

A novel approach to recognize and extract features from a 2.5D prismatic part for an interactive G-CAPP system

Viswa Mohan Pedagogu¹, Manish Kumar², & Saroj patel³

¹PhD scholar ^{2&3}Associate Professor,
JNVU & JNU - Jodhpur, India

Abstract- The efficient integration of computer aided design, computer aided process planning and computer aided manufacturing systems is very difficult in fully automated manufacturing industries, leading to an open area for research. This paper presents a new technique; the Concentric Contrasting Circles Approach Methodology (3'CAM) has been introduced for extraction of features to generate a variety of modules for a computer- aided process planning (CAPP) system. The 3'CAM is a novel mathematical approach technique developed to surmount the difficulties in existing interface and interactive methods. A 2.5D prismatic part has been adopted for the research work and feature coordinates as input under various constraints. This technique is employed as a powerful tool to build a flexible, user friendly, an uncomplicated and expandable with reduced time and cost requirements compared with other proficiencies for interactive G-CAPP system. This technique can be applied instead of interface systems similar to Boundary representation (B-Rep) or constructive solid geometry (CSG) for feature recognition and extraction. In this paper, feature recognition, extraction methodology has been introduced and demonstrated with a simple part. The extraction of features is developed by using Visual Studio at front end, supported by SQL-Server and C# software at back end respectively.

Keywords — Computer Aided Design, Computer Aided Process Planning, Computer Aided Manufacturing, Concentric Contrasting Circles Approach Methodology and Feature

ABBREVIATIONS AND ACRONYMS

LCL - Locus Check Loop

CCL - Concentric Contrast Loop

CDM -Circum Dia Module

CCM - Circum Centre Module

DBM - Drill Bit Module

++ - Variable radius for incrementing of concentric circle pair.

l - *Node* is a connection point that represents the distance between local co-ordinates [x, y, z] w.r.t. origin [X, Y, Z]

a - Radius of the circle

(a++) - Incremental radius with l

m - A variable, assigned for representation of a node

n - Denotes the total no of nodes

p - Nearest node or centre of the circle

h - Nearest node distance from origin in x-axis

k - Nearest node distance from origin in y-axis

x_m - Node distance along with the x-axis (variable)

y_m - Node distance along with the y-axis

z_m - Node distance along with the z-axis by default, $z=0$ later on equal to T

T - Thickness of work piece

L - Length of the work piece

I. INTRODUCTION

The core task in any modern industry is to manufacture a product with uncompromised quality, justifiable cost, and timely delivery of the manufactured goods. Owing to the increased global competition, modern industry has been forced to utilize the latest implementations and updated technologies instead of traditional approaches to

acquire and bring forth fresh merchandise in the most effective way. Computer Aided Process Planning (CAPP) system is an Information Processing System (IPS) can avail to shorten planning time and increase consistency and efficiency. On the other hand, the demand for high productivity and character is inevitable owing to the challenges of global competition. This in turn determines the need to shorten the Product Development Life Cycle (PDLC). A hugely diverse customized manufacturing system that can produce a part within a stipulated time has been developed. [1]

In recent years, the integration of Computer Aided Design (CAD) and Computer Aided Manufacturing (CAM) systems in production cycles has revolutionized modern automated manufacturing industries. Moreover, CAPP, a key step in the PDLC, has attracted much research interest. However, correct sequencing of machining features still remains a challenge in CAPP owing to its complexity and adherence to the class of non-deterministic polynomial-time problems [2]. In the most previous research, the task involves reading, recognition and identification of design specifications and interprets into appropriate machining information that is required for product fabrication. In this regard, the purpose of a characteristic based information model has been seen as an adequate technology to join the engineering domains and various computer systems. Consequently, the application of feature-based process planning has gained much attention in supporting an efficient integration of CAD/CAM.

The problem of feature interactivity has led to the proposition of a characteristic based approach for generating hole series machining features from a design feature model. The hole series features are important machining features for gear box components used in machine tools, difficult and face challenges in creating a feature database. A restraint-based method has been evolved to fix a hole-series feature model, based on geometric and topological information extracted from the design data base.

Moreover, a Standard for Exchange of Product (STEP) file is generated to link the machine features to downstream activities in the CAPP environment by making a prototype CAPP system. A study of existing literature indicates that despite substantial progress, events linked to system integration of CAD and CAPP in particular, have not even been fully resolved.

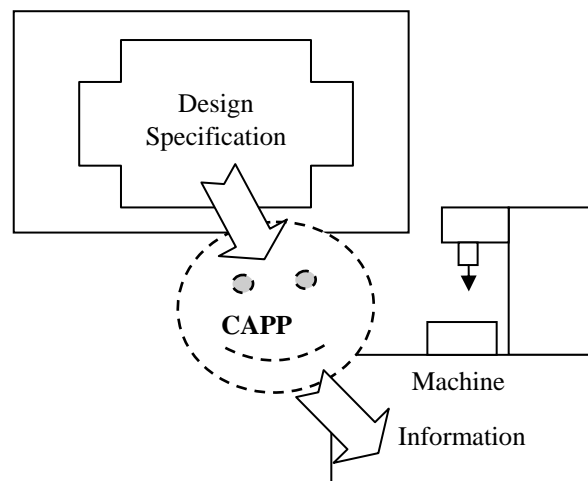


Fig.1: Basic function of a CAPP System

A lot of the current research endeavour is dedicated to feature recognition, which acts as a significant part in the integration of CAD, CAPP and CAM [3]. An adequate system has been focused using a rule based method and particular algorithm to realize a form of non-intersecting features and convert these into machining features. However, it was restricted to some unique features and not projected to all machining features [4]. An up-to-date survey has been introduced along with graphical representation for easy understanding of the past, present, and future of CAPP system with its advances, methods/technologies, discussions, and conclusions [5]. The subsystems of computer integrated manufacturing include computer aided design, computer aided process planning, computerized numerical control, computer-aided quality assurance, computer-aided robot control, computer-aided inspection and planning, computer-aided assembly computer-aided transport and storage finally, computer-aided maintenance.

Many commercialized software solutions and industrial arrangements are presently available to help the integration of subsystems with one another to achieve optimization [6]. Furthermore, the cost and time of delivery of the product impacts on dimensions of the product such as 2-dimensional, 2.5-dimensional and 3-dimensional machining. The 2-dimensional part is composed of a series of two dimensional (2D) paths in the XY plane. Whereas, the 2.5-dimensional part approaches composed of XY plane at constant Z levels or through. Here, the depth of cut is equal to thickness of the part. Similarly, 3-dimensional parts composed of XYZ planes.

Various mechanical components are manufactured by milling, drilling, and other machining process of 2.5D prismatic components. Rough milling is responsible for removing most of the material, while the finishing stage can be performed by machines with up to 5 axes, and is achieved by the 2D part. High material removal obviously reduces production lead time, there by speeding up the fabrication procedure and providing flexibility. The probe has been taken out along the advantages of getting a tool path for a 2.5D milling operation by comparison with 2D and 3D process [7].

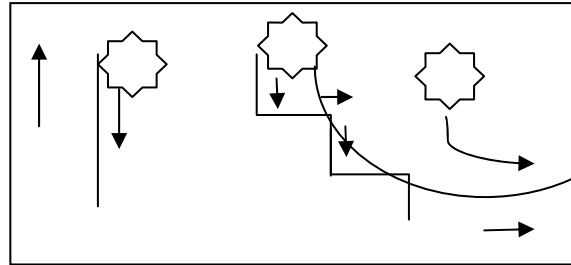


Fig. 2: Variety of dimensional machining

A several systematic steps involved in up-to-date approaches are 1.Design, 2. Identification of features, 3.Selection of processes, tools and machines, 4. Machining and inspection. Machining features generated in the design phase are recognized and stored using visual basic in the CATIA software active X interface. The algorithms have been built-up for individual features, and are embedded in visual basic courses. Although, this arrangement is suited to the 2.5D, part approach it can be extended to 3D prismatic part and complex feature machining. Eventually, a process planning chart is introduced for model process planning [8]. Global competition among manufacturing industries has increased the need to shorten production lead times and increase the quality of merchandise. Attempts at automating the processing of geometric data and scaling down the manual effort in managing complex and huge geometries persistently been pursued over the two decades. This quest has resulted in a drastic improvement in CAD systems that facilitate the integration of complex geometries into the design models. Algorithms play an important function in tool generation and machining of prismatic parts. This full procedure has been executed as a case study considering a prismatic part, while various parameters such as machine time and power are determined using creo/pro-e design software. The results of simulating the prismatic part using depth 1 CNC milling software have also been demonstrated [9]. Nevertheless, the software are expensive, time consumed and difficult to understand.

A novel optimization algorithm, the cuckoo search algorithm (CSA), has been suggested to solve manufacturing optimization problems. Comparison of the results of solving a milling optimization problem using the CSA with those obtained using other well-known optimization techniques such as ant-colony algorithm, the immune algorithm, hybrid immune algorithm, hybrid particle swarm algorithm, genetic algorithms, and feasible direction method, establish that the CSA is really efficient and robust plan of attack for solving machining optimization problems, Though, the CSA has not feasible and suitable for 2.5D prismatic part.[10].

A rule based approach has been proposed to solve the major feature recognition extraction problems, of geometric primitives from a CAD model, however the method was restricted to faces, borders and soon Further, A unique approach has been introduced using graphic interactive programming and a unigraphics solid modeller. Further, other researches introduced a method that utilizes feature-based modeling to define a pre-processing plan that specifies the required capabilities at a high level helps to the feature reorganization for the both geometric and non-geometric information nevertheless, identified as expensive method. A novel case retrieval method has been proposed for rotational parts to combine feature and characteristics of partial information to reuse process characteristics of manufacturing process information models. Further than, a feature detection algorithm method has been produced for a stereo-lithography parts specification [11-15].

An state of art algorithmic approach has been proposed for observe thirty three variety features based on their geometry, curvature, location, orientation and process parameters within the expert CAPP system for single point incremental forming along with feature coding developed feature based modeling method in a process planning environment. Although, the method was well proved but difficult in understanding. A feature conversion algorithm for coaxial hole series machining features was also proposed based on the design feature model. The converted machining feature model can be changed to a downstream CAPP application through the developed STEP file interface. Blended with our planner type form feature conversion and dimension and attribute feature conversion algorithms, the complete machining feature model can be gendered from the design feature model and can also be transmitted directly to a CAPP system through the STEP file. Implemented in the pro/engineer environment on a sun Sparc-20 workstation, using the pro/develop tool. Nonetheless depending on the downstream application, it may be necessary to convert design features into machining features through feature conversion or feature extraction. Feature extraction defines machining features from a geometry-based

design model, whereas feature conversion converts machining features from a feature-based design model nevertheless, the method was not able to convert all features [16-19].

II. FINDINGS OF PRECEDING RESEARCH

Conversely, the proposed methodologies are intensely acknowledged costly and complex in nature. In addition to that most of them difficult to understand and development of a software. In order to overcome all the erroneous, a solution has been implemented i.e. the concentric contrasting circle approach methodology (3'CAM) system. The 3'CAM technique is more effective in optimizing cutting parameters for milling operations, compared with all other proficiencies. Initially, the method is proposed for development of CAPP module that based on feature and 2.5D prismatic part. However, the entire proposal can be extended for 3D and rotational parts.

The feature based modeling has been proved itself as an efficient approach than other proficiencies. Therefore, in the 3C'AM technique has been developed on feature based modelling methodology. After experienced with all the preceding proposals a 3'CAM technique has been initiated to overcome all the messes.

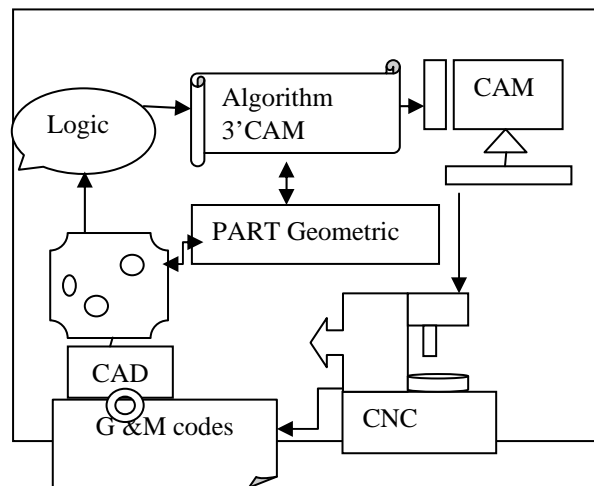


Fig.3.Flow of 3'CAM Technique

III. PROCEDURE FOR PROPOSED METHODOLOGY

In the proposed methodology the features are identified, by considering a CAD model as input. However, it is a trial and error based methodological approach till the result is declared. The task is accomplished by performing a continuous magnification of a couple of concentric circles until the required locus condition at a particular radius from a pre-selected coordinate (node) is met. In addition to this a separate logic has been developed in recognition of each feature depending on their geometry specification and characteristics. A typical logic can be generated automatically after satisfying some initial conditions geared up for individual feature. The entire consistent approach has been divided into three sections, i.e., input, processing and output.

