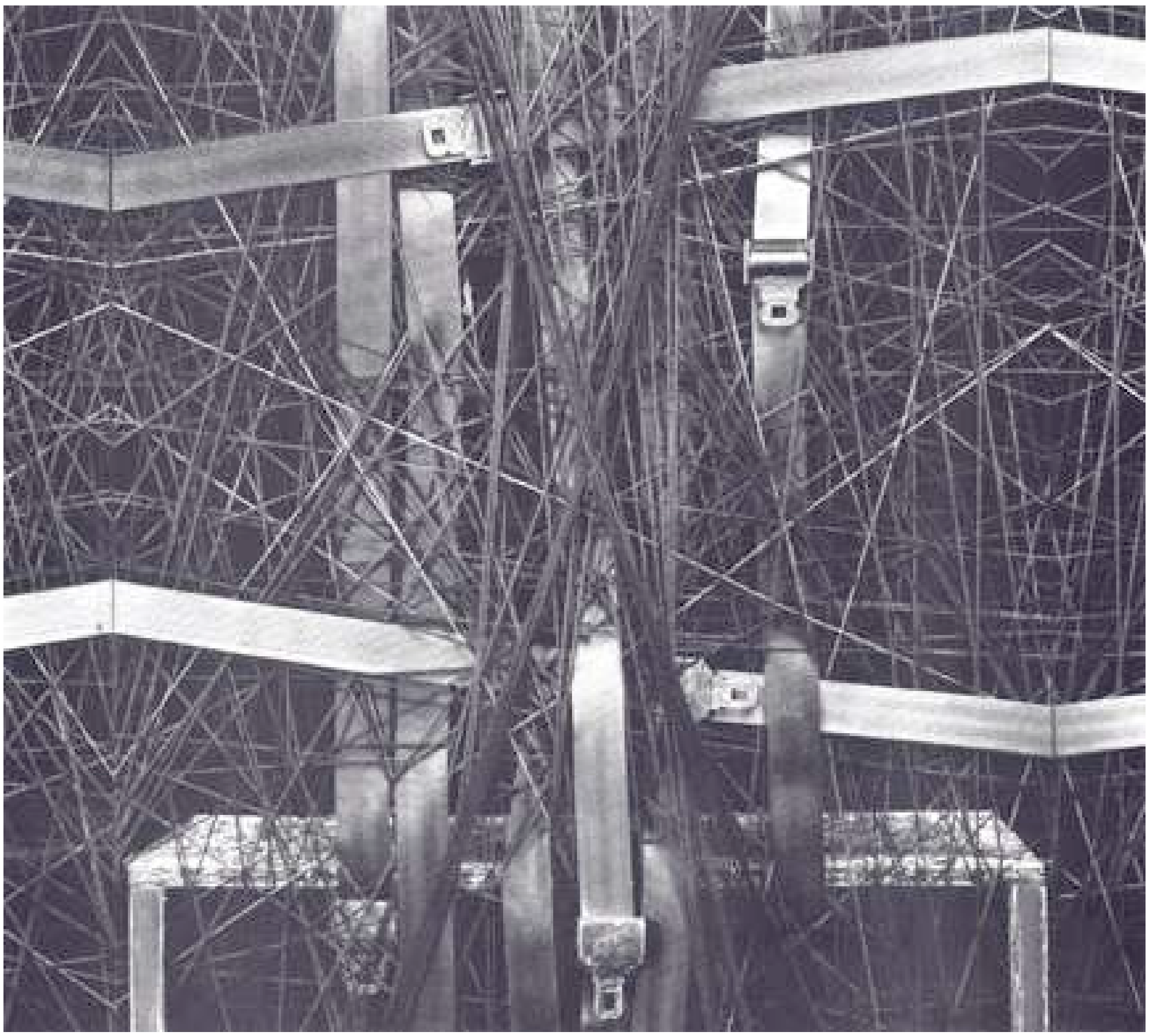
MSP



A Primer on Compression

Copyright and Trademark Notices

Credits

Compression tutorial and tutorial patches: Peter Elsea

Cover Design: Lilli Wessling Hart

Graphic Design: Gregory Taylor

About This Manual

This manual contains a series of interactive tutorials that explain the mechanics and applications of audio compression using a set of four objects licensed by Cycling '74 from Octiv, Inc. in Berkeley, California. For more information about Octiv's products, visit their web site at

