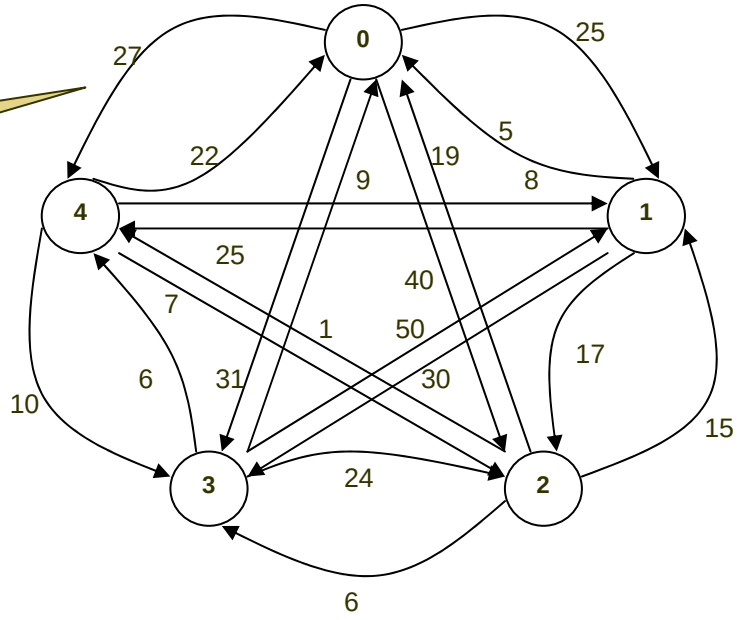


ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

USE OF CPAN FARMBB FOR THE SOLUTION OF THE TSP:

The TSP to solve is defined by the graph of the fig that represents a network of 5 cities that the travelling salesman must visit, each one represented by a node of the graph labeled with a integer number. The directed arcs represent the existing ways to go from a city another one; each one of these labeled with a nonnegative cost

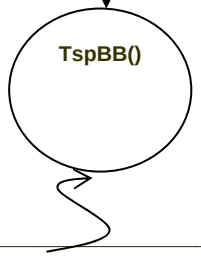


ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

USE OF CPAN FARMBB FOR THE SOLUTION OF THE TSP:

```
CLASS CONCRETE Tsp
  {
   .
   .
   .
    PUBLIC ANYTYPE resuelve(ANYTYPE data)
      {
       .
       .
       .
      }
  };
```



The initial slave object is create like an instance of the class that defines the problem, in this case the one of the TSP. This object will be associated to each stage of the CpanBB

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

USE OF CPAN FARMBB FOR THE SOLUTION OF THE TSP:

```
Object obj[]= {
               Tsp CREATE(. . .)
              };
```

The Slave Object is created and represents the algorithm of the TSP

```
method meth[]={
               method CREATE (. . . resuelve),
              };

asociacion pareja=crea_asociacion(obj,meth,1);

FarmBBManager CpanFarmBB[] = {
                               FarmBBManager CREATE(pareja);
                               . . .
                               FarmBBManager CREATE(pareja);
                              }
```

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

USE OF CPAN FARMBB FOR THE SOLUTION OF THE TSP:

```
Object obj[]= {
                Tsp CREATE(. . .)
              };

method meth[]={
                method CREATE (. . . resuelve),
              };

asociacion pareja=crea_asociacion(obj,meth,1);

FarmBBManager CpanFarmBB[] = {
                               FarmBBManager CREATE(pareja);
                               . . .
                               FarmBBManager CREATE(pareja);
                             }
```

The object is created that represents the associated method the enslaved object

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

USE OF CPAN FARMBB FOR THE SOLUTION OF THE TSP:

```
Object obj[]= {
               Tsp CREATE(. . .)
              };

method meth[]={
               method CREATE (. . . resuelve),
              };

asociacion pareja=crea_asociacion(obj,meth,1);

FarmBBManager CpanFarmBB[] = {
                              FarmBBManager CREATE(pareja);
                              . . .
                              FarmBBManager CREATE(pareja);
                             }
```

A list of associations is created (Slave Object, method)

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

USE OF CPAN FARMBB FOR THE SOLUTION OF THE TSP:

```
Object obj[]= {
               Tsp CREATE(. . .)
              };

method meth[]={
               method CREATE (. . . resuelve),
              };

asociacion pareja=crea_asociacion(obj,meth,1);
```

The instances of the Cpan FarmBB are created initializing them with the list of associations

```
FarmBBManager CpanFarmBB[] = {
                              FarmBBManager CREATE(pareja);
                              . . .
                              FarmBBManager CREATE(pareja);
                             }
```

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

```
int mat_ady[][] = { ∞ ,  25,  40,  31,  27
                     5,   ∞,  17,  30,  25
                    19,  15,   ∞,   6,   1
                     9,  50,  24,   ∞,   6
                    22,   8,   7,  10,   ∞ }

ANYTYPE data[]= { Nodo CREATE(mat_ady) };
```

```
FUTURETYPE resul[num_problems];
FOR i=(0,num_problems)
    resul[i]= THREAD CpanFarmBB[i].execution(
```

```
ANYTYPE resultados[num_problems];
FOR i =(0,num_problems)
   {
    resultados[i]=resul[i];
    imprime_datos(resultados[i]);
   }
```

