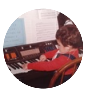
Hybrid Sonification for the Relative Strength Index in Technical Analysis of Financial Markets

Conference Paper · August 2013

1 author:



Pablo Cruz
Universidad Técnica Federico Santa María

Hybrid Sonification for the Relative Strength Index in Technical Analysis of Financial Markets

Pablo Cruz Navea
Departamento de Informática
Universidad Técnica Federico Santa María
Avenida España 1680, Valparaíso, Chile
pcruz@inf.utfsm.cl

***Abstract*—Technical indicators have been widely used in the history of technical analysis of financial markets. In general, these indicators are used in order to summarize data from the market. Technical indicators are often difficult to understand for the non-experienced investor and graphical interpretations are exposed in order to help the understanding. In this paper, a hybrid sonification is presented for the Relative Strength Index as an aid to the understanding of this technical indicator. As a proof of concept, a prototype developed in a real-time sound synthesis language is presented. The prototype is used with data publicly available from a particular stock in the chilean market.**

Keywords: Parameter mapping sonification, earcons, technical analysis, sound synthesis.

I. Introduction

Technical analysis of financial markets is defined by J. Murphy as the study of the market action for the purpose of forecasting future price trends [1]. This common-accepted old definition of technical analysis also indicates that the study is done primarily through the use of charts [1]. Despite the generality of this definition, several techniques have been developed and deployed in order to support the general methodology of technical analysis of financial markets. One example of these techniques is the use of technical indicators in order to summarize data from the market and to provide some guidance to the actions of the investors.

One commonly-used technical indicator is the Relative Strength Index (RSI) developed by J. Wilder [2]. RSI is defined as follows [1] [2]:

$$RSI = 100 - \frac{100}{1 + RS}$$

where RS is defined as follows:

$$RS = \frac{\overline{X_u}}{\overline{X_d}}$$

In RS, $\overline{X_u}$ is the average of X days' up closes and $\overline{X_d}$ is the average of X days' down closes. The value of the RSI fluctuates between 0 and 100. Setting the number of days (X) is left to the investor (fourteen days was the value proposed initially by [2]). In general, movements below 30 are considered *oversold condition* and movements over 70 are considered *overbought condition* [1].

Using technical indicators such as the RSI is a difficult task for the non-experienced investor. Even for the experienced investor, the interpretation of RSI can be difficult as technical analysis tools often have multiple indicators running at the same time. It is in this context in which a hybrid sonification is presented for the creation of acoustic displays as an aid to the interpretation of this indicator. Although the main goal of this sonification is to serve as an aid to the interpretation of this indicator, there is no restriction in using this proposal as an alternative to the numerical and graphical visualization.

As a proof of concept, a simple implementation in the real-time sound synthesis language SuperCollider [3] is presented. The implementation uses real and official data gathered from Bolsa de Santiago de Chile which is the main stock market in Chile.

In the following, the paper is structured as follows. Section II presents the related work, mainly focused in what sonification is, what has been done with sonification and a brief overview of which sonification techniques are used in the development of this work. Section III presents the development of the sonification. Section IV presents details of the implementation developed as a proof of concept. Finally, section V presents the conclusions of this paper and future work.

II. Related work

Sonification is a part of auditory displays [4]. Auditory displays is a young research field in which the potential of sound to support human activity, communication with technical systems and to explore complex data is recognized [5][6].

One of the first definitions of sonification is the following: a subtype of auditory displays [4] in which data relations are transformed into perceived relations in an acoustic signal [5]. Definition for sonification has evolved through the years. Hermann goes beyond this initial definition and states that a transformation from data to sound signals can be considered sonification if and only if the sound reflects objective properties or relations in the input data, the transformation is systematic (i.e., there is a clearly defined specification of how data changes the sound), the sonification is reproducible and the system can be used with different data [6].

Hermann also states that sounds generated by sonification are different from other ones we hear in real life. In this context, sounds generated by sonification are a subset of functional sounds, that is, they are organized (they are not random) and serve a certain function or goal [6]. Randomness can be used to create specific sounds, but Hermann insists that if randomness is used in the creation of auditory displays, the procedure which uses randomness must be clearly defined and specified [6].