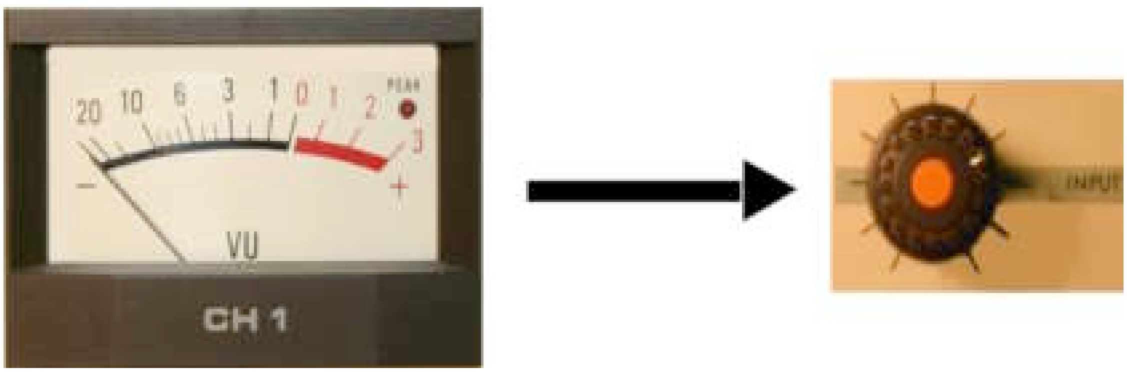
http://www.octiv.com.

These tutorials assume basic working knowledge of Max and MSP, so before delving into these tutorials, make sure you've glanced at the regular Max and MSP Tutorials first.

Index

Introduction: What is Compression?

Dynamic compression is about level control. As an example, imagine recording a trumpet playing with a guitar. If you set the levels for the trumpet, the guitar will sound very quiet when played alone. If you set the levels for the guitar, the trumpet will be distorted when it comes in. So you ride gain, watching the trumpet player and turning the knob down when he's about to play. Compressors try to do this for you, adjusting the gain to keep the signal at a good strong level.



VU&Knob

Any compression device has two parts: a circuit that measures the incoming signal, and a controlled amplifier that adjusts the gain of the output signal. The level measurement is connected to the gain control of the amplifier. This measurement is compared to a *threshold setting* to determine what the device will do. There are several options:

- Most compressors used in recording reduce the gain of signals that are above the threshold. A compressor that completely flattens out all signal above the threshold is called a *limiter*.

- Some compressors increase the gain of signals that are below the threshold, leaving signals above the threshold alone.

- Compressors used in radio broadcasting boost soft signals and limit overly strong ones. These devices may be called *comp/limiters* or AGCs for automatic gain control.

- *Multiband compressors* combine compression and graphic EQ, treating each frequency band independently. Multiband limiters are also available.

- If the device reduces the gain of signals that measure below the threshold, it is expanding. The most common example of expansion is the *noise gate*, which has a very low threshold to shut off grunge in quiet moments.

- In communications systems, signals are often compressed before they are sent and expanded upon reception. This helps reduce noise. The whole process is called *companding*.

This is all done to lesser or greater degrees based on a control that sets the amount of gain change. Gain change is expressed as a ratio. If this ratio is 2:1, a compressor would reduce a signal that is 6 dB above the threshold by half, until it is 3 dB above the threshold. If the ratio is 3:1, a +12 dB signal would be reduced to +4 dB.

The time it takes a compressor to respond to a change in signal level is important. It needs to be at least as long as a cycle of the lowest frequency coming in, which would be 1/30th of a second for a 30Hz tone. On the other hand, a rim shot is over in the same amount of time. A compressor fast enough to cover that would just pump up and down on the low tone. To handle this variety in material, compressors have controls for the speed of response. Usually there are two, *attack time* and r*elease time*, because musical sounds tend to start quickly and end slowly.

Threshold, ratio, attack time and release time are the vital parameters of a compressor. These determine what is heard and when. Most compressors have additional controls that set the details of measurement, input and output level, range of gain change and other options.

The patcher *C0. Building a Compressor* shows how a compression system is built. The signal coming in is split to a level measuring circuit and a gain control circuit. The measurement is done using the **average~** object in rms mode. This is converted to dB for the gain calculations, which aren't too complicated. In the subpatcher, the difference between the signal level and threshold is multiplied by a factor derived from the ratio. The result is converted back to absolute level and offset to create a gain correction value.

Back in the main patcher, a **>~** object determines if the correction will be applied. This will switch abruptly from unity to corrected gain, a technique known as "hard knee" compression. (Most commercial units ease gently into gain reduction mode, providing "soft knees".) The **rampsmooth~** object slows down the gain changes. Note that the attack time of the compressor is set with a rampdown message. That's because the gain is reduced when the compressor kicks in.

The final gain is multiplied by the input signal to provide the output.

> *Decibels* measure the ratio between sound levels. Sound levels vary tremendously, from about a watt down to a picowatt. This is a 1,000,000,000,000:1 ratio. With this wide a range, all you really need to know is the number of zeros in the ratio. A handy way of taming this number is to use logarithms. The formula for the dB relationship between two signal amplitudes is:
>
> $$20 \log (A/B)$$
>
> Usually B is some standard reference. With this formula, the dB value is positive if A is larger than the reference; if A is smaller, the value is negative. When the signals are the same, you get 0 dB. The reference in digital systems is almost always the full strength, so most signals are negative dB. In MSP a signal of 1.0. is 0 dB.
>
> Gain is a ratio between the signal coming in and the signal going out. This is usually in dB. If you chuck some common numbers into the dB formula, you find a doubling of the signal amplitude is a gain of 6 dB, and if it is increased tenfold, the change in dB is +20. Multiplying an MSP signal by 0.001 produces a reduction of 60 dB.
>
> Decibels are a useful way to measure signals because a change of 6 dB sounds about the same with soft sounds as with loud ones.