The initial datas are specified creating a TDA like an object that defines the user

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

ETSII-DLSI - FCC-BUAP

```
int mat_ady[][] = { ∞ ,  25,  40,  31,  27
                    5,   ∞,  17,  30,  25
                   19,  15,   ∞,   6,   1
                    9,  50,  24,   ∞,   6
                   22,   8,   7,  10,   ∞ }

ANYTYPE data[]= { Nodo CREATE(mat_ady) };

FUTURETYPE resul[num_problems];
FOR i=(0,num_problems)
    resul[i]= THREAD CpanFarmBB[i].execution(data[i]);

ANYTYPE resultados[num_problems];
FOR i =(0,num_problems)
   {
    resultados[i]=resul[i];
    imprime_datos(resultados[i]);
   }
```

They execute in parallel the CpanBB by means of a request

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

```
int mat_ady[][] = { ∞ ,  25,  40,  31,  27
                     5,   ∞,  17,  30,  25
                    19,  15,   ∞,   6,   1
                     9,  50,  24,   ∞,   6
                    22,   8,   7,  10,   ∞ }

ANYTYPE data[]= { Nodo CREATE(mat_ady) };

FUTURETYPE resul[num_problems];
FOR i=(0,num_problems)
    resul[i]= THREAD CpanFarmBB[i].execution(data[i]);
```

```
ANYTYPE resultados[num_problems];
FOR i =(0,num_problems)
   {
    resultados[i]=resul[i];
    imprime_datos(resultados[i]);
   }
```

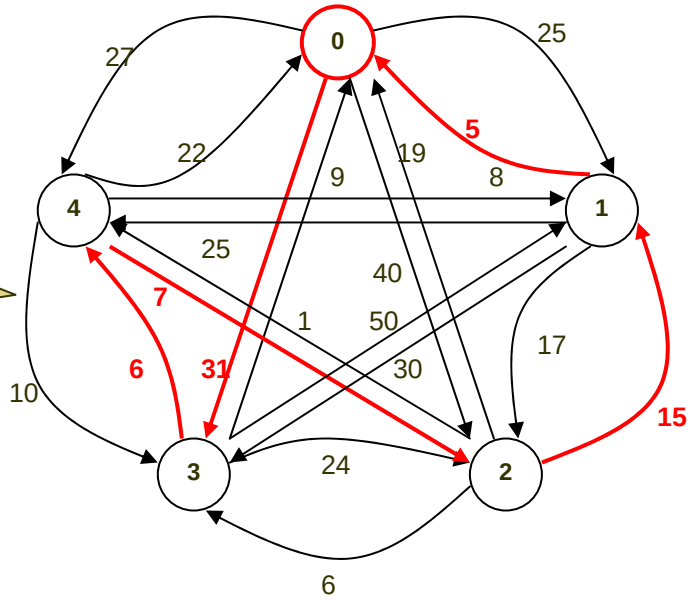
The final results are obtained and printed

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

ETSII-DLSI - FCC-BUAP

The solution that obtained the Cpan FarmBB for the enunciated problem, supposing that the Travelling Salesman part of the city labeled with number 0, was: Node 0 → Node 3 → Node 4 → Node 2 → Nodo1 → Node 0 With a minimum cost of 64



DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

ETSII-DLSI - FCC-BUAP

What calculates the Cpan FarmBB?

1. It finds the best solution of between all the solutions of a given problem.
2. It obtains the number of generated nodes (between smaller it is east value, less part of the expansion tree will have had to construct and therefore faster it will be the execution)
3. It obtains the number of analyzed nodes (it is the number of nodes that the CPAN has had to analyze). He is desirable that this number is small since it represents the number of nodes of the expansion tree that are crossed of effective form.
4. It obtains the number of pruned nodes. This value indicates the effectiveness of the pruning. While greater it is east number, less work will have to make the CPAN.

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

ANALYSIS OF SPEEDUP

The analysis of Speedup of the CPANS B&B that appears in the table (following slide), was carried out in a Parallel System Origin 2000 Silicon Graphics (of 64 processors) available in the European Center for Parallelism of Barcelona CEPBA.

