

# A new approach in Tube Count For Tubular Heat Exchangers

Ardalan Sadighian, Leili Mirmontazeri, Mohammad Toossi , Fatollah Farhadi  
Chemical and petroleum Engineering  
Sharif University of technology  
Azadi Ave, Tehran  
Iran

*Abstract* - This paper suggests a solution for calculating the Tube Count in a shell and tube heat exchanger. A visual C program developed for this purpose computes the number of tubes with respecting several constraints; including the shell ID, number of passes, center to center distance of tubes and tube outer diameter. Furthermore the results of this program is compared with existing tube count methods and developed by experiment. The algorithm which is used, the capability and advantages of this program are discussed in this paper.

*Key words:* Heat exchanger, Tube count, Tubesheet, Shell

## 1 Introduction

A "heat exchanger" is a device for transferring heat from one fluid to another, where the fluids are separated by a solid wall so that they never mix. They are widely used in refrigeration, air conditioning, space heating, power production, and chemical processing [1]. The most common heat exchanger of the various types of unfired heat transfer equipments are the shell and tube heat exchanger which consists of a series of tubes, through which one of the fluids runs. The second fluid runs over the tubes to be heated or cooled. It is robust and its shape makes it well suited to pressure operation.

The number of tubes which can be accommodated within a given shell inside diameter is termed the **Tube count**. For a given shell inside diameter, Tube outside diameter, pitch and pitch angle, tube count depends on the factors listed below:

**Type of heat exchanger.** This determines the OTL [5].

**Design pressure.** This may reduce the OTL for floating-head type exchangers.

**Nozzle diameter.** Adequate escape area around an internally fitted impingement baffle at the inlet nozzle must be provided which can not be filled with tubes. The greater the nozzle diameter the smaller the tube count. Similar Consideration apply at the outlet nozzle .The tube count is independent of nozzle diameter if an externally fitted impingement plate or distributor is used.

**Number of tube side passes.** As the number of passes increases, tube count decreases. The area occupied by pass partitions can not be accommodated by tubes.

**Tube-end attachments.** Welded tube ends may reduce the tube count in multi-pass exchangers of all types because tubes closest to the pass partitions may have to be pitched at greater distance than normal to accommodate the weld.

**Tie rods, Spacer and sealing devices.** The tube count may be reduced to accommodate these items.

**Reduced tubesheets** .In some cases the tubesheets may have reduced hubs at the periphery for welding to the shell or head barrel. The OTL may be reduced as tubes should not encroach into the radii.

**Rotatable Bundles.** Rotatable bundles with intently fitted impingement plate require escaping areas on opposite sides of the bundle, resulting in a reduction in tube count. [2]

Because of so many variables it is not possible to calculate the exact number of tube counts in usual tube count tables. After a literature review, an algorithm is presented to find out the tube count. A typical input, output of this program is shown and finally the results are compared with other methods.

## 2 Related works

All methods to compute tube count are based on experiment. In many tube count tables available in reference books,[2,3,6,7] authors have derived them from experiments. As complexities like tube row distance, difference in OTL (which is related to many parameters), are not mentioned, so results gained from these methods should be recomputed with related formulas. There are also some formulas like

$$Tubecount = \frac{0.78D_{ctl}^2}{C_t L_{tp}^2}$$

( $D_{ctl}$  = shell diameter,  $L_{tp}$  = tube layout pitch,  $C_t$  = tube field layout constant)

This can calculate tube count. But these formulas are obtained from fitting a curve to experimental data [3].

## 3 The newly proposed algorithms

We have developed several algorithms to solve this problem. Here, we will discuss three of our most efficient and straightforward algorithms along with their advantages and disadvantages. In our initial problem, we had a plane full of the heat exchanger tubes and a shell (Fig 1a) and we wished to find a horizontal disposition  $l_h$  and a vertical disposition  $l_v$  for this plane so that it maximizes the number of tubes fitting in the shell. For simplicity, we replace each tube in the plane by the point at its center and subtract the tube diameter from the shell diameter (Fig 1b). Clearly, the problem of finding the maximum number of points we can fit into this new shell is equivalent to our previous problem. From now on, we will refer to the distance between two adjacent points as  $d$ , and to the largest number of points fitting in a single row in the shell as  $n$ .