Measuring Signals

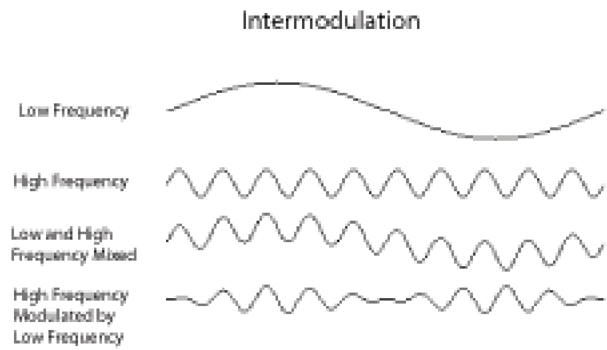
The level of a signal is tricky to measure. Since the waveform is a rapidly changing curve, there are various things to measure:

- *Peak amplitude* is the highest sample value seen. (Or lowest- since samples are positive and negative, it's the absolute value we are interested in.)
- *Average amplitude* is the average of the samples taken over some period of time.
- *RMS* is the “root mean square” of absolute values of the sample values. That means square each sample, average those, then take the square root of the average. This math duplicates the action of an analog circuit that measures level.

The type of measurement made will also affect the action of a compression device. Usually limiters respond to peak levels, and compressors respond to averages.

Intermodulation Distortion

Intermodulation distortion, or IM, occurs when low frequency signals modulate the level of high frequency signals, or vice versa.



This is a pretty raw sound, as it includes components that are the sum of the two frequencies and the difference between the two frequencies. IM is quite common in electronic circuits and speakers, but it is usually a fraction of a percent of the original signal. Anything over 1% IM is noticeable, and 3% is downright annoying.

Tutorial 1: Peak Limiting

The patcher *C1. Peak Limiting* illustrates the use of the **omx.peaklim~** object. You can see it in action by applying any signal with varying levels. Most pop music is heavily compressed, and won't do much here. Try raw drum recordings if you have any, or classical music or recordings of spoken word.

The meters show the level of the input, the action of the amplifier, and the output level. If you increase the input gain, you will see the level increase to a point, but then the gain slams down so that the output won't be allowed to reach the distortion point. You can then lower the output gain to get the original signal level with the peaks removed, as shown in figure 1.

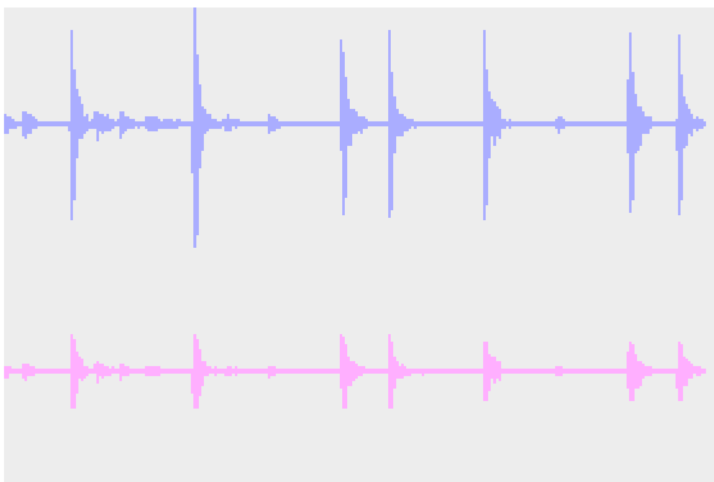


Figure 1. Peak limiting before (top) and after (Input +10 Threshold -1, output –10.)

If you raise the threshold, distortion can occur and will be very audible. If you lower the threshold, the gain indicator will be mostly in the low position, and the overall effect will be a lot quieter. Raising the output gain will restore the original signal level, but you will find quiet spots in the input contain things you didn't notice before. The overall impression of loudness should be much stronger than before. This is heavy limiting.

Figure 2 shows the before and after of heavy limiting on a drum track. You can see how the pop of the drums is stretched into a nearly continuous sound.

