

Errors in the Erlang97 Excel add-in module

Karin van Eeden^a, Edwin van der Hilst^b en Ger Koole^{a,c}

^aCCmath bv, Amsterdam

^bplanners.nu, Groningen

^cVU University Amsterdam

vaneeden@ccmath.com, edwin.vander.hilst@planners.nu, ger.koole@vu.nl

The Erlang97 Excel add-in module¹ is often used for capacity calculations in call centers. However, there are conceptual errors in some of its functions. In this article we explain the errors we encountered.

Service level calculations

When we do calculations using the Erlang formula to obtain the required staffing level or the number of agents then we usually work with an integer number of agents. For example, consider a call center with 100 calls per hour, an average handling time of 180 seconds, and a required service level of 80% answered within 20 seconds. The Agents function of the Erlang97 module tells us that we need 8 agents: =Agents(0.8,20,100,180) gives 8 as result. This is the minimal number of agents required, because $SLA(8,20,100,180)=88\%$ and $SLA(7,20,100,180)=74\%$. Note that in the Erlang97 module the average handling time and the "ServiceTime" (the 20 of 80/20) are both given in seconds, the forecast is in number of calls per hour.

Sometimes it makes sense to use a non-integer number. For example, let us consider calculations that include shrinkage. Suppose that in a given quarter 22 agents are scheduled and that the shrinkage is 30%. Then one expects an availability of 70% of 22 is 15.4 agents. It would be wrong to round to 15 or 16 agents. The Erlang97 module is meant to deal with these non-integer numbers. However, the way it is implemented is wrong. We illustrate this with an example.

Consider a call center with on average 50 calls per hour and an average handling time (AHT) of 300 seconds. The call center wants to attain a 80/20 service level,

meaning that 80% of the calls will be answered within 20 seconds. With the FractionalAgents function we can calculate the number of agents that is required exactly to achieve a certain service level; this number is often non-integer. In the example this function gives 6.57 agents. We use this number of agents to calculate the service level using the SLA function. Of course, this should be exactly 80%, but the answer is 75%!

The error is in the SLA function, and only occurs when the number of agents is fractional. Part of the computations are done with a rounded number of agents, while at other places the fractional number is used. The following interpretation of the fractional number should be used. When, as in the example, 6.57 agents are used, then we should interpret this as if 57% of the time there are 7 agents and 43% of the time there are 6 agents. Not only SLA doesn't use this consequently, but we see the same error in the functions Abandon, ASA, Queued, QueueSize, ServiceTime en Trunks. For this reason the Erlang97 module should not be used for a fractional number of agents.

The Abandon function

Another erroneous calculation is done in the Abandon function. It is meant to calculate the percentage of abandoned calls. When we take into account abandonments in our calculations then it is important to know how much patience they have before they abandon. However, analysis of call center data always shows that there is variability in the patience²: there are callers with much patience and there are callers with less patience. Especially the latter are prone to abandon. Another important aspect of abandonments is that they shorten the waiting time of other calls. Both aspects are not taken into account in the Abandon function of the Erlang97 module. To make this clear we perform computations with an example with 50 arriving calls per hour, an AHT of 300 seconds, 6 agents, and an "AbandonTime" of 120 seconds. According to the Abandon function 15,7% abandons, thus 84,3% of the calls is answered. This is exactly the service level (SL) in the Erlang C model with a "ServiceTime" of 120 seconden: $SLA(6,120,50,300)$. The service level at 20 seconds is 71,1%.

A correct calculations takes variability in patience and the consequences of abandonments on future calls into account. The Erlang X formula of the CCmath Erlang module³ makes such a computation. This formula predicts for the same number a much better performance: 7.9% abandonments and 83.8% served within 20 seconds. If we employ one agent less then we are closer to the SL

prediction made by the Erlang97: 14.8% abandonments and 71.8% answered within 20 seconds. A considerable difference which leads to the fact that the Erlang X module will schedule less agents.

Referenties

1. Erlang97 Excel add-in module, downloaded from <http://www.erlang.co.uk/excel.htm>
2. Ger Koole. Call Center Optimization. MG books, Amsterdam, 2013.
3. <http://www.ccmath.com/en/calculators.html>