A. Input

Initially, *nodes*¹ from the 2D design of a part are the inputs to the proposed method. Next to that a necessary nodes are selected and the data is stored in a node bank as a data base. The nodes are then arranged in ascending order, conditionally, with the node nearest to the origin (0, 0, 0) placed first and next nearest as second and so on. Conversely, at each iteration, a certain number of nodes are deleted after satisfying the logical condition of the feature. Therefore, the total number of nodes in the node bank available for the iterations is reduced. Eventually, if the number node in the node bank is zero, means all the features of the given part are executed. In addition to that, when the node bank is empty then the process of extraction of feature terminates.

B. Processing

The node nearest to the origin has been considered as the centre of an imaginary circle pair i.e. [a, a+1], with an initial radius (a) value is 1 (by default). Then the Locus check starts by substituting every node in the node bank (Data Base). If the definite conditions are satisfied, then the concern feature is extracted along with the dimensions. However, the total procedure is extended out in two stages.

Phase 1: A trial and error phase used to identify the Circum-center and Circum-radius. The phase 1 further depends on the sort of feature whether it belongs to logical case i.e. Type1 and Type2.

Type 1: If 3 nodes are satisfied with the locus condition for centroid identification, and the final feature identified then the feature is round (circular) or hexagon. For instance,

```

Start
IF_Circle
{3nodes? "Circum_Center"}
{Atleast 12 nodes? "Feature_Recognition"}
{ Atleast 13 nodes? "minimum_input_nodes"}
{'Type1'}
Feature type _ Dimensions
Circle_ 6.0
End
    
```

Case 2: If 1 node is satisfied with the locus condition for centroid identification, and the final feature identified then the feature is triangle, square, or rectangular Phase 2: A recognition of the feature Identification phase, number of nodes on the Circum Circle. After recognition of a feature, the corresponding nodes are deleted from the node bank, and the remaining nodes are arranged in a new series. The new list of nodes is automatically arranged in ascending order and hence, again the first node nearest from the origin (0, 0, 0) is (h, k, 0) denote by letter 'p' then, process repeats.

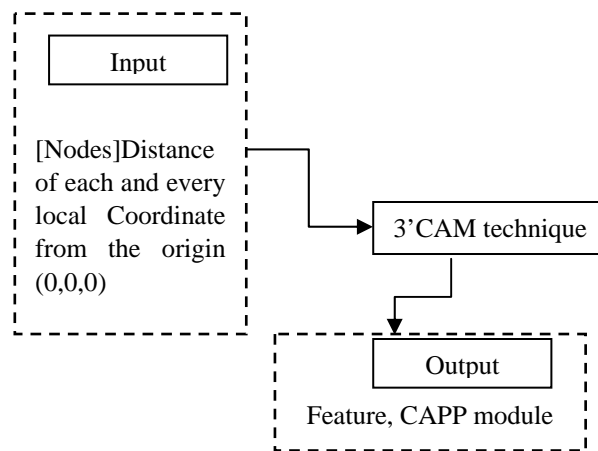


Fig.4 Core-methodology

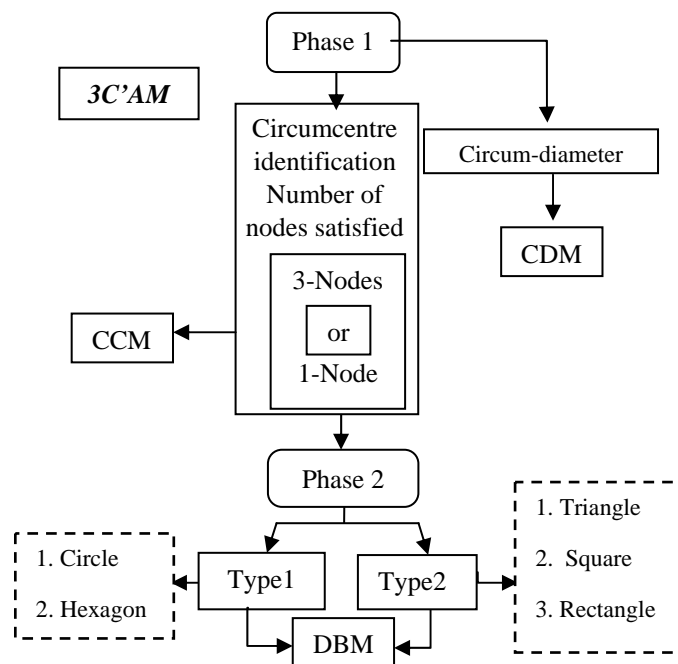


Fig.5. 3-CAM Technique for feature extraction

C. Output

Since, a feature is identified by a logical approach, the output includes the type of feature, its geometric Structure, dimensions, type of operation, and the nature of tool used for the generative- CAPP system.

IV. CONSTRAINTS OF THE 3'CAM TECHNIQUE

1. Symmetrical geometric structures.
2. Pre-determined nodes collection (see table 1)
3. Non-overlapping features
4. The Circum-Centre nodes include input data
5. By default initially, the value of Z (direction) =0
6. 2.5D prismatic part
7. Regular shapes, hexagon, equilateral triangles, circles, and quadrilateral shaped features with various dimensions and orientation can be extracted simply and easily.
5. Required Input data conditions

For type-1 features like a circle, the prerequisite condition is that a minimum of 13 nodes are required, 3 nodes must be satisfied for Circumcenter identification and 12 or more nodes must be satisfied for the recognition of the feature (i.e., a circle). Similarly, for identification of a feature as a hexagon, the minimum number of nodes is 7, with 3 nodes needed for Circumcenter identification and 6 nodes for identification of the feature. For a triangle, the numbers of nodes to be satisfied for Circumcenter identification and feature recognition, nodes required are 1, 3, and 4, (1+3), respectively. The corresponding numbers in a square and rectangle are 1, 4, and 5(1+4) as given in table 1.



Fig. 6. Type of features

V. INFORMATION FLOW SEQUENCE

The information flow sequence of the proposed CAPP system is shown in fig 7. The various modules used for generating the process plan are described in the following section. The required coordinates from the CAD model are the first input to the 3' CAM technique for Generative-CAPP module. Using the list of coordinates (Nodes) from the 2.5D prismatic model, by considering (x, y) plane nodes and where the z = 0 (by default), the 2.5D model can be transformed into 2D. All the nodes are then listed in the node bank. Next, all nodes from the node bank are collected and arranged in ascending order of their distances from the origin and a corresponding serial number is assigned to each node. The decision- making concentric contrast loop determines whether it belongs to type1 (circle or hexagon) or type2 (triangle, square or rectangle) feature.

VI. HIERARCHY OF THE PROPOSED METHODOLOGY

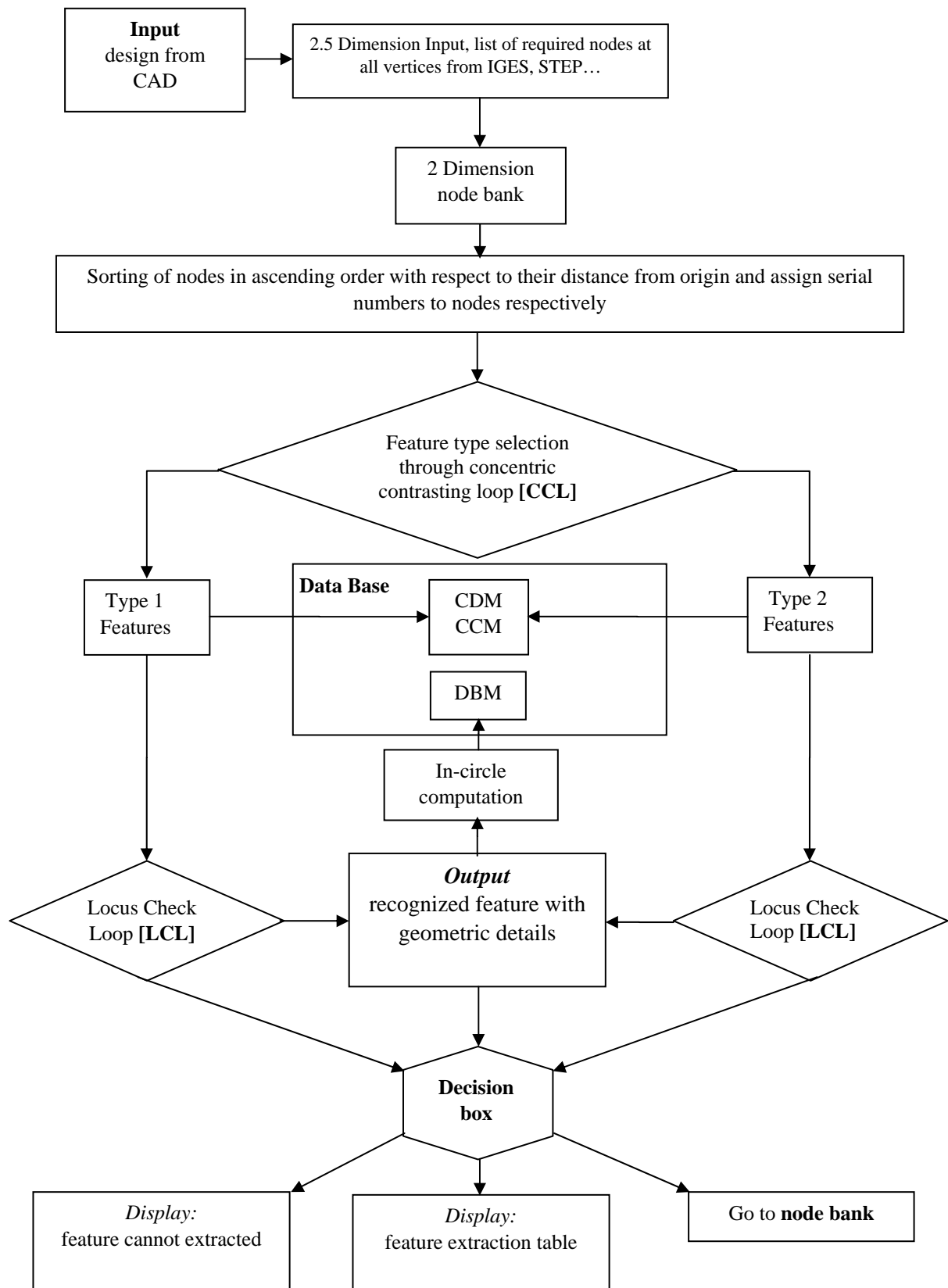


Fig.7. the information flow sequence for CAPP

Table 1. Pre-requisitions for feature identification

S.No	Type	Feature	No. of Nodes should be Satisfied for		No. of Nodes required
			Circum Centre	Feature Recognition	
i	1	Circle	3	≥12	≥13 (≥12 +1)
ii		Hexagon	3	6	7 (6+1)
iii	2	Triangle	1	3	4 (3+1)
iv		Square	1	4	5 (4+1)
v		Rectangle	1	4	5 (4+1)

The features are then held in the locus check loop (LCL), concentric contrasting loop (CCL), for extraction of a feature and Circumcenter Module (CCM), Circum Diameter Module (CDM), and Drill Bit Module (DBM), are needed to develop a CAPP module. Further, detailed algorithms are presented for each iteration and activity. Ultimately, a decision box determines whether the given nodes satisfy the logic. If so, the extraction process terminates and the relevant feature is extracted. At the next level, the machining setup module is invoked. This module covers all the machining details, including the materials, tools, machines, operations, speed, feed, and depth of cut for the CAPP.