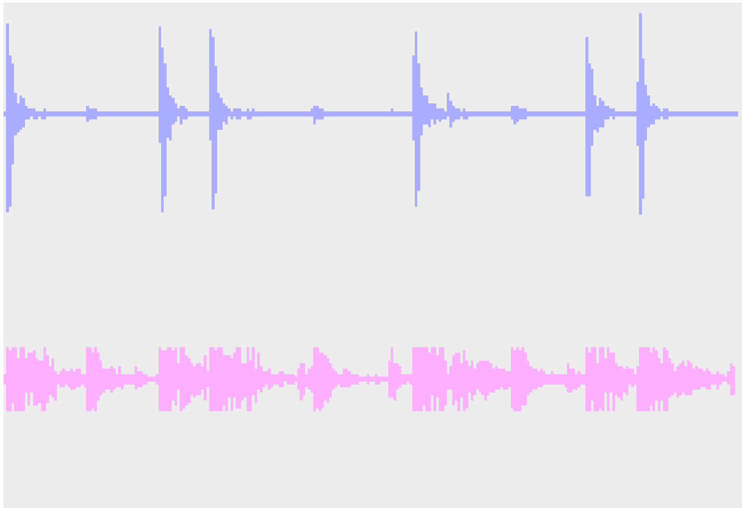


Figure 2. Input + 20, threshold - 12, output 0

The **omx.peaklim~** object has two modes that switch response times. Mode 0 is very fast and will suppress the shortest of transients. However, if the limiting is kicking in and out a lot, you will hear the signal get chopped up. Mode 1 is a bit more leisurely. This will allow peaks through in percussion tracks, but will sound nicer with vocal or instrumental material. The change is too subtle to show on the meters, but you should be able to hear it.

Peak limiting is often used as a safety net when recording unpredictable musicians. Many recorders have peak limiting built in. Heavy limiting gives a very full sound, but the increase in between- the-notes grunge makes the mix muddy. Generally, for sweetening sounds you will want to use a compressor that has more finesse.

Tutorial 2: Basic Compression

The *C2. Basic Compression* patch demonstrates some of the features of the compressor section of the **omx.comp~** object, which produces broadcast type compression and limiting. The signal source for this patch is in the **steptone** subpatcher. It generates a simple pattern of tones that gets louder and softer in 6 db steps. The level of signal is shown in the meter labeled input. When the compressor is bypassed, this set of tones is heard with no processing. Uncheck the bypass box to hear the compressor in action.

Threshold

With the opening settings in this patch, the compressor will keep all of the tones pretty much the same level at the output. Play with the threshold and you will see this level change. The center meter shows the internal action of the compressor. Gain is reduced as the bar drops.

If you have good speakers, you will notice the step tone generator produces a bit of distortion on the loudest notes. When the compressor is working, this distortion is heard on those same notes, no matter what the threshold setting. The moral illustrated here is that compression cannot fix problems that occur upstream in the recording process.

Ratio

If you adjust the ratio down from the extreme setting, you will find the output levels begin to vary when input is below the threshold. (Parts of the signal above the threshold are still limited.) At a 1:1 ratio the output will follow the input. The most usable settings are generally 2:1 to 5:1, since this will maintain some shape to the phrase. You will see this when we start compressing actual signals.

There is a practical limit to how much a signal can be amplified. Below a certain point, all you are doing is bringing up noise. This limit is adjustable on the **omx.comp~**, from 36 dB to 0. You'll find the range adjustment in the subpatcher labeled tweaks, which contains some (but by no means all) of the special parameters of the **omx.comp~** object. In hardware these are usually internal adjustments.

Attack

Set the threshold to -10 and the compression ratio around 20:1. Now listen carefully to the attack portion of the tones. Compare this with the bypassed tone. It seems that the compressor is somehow accentuating the percussive aspect of the tone. Now reduce the attack rate a bit (this is the same as increasing attack time). You will hear even more pop in the sound. In fact, it will distort badly when the attack rate is too slow. Why? At the start of the sound, the compressor is amplifying as much as it can because there's no signal. When the sound level coming in increases faster than the compressor can respond, the internal gain will briefly be in the wrong mode, amplifying when it should be reducing. This will exaggerate the percussion effect. This kind of setting is known in the trade as “punching”, and is often applied to kick drums to give a strong sense of beat without overwhelming the band. This effect can be achieved with a bit more subtlety using the delay parameter as found in the tweaks subpatcher. The delay control delays the signal to the level detector. Naturally, threshold and ratio strongly affect this effect.

There is another tweak called smoothGain that will also affect this. Gain smoothing applies an envelope to the gain control. If you set gain smoothing to zero you will immediately hear why— the gain is actually changing in steps which produce a noise like a zipper. Gain smoothing affects the attack and release rates identically.

Release

The step tone generator will also produce a steady low-pitched drone at -30 dB. If you turn it on while the step tones are running, you will hear the effect of the release rate Notice that the drone disappears when the step tone is above the threshold. That's to be expected, since the compressor is turning the entire signal down. In fact, if you watch the gain meter, you will note it accurately indicates the level of the drone. Listen to how the drone comes in—it comes up gradually, as the compression effect goes away. Now increase the ratio. The drone is louder between notes, and the pumping effect is more pronounced. Playing with the release time will provide some entertaining effects. At very short releases the drone pumping is quite noticeable, but if the release is too long, the step tones never get very loud and lose sync with the input. At the longest settings the drone goes away altogether.