Several techniques exist for sonification. Here, we briefly describe audification, auditory icons, earcons, parameter mapping sonification and model-based sonification. Audification aims to directly translate a *data waveform* into sound [7] and it should be noted that it is not a requirement that data belongs to the sound domain. Auditory icons mimic some frequently-used non-speech sounds that might be familiar with the sonification user due to the experience in the real world [8]. One known example of auditory icon is the typical sound associated with the operation *"delete a file"* which tries to emulate the sound of a paper being destroyed. Similar and complementing auditory icons, there is the category of earcons. Earcons are also non-speech sounds with the intention of providing information to the user about some computer object [9], but they are musically structured and the relationship between the sound and the information that represents is not known in advance [10] (i.e., the relationship must be learned by the user of the sonification). Parameter mapping sonification is a technique in which data or information is mapped to sound parameters [11]. Parameter mapping sonification is based in the idea that sound is a multidimensional entity. A more interactive alternative is model-based sonification in which acoustic responses are related to user's actions [12][13], that is, the user excites the sonification model and waits for an acoustic response [6]. This interactive sonification allows, for example, a more interactive alternative for data exploration.

Audification, auditory icons and earcons are mostly useful in sonification of low-dimension data while parameter mapping and model-based sonification are better for high-dimension data [12], being model-based sonification the most suitable for high-dimension data overcoming some problems that appear when using parameter mapping in high-dimension data [12][14].

Some of the functions of sonification generally mentioned in the literature are: data exploration, alerts, warnings, status indicators [15] and process monitoring [22]. For example, sonification has been used as an aid in the process of analysis and interpretation of seismic data [16], as a complement for statistical analysis when studying social data [17], for network activity monitoring by expressing certain events through different sounds [18], to create a tool for highlighting normal and pathologic rhythmic activity in electroencephalography [19], for listening phase transitions in spin models as a complementary data analysis method in statistical physics [20] and for identification of autocorrelation in capital market trading data [21].

In this work, a hybrid sonification is presented for the Relative Strength Index briefly described in section I. First, earcons are used to indicate when the RSI enters in oversold and overbought areas. Second, parameter mapping sonification is used for creating acoustic displays that convey acoustically the RSI value when it is in oversold or overbought areas.

To this date, no other work has presented a hybrid sonification for the RSI index.

III. The Sonification

The sonification proposed in this paper is a hybrid one in the sense that it uses two types of sonifications. The first type is composed of two earcons. Earcons are used in this sonification in order to indicate in a distinctive way when the RSI enters in an oversold or in an overbought condition. The second type is a parameter mapping sonification which maps the RSI value in oversold and overbought conditions to the frequency of a real-time generated sound.

A. Earcons

As described in section II, an earcon is a non-speech sound that is created for conveying some information to a user in an acoustic way. In the sonification proposed here, two earcons are necessary:

1) Oversold line crossing earcon. This earcon appears as soon as an oversold line crossing is detected. The earcon is intended to appear only in direction from up to down, that is, the earcon appears **only when the RSI enters in the oversold section**.
2) Overbought line crossing earcon. This earcon appears as soon as an overbought line crossing is detected. Analogously to the oversold line crossing earcon, the earcon for this case appears **only when the RSI enters in the overbought section**.

As the relation between the earcon and the information conveyed is not known in advance, some brief training of the investor using the trading system with the sonification may be needed. The selection of particular sounds that represent the earcons is left as a choice in the particular implementation. However, there are two restrictions in this side of the sonification:

- The earcons are different. The two earcons are not allowed to be the same sound.
- Earcons appear only when the RSI enters in an overbought or oversold condition. Earcons are not intended to inform that the RSI value has left one of these areas.

B. Parameter Mapping

The parameter mapping side of this sonification allows the user to gain insight regarding the value of the RSI near the overbought and oversold lines, but inside these areas. The parameter proposed to be the main characteristic of the acoustic display is the frequency of the sound. Thus, the mapping is between the value of the RSI and the frequency of the sound. We use the initial and commonly-used values for defining the oversold and overbought lines, that is, an RSI value of 30 is set to the oversold line and an RSI value of 70 is set to the overbought line.

