Because models with causal convolutions do not have recurrent connections, they are typically faster to train than RNNs, especially when applied to very long sequences. One of the problems of causal convolutions is that they require many layers, or large filters to increase the receptive field. For example, in Fig. 2 the receptive field is only 5 (= #layers + filter length - 1). In this paper we use dilated convolutions to increase the receptive field by orders of magnitude, without greatly increasing computational cost.

A dilated convolution (also called *à trous*, or convolution with holes) is a convolution where the filter is applied over an area larger than its length by skipping input values with a certain step. It is equivalent to a convolution with a larger filter derived from the original filter by dilating it with zeros, but is significantly more efficient. A dilated convolution effectively allows the network to operate on a coarser scale than with a normal convolution. This is similar to pooling or strided convolutions, but here the output has the same size as the input. As a special case, dilated convolution with dilation 1 yields the standard convolution. Fig. 3 depicts dilated causal convolutions for dilations 1, 2, 4, and 8. Dilated convolutions have previously been used in various contexts, e.g. signal processing (Holschneider et al., 1989; Dutilleux, 1989), and image segmentation (Chen et al., 2015; Yu & Koltun, 2016).

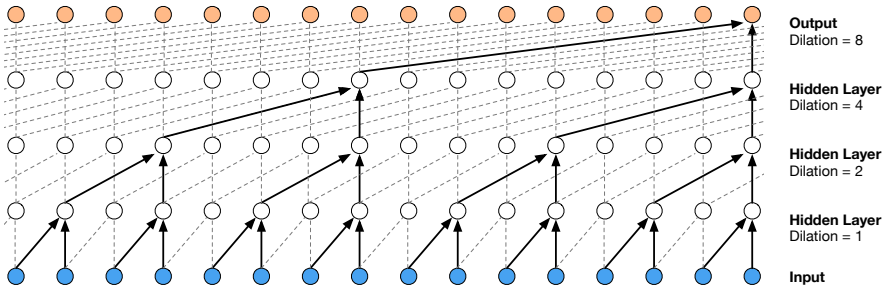

Figure 3: Visualization of a stack of *dilated* causal convolutional layers.

Stacked dilated convolutions enable networks to have very large receptive fields with just a few layers, while preserving the input resolution throughout the network as well as computational efficiency. In this paper, the dilation is doubled for every layer up to a limit and then repeated: e.g.

$$1, 2, 4, \ldots, 512, 1, 2, 4, \ldots, 512, 1, 2, 4, \ldots, 512.$$

The intuition behind this configuration is two-fold. First, exponentially increasing the dilation factor results in exponential receptive field growth with depth (Yu & Koltun, 2016). For example each $1, 2, 4, \ldots, 512$ block has receptive field of size 1024, and can be seen as a more efficient and discriminative (non-linear) counterpart of a 1×1024 convolution. Second, stacking these blocks further increases the model capacity and the receptive field size.

2.2 Softmax distributions

One approach to modeling the conditional distributions $p\left(x_t \mid x_1, \ldots, x_{t-1}\right)$ over the individual audio samples would be to use a mixture model such as a mixture density network (Bishop, 1994) or mixture of conditional Gaussian scale mixtures (MCGSM) (Theis & Bethge, 2015). However, van den Oord et al. (2016a) showed that a softmax distribution tends to work better, even when the data is implicitly continuous (as is the case for image pixel intensities or audio sample values). One of the reasons is that a categorical distribution is more flexible and can more easily model arbitrary distributions because it makes no assumptions about their shape.

Because raw audio is typically stored as a sequence of 16-bit integer values (one per timestep), a softmax layer would need to output 65,536 probabilities per timestep to model all possible values. To make this more tractable, we first apply a μ-law companding transformation (ITU-T, 1988) to the data, and then quantize it to 256 possible values:

$$f\left(x_t\right) = \operatorname{sign}(x_t) \frac{\ln\left(1 + \mu\left|x_t\right|\right)}{\ln\left(1 + \mu\right)},$$