Pumping makes background noise more annoying and can affect the reverberation on a voice or the sustain of a guitar. When an instrument with bass frequencies is mixed in with a sustaining instrument, the pumping effect is even worse. This is because a bass signal has to be stronger to 'sound' as loud as a signal in the treble range. As a result, even a reasonably soft bass sound will affect a piano.. You can hear this effect by setting the swap toggle. This makes the stepped tone low frequency and the steady tone high frequency.

The **omx.comp~** object overcomes this problem by dividing the bass signals off and treating them separately. The message dualBandEnable 1 as found in the **tweaks** subpatcher will turn this feature on. When dual band is enabled, bass signals have a lessened effect on the main band. (The gain indicator splits into two sections to show this.) Note that middle frequency signals will still control the bass level. Compression is often applied to individual bass tracks to control the resonance of open strings and add a bit of punch to uprights. If you are doing that, you don't want to be in dual mode.

Very short attack and release times will also produce intermodulation distortion. You can hear what it sounds like by setting the rates as high as possible and disabling dual band mode. With the tones swapped, you will hear this as roughness in the bass.

Tutorial 3: Tweaking Compression

There are more messages used to control the **omx.comp~** object than are found on many compressors. Using these messages gives you unprecedented access to the inner workings of the object and let you adjust the behavior for a wide variety of sounds. In addition to Threshold, ratio, Attack, and Release, the following messages are available: (Try them out in patcher *C3. Tweaking Compression*.)

channelCoupling Stereo signals naturally have different levels on each side. If one channel triggers compression but the other does not, the relative levels of the channels will be changed. This will result in a movement of the sound image. Therefore equal compression is applied to both channels even if only one needs it. Normally, the amount of compression is determined by the stronger channel at any moment. If channelCoupling is set to 1, compression will follow the left channel. On 2 the right channel determines the action.

SmoothGain Since the level detector works in discrete steps, applying the level measurement directly to the gain control would result in stepped action or "zipper noise". To prevent this, an envelope (similar to **line~**) is applied to the control signal. SmoothGain controls the time of this envelope.

Delay delays the application of the measurement to the amplifier control. This will allow fast peaks through for an extra punchy sound. Combined with a fast attack time, this will produce strong but very short drum kicks, perfect for hip hop.

Sidechain A Sidechain filter applies the inverse vocal EQ described in tutorial 4 to the measurement circuit. This will reduce the effect of vocals on the output level, useful if you are compressing a full mix. It will keep the voice from pushing down sustaining parts such as organ.

Noise Gate Noise gating is almost always necessary when you compress audio from the real world. For instance, if you are recording a guitar amp, the compressor will emphasize the amp hum between notes. To help prevent this, the **omx.comp~** has a built-in noise gate. When enabled, this applies downward expansion to any signal below the ngThreshold. There are no attack and release controls because gating is normally applied to very quiet signals.

Release The three messages gatingLevel, freezeLevel and progressiveRelease make automatic adjustments to the release timer based on the input signal. Signals below the freeze level will not trigger any amplification. Signals above this but below gating level will have a slow release. Progressive release speeds up the release time on stronger inputs. This means that softer signals (where the ear is less sensitive to dynamic change) will not be as compressed as loud ones.

LimEnabled puts an absolute limit on the strength of the output signal. This prevents a signal overload when the controls are set for punching.

Parameter Messages

The omx objects from Octiv, Inc. send messages from the third outlet when parameters are changed. These are in parameter lists, which contain 7 fields:

1. Scope 0 = global, 1 = presetable, 2 = end of list.

2, Name of parameter.

3. Current value.

4. Maximum value.

5. Minimum value

6. Display value.

7. Units – this is “arbitrary” for most items.

The parameter values are arbitrarily scaled, most from 0 to 100. Some of these can be converted to familiar units. Any dB parameter can be converted by looking at the value shown when set at 0 – that’s the minimum value (100 corresponds to 0 dB). Divide 100 by this value to get the scaling factor. Then to set a value, add the absolute of the minimum and multiply by the absolute of the scaling factor, like this:

- To convert agcThreshold (range from –36 to 0) use an **expr** object with the following arguments: (36 + $i1) * 2.77.

- To convert ngThreshold (range from –90 to 0) use an **expr** object with the following arguments: (90 + $i1) * 1.11.

It’s probably easier to capture the numbers coming out in the parameter messages and display those.

- Ratio uses exponential values. To convert the numbers given to ratio use an **expr** object with the following arguments: ln($f1) / ln (1.04).

Presets

With so many parameters, some method for recalling combinations of settings is essential. The compressors have a few presets built in (selectable using the ChoosePreset message), but the ultimate solution will be to save the entire setup in a **coll** or **pattr** object. The SaveSettings message will give you a dump of all of the parameters.

