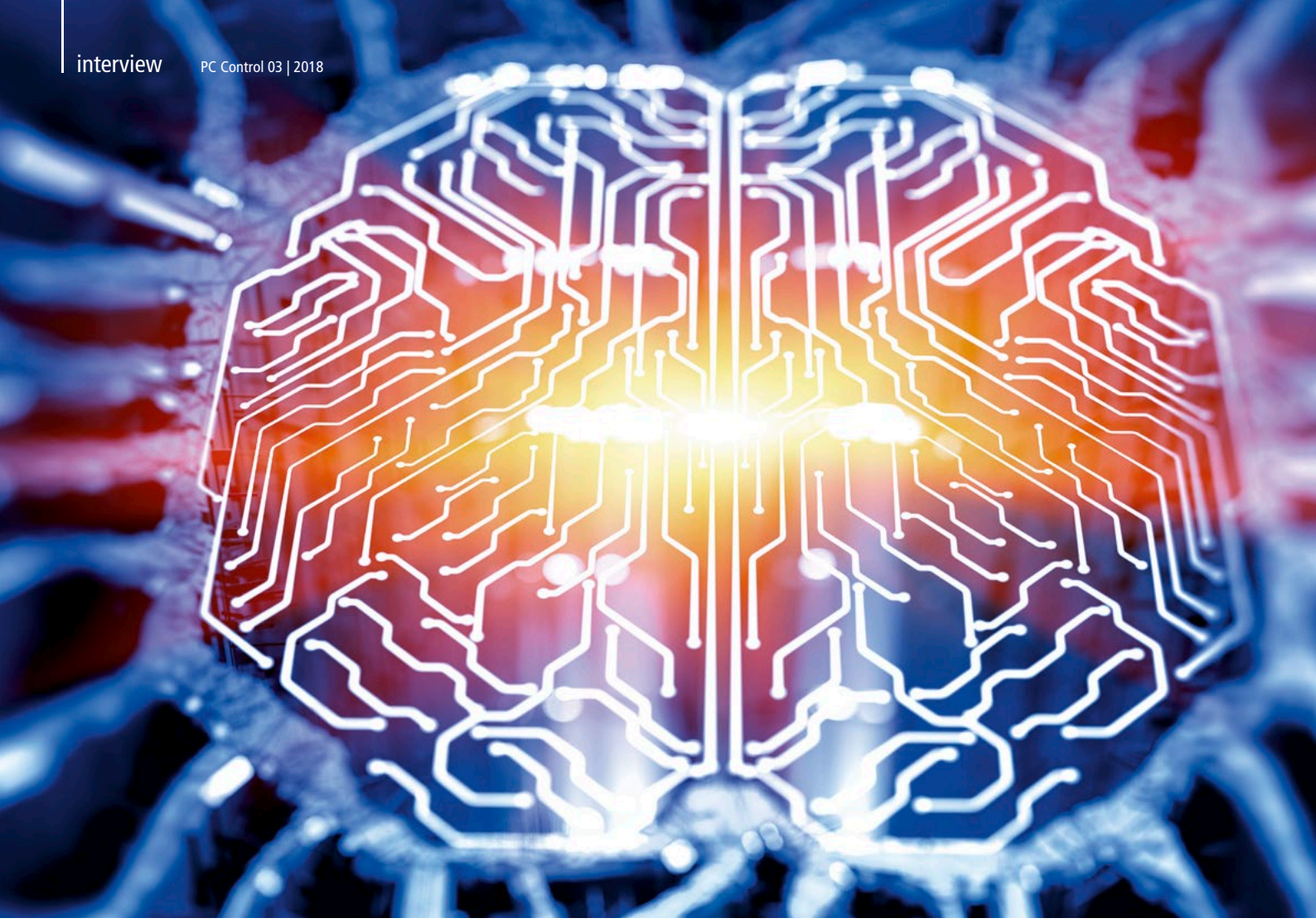


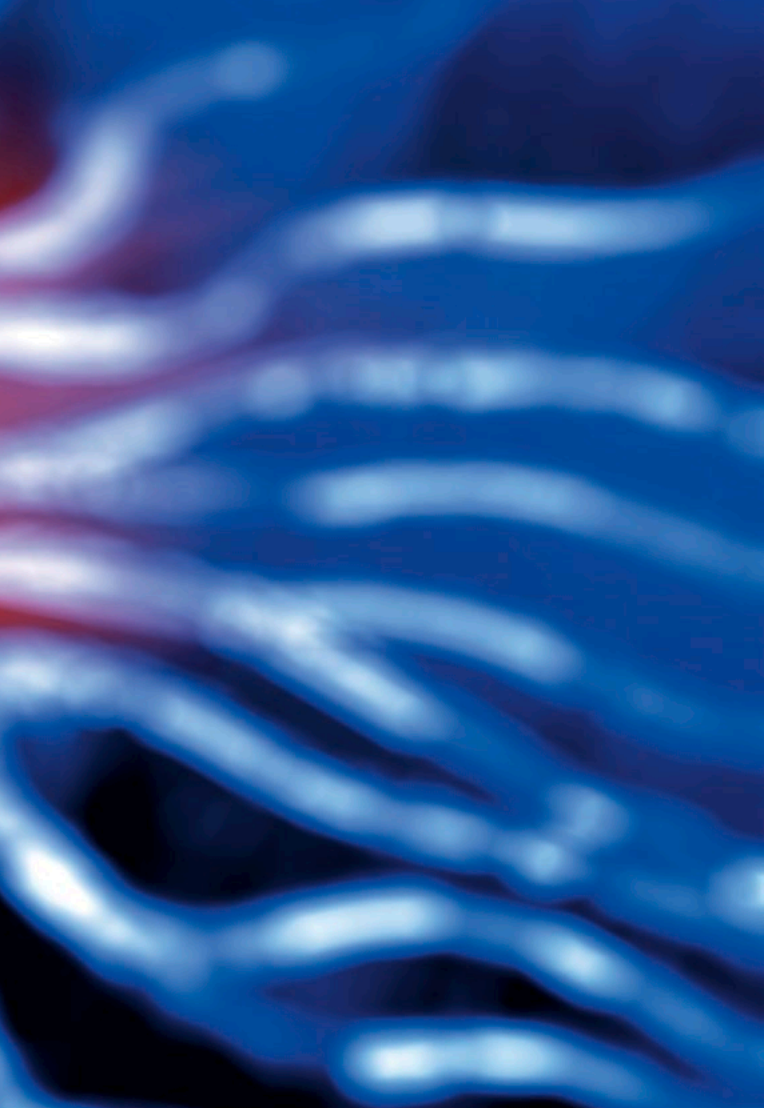
Trend interview: Should AI algorithms run on local control systems?

A differentiated approach to Artifical Intelligence, the cloud and the edge

In the mechanical engineering sector in particular, many Industrie 4.0 initiatives that users are rolling out rely on data evaluation at the machine level. With scalable computing power available in control systems and edge devices, sufficient resources can be allocated to evaluating data locally and directly on production lines. But how much intelligence can be reasonably provisioned at the boundary to the cloud, and how can tasks be divided effectively between the cloud and the edge? Dr. Josef Papenfort discusses these forward-looking questions with Andreas Gees, deputy editor-in-chief at Elektro Automation magazine, in the following trend interview.

Elektro Automation: Are there sound reasons in favor of integrating AI algorithms into local applications, that is, control systems or edge devices? What might be the advantages over cloud-based solutions, and how important a role does data security play?

Dr. Josef Papenfort: Where AI algorithms should be executed depends to a large extent on the acceptable latency and on the cost ceiling. AI algorithms rely on data and, here, it's important to distinguish between the data that's needed just once, initially, during the training process, and the data needed continuously in a trained network that is operating predictively in inference mode. For both machine learning and inference to take place in the cloud, sufficiently high data rates must be available. In cloud-based systems, this can be costly, and if the internet bandwidth available is inadequate, this may result in restrictions to the functionality. Running inference on an edge device, on the other hand, requires a one-time investment in hardware that then allows data to be shared at high speeds over a local network. The learning process, by contrast, can be conducted in the cloud: Data only needs to be uploaded once, because the process is finite, not continuous. The learning process may, of course, require considerable computing resources, but only temporarily; in this case, cloud resources charged on a pay-as-you-go basis are a good fit.



"The learning process may, of course, require considerable computing resources, but only temporarily; in this case, cloud resources charged on a pay-as-you-go basis are a good fit."