VII. FEATURE RECOGNITION ALGORITHMS

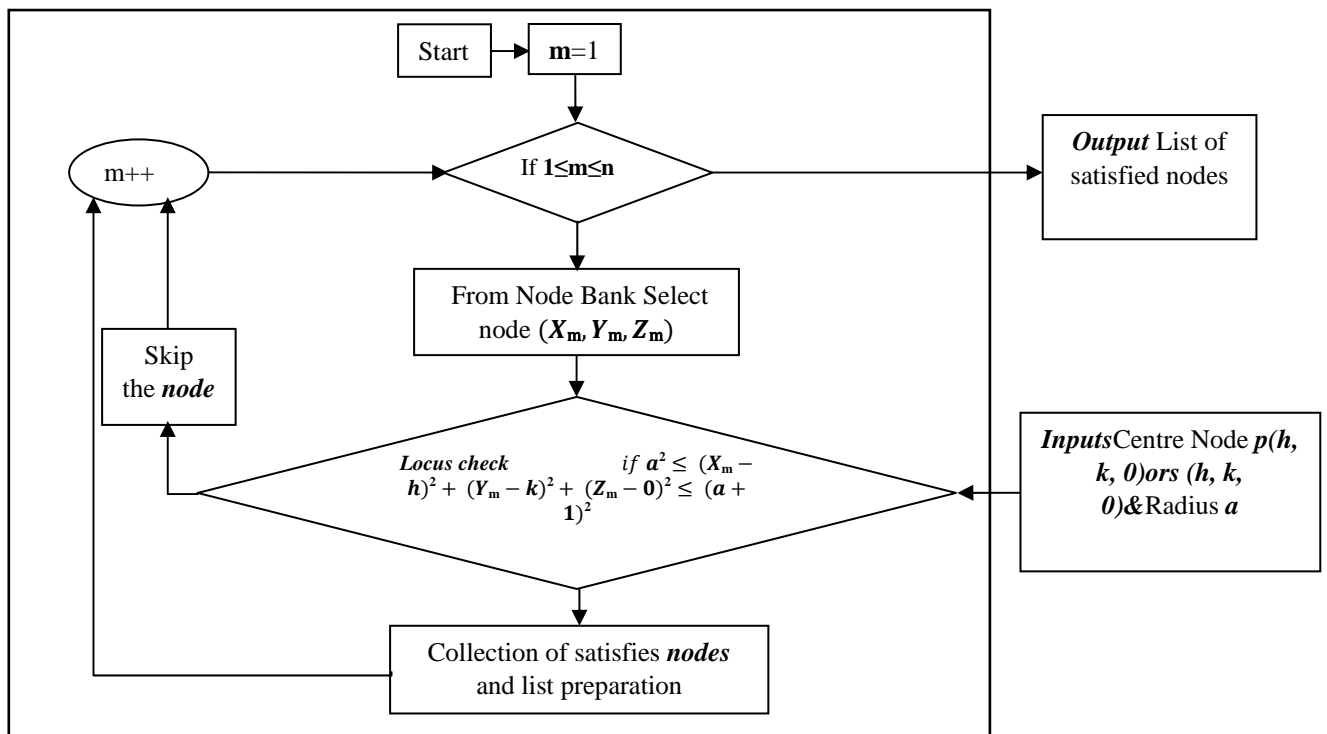


Fig.8. Locus Check Loop [LCL]

A. Locus checks loop

1. Let m and n denote all nodes in the node bank and the maximum number of nodes (total number of nodes), respectively. A node is a common term used to denote the coordinates $(x, y, \text{ and } z)$ as a set. Here the initially, the z value has been considered as zero later on the value equals to the depth of cut or thickness of work piece obviously, in negative and vertically down ward direction.

Example, let the total no of nodes in a node bank is 50 i.e., $n = 50$ and m value varies from 1 to 50 means, $m = 1$ or 2 or 3 or 450. Then,

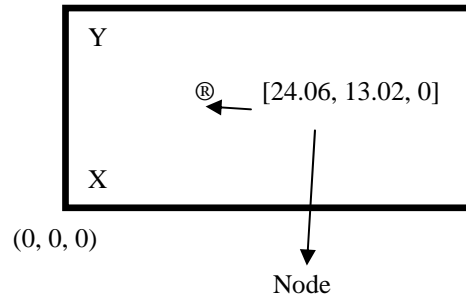


Fig 8. Basic node details

2. Start off the loop by considering the initial value of $m=1$, apply the governing condition, $1 \leq m \leq n$.
3. Produce a list of nodes in ascending order and select the nearest node from the origin, denoted by letter 'p', consider as the center of the circle.
4. Let 'a' denote the radius of the circle; then the locus check is carried out by substituting each and every node.
5. Perform the logical locus condition check modulus operandi for identification of the node

$$\text{Locus check if } a^2 \leq (X_m - h)^2 + (Y_m - k)^2 + (Z_m - 0)^2 \leq (a + 1)^2$$

If any node exists between 'a' and $(a+1)$ radial zone for a particular m value ($m \leq n$) (loop continues), the satisfied nodes are selected and a list is prepared. However, the nodes may be within the circular zone or on the pair of the circles.

6. Output if the number of nodes satisfying the above condition, the node list is executed If, else go to increment 'a' by 1 till $1 \leq a \leq L/4$ then, go to LCL, else go to the next nearest node.

B. Concentric contrasting loop

Implementation of the CCL is initiated by accepting a nearest node as input say, $p(h, k, 0)$ where p is the centre of the circle.

1. An imaginary concentric circle pair is created with radii, 'a' and $(a+1)$.
2. Using a default single assignment with an initial value $a=1$, a radial zone where 'a' as the lower limit and $a+1$ as the upper limit as input to the LCL.
3. In LCL, each active node in the node bank has undergone the locus check, the nodes located about the radial zone $[a, a+1]$ closed interval are recognized and identified.
4. Else, if there is/are no node/nodes within the radial zone, go to the radial increment $(a+1)$. The radius value 'a' should fall within the interval $[1, L/4]$, (where, L is the length of block in x -direction) about the same centre p ; increment the radius by 1. Then If $1 \leq a \leq L/4$, go back to LCL, else deactivate the $p(h, k, 0)$ node and move to STEP #
5. The output of LCL, If 3 nodes are satisfied go to Algorithm-type1 else, if 1 node is satisfied go to Algorithm type2; else if no-node is satisfied, go to radial increment, $a = (a+1)$.
6. If the largest of all the distances of the satisfied nodes, from the parentage is not more capital than the distance of node $S_0 \rightarrow P$ from the beginning, that is, if $S_0 \rightarrow S > S_0 \rightarrow P$, else, radius 'a' is incremented by 1.

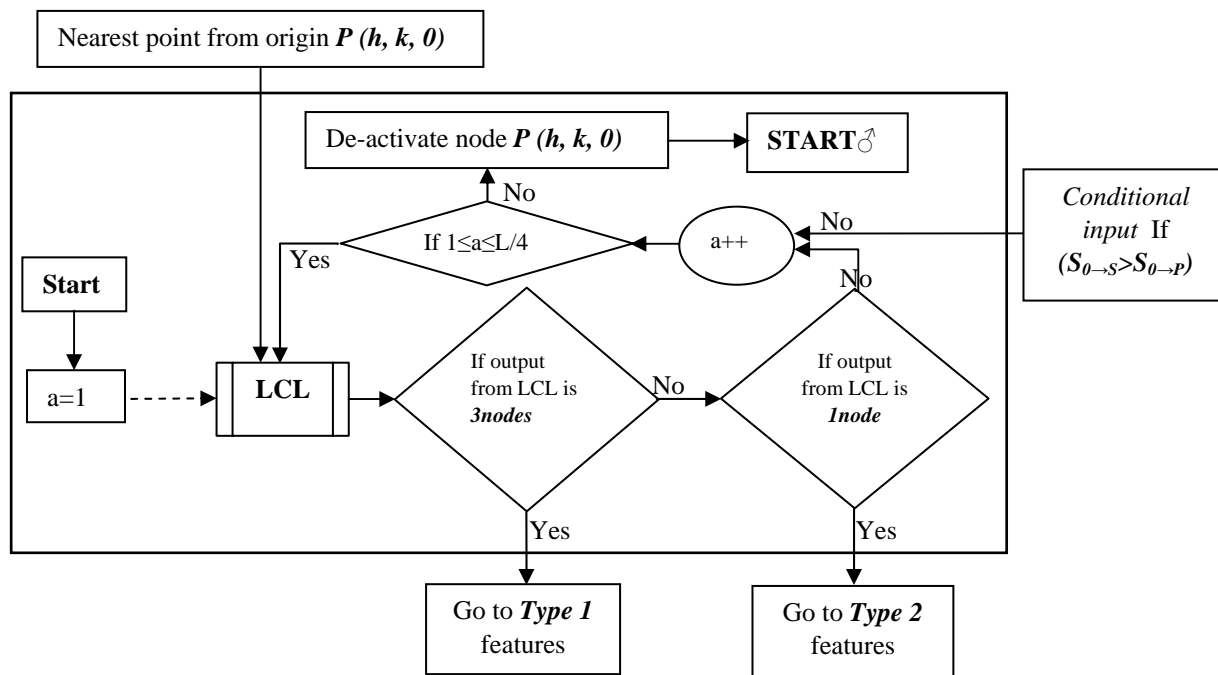


Fig.9 CONCENTRIC CONTRASTING LOOP [CCL]

VIII. EXPLANATION OF SALIENT FEATURES

An artificial intelligence system has been introduced to assure consistency and continuity in the proposed logic, through an error solving prediction and improvement techniques.

A. Decision Box

Decision box plays a keyhole, because,

1. It find outs whether the proposed logic is capable of extracting all features from a given model or design. When the far –off node (nth node) is de-activated and unfortunately if any balanced nodes are in active mode.
2. It controls the number of iterations.
3. When all the features are extracted it displays ‘feature extraction is completed’ and further proceeds to the ‘Machine set up the module’.

B. Conditional Inputs to CCL

There is a probability of any foreign *node/nodes* can be captured while tracing the centroid. However, in such cases a distance check condition i.e., $(S_{0 \rightarrow S} > S_{0 \rightarrow P})$ should be satisfied if else the *logic steps back* to CCL, accepting a radial increment.

C. Radial Upper Limit (L/4)

In economical and machine capability point of view, the most of the CAM applications, a variety of jobs having multi-facet features consists of a variety of centroid or Circumcentre ranging within L/4. Thus, here radius limits are taken as $1 \leq a \leq L/4$.

D. Deactivation Technique

Deactivation means without deleting the nodes later on every iteration, they are temporarily made as dummy nodes. Deactivated nodes will not participate in further iterations. Thus, such nodes will not intervene in the capturing of successive feature.

E. Using 2.5D as an Advantage

The main criteria for 2.5D model is *constant depth for all features*. Here only surface nodes are collected as node bank for feature extraction with *default depth (T)* for all features. Therefore using *2D in-equation* of *concentric circle radial zones*, feature extraction is carried out.

F. Using Closed Intervals of Radial Zones

Since, Circumcentre of a feature may be an integer or a fraction in nature. Instead of finding with a single circle a **concentric pair** is used in tracing of Circumcentre. The node may be on the periphery of circle pair or in between the radii $[a, a+1]$ can de captured.

G. Model Of Feature Extraction Table:

Table operating function ID: FET_[R (1...f...F), C (1.....4)]table call code FET

Table.2. model of Feature information table

C0	C1	C2	C3	C4	
Sr.no	Feature title	Feature size & location in XY-plane	Circumcentre location	Depth of pocket	R0
1					R1