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

ETSII-DLSI - FCC-BUAP

RESULTS

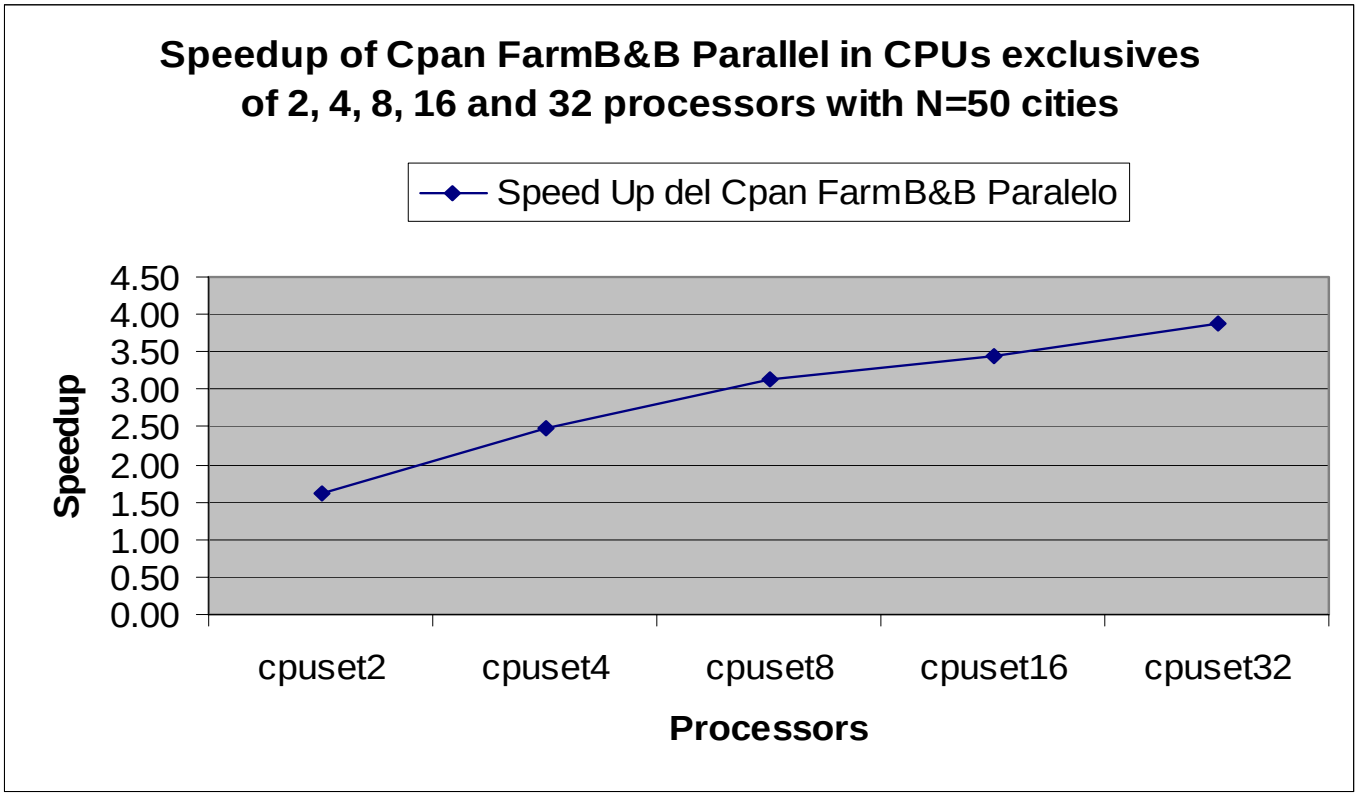
Excecution of Cpan FarmBB Parallel in cpus exclusives of 2, 4, 8, 16 and 32 processors with N=50 cities

	Cpan FarmBB Secuencial	cpuset2	cpuset4	cpuset8	cpuset16	cpuset32
Time of excecution in secs.	*35.42*	21.88	14.21	11.34	10.27	9.10
Time in secs. Usuary mode	***25.80***	**22.14**	**20.16**	**18.16**	**21.67**	**21.14**
Time in secs. System Mode	***1.30***	**1.11**	**1.01**	**1.03**	**1.02**	**1.04**
Time in secs. of CPU	*27.10*	*23.25*	*21.17*	*19.19*	*22.69*	*22.18*
Number of cycles	**266948719**	**69877161**	**69073500**	**67663433**	**66942422**	**65992570**
Number of instructions	**202106112**	**733380571**	**73219450**	**72932731**	**72454022**	**72235919**
CPI	*1.321*	*0.952*	*0.943*	*0.928*	*0.924*	*0.914*
Speedup	*1.00*	*1.62*	*2.49*	*3.12*	*3.45*	*3.89*
Amdalh	*1.00*	*1.68*	*2.55*	*3.43*	*4.16*	*4.64*

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

RESULTS:



ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

C O N C L U S I O N S

1.The technique of Branch & Bound as a High Level Parallel Composition or CPAN has been implemented.

2.The utility of the library of CPANS proposed in (Rossainz and Capel 2004) which serves to make compositions of CPANS and to define new CPANS models as in the *Cpan Branch & Bound* has been demonstrated.

3.With the model of the *Cpan Branch & Bound* we have been able to offer an optimal solution of a TSP NP-Complete problem.

4.The CPANS Pipe, Farm, TreeDV and Farm-Branch-&-Bound constitute the library of classes of the Cpans.

ETSII-DLSI - FCC-BUAP

DESIGN AND USE OF THE CPAN BRANCH & BOUND FOR THE SOLUTION OF THE TRAVELLING SALESMAN PROBLEM (TSP)

THANKS ! ! !