Metering

The right outlet provides gain metering if enabled with the meters message. The outputs are a list ready to be connected to a **multislider** object. The list for the **omx.comp~** shows left and right values for AGC gain, noise gate and limiter. The meterRate message determines how often these are sent. Alternately, you can get the values using the meterData message. The meters for the 4-band and 5- band compressors are quite elaborate.

Compressors can be used on any recording where there is a need for them. In the recording industry compression is routinely used on vocals, drums and bass. To try these out, you can open and use the the *C0. Building a Compressor* tutorial patch again, get some raw materials and create your own compressor patch with an **omx.comp~** object, or use your favorite VST plug-in compressor.

Vocals

Using compression on vocal tracks can help make the lyrics more understandable. Singers will typically shape a phrase something like this:

"Oh give me a home, where the Cadillacs roam...."

This might sound lovely taken by itself, but when mixed with a typical band or piano accompaniment the weaker syllables "oh" and "where" will be masked out. Either that or you will get a Bing Crosby mix where the band sounds like it is in the next room. If you ask the singer to even the phrase out, they will be so uptight that there will be no expression at all. You can help these folks out with a bit of subtle processing.

The first processing you can apply is equalization. If the tracks are equalized properly, they can be made clear with minimal compression. Vocal equalization is a mild boost (3 dB or so) somewhere between 2 and 4 kHz. You tune it to the specific voice and what you are listening for is clear vowels. You then put the inverse equalization (a 3 dB cut at the same frequency) on all other tracks that might interfere (such as guitars), making a window in the spectrum for the voice to come through. With only a 3 dB change, you haven't made an obvious difference in the voice or any single instrument. Many microphones already have this sort of EQ boost built in, but you still need to cut the other tracks. (In fact, if they were recorded with the same microphones, you'll need to cut them even more.)

The next thing to do is to add some compression. What you want to do is maintain as much of the singer's inflection as you can, while bringing out every syllable—or nearly every syllable. Play the track in bypass mode first and watch the meters. Make a note of the range covered and particularly the reading of the loudest spots. Set the threshold a bit below the loudest mark so the compressor is not working very hard. It will recover fastest and distort the least if it never gets into extreme change. (Remember, the **omx.comp~** object is a limiting compressor. Set the threshold at 0 and boost the input until the high point is just below this.)

Start with gentle (2:1) ratio and fast attack and release. Increase the ratio until the meters show the phrases have flattened out enough, generally within 12 to 6 dB throughout, and then slowly increase the attack time (reduce the rate). There will be a point where the voice suddenly starts jumping out. Keep it if you like it, but it will sound more natural if you back off of the setting a bit. Now increase the release time (decrease the rate) until the notes sustain well but you aren't getting bursts of breath.

For some singers, this procedure may be totally wrong—what you need to do is learn how the ratio, attack and release work with voices in general so you can respond to what you hear. Here are some situations to watch for:

- A singer who does not sustain tones will benefit from short release times. Short release works like a guitar sustain pedal. On the **omx.comp~** you may want to tweak the gating threshold or progressive release parameters when you do this.

- Some singers belt certain syllables. This is where the **omx.comp~** object shines—just reduce the threshold until the phrase evens out. With other devices, you may have to patch in a second limiter.

- Some singer/microphone combinations will produce a lot of sibilance. A few compressors have a “de-essing” feature, which boosts the highs going into the measuring circuit. The hiss will then trigger extra gain reduction. The **omx.4band~** object can be used to do this.

Bass

Bass players have to work hard to keep all of the notes even on their instrument. One reason for this is that the open notes are much more resonant than fretted ones. Amplifiers contribute their share of problems, and by the time you add the strangeness of close miking a loudspeaker cone, the levels can vary widely. Compression can help; the amount of compression to use depends on the individual player, but it generally doesn’t have to be too heavy. When working with rock, you’ll need a lot to keep the bottom solid. The attack will vary according to the type of music. Jazz bass requires a smooth sound, so a quick attack will be used, but in hiphop the bass is almost a percussion part. Slow down the attack time to get a punchy sound. Avoid stretching the notes with slow release—it’ll sound muddy.

Drums

If you only use overhead microphones for a drum set, all you want is peak limiting, because compression on cymbals usually sounds terrible. A snare microphone may be compressed if the head is dead or you want extra snare rattle, but a more effective approach is gating with a rather high threshold. That way most of the snare sound comes from the overheads, but the pop from the snare microphone will happen just a bit earlier and will give the rhythm a crisper sound. The gate keeps any leakage from the high hat out of the mix. To gate with the **omx.comp~**, object, enable noise gating but not AGC.

Kick drum processing could fill an entire chapter of a book. In general, you are trying to find a balance between punch and mud. The compressor is used to suppress the ring while keeping or even amplifying the strike. A slow attack is indicated here, with the amount of compression determined by the length of ring from the drum.