IX.FLOWCHART OF THE PROPOSED METHODOLOGY

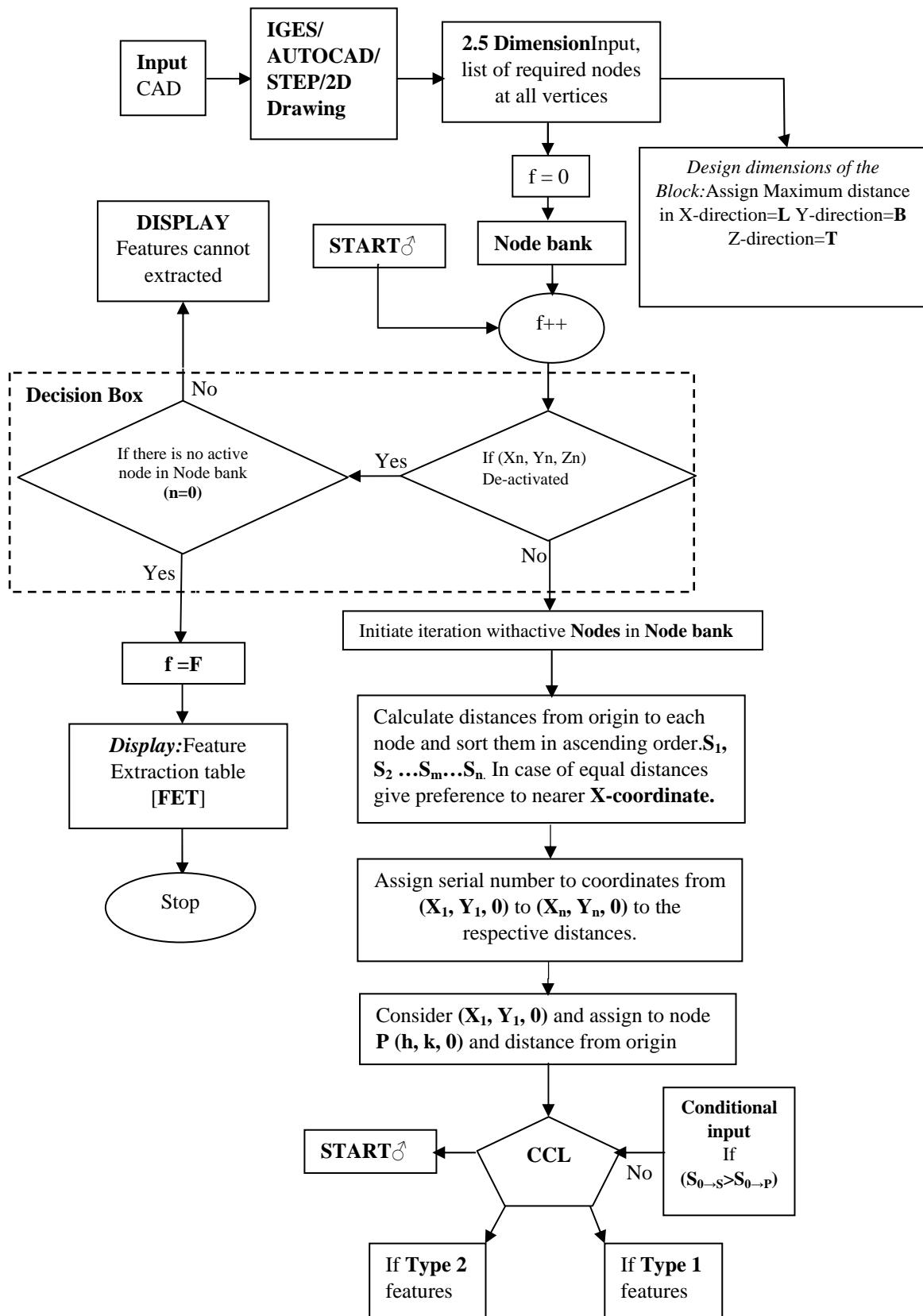
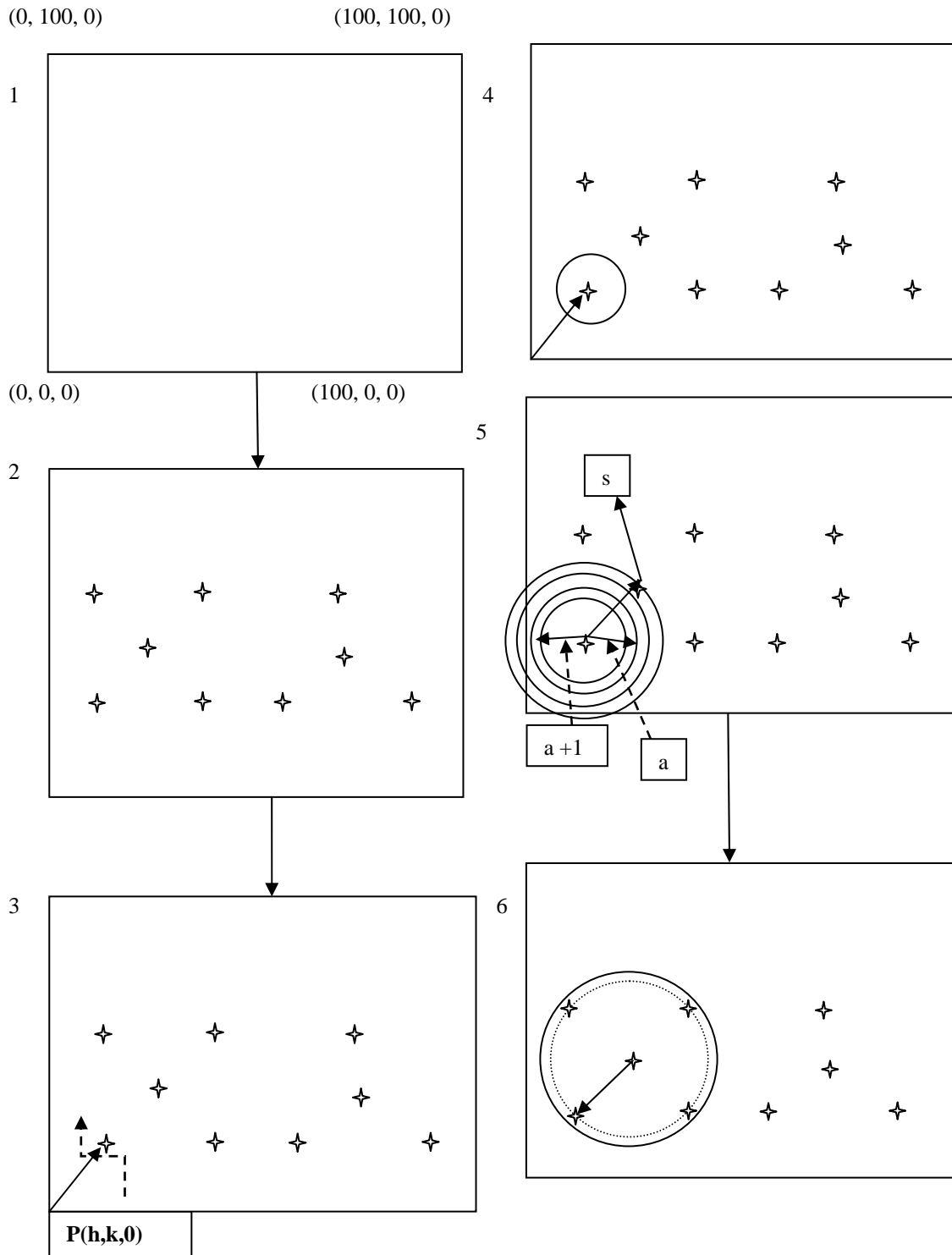
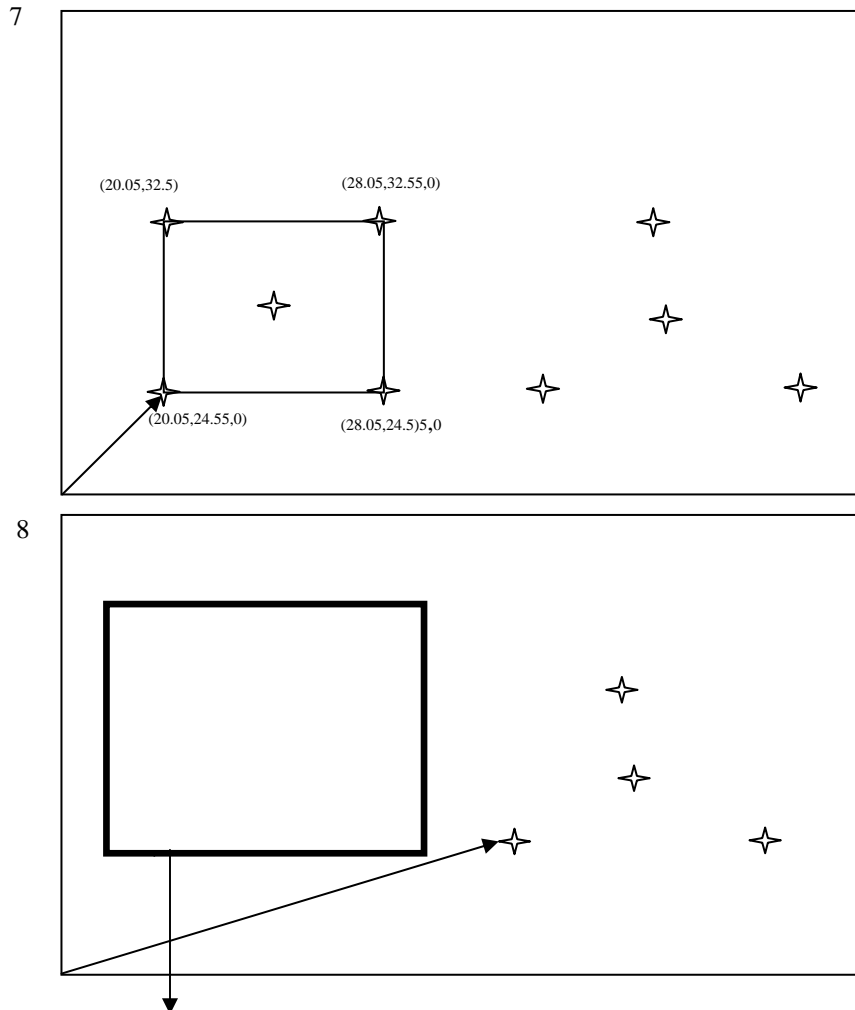


Fig10: flowchart of the proposed methodology

X. PICTORIAL REPRESENTATION OF PROPOSED METHODOLOGY





S.No	Feature	XY-Plane	Circum center	Depth of Cut
1	Square	(20.05,24.55,0) (28.05,32.55,0) (28.05,24.55,0) (20.05,32.55,0)	(24.05,28.55, 0)	-40

Sr`.No	Feature	No of Deactivate Nodes	Number of different values	Circum Dia Module	Dimensions of features	Drill Bit Module
1	Square	5	2	2a	8*8mm	Least of 4C_2 Distance = L_s

Fig.11. Pictorial representation of 3`CAM technique

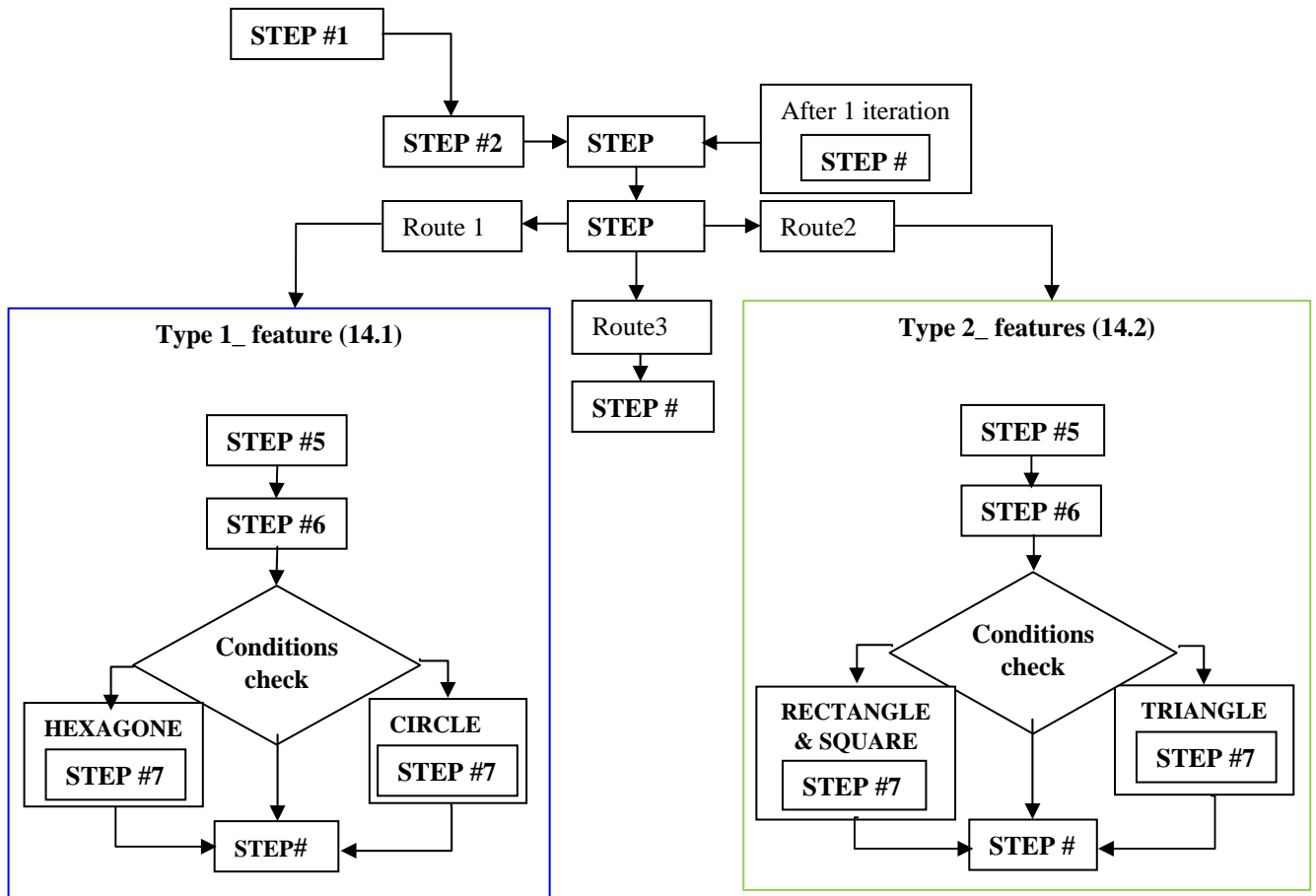


Fig.12 Information flow sequence in proposed methodology

XI. INDEPENDENT ALGORITHM FOR PROPOSED PROCEDURE

STEP #1

- **Input** component design from **CAD**.
- **Input**, all required nodes as per requirement table for **2.5DNode list** preparation.