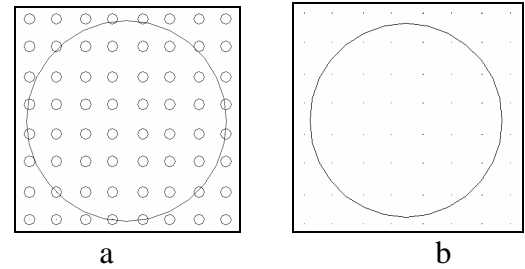


Fig 1 converting the tubes in to the spots

Without loss of generality, we can assume that  $l_h$  and  $l_v$  are both real numbers in the interval  $[0,d)$  because the tube pattern on the plane will repeat if we shift the plane more than the distance  $d$  in the vertical or horizontal direction.

The first and the simplest algorithm is based on shifting the points in several small predefined steps and counting the number of points inside the circle (or shell) each time. Let  $m$  be a large integer denoting our resolution of our search. We divide the interval  $[0,d)$  into  $m$  segments of size  $d/m$  (fig 2) and only search the cases where  $l_h$  and  $l_v$  are on the boundaries of these segments. Considering that  $l_h$  and  $l_v$  can only have  $m$  different values each, there will be a total of  $m^2$  combinations resulting in a running time of  $O(m^2 n^2)$ . Our first algorithm is both slower and less accurate than the next two algorithms. However, its simplicity and its close to optimum results make it a good choice nonetheless.

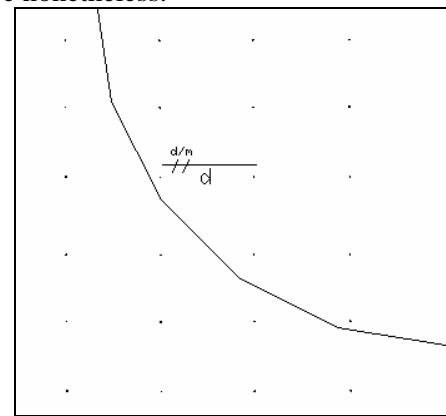


Fig 2 dividing the distance in to the smaller pieces

In our second algorithm, we shift the tubes vertically in  $m$  steps similar to the first algorithm but use a different method for determining the optimum horizontal disposition with infinite precision. Assume that  $l_v$  is

currently one of the  $d/m$  multiples as described in the previous algorithm. To compute the optimum value for  $l_h$  we have to consider the movement of the points in each row as we increase  $l_h$  from 0 to  $d$ .

At some point during the  $[0,d]$  interval, the last point in the row which was inside the circle when  $l_h$  was 0 will move out of the circle. We call the interval beginning at this point and ending at  $d$  a -1 interval. On the other hand, a new point will enter the shell as we increase  $l_h$ . The interval beginning at this point and ending at  $d$  will be called a +1 interval. Clearly, if the value chosen for  $l_h$  falls inside a -1 interval, we will have one less point in the shell, and if it is within a +1 interval, there will be one additional tube in the shell.

There is exactly one +1 interval and one -1 interval associated with each row of tubes (fig 3) because exactly one tube will enter to and exit from the shell if we shift the tubes with the distance  $d$  in a horizontal direction. Considering that there are a total of  $n$  rows of tubes within the shell, we will have  $2n$  intervals inside our  $[0,d]$  interval. To calculate the optimum value for  $l_h$  we have to find a value which is within as many +1 intervals as possible and as few -1 intervals as possible. This can be done in  $O(n \log n)$  by sorting the starting and ending points of these intervals and moving through them from 0 to  $d$ , keeping track of the number of the overlapping intervals as we enter to and exit from each interval.

By having a running time of  $O(mn \log n)$ , this algorithm is the fastest of all three algorithms and is relatively easy to implement. It also has the second best precision.

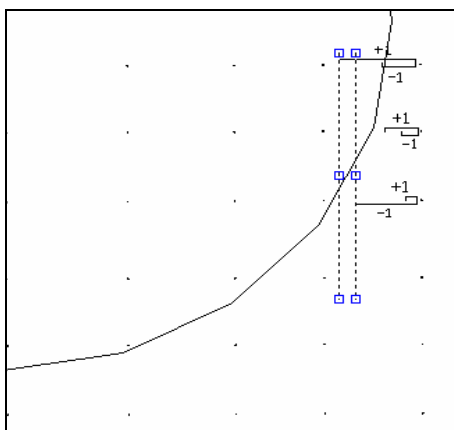


Fig 3 +1 and -1 spans for each row

In the third algorithm, we extend the method we used for determining  $l_h$  in the previous algorithm to the second dimension, allowing us to compute both  $l_h$  and  $l_v$  with infinite precision.