Many times you will be trying to make up for the strange things drummers do to the kick drum to keep the sound from bleeding into other tracks. An untamed bass drum will ring quite a long time, and that very low frequency gets in everywhere. The result is you often see kicks with heads removed, holes cut in them, and various mutes attached. (The last one I recorded had a ten pound pillow in it.) The resulting sound is best described as "phlub". One common trick is to use gated reverb. A fairly thick reverb is applied to the drum and fed through a compressor set up for external gating. The gate is controlled by the unprocessed sound of the drum, so you get a lot of extra sound, but it does not tail off like normal reverb. Some reverbs have gated settings, but you have little control over the effect.

Tutorial 5: Multiband Compression

It's usually the case that the bass, drums, vocals and other instruments are individually compressed before they are mixed together. Sometimes this is not possible or desirable. For instance, a radio station or a DJ may wish for a louder and more consistent sound than assorted CDs happen to provide. Simple compression is seldom satisfactory for this situation, because sustained parts will be pumped by the vocals and drums. To prevent this, a multiband compressor treats each section of the audio spectrum independently; you can have a full bass and constant rhythm while still letting the voice be heard.

The multiband compressor has found creative uses beyond the original intent of spicing up FM radio. If you start with a fairly intense broadband headbanger mix, you can use the device to cram every last dB into every octave for that solid wall-of-sound effect. Or, with a bit of restraint, you can master a recording to give a comfortable experience on a wide range of playback systems.

Spectral Balancing

Good arrangers know how to orchestrate a score to give a balance between bass and treble that is satisfying and keeps the listener's attention. This is called *spectral balance*. A similar effect can be achieved by compressing bass, midrange, treble, and high frequencies for consistency and adjusting the levels of each to fit a curve that matches mid and treble, leaves slightly less bass, and somewhat lower top end. This curve comes from analyzing the overall response of many successful albums.

The 4-band compressor does a good job of solidifying a mix. The patch *C5a. Multiband Comp 4band* shows the **omx.4band~** object in action. This is a really complex gadget, far more than four compressors lashed together. There is quite a bit of processing before the band-specific auto gain controls (AGCs).

There is a downward expander first, with an adjustable threshold to sweep away noise and dirt. This is followed by an overall AGC, then the signal is split into four bands that are (approximately) deep bass, normal bass, midrange (where most of the music occurs) and highs. There is a second downward expander on the high section, since high frequency noise is especially annoying. Each band has a control for drive (the threshold is fixed at 0, so more gain here means more compression), attack, release, and level into the final mix. There's a final limiter to prevent any peaking.

Start by trying the three presets on some commercial music (probably already sort of squashed). Play with the outmix faders to get a sense of where the bands are. Now try some raw tracks and adjust drive, attack and release to get the sound as full as possible. As always, you get the most obvious effects when the meters are dancing.

The meters are rather enigmatic, but here's what they mean, left to right:

- Left/right input. Provide enough signal to get these pretty high.
- Release gating. This locks all release times when the input gets low—it helps prevent pumping of background noise. The indicator goes high when release gating is on. (Note how the gain bars all stall.)
- The two expanders, wideband and highs. This shows the gain of the expanders, so when they drop, signal is being shut down.
- The left and right master AGC gain. Usually you want the attack and release fairly slow, as these are just meant to keep the overall level in the center of the dial.
- The gain in the four bands. Unity gain is the lowest position.
- Left and right limiting action. If these are pushing the floor, back off on the input.
- Left and right output levels.

After a bit of practice you should be able to produce a full satisfying sound on any kind of material. The best test is to get it to sound like the same music when played back softly or loud.

Mastering

Mastering for different qualities of playback gear is a special art. The major difference between systems is the bass response of the speakers. We would all like true response down to the limit of hearing, but physics and economics work against that. Most budget speaker systems bottom out at 100 Hz or so, the bass driver's resonant frequency. If a speaker is resonant at 100 Hz, a 50 Hz signal will excite that resonance, and so will a 55 Hz signal. The result is that anything lower that the speaker is equipped to handle will just add mud to the low end. Mastering engineers know this and roll off the signal below 75 Hz or so. So, what's left to offer someone who spends the money to get speaker with a good bass response? Instead of just throwing out the deep bass, we want to control it so that there's something for good speakers to reproduce without getting the less expensive speakers overly excited. A multiband compressor is just the thing, because we can keep the bass full without crossing the line. (To do this properly you have to listen to the mix on appropriate speakers, which is why you often see low-end Radio Shack and Yamaha boxes in million dollar recording rooms.)

We also usually roll off the high end just a bit, as compressed highs keep the attacks crisp. However, if the level is too high the effect is overly bright.