The mapping between the RSI value and the frequency of the sound is linear. Using the two-point form of the linear equation shown in (1), the frequency formulas for the overbought and oversold areas are determined.

$$f_{area} = m_{area}(RSI - MinRSI_{area}) + MinFreq_{area} \quad (1)$$

Where m_{area} is defined in equation (2) and $area$ represents the overbought (ob) or oversold (os) area.

$$m_{area} = \frac{MaxFreq_{area} - MinFreq_{area}}{MaxRSI_{area} - MinRSI_{area}} \quad (2)$$

For an oversold area, the minimum frequency proposed is 200 [Hz] and the maximum is 300 [Hz]. For an overbought area, the minimum frequency proposed is 400 [Hz] and the maximum is 500 [Hz] (a summary of values used for the frequency mapping is given in table I). Given these values (and the values of RSI that determine oversold and overbought areas), the values of m_{area} can be determined as follows:

$$m_{ob} = \frac{300 - 200}{30 - 0} \approx 3.3$$

$$m_{os} = \frac{500 - 400}{100 - 70} \approx 3.3$$

By using the linear equation, the frequency function for an oversold area is determined in the following way:

$$f_{os} = m_{os}(RSI - MinRSI_{os}) + MinFreq_{os}$$

$$f_{os} = 3.3(RSI - 0) + 200$$

$$f_{os} = 3.3RSI + 200$$

For the overbought area, the frequency function is obtained in the following manner:

$$f_{ob} = m_{ob}(RSI - MinRSI_{ob}) + MinFreq_{ob}$$

$$f_{ob} = 3.3(RSI - 70) + 400$$

$$f_{ob} = 3.3RSI + 169$$

In words, the mapping can be expressed as follows:

- If the RSI value is below 30 (that is, the RSI is in an oversold area), the RSI is mapped to the frequency as indicated by:

$$f_{os} = 3.3RSI + 200$$

- If the RSI value is above 70 (that is, the RSI is in an overbought area), the RSI is mapped to the frequency as indicated by:

$$f_{ob} = 3.3RSI + 169$$

These functions are used to specify the frequency value that is mapped to the RSI value.

No other parameters of sounds are used in the parameter mapping side of the sonification. Thus, duration, amplitude, timbre and other parameters are left to specification in the particular implementation.

Although it is not indicated which type of sound is used to express the frequency change, a simple sine wave oscillator (as shown in section IV) can be used. The important requirement here is the mapping of the RSI value to the frequency of the sound.

TABLE I. VALUES USED FOR FREQUENCY DEFINITION

-	Oversold Area	Overbought Area
MinRSI	0	70
MaxRSI	30	100
MinFreq [Hz]	200	400
MaxFreq [Hz]	300	500

Values for the minimum and maximum frequencies were selected using the experience of the author and after testing the generation of sounds with the minimum and maximum values of the RSI (i.e., testing a sound with frequency 200 [Hz] and with frequency 500 [Hz]). These tests were done in order to select suitable frequencies to avoid damaging human ear and computer speakers.

IV. Implementation and Proof of Concept

As a proof of concept, an implementation was made in the real-time sound synthesis programming language called SuperCollider. SuperCollider is a domain-specific programming language and it is a blend of Smalltalk, C and ideas from several other languages [3]. In this section, a conceptual description of the program is presented. Source code of the program can be downloaded (along with brief instructions) at http://www.temporubato.net/pcruzn/computerMusic/RSISon.

The RSI Sonification program is composed of three big components, as depicted in figure 1.

