International Journal of Science and Engineering Applications
Volume 7–Issue 04, 39-44, 2018, ISSN:-2319–7560

Exploration Geochemistry Data-Application for Cu Anomaly Separation Based On Classical and Modern Statistical Methods in South Khorasan, Iran

Aref Shirazi
Amirkabir University of Technology (Tehran Polytechnic)
Mining and Metallurgical Engineering Department
Tehran, IRAN

Ardeshir Hezarkhani
Amirkabir University of Technology (Tehran Polytechnic)
Mining and Metallurgical Engineering Department
Tehran, IRAN

Adel Shirazy
Shahrood University of Technology
Faculty of Mining, Petroleum and Geophysics
Shahrood, IRAN

Abstract: The polymetal mining area is located 30 kilometers northwest of Birjand, South Khorasan Province of Iran. Considering the importance of recognizing the geochemical limit value for post-analysis studies, the limit value (= non-normative visualization) in the data of the stream was identified and described using the classic and modern statistical method. Sampling method in this area was lithogeochemical samples. Simple statistical methods, K-Means, K-Medoids, Fuzzy C-Mean (FCM), Self-Organized Map (SOM), have been used in this study. Anomaly maps are depicted in each method and separated from the background. Each method showed different anomalies, but the K-Mean and K-Medoids methods had similar responses.

Keywords: FCM; SOM; K-Means; Classical statistics; Anomaly separation; Exploration geochemistry; Copper

1. INTRODUCTION

Separation of anomalies from the background is one of the most important and key steps in geochemical exploration. There are several ways to identify and separate anomalous areas from the field. These methods include the classical statistics, K-Means, K-Medoids, Fuzzy C-Mean (FCM), Self-Organized Map (SOM) used in this study. In general, anomalous areas are also useful in surface surveys to determine the location of exploratory drilling. Also, by studying more precisely, the resulting maps can be used to structure the sampling grid for later, more coherent steps as well. [1,2]

2. Geolocation of the area

This area is located in the geographical location of 32° 58' 23" northern latitudes and 58° 56' 7" eastern longitudes in the south Khorasan province of, Iran. It is called Siojan. Siojan mining area is located 30 kilometers from Birjand city (see Figure 1 and 2).

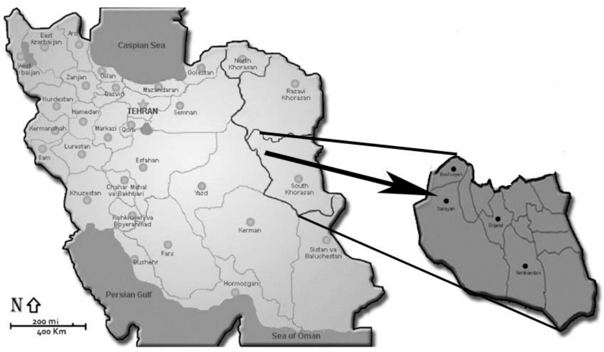


Figure. 1 South Khorasan Province Location and Siojan, Iran Map [3]

Figure. 2 Geolocation and access way of studied area.

2.1 Geological setting

This region is located in the northwest of the Lut zone in terms of the division of the structural states of the Iranian crust in the east of the Central Block of Iran. The Nehbandan fault system, which covers all the Sistanian states, enters into the Lut zone in the northern part as a direction to the west. [4] The area generally consists of volcanic and intermediate cenozoic rocks in the form of andesitic masses, basaltic andesite, dacite and intrusive rhyodacite. In some cases, this formation has caused alteration and mineralization in the region. [5] Pyroclastic rocks such as altered acid tuffs, and cuttings with andesitic elements are also present in the region. Non-volcanic units such as conglomerate, sandstone, salt salts and young alluvium are found in the area. [6,7]

You can see the location of Siojan area in geological map in figure 3.

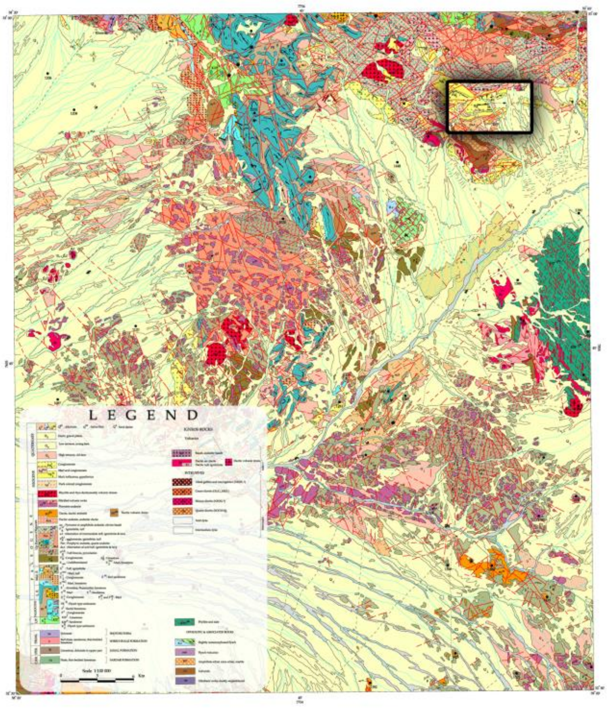


Figure. 3 Geological map of Khusf on scale 1:100000, The Siojan area is shown in the map. [8]

3. Sampling

The area was covered with 120 samples of lithogeochemicals. The samples were analyzed as 44 elements. The sample preparation method was Aquaragia. And samples were analyzed by ICP-MS method. You can see the location of the samples in the UTM system in Figure 4.