The **omx.5band~** object compressor is detailed enough to provide this kind of fine control. (See the patcher *C5b. Multiband Comp 5band*)

In addition to the features of the 4 band, the 5 band features:

- Five bands. The midrange is split into two, and the low bands are a bit lower.
- An individual downward expander (noise gate) on each band.
- Individual threshold and limiting on each band.
- A switch on each band that sets what happens at the threshold. Amplification happens below the threshold, but you have your choice of limiting or unity gain above.
- The Overall AGC is split into two bands, with the lower labeled "Bass Enhancement." These are slow-acting circuits that keep the input where the compression can have best effect.
- An extra "soft" clipper on the deep bass. Set this so small speakers aren't thrown into resonance by notes below their range.
- A "spatial enhancer." This separates the channels of mixes that are nearly but not quite mono. (A very common mixing style, especially over the radio.) It does so by subtracting a bit of left from the right signal and vice versa, but only when the two are nearly the same. The parameters for this are desired difference between the side signal and the combined (mono) signal, maximum gain on the difference signals, and speed of response to changes in separation.

The presets in the for the **omx.5band~** object illustrate various kinds of processing curves. The "Universal" setting is just a general boost to the sound, with a 3-to-1 compression and no limiting. Bass enhancement is switched in for a bit of warmth. The "Pop" setting, on the other hand, sets a 50:1 compression with hard driven limiting in the lowest two bands. The gives a solid bottom to the sound. The "Hit Radio" settings are a compromise between the two. You will note that the times are all about the same, with appropriately slower attack and release in the low bands. "FM Radio" differs primarily in the high end, which is compressed a bit tighter than the pop and hit settings.

Since the **omx.5band~** object uses 7 times as much CPU as the four-band version, it may not be appropriate for everyday use—but it will be perfect for a few tough tasks.

Gating and Keying

In *keying*, one sound turns another on (keying is sometimes referred to as “side-chain gating” and the “key” is sometimes called the 'side-chain' signal). This can produce some interesting electronic music effects, such as drum rhythms imposed on a chorus. The patcher *C6a. Keying* shows the compression patcher modified for keying. This did not require many changes—simply the separation of the control signal from the processed signal and the removal of the switch that triggers unity gain at the threshold. One more thing—the sense of the gain calculation (difference between the control and the threshold) is reversed by exchanging the leads on the math subpatch. Thus the amplification of the left signal is set by the level minus the threshold. When the control level is below the threshold this is negative, giving a reduction in gain.

This kind of patch can be fussy about levels, so an output gain is provided to give extra boost if needed.

With this setup, a signal in the left channel will not pass through unless a signal is present on the right channel. The ratio determines how much effect the control has on the left signal. If the ratio is set to 1:1, there is no effect at all. For ratios much above 2:1, there may be some distortion. The threshold determines how strong the control has to be to turn the left input on.

Microsounds

We usually use the noise gate to keep from amplifying things that should not be heard when compressing. Sometimes these in-between sounds are very interesting in themselves. I like to record very tiny sounds, like the ticking of a mechanical stopwatch. When doing that, I use quite a lot of compression to bring up the ting of the spring while keeping the grind of the winding at a reasonable level. The patcher *C6b. Microsounds* is set up for this. Compression is applied to signals that are below the threshold, and signals above are unchanged. This leaves the notes essentially alone, but the sounds between the notes are brought out. (This is how a guitar sustain pedal works.) Of course there is a lot of junk at very low levels we don’t want to hear, so the patch has a second threshold; levels below this threshold are unaffected.

Expansion

The *C6b. Microsounds* patcher has one more trick. The ratio can be set below 1:1. Fractional ratios give expansion, where soft sounds are made softer. This can restore tracks that have been over compressed, or separate out the loudest sounds in a pattern.

Ducking

Sometimes you use one sound to control the loudness of another. An example would be the automatic ducking you hear on talk radio. The caller alone sounds loud and clear, but as soon as the host speaks, the caller is turned down. This can be done using the patcher *C6c. Ducking*. It's just like the previous patcher, but with the control level again subtracted from the threshold. The action of the patch is reversed, with the signal getting through the left only on the lack of control. Again, the ratio affects the amount of gain change and the threshold sets the trip point.

Controlling Feedback

Delayed feedback is one of the most common MSP tricks, but we all know how hard it can be to keep the echoes from building up and overwhelming everything. If you add a **omx.comp~** object in the feedback path, as shown in the *C6d. Feedback* patcher, you can forget about the levels and concentrate on getting the rhythm right. Compressing and gating have a lot to offer looping patches too.

This is just the tip of the compression iceberg. Most engineers practice and experiment with compressors throughout their careers, and develop a wide repertoire of tricks. Compression can add impact to sound effects like thunder and breaking glass, make a sleepy announcer sound lively, and balance the backup singers in a band. Many groups and producers owe their "signature sound" to some magic compression setting, and it helps transmit the voices of operatic tenors safely over television. In short, compression is one of the fundamental tools of modern audio technology.

Index