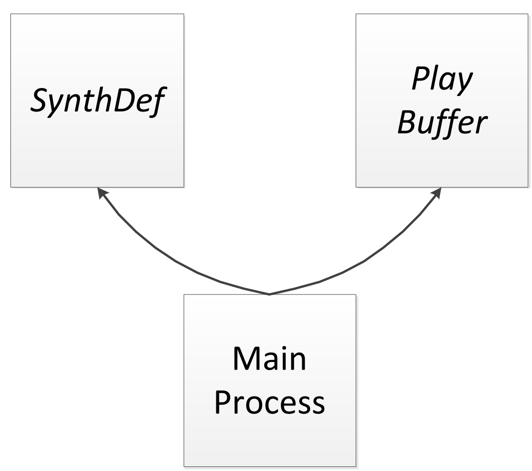


Fig. 1. Graphical view of the components of the implementation

The first component to be described is the *SynthDef*. A *SynthDef* is an encapsulation of *UGens* and their relations (for more information about *UGens*, the reader can review [3][25]).

This *SynthDef* is used for creating the acoustic display that is associated with the value of the RSI. Recall from section III that the value of the RSI is mapped to a frequency value that is passed to the *SynthDef*. In this particular implementation, the *SynthDef* creates a sine wave (see code in figure 2). A sine wave oscillator (which generates a sine wave sound) can be expressed in continuous time as a function $u(t)$:

$$u(t) = Asin(2\pi f_0 t + \phi)$$

in which A is the amplitude of the wave (volume, in terms of sound), f_0 is the frequency of the wave and ϕ is the initial phase of the wave. It should be noted that in one sounding wave, the phase has no audible effect. Detailed information about sound synthesis can be reviewed in [23] and in [24].

As figure 2 shows, the frequency given by the mapping described in section III-B is a parameter for the oscillator. The exact mapping in the program is the same described in section III. Once the mapping between the RSI value and the frequency has been done, the frequency value is used as a parameter for the *SynthDef* (see code in figure 3).

```
SynthDef ("sineWave", {
      arg freq;
      var out;

      out =
      SinOsc.ar ([freq, freq], 0, 0.5)
      * EnvGen.kr (Env.perc(0, 2), doneAction: 2
      );

      Out.ar (0, out);
      }
).add;
```

Fig. 2. *SynthDef* for the sine wave oscillator

The second component is called *Play Buffer*. A *Play Buffer* is simply a location in memory which holds the sound wave that represents the earcon. In this implementation, two *Play Buffers* are used as two earcons are necessary. As code in figures 4 and 5 shows, one earcon is necessary for the acoustic display that represents the crossing of the overbought line ($RSI > 70$) and the other for the acoustic display that represents the crossing of the oversold line ($RSI < 30$).

The third component is called the *Main Process*. This component, which is a Task object in SuperCollider with time waiting of one second between cycles, is responsible for (1) taking the value of the RSI and evaluating if there existe one of the two line crossings described before, (2) mapping the RSI value to a frequency value and (3) when (1) and (2) are completed, calling the *Play Buffer* and *SynthDef*, respectively, to generate the acoustic displays.

Data used for testing this implementation corresponds to the RSI value for the chilean stock Invercap (short trading name) publicly available at Bolsa de Santiago de Chile[1]. The time window begins at 04/16/2013 and ends at 05/31/2013.

[1]http://www.bolsadesantiago.com

```
if (dataArray.at(i).at(6).asFloat <= 30.0, {
      a = Synth ("sineWave");
      a.set ("freq",
                  dataArray.at(i).at(6).asFloat
                  * 3.3 + 200);
      h = dataArray.at(i).at(6).asFloat
                  * 3.3 + 200;
      h.postln;
      1.wait;
      }
);

if (dataArray.at(i).at(6).asFloat >= 70.0, {
      a = Synth ("sineWave");
      a.set ("freq",
                  dataArray.at(i).at(6).asFloat
                  * 3.3 + 169);
      h = dataArray.at(i).at(6).asFloat
                  * 3.3 + 169;
      h.postln;
      1.wait;
      }
);
```

Fig. 3. Mapping between RSI value and frequency value

```
~oversold = Buffer.read(s, "oversold.wav");
~overbought = Buffer.read(s, "overbought.wav");
```

Fig. 4. The two earcons are loaded in environment variables

Invercap was selected as it was one of the few stocks that has interesting movements of the RSI value in a short period (i.e., there was several line crossings and RSI values falling in almost the complete range). Table II shows the data points used in the implementation. In the table, column one corresponds to the value of the RSI and column two to the frequency mapped to the RSI value. Column three indicates if an earcon appeared or not.