As we have shown, our search space is a  $d*d$  square corresponding to the possible combinations of values assigned to  $l_h$  and  $l_v$  because they are both in the interval  $[0,d]$ . Now, we consider a single tube as we move the answer point  $(l_h, l_v)$  in this  $d*d$  square. If the tube is close enough to the shell, we will have two areas in the square. If  $(l_h, l_v)$  resides in one of these areas, the tube will be inside the shell and it will be outside the shell otherwise.

It is easy to see that these two areas within our search space – the  $d*d$  square – are always separated by an arc whose radius is the same as the radius of the shell. Putting  $(l_h, l_v)$  at a particular side of this arc means that one more tube is going to fit in the shell.

If we consider all of the arcs associated with each of the tubes close to the shell, we will have a total of  $O(n)$  arcs in the square (fig 4a). Similar to the previous algorithm, we now have to find an area in the square that has the highest number of tubes (fig 4b). This allows us to compute the best possible answer with infinite precision but this algorithm is considerably harder to implement than the previous ones and is also usually slower than the second algorithm.

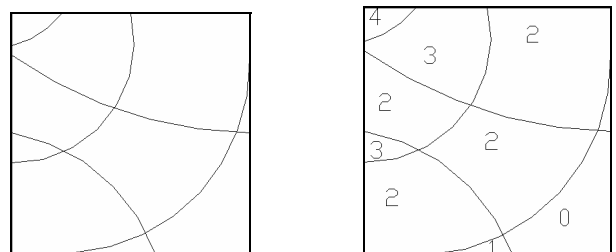


Fig 4 drawing the curve near each tube in a square

The problem here is a trade off between accuracy and speed. Normally in engineering problems we do not need infinite accuracy, but time is very precious for us. As a result we choose the second algorithm which provides a good balance between these two.

## 4 Appearance

In this section appearance, inputs, outputs and different abilities of this program will be discussed. The general appearance of the total sheet layout is shown in Fig 5. As inputs, this program gets: 1-Tube pitch, 2-Shell ID, 3-OTL, 4-OTD, 5-Center to center distance of tube, 6-Accuracy, 7-Longitudinal baffle, 8-Number of passes, 9-Tube row distance.

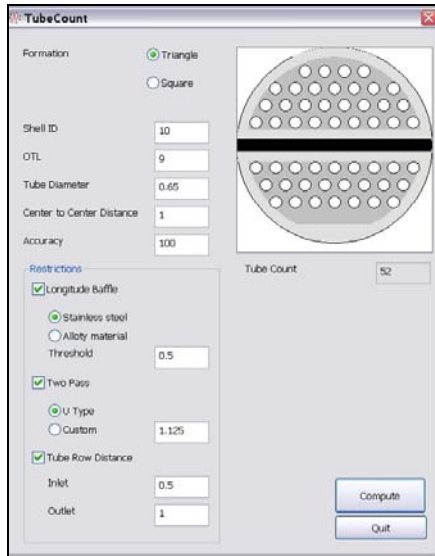


Fig 5 General layout of the tubesheet

**1. Tube pitch:** There are four common tube layout patterns: triangular (60°) (fig 6-a), rotated triangular (30°) (fig 6-b), square (90°) (fig 6-c) and rotated square (45°) (fig 6-d) [4]. As the two rectangular and the two squares each have a similar group of tube count, so in the part which the user chooses the formation there is only two patterns available, square and triangular.