STEP #2

- Display Maximum distance in **X, Y & Z** is taken as block dimensions **L, B** and **T** respectively.
- Now prepare **Node bank [Z=0]** by default].
- Take total nodes in node bank as **single turn input** for initiation of **iterations**.
- **Go to step 3**

STEP

Iterations begin in this step. From certain terminals of logic, linked to **START** then to **Decision box**.

Decision box:

- **If node (X_n, Y_n, Z_n) is De-activated.**
- **Go to the condition 'If all nodes in Node bank are De-activated $(n=0)$ '.**
- **Else, go to step 3.**
- **If, n^{th} node and all nodes in Node bank are De-activated $(n=0)$.**
- **Display:** feature extraction is completed **go to** machining set up module.
- **Else, Display:** features cannot be extracted.

STEP #3

- Net **Nodes** available in **Node bank** will undergo iteration.
- Calculation of distance from origin to every node and sorting them in ascending order from **S₁, S₂...S_n**, where **n** is number of nodes available in **Node bank**.
- Assign serial number to nodes from **(X₁, Y₁, 0)** to **(X_n, Y_n, 0)** with respective to distance.
- Consider **(X₁, Y₁, 0)** to node **p (h, k, 0)** and distance from origin **S_{0-p}**.

STEP #4

Go to **CCL** for feature **type** selection.

- **If, type1** features are selected then go to **type 1** features algorithm.(14.1)
- **Else, type2** features are selected then go to **type 2** features algorithm.(14.2)
- **Conditional input** to **CCL**, If ($S_{0 \rightarrow s} > S_{0 \rightarrow p}$) is *not satisfied*.

A. Algorithm for type 1 features

If output from **step 4** is **type1**.

STEP #5

- Copy those **3nodes** and calculate individual distances from origin to each **node**.
- Arrange them in ascending order and consecutively assign $S_{0 \rightarrow q}$, $S_{0 \rightarrow r}$, $S_{0 \rightarrow s}$ and respective **nodes** as **q, r and s**.
- If $S_{0 \rightarrow s} > S_{0 \rightarrow p}$
 - Then, Take **node 's'** as centre of circle and assign $s(h, k, o)$ as **new centre**. Store this node in Circum Centre Module (**CCM**). Distance $S_{p \rightarrow s}$ is consider as radius of circum circle and ' $2 * S_{p \rightarrow s}$ ' store in Circum Diameter Module (**CDM**).
- **Else**, go for **increment** to **CCL**.

STEP #6:

- Go to **LCL**.
- In **LCL**,
 - With $a = S_{p \rightarrow s}$ as radius, **node 's'** as centre, **Locus check** is carried out by substituting each and every **node** in radial zone $a, a+1$.

If Output from **step 6** is

$\left\{ \begin{array}{l} \geq 12 \text{ nodes, for step 7, go to Circle (14.1.1).} \\ 6 \text{ nodes, for step 7, go to hexagon (14.1.2).} \\ \text{Other than } \geq 12 \text{ or } 6 \text{ nodes, De-activate nodes (h,k,0), go to step\#.} \end{array} \right.$

B. Circle

Step #7:

- **Display:**
 - In Feature column as **Cylinder**
 - Feature location: Circumcenter $s(h, k, 0)$.
 - Feature size: Hole diameter $d = 2a$.
 - Depth of hole is the thickness of block '**T**'.
- Store ' $2a$ ' in Drill Bit Module (**DBM**).
- De-activate **boundary nodes** and **1 centre node**.

Go to **step#**.

C. Hexagon

Step #7:

- **Display:**
 - Feature: **Hexagon**.
 - Feature location: Circumcenter $s(h, k, 0)$.
 - Feature size: **Satisfied nodes**.
 - Depth of pocket is the thickness of block '**T**'.
- Calculate **node to node** distance in 6C_2 combination and arrange in **ascending order**.
- Smallest distance as **side length (Ls)** and ' $2 * 0.866 * Ls$ ' store in Drill Bit Module (**DBM**).
- De-activate **6 boundary nodes** and **1 centre node**.

Go to **step#**.

If output from **step 4** is **type 2**.

D. Algorithm for type 2 features

STEP #5:

- Copy that **node** and assigns to it. Calculate distance $S_{0 \rightarrow s}$ from origin.
- **If** [$S_{0 \rightarrow s} > S_{0 \rightarrow p}$].

- Then Take node 's' as centre of circle, $s(h,k,0)$ as new centre and Store this *node* in Circum Centre Module (CCM). Distance $S_{p \rightarrow s}$ is consider as radius of circum circle
- Store $2 * S_{p \rightarrow s}$ in Circum Diameter Module (CDM).
- **Else**, go for increment to CCL.

STEP #6:

- With **input s (h, k, 0)** as centre and radius $a = S_{p \rightarrow s}$. Go to LCL.
- In LCL.
 - With '**a,a+1**' as radial zone, nearest node from origin 's' as centre Locus check is carried out by substituting each and every remaining *nodes*.

If Output from **step 6** is

{	<p>3 nodes, for step 7, go to Triangle (6.2.1).</p> <p>6 nodes, for step 7, go to Rectangle & Square (6.2.2).</p> <p>Other than 3 or 4 nodes, De-activate <i>nodes (h,k,0)</i>, go to step#.</p>
---	--

E. Triangle**STEP #7:**

- **Display:**
 - Feature: **Triangle**.
 - Feature location: Circumcenter $s(h, k, 0)$.
 - Feature size: **3 satisfied nodes**.
 - Depth of pocket: '**T**'.
- Calculate *node to node* distance in 3C_2 combination and mode is taken as side length (**Ls**).
- Store $Ls/\sqrt{3}$ in **DBM**.
- De-activate **3 boundary nodes** and **1 centre node**.

Go to **step#**.**F. Rectangle and Square****STEP #7**

- Calculate *node to node* distance in 4C_2 combination and arrange in **ascending order**.
- **Least distance** is taken as side length (**Ls**).
- Store '**Ls**' in **DBM**.
- **If** the number of different distances = **3**,
 - Display:**
 - Feature: **Rectangle**.
 - Feature location: Circumcenter $s(h, k, 0)$.
 - Feature size: **4 satisfied nodes**.
 - Depth of pocket: thickness of block '**T**'.
- De-activate **4 boundary nodes** and **1 centre node**.

Go to **step#**.**Else,**

- **If** number of different values = **2**.
 - Display:**
 - Feature: **Square**.
 - Feature location: Circumcenter $s(h, k, 0)$.
 - Feature size: **4 satisfied nodes**.
 - Depth of pocket: thickness of block '**T**'.
- De-activate **4 boundary nodes** and **1 centre node**

Go to **step#**

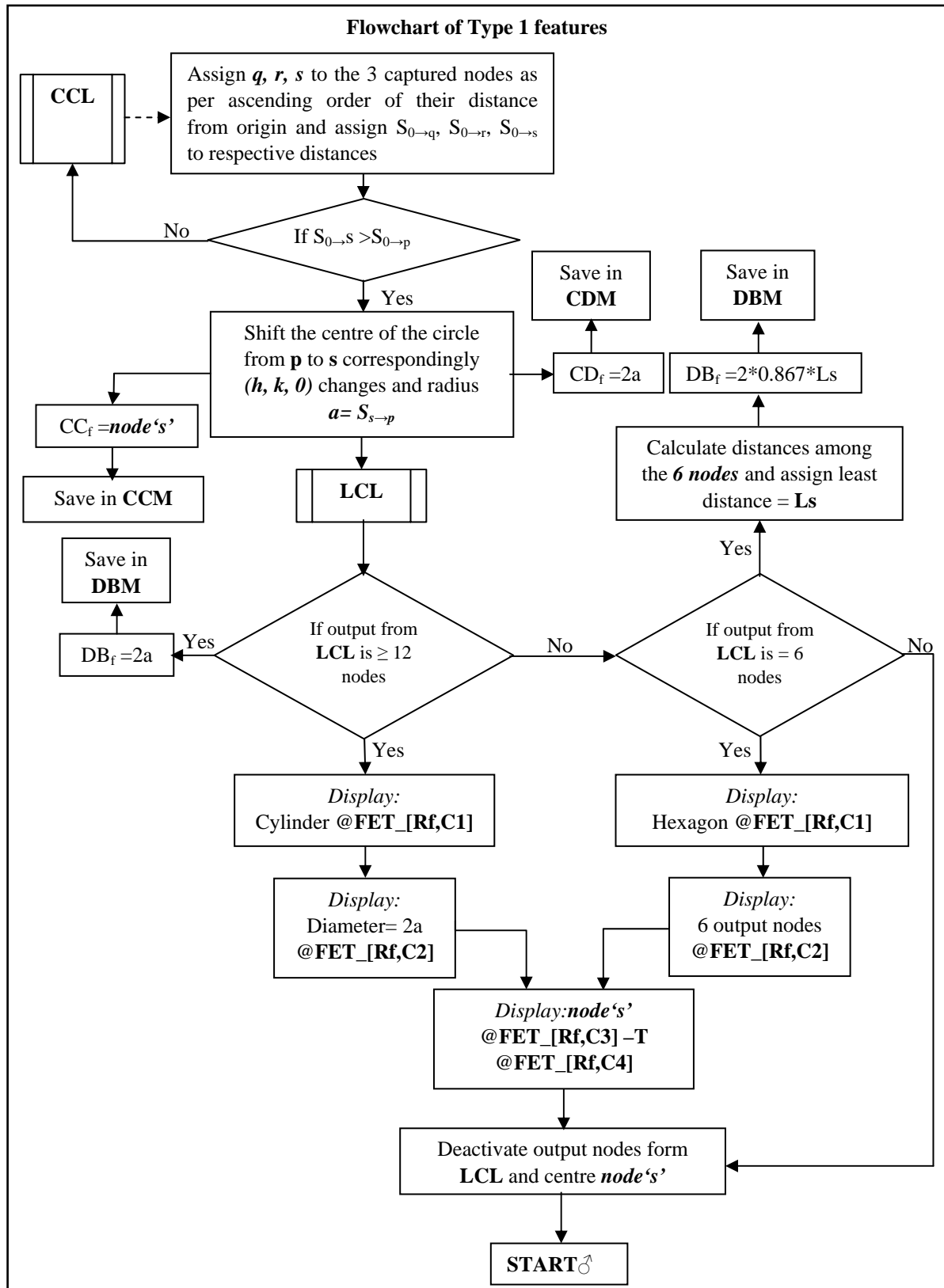
- **Else**, De-activate **4 boundary nodes** and **1 centre node**.

Go to **step#**.

G. Combinational Identification Of Features

The square and rectangle consists of 4 nodes on their Circum Circle. However, differentiates in side lengths. In square, the side length is equal and in rectangle opposite sides is equal.

On the basis above criteria, after getting 4 nodes on circum circle. Node to node distance, among those 4 nodes are computed such that considering **2 nodes** at a time randomly without repetition i.e., 4C_2 combination, remain with **6 values**, out of them if there are 2 different values (4 equal sides & 2 equal diagonal values), represents **Square** feature. If 3 different values (2 lengths, 2 breadths & 2 diagonal values) as a **Rectangular**.