A. *Intraday Trading and the Sonification*

The Relative Strength Index was defined initially for operating over days. That is, the RSI is calculated using closing prices data from a particular stock in a daily basis. The sonification proposed in this paper is intended to follow this initial definition.

Today, several other approaches to the use of the RSI exist. One of these approaches is the use of the RSI for intraday trading purposes. Although the formula of the RSI does not suffer changes and the value will always fall between 0 and 100, the use of this sonification can also be approached differently. In the case of intraday trading, the use of this sonification can be approached as a real-time monitoring in which the earcons and the sounds generated with the frequency mapped to the RSI value are used to direct the attention of the investor to the trading system.

```
if (((dataArray.at(l).at(6).asFloat >
      30.0).and(dataArray.at(i).at(6).asFloat < 30)),
      {
      playBuffer = {PlayBuf.ar([1, 1], ~oversold,
                              doneAction: 2)};
      playBuffer.play;
      2.wait;
      }
);


if (((dataArray.at(l).at(6).asFloat <
      70.0).and(dataArray.at(i).at(6).asFloat > 70)),
      {
      playBuffer = {PlayBuf.ar([1, 1], ~overbought,
                              doneAction: 2)};
      playBuffer.play;
      2.wait;
      }
);
```

Fig. 5. Play Buffers for oversold line crossing ($RSI < 30$) and overbought line crossing ($RSI > 70$)

V. Conclusions

Technical indicators are widely used to summarize data of financial markets and to provide some guidance to the actions of the investors as a support for the methodology of technical analysis of financial markets. Using such indicators is not an easy task. Even experienced investors often have difficulties in interpreting the data and using these indicators.

Sonification, a kind of auditory displays, can be used to convey information in acoustic ways. In this paper, a hybrid sonification for the relative strength index (RSI) is presented. The sonification is hybrid because it is composed by (1) two earcons that indicate when oversold and overbought areas are reached and by (2) a parameter mapping between the RSI value and the frequency of a sound when the RSI is in an oversold or in an overbought area.

A proof of concept is presented by means of an implementation in a real-time sound synthesis programming language. This language is SuperCollider, a widely known domain-specific language in the sound synthesis context.

Technical indicators are also used with closing prices to determine possible divergences. Buy or sell signals appear when a set of particular rules for the divergences are met. As a future work, we are interested in exploring and expanding the sonification proposed in this paper to create acoustic displays for these divergences in order to indicate acoustically when a buy or sell signal is generated. Also, as this paper presented a proof of concept of a particular sonification design, we are interested in conducting experimental studies in order to gain insight regarding the use of the sonification, the implications of the design and the *fine-tuning* of the sounds (pitch and timbre) used.

TABLE II. Data points from Invercap (04/16/2013 to 05/31/2013)

RSI value	Frequency [Hz]	Earcon?
25.25	283.3	No
26.81	288.5	No
36.75	N/A	No
35.26	N/A	No
35.29	N/A	No
35.13	N/A	No
32.09	N/A	No
32.09	N/A	No
35.49	N/A	No
46.95	N/A	No
53.61	N/A	No
57.44	N/A	No
62.45	N/A	No
71.94	406.4	Yes (overbought)
72.87	409.5	No
55.25	N/A	No
48.27	N/A	No
48.27	N/A	No
41.78	N/A	No
29.8	298.3	Yes (oversold)
26.97	289	No
26.97	289	No
24.34	280.3	No
30.18	N/A	No
29.55	297.5	Yes (oversold)
26.62	287.8	No
26.56	287.6	No
24.87	282.1	No
24.87	282.1	No

References

[1] J. Murphy, *Technical Analysis of the Financial Markets*, New York Institute of Finance, 1999.

[2] J. Wilder, *New Concepts in Technical Trading Systems*, Trend Research, 1978.

[3] S. Wilson, D. Cottle, N. Collins, *The SuperCollider Book*, The MIT Press, 2011.