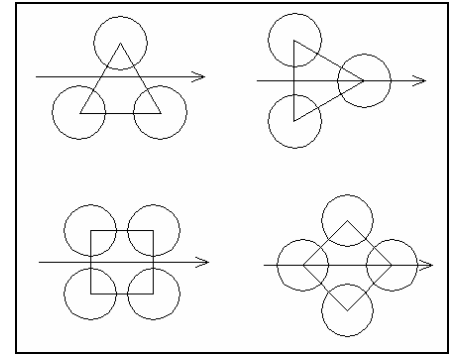


Fig 6 The pitch types: a-triangular, b-rotated triangular, c-square, d-rotated square

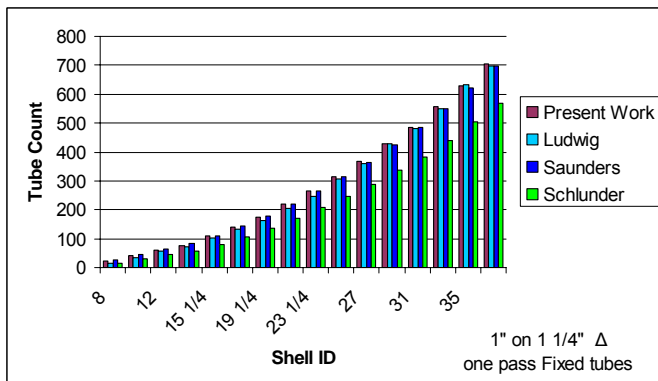
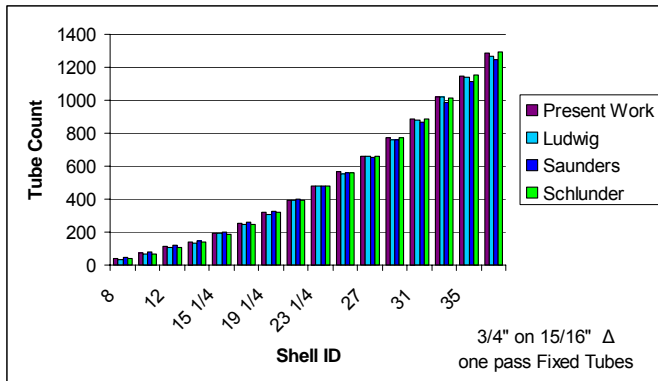
2. **Shell ID:** Shell ID is the inner diameter of shell. In this program shell ID is determined by a number in inch.
3. **OTL:** Outer Tube Limit, is the diameter of the circle tangent to the outer most tube .
4. **OTD:** the outer diameter of the tube.
5. **Center to center distance:** the distance between the centers of two tubes next to each other.
6. **Accuracy:** the number inputted as the accuracy shows the number of the divisions defined in algorithm chapter. For instance for the center to center distance of 1 inch and accuracy of 100 the pace of our movement is 0.01 inch.
7. **Longitudinal baffle:** This program offers two types of longitudinal baffle 1.stainless steel, 2.alloy material .The thickness of the baffle for stainless steel is 5/8 inch and for alloy material is 1/2 inch [5].The threshold number defines the distance in which tubes can not be closer to the baffle.
8. **Type of passes:** This program support two type of passes U Tubes and Fixed Tubes .The restriction that U pass adds to the program is that the central rows of tubes can not be closer to each other less than 2.5 times of the tube diameter. The Fixed tubes type adds a pass partition in which we can not put

tubes in the place of pass partition. The thickness of the partition is defined by the user.

- 9. Bundle to shell clearance:** It is a distance from shell ID and the tube bundle OD (OTL), where no tube can be placed in it. Usually this distance is not equal in top and bottom section of the bundle clearance so the program gets two numbers as inputs that determine these two sections.

## 5 Conclusions

All complexities such as split backing floating-head and fixed tubes affect OTL. This method can get the OTL and calculate the tube count as a result it could cover these complexities. We have no information about the number of the OTL in tube count tables. In below graphs OTL is considered from heat exchanger type and available standards in order to compare the results of this program with tube count tables.



The advantage of this program is that despite other tube count methods which consider a fix OTL for a type, it can change the OTL when desired.

## 6 References

- [1] www.Britanica.com
- [2] E.A.D. Saunders, Heat exchanger selection design and calculation, Longman scientific and technical, spring 2002
- [3] Schlunder, E.U., Heat Exchanger Design Handbook E D H, Hemisphere, USA, 1983
- [4] Rajiv Mukharej, Effectively Design Shell and tube Heat exchangers, Chemical Engineering Progress, February 1998
- [5] Tubular Exchanger Manufacturers Association, Standards of the tubular exchangers and manufacturers association, seventh edition, TEMA, New York 1998.
- [6] Ernest E. Ludwig, Applied Process Design for Chemical and Petrochemical Plants, Gulf Publishing company, USA, June 1977