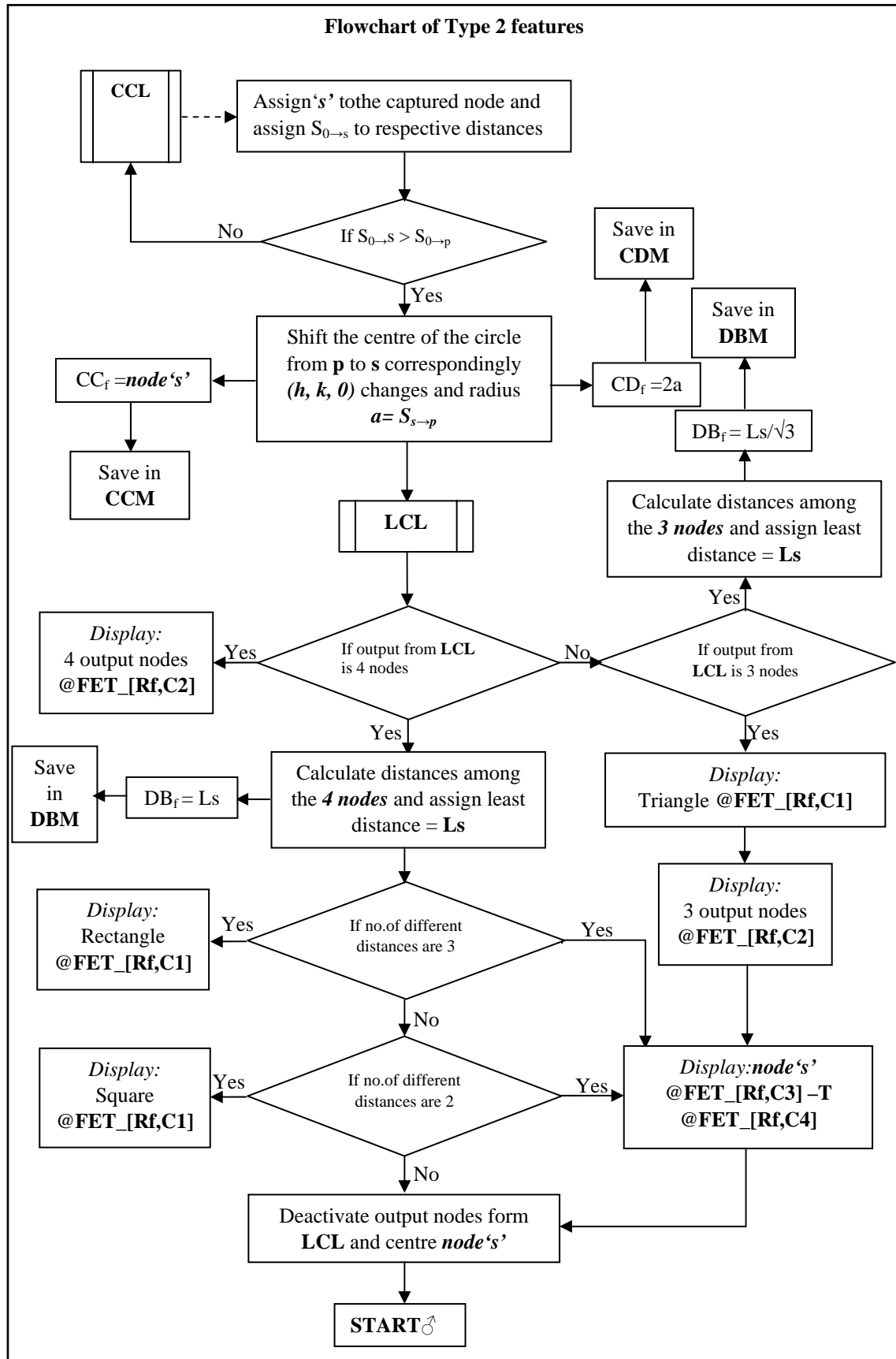


Fig. 14. Flowchart for type-2 features

CIRCLE [Step #7]**Display:** Feature**Type** - Cylinder**Location:** Circum-Centers ($h, k, 0$).**Size:** Hole diameter $d=2a$.**Depth of CUT**= block thickness [T].**Drill Bit Module (DBM)** = $2a$

Deactivate boundary nodes and 1 centre node. THEN, Go to step#.

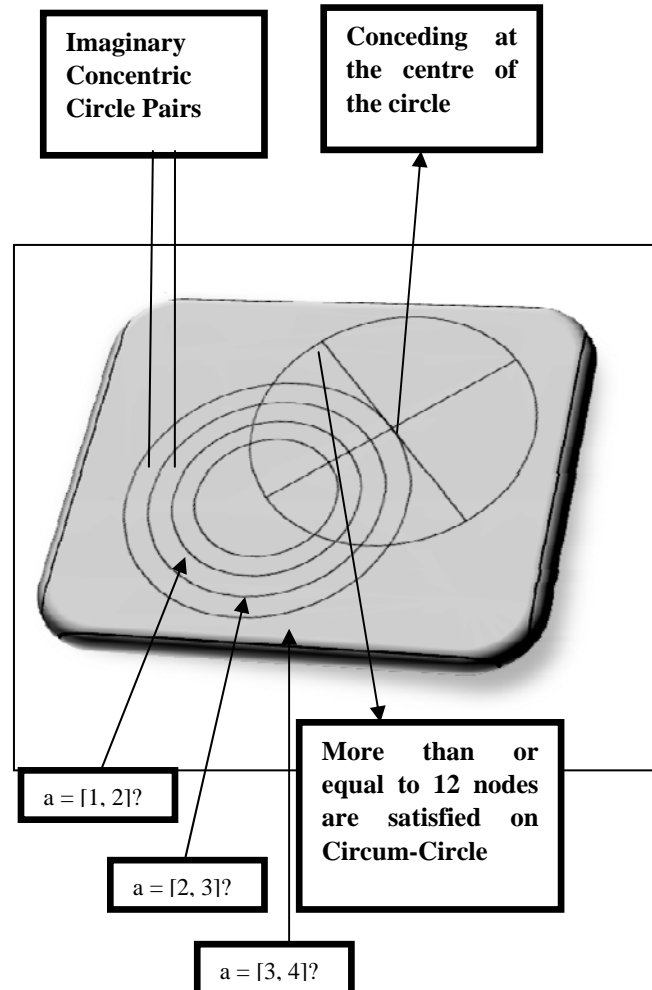


Fig. 15. Capturing of a cylindrical feature

HEXAGON [Step #7]**Display:** Feature**Type:** Hexagon.**Location:** Circum-Center s ($h, k, 0$).**Size:** Satisfied nodes.**Depth of CUT**= thickness of block ' T '.Calculate node to node distance in 6C_2 combination and arrange in ascending order.Smallest distance is considered as the side length (L_s) and ' $2*0.866*L_s$ ' store at (DBM).

Deactivate 6 boundary nodes and 1 center node. THEN, Go to step#.

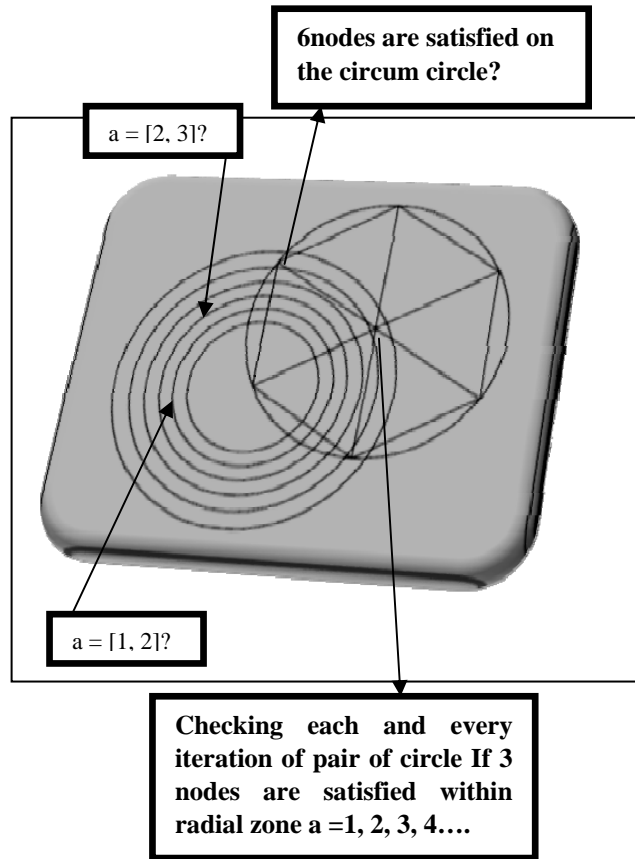


Fig.16. Recognition of Hexagonal feature

SQUARE [Step #7]

If number of different values =2.

Display: Feature:

Type **Square block**

Location: Circumcentres (**h, k, 0**).

Size: **4 satisfied nodes**.

Depth of pocket: thickness of block '**T**'.

De-activate **4 boundary nodes** and **1 centre node** Go to step#

Else,

De-activate **4 boundary nodes** and **1 centre node**. Go to step#.

Step#4

Copy that **node** and assign **s** to it. Calculate distance $S_{0 \rightarrow s}$ from origin. If [$S_{0 \rightarrow s} > S_{0 \rightarrow p}$].

Then, Take node '**s**' as centre of circle, **s(h,k,o)** as new centre and Store this **node** in Circum Centre Module (**CCM**).

Distance $S_{p \rightarrow s}$ is consider as radius of circum circle, Store $2 * S_{p \rightarrow s}$ in Circum Diameter Module (**CDM**).

Else, go for increment to **CCL**.

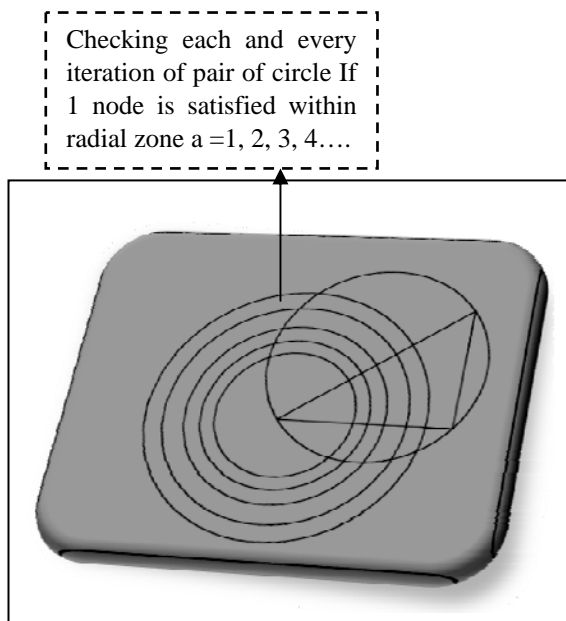


Fig 17. Recognition of Triangular feature

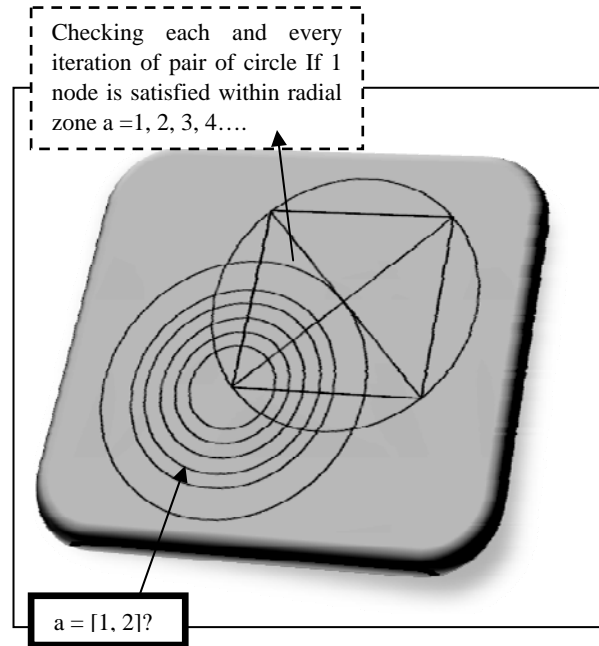


Fig 18. Recognition of Square feature

Step#4

Copy that **node** and assign **s** to it. Calculate distance $S_{0 \rightarrow s}$ from origin. **If** $[S_{0 \rightarrow s} > S_{0 \rightarrow p}]$.

Then, Take node 's' as centre of circle, $s(h, k, o)$ as new centre and Store this **node** in Circum Centre Module (**CCM**).

Distance $S_{p \rightarrow s}$ is consider as radius of circum circle, Store $2 * S_{p \rightarrow s}$ in Circum Diameter Module (**CDM**).

Else, go for increment to **CCL**.

Step#6

With **input s** ($h, k, 0$) as centre and radius $a = S_{p \rightarrow s}$. Go to **LCL**.

In **LCL**. With ' $a, a+1$ ' as radial zone, nearest node from origin 's' as centre Locus check is carried out by substituting each and every remaining **nodes**.

TRIANGLE [Step #7]

Display: FeatureType: **Triangle**.

Location: Circumcenter $s(h, k, 0)$.

Size: **3 satisfied nodes**.

Depth of pocket: '**T**'.