[4] B. N. Walker, M. A. Nees, M. A., *Theory of Sonification*, The Sonification Handbook (pp. 9-39), Logos Publishing House, 2012.

[5] G. Kramer, B. N. Walker, B. N., T. Bonebright, P. Cook, J. Flowers, N. Miner, et al., *The Sonification Report: Status of the Field and Research Agenda*, Report prepared for the National Science Foundation by members of the International Community for Auditory Display (ICAD), 1999.

[6] T. Hermann, *Taxonomy and Definitions for Sonification and Auditory Display*, Proceedings of the 14th International Conference on Auditory Display, 2008.

[7] F. Dombois, G. Eckel, *Audification*, The Sonification Handbook (pp. 301-324), Logos Publishing House, 2012.

[8] E. Brazil, M. Fernstrm, *Auditory Icons*, The Sonification Handbook (pp. 325-338), Logos Publishing House, 2012.

[9] M. Blattner, D. Sumikawa, R. Greenberg, *Earcons and Icons: Their Structure and Common Design Principles*, Human Computer Interaction 4 (1) (pp. 1144), 1989.

[10] D. McGookin, S. Brewster, *Earcons*, The Sonification Handbook (pp. 339-361), Logos Publishing House, 2012.

[11] F .Grond, J .Berger, *Parameter Mapping Sonification*, The Sonification Handbook (pp. 363-397), Logos Publishing House, 2012.

[12] T. Hermann, H. Ritter, *Listen to Your Data: Model-based Soni?cation for Data Analysis*, Advances in Intelligent Computing and Multimedia Systems (pp. 189194), International Institute for Advanced Studies in System Research and Cybernetics, 1999.

[13] T. Hermann, *Model-Based Sonification*, The Sonification Handbook (pp. 399-428), Logos Publishing House, 2012.

[14] T. Hermann, H. Ritter, *Model-Based Sonification Revisited - Authors' Comments on Hermann and Ritter, ICAD 2002*, ACM Transactions on Applied Perception (TAP) 2(4) (pp. 559-563), 2005.

[15] B. N. Walker, G. Kramer, *Ecological Psychoacoustics and Auditory Displays: Hearing, Grouping, and Meaning Making*, Ecological psychoacoustics (pp. 150175), Academic Press, 2004.

[16] C. Hayward, *Listening to the Earth Sing*, In Auditory display: Sonification, audification, and auditory interfaces (pp. 369-404), Addison-Wesley, 1994.

[17] A. de Campo, M. Egger de Campo, *Sonification of Social Data*, In Proceedings of the 1999 International Computer Music Conference, Beijing, 1999.

[18] M. Gilfix, A.L. Couch, *Peep (The Network Auralizer): Monitoring Your Network with Sound*, Proceedings of the 14th USENIX conference on System administration (pp. 109-118), 2000.

[19] G. Baier, T. Hermann, U. Stephani, *Event-based Sonification of EEG Rhythms In Real Time*, Clinical Neurophysiology 118(6) (pp. 1377-1386), 2007.

[20] K. Vogt, W. Plessas, A. de Campo, C. Frauenberger, G. Eckel, *Sonification of Spin Models: Listening to Phase Transitions in the Ising and Potts Model*, In Proceedings of the 2007 International Conference on Auditory Display (pp. 258-265), 2007.

[21] D. Worral, *Using Sound to Identify Correlations in Market Data*, Proceedings of the 6th international conference on Auditory Display (CMMR/ICAD'09), Springer-Verlag, 2010.

[22] P. Vickers, *Sonification for Process Monitoring*, The Sonification Handbook (pp. 455-492), Logos Publishing House, 2012.

[23] M. Russ, *Sound Synthesis and Sampling*, 3rd ed., Focal Press, 2008.

[24] S. Bilbao, *Numerical Sound Synthesis: Finite Difference Schemes and Simulation in Musical Acoustics*, 1st ed., John Wiley and Sons, 2009.

[25] C. Roads, *The Computer Music Tutorial*, The MIT Press, 1996.