Dr. Josef Papenfort,
TwinCAT Product Manager at Beckhoff

Elektro Automation: Control tasks generally call for hard real-time systems whereas AI algorithms aren't necessarily real-time-dependent. Can these two different requirements be reconciled at the local level?

Dr. Josef Papenfort: The type of connection depends on the application implemented or supported by an AI algorithm. Such an algorithm obviously relies on data received from the real-time process – in other words a stream of data moving from the real-time to the non-real-time domain. The essential question is whether the non-real-time algorithm needs to intervene in the real-time process in response. This is not necessary if, for instance, the AI algorithm is used in predictive maintenance to gauge, say, a part's remaining useful life. As a rule, such an application will not require direct intervention in the real-time process. In a closed-loop optimization scenario, though, the AI algorithm needs to feed results back to the real-time process continuously.

Elektro Automation: What kinds of applications are best suited to running AI algorithms locally, and where are the likely computing performance limits?

Dr. Josef Papenfort: There is no straightforward answer to that question – it depends on the hardware deployed whether an edge device has the power to handle deep neural network computations in a suitable time frame. In PC-based control systems, however, the hardware can be scaled to match the demands of the application.

Elektro Automation: From your perspective, are there application scenarios – cross-site analyses, for instance – that only really make sense if run in the cloud?

Dr. Josef Papenfort: Again, it's important to distinguish here between the learning phase and inference mode. Learning is based on a set of data. If a given site generates insufficient machine-learning data on its own, data from multiple sites that perform the same process can be combined to obtain an adequate database, and the cloud offers a suitable means of merging that data. The cloud can also provide the computing power needed to initiate the learning process at a given time. Whether or not inference should be performed in the cloud as well depends, as I already mentioned, on the given application, the latency and the data rate.

Further information:
www.beckhoff.com/iot