Calculate **node to node** distance in 3C_2 combination and mode is taken as side length (**Ls**).

Store: $Ls/\sqrt{3}$ in **DBM**.

De-activate **3 boundary nodes** and **1 centre node**. Go to **step#**.

RECTANGLE

Step#7

If the number of different distances = **3**,

Display: Feature

Type: **Rectangular block**

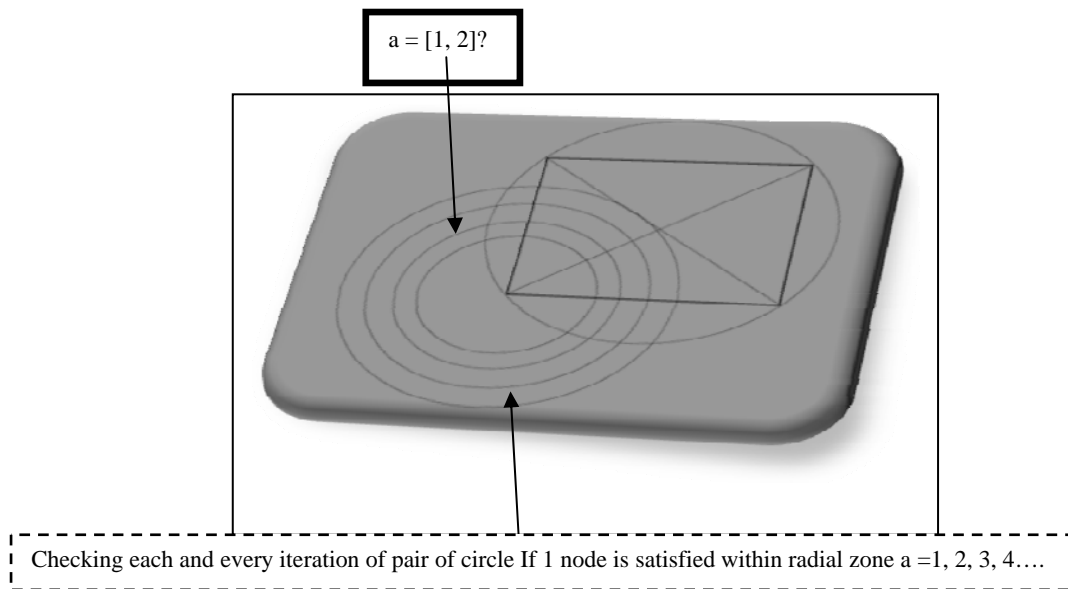
Location: Circumcentre $s(h, k, 0)$.

Size: **4 satisfied nodes**.

Depth of pocket: thickness of block '**T**'.

De-activate **4 boundary nodes** and **1 centre node**. Go to **step#**.

Else,



XIII. A Case Study

A. Concentric Contrast Circle Approach Is Applied To A Model

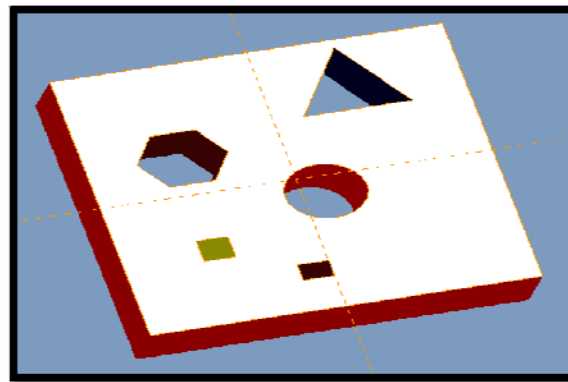


Fig.20. Isometric view of prismatic part model

Component designed model is imported from AutoCAD software to Autodesk Design Review, and various required nodes are marked under standard DWF format dimensional view of the part is as shown in fig. 24. Node list is prepared, which consists of all required nodes as per constraints and requirements as shown in the fig. 26. Maximum distance in X, Y, Z is taken as block dimensions L, B, and T, respectively: L=100 mm, B=100 mm, T=-40 mm. Nodes are selected from edge representation with 13 nodes selected for the circle. As the remaining features, all corner nodes and the corresponding centroid are selected and the list is prepared.

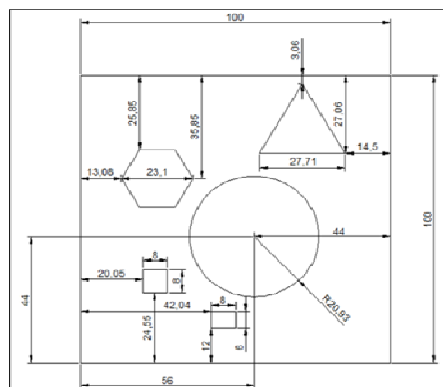


Fig.21: 2 Dimensional view of the Part

B. Edge representation and node collection

All required nodes can be collected Through AUTO CAD software

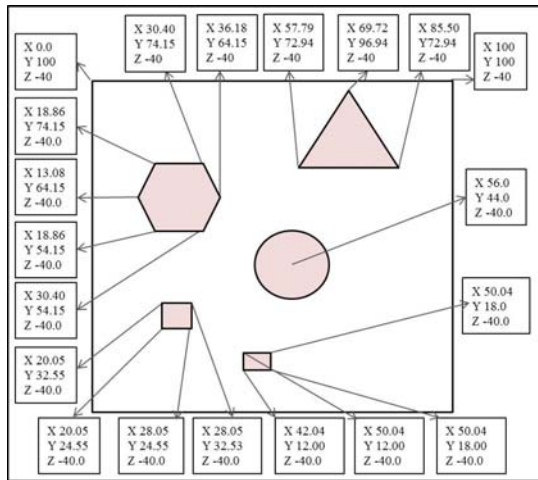


Fig.22. required nodes Edge-rep

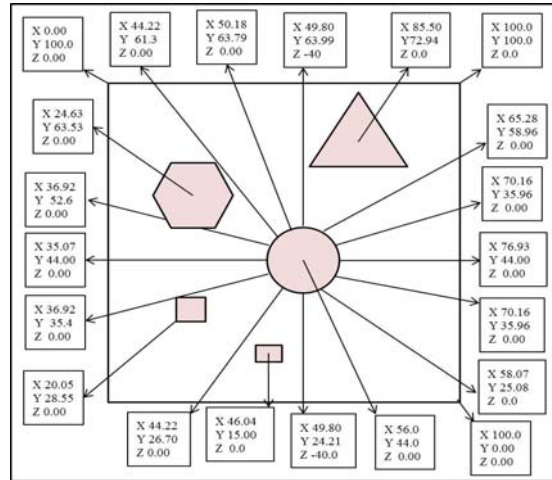


Fig.23. Centroid and circle nodes Edge-rep

B. List of required nodes and node bank preparation

List of required nodes Node Bank preparation

All required nodes at surface **Z=0** (default)

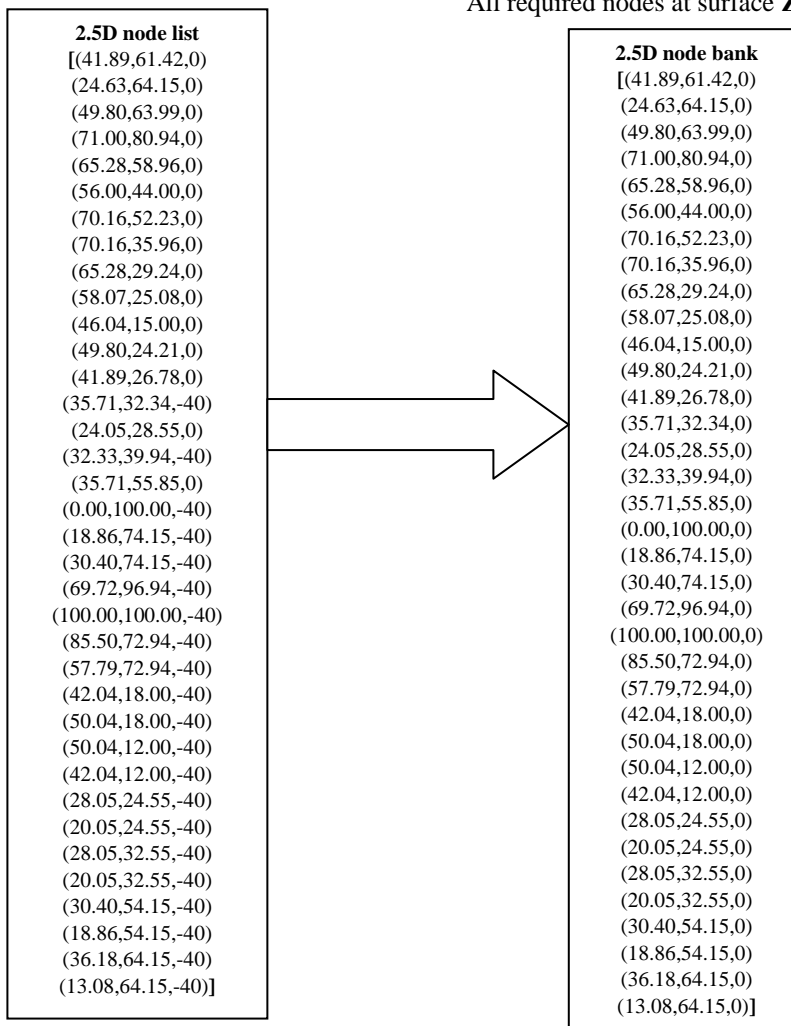


Fig. 24. Node Bank preparation

C. Type - 1 Feature

2.5D Node Bank	
Circle	
•	(56.0, 44.0, 0.0)
•	(76.93, 44.0,-40.0)
•	(70.16, 35.96,-40.0)
•	(65.28, 58.96,-40.0)
•	(49.80, 63.99,-40.0)
•	(50.18, 63.79,-40.0)
•	(44.22, 61.30,-40.0)
•	(36.92, 52.60,-40.0)
•	(35.07, 44.0,-40.0)
•	(36.92, 35.4,-40.0)
•	(44.22, 26.70,-40.0)
•	(49.80, 24.21,-40.0)
•	(58.07, 25.08,-40.0)
•	(70.16, 35.96,-40.0)
Hexagon	
•	(24.63, 63.53, 0.0)
•	(18.86, 74.15,-40.0)
•	(30.40, 74.15,-40.0)
•	(36.18, 64.15,-40.0)
•	(30.40, 54.15,-40.0)
•	(18.86, 54.15,-40.0)

D. Type - 2 Features

2.5D Node Bank	
Rectangle	
•	(46.04,15.0,0)
•	(42.04,12.0,-40.0)
•	(50.04,12.0,-40.0)
•	50.04,18.0,-40.0)
•	(50.04,18.0,-40.0)
Square	
•	(24.05,28.55,0.0)
•	(28.05,24.55,-40.0)
•	(28.05,32.53,-40.0)
•	(20.05,32.55,-40.0)
•	(20.05,24.55,-40.0)
Triangle	
•	(85.50,72.94,0.0)
•	(57.79,72.94,-40.0)
•	(69.72,96.94,-40.0)
•	(85.5,72.94,-40.0)

Fig. 25. Required nodes and order

XIV. RESULT

Table3. Feature information table

Sr.no	Feature title	Feature size & location in XY-plane	Circumcentre location	Depth of pocket
1	Square pocket	(20.05,24.55,0) (28.05,32.55,0) (28.05,24.55,0) (20.05,32.55,0)	(24.05,28.55,0)	-40
2	Rectangular pocket	(42.04,18.00,0) (50.04,18.00,0) (50.04,12.00,0) (42.04,12.00,0)	(46.04,15.00,0)	-40
3	Cylinder	Diameter = 40mm	(56.00,44.00,0)	-40
4	Hexagon	(18.86,54.15,0) (18.86,74.15,0) (30.40,74.15,0) (30.40,54.15,0) (36.18,64.15,0) (13.08,64.15,0)	(24.63,64.15,0)	-40
5	Triangle	(69.72,96.94,0) (85.50,72.94,0) (57.79,72.94,0)	(71.00,80.94,0)	-40

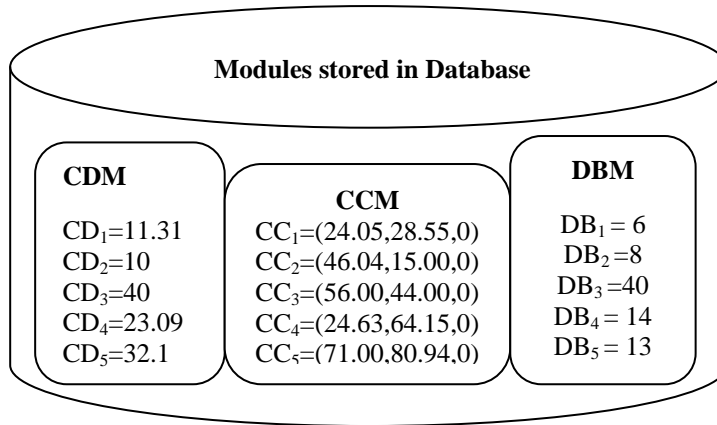


Fig26:modules output from 3'CAM technique

XV. WINDOWS OF 3'CAM TECHNIQUE

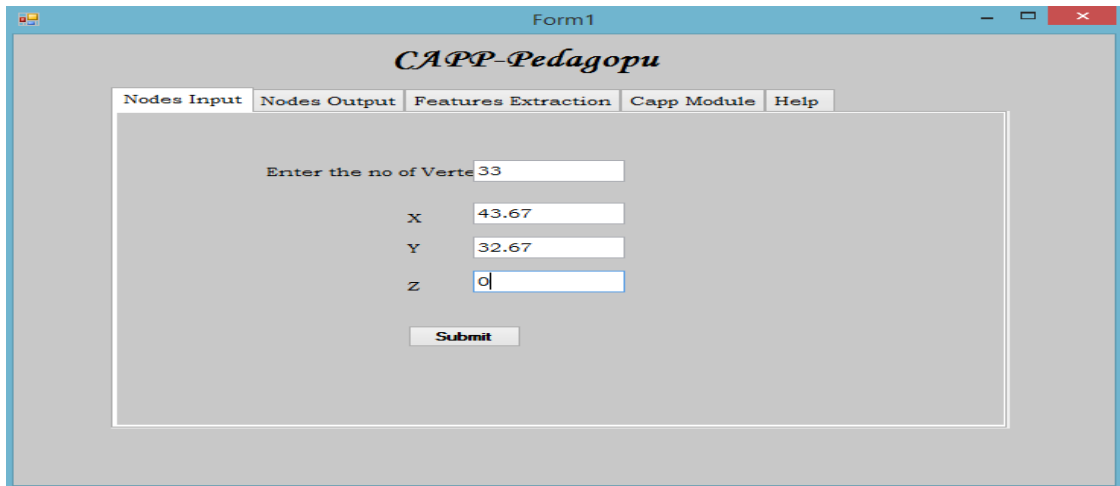


Fig.27A Input to 3'CAM technique

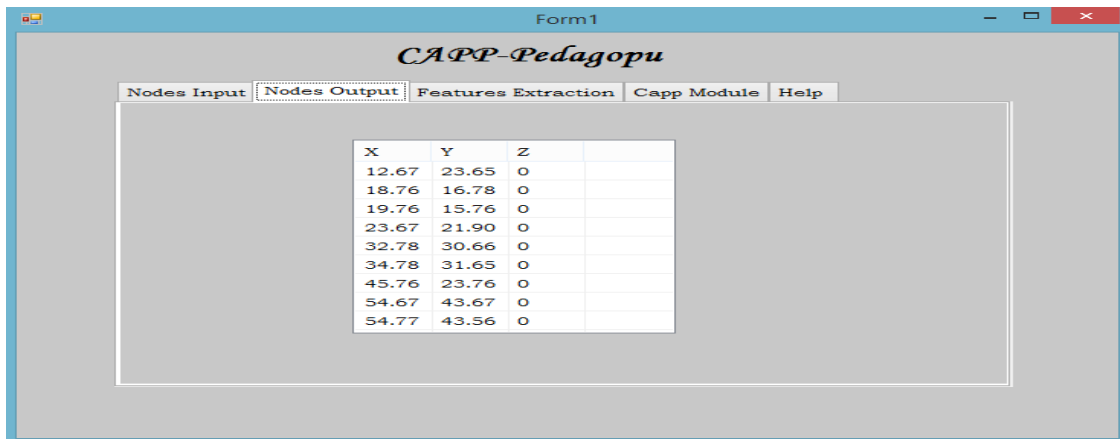


Fig. 27B output of nodes

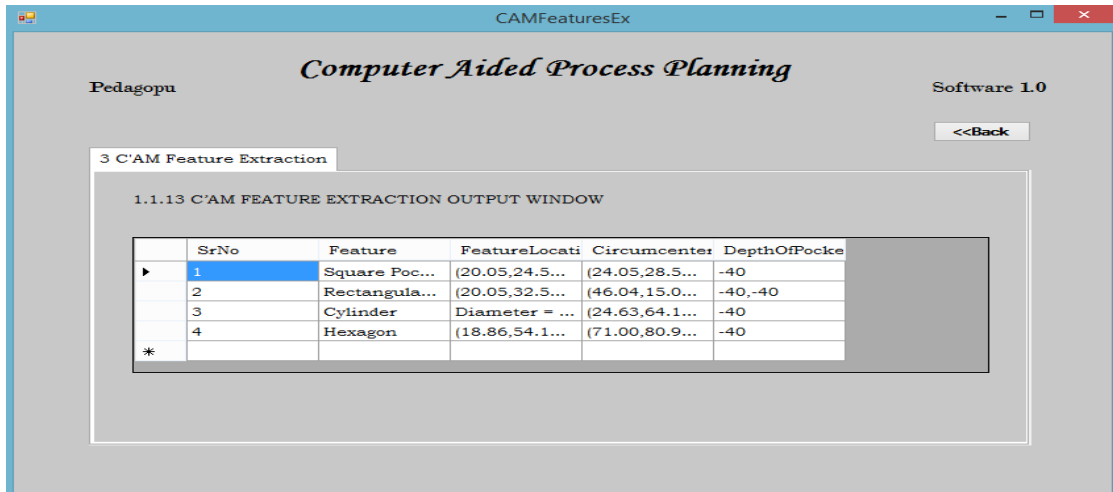


Fig. 27C. 3'CAM technique feature extraction output window

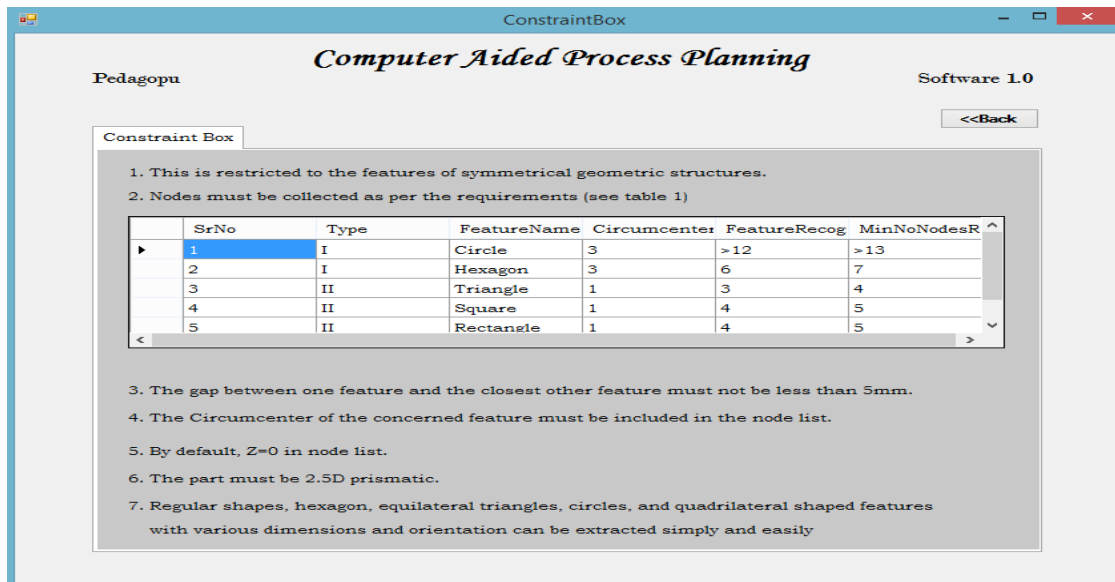


Fig. 27D Constraint of 3'CAM technique

XVI. CONCLUSION

In this paper, a new Concentric Contrasting Circles Approach Methodology (3'CAM) technique has been introduced for recognition and extraction of various symmetric features from 2.5D prismatic part. The necessitated coordinates are imported from a standard IGES file as an input from creo2.0 designed part and recognition of features along with their geometric dimension as output has been successfully achieved. In general, the extraction of features plays a major role to build a CAPP module for machining a part. The geometric information of the 2.5D part that features to be extracted uses 3'CAM technique instead of Boundary representation (B-rep) or Constructive Solid Geometry (CSG). In this paper the recognition and extraction of a 2.5D part is demonstrated by a simple case study. The extraction of features is developed by using Visual Studio at front end, supported by SQL-Server and C# software at back end respectively. Finally, the 3'CAM technique has been applied, verified and finally approved for selected 2.5D part.

XVII. REFERENCE

- [1] Zheng (2008). Systematic modeling and reusing of process knowledge for rapid process configuration, robotics and Computer-Integrated Manufacturing, 24,763-772.
- [2] Mokhtar, A. et.al (2011). Machining precedence of 2D/2.5D interesting features in a feature-based data model. Journal of Intelligent Manufacturing, 22 (2), 145-161.
- [3] Gao, J. et.al (2004). Extraction of machining features for CAD/CAM integration. International Journal of Advanced Manufacturing Technology, 24,573-581.
- [4] M.T.Hayasi et.al (2009) 'Extraction of manufacturing information from design-by-feature solid model through feature recognition' Int J AdvManufTechnol (2009) 44:1191-1203 DOI 10.1007/s00170-008-1922-6
- [5] International Review of Applied Engineering Research. ISSN 2248-9967 Volume 4, Number 2 pp. 171-176

- [6] V.M. Pedagogu. Manish Kumar (2014) An Outline of 2.5 Dimensional Prismatic Parts Machining International Review of Applied Engineering Research. ISSN 2248-9967 Volume 4, Number 2, pp. 185-190
- [7] V.M. Pedagogu. Manish Kumar (2014) An Integrated Geometric Modeling Methodology for 2.5D Cylindrical Prismatic Part for Computer Aided Process Planning Intelligent Control and Automation, 2014, 5, 183-189 in SciRes.
- [8] V.M. Pedagogu. Manish Kumar (2015) A Prismatic Part Machining and Simulation for Computer Aided Process Planning System International Journal of scientific Research, Volume: 4 | Issue: 1 • ISSN No 2277 – 8179. Volume: | January 2015 • ISSN No 2277 – 817
- [9] Ali R. Yildiz (2012) ‘Cuckoo search algorithm for the selection of optimal machining parameters in milling operations ‘Int J AdvManufTechnol DOI 10.1007/s00170-012-4013-7
- [10] Babic B, et.al (2008) A review of automated feature recognition with rule-based pattern recognition. Computers in Industries 54 (4):321-337
- [11] Abu R, et.al (2010) Attribute based feature recognition for machining features. JournalTechnologyi 46:87 103
- [12] Gracia F, et.al (2011) Process planning based on feature recognition method. In: 2011 IEEE international symposium on assembly and manufacturing (ISAM), IEEE, pp 1-5
- [13] Yu H, et.al (2012) Research on CAPP method for rotational parts based on case retrieval. Machinery 1:014
- [14] Behera AK, et.al (2012) Advanced feature detection algorithms for incrementally formed sheet metal parts. Trans Nonferrous Metals Soc China 22:315-322
- [15] Celik I, Unuvar A (2013) A new object coding system for computer-aided process planning applications. Arab J SciEng 38(5):1 103-113.
- [16] J. Gao et.al (2004) Extraction of machining features for CAD/CAM integrationInt J AdvManufTechnol 24: 573–581DOI 10.1007/s00170-003-1882-9
- [17] Lee JY, Kim K (1999) Generating alternative interpretation of machining features. Int J AdvManufTechnol 15:38–48
- [18] Baskar N, et.al (2005) Optimizarion of machining parameters for milling operations using nonconventional methods. Int J AdvManuf Tech 25:1078–1088

AUTHORS PROFILE

Viswa Mohan Pedagogu did his Diploma, Graduation and Post Graduation in Mechanical Engineering. He received his ME in Mechanical Engineering specialisation in Production Engineering from University College of Engineering, Osmania University, Hyderabad, and Telangana State, India. He is presently under the PhD degree in JNU-Jodhpur. He has over 20 years of experience in academic, industry and research. He has worked various govt and private sectors.

Dr. Manish Kumar has completed his PhD in JNVU-Jodhpur and working as an associated professor. He has many publications in national and international journals.

Dr. Saroj Patel completed her PhD in mathematics and working as an associate professor in JNU Jodhpur and she has many publications in national and international journals.