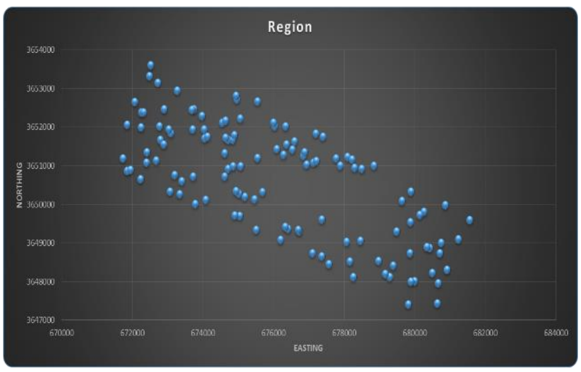


Figure. 4 locations of lithogeochemical samples in the area.

4. Geochemical Anomaly Separation Methods

Separation of geochemical anomalies from background has always been a major concern of exploration geochemistry. The search for methods that can make this analysis quantitative and objective aims not only at the reduction of subjectiveness but also at providing an automatic routine in exploration, assisting the interpretation and production of geochemical maps. [9,10]

4.1 Statistical Methods

In classical methods, anomalies are usually detected, regardless of the location of each instance, and only by formulating relationships. Commonly updated methods can be found in Formula 1. [1,9]

$$Anomaly = (\overline{X})Or(median) + 3(S)Or(MAD) \quad (1)$$

Here, x is the mean, median is the same. And S is the standard deviation and the MAD is the mean / median with the absolute value of the deviation difference [11,12], which has the following two formulas.

$$MAD = Mean|\,Xi - Mean\,|$$
$$MAD = Median|\,Xi - Median\,|$$

By placing the above values in Formula 1, we can reach different ranges.

4.2 K-Means and K-Medoids

The K-Means method, despite simplicity, is a basic method for many other clustering methods (such as fuzzy clustering). This method is a monolithic and flat method. For this algorithm, different shapes are expressed. But they all have repetitive routines that try to estimate the following for a fixed number of clusters [13]:

- Getting points as centers of clusters, which are actually the same average points belonging to each cluster.
- Assigning each data sample to a cluster that gives the data a minimum distance to the center of that cluster.

In this method, first, the number of clusters needed for points is randomly selected. Then the datas are attributed to one of these clusters according to the degree of similarity. And so new clusters are achieved. By repeating the same procedure, it is possible to calculate new centers for each replication by averaging the data, and the data are re-assigned to new clusters. This process continues as long as there is no change in the data. If this method is such that each cluster is displayed with one of the objects located near the center, then the " K-Medoids " is called. [14,15]

Calculation flowchart of the method algorithm is shown in figure 5.

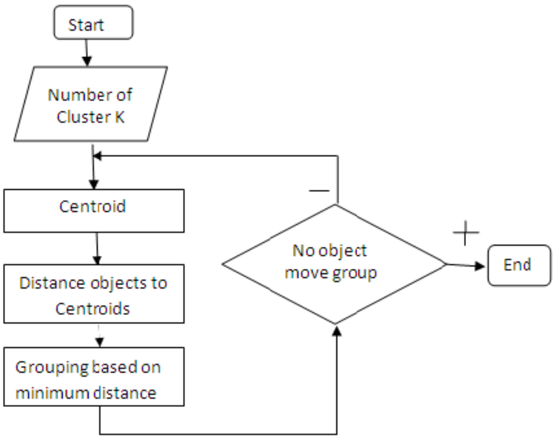


Figure. 5 Calculation flowchart of method algorithm. [16]

4.3 Fuzzy C-Means (FCM)

FCM is a separation clustering method that uses the Euclidean distance similarity criterion to measure the similarity of data and clusters. In other words, this algorithm identifies spherical clouds of points in a p-dimensional space. These clusters are approximately equal in size. Each cluster is displayed with its center. This mode of displaying clusters is also called a model or example, because it is often referred to as a representative of all the data assigned to the cluster. [1,9]

Calculation flowchart of fuzzy C-Means clustering algorithm is shown in figure 6.

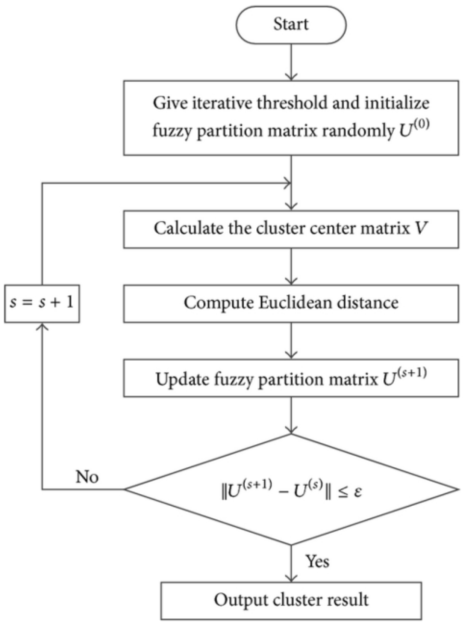


Figure. 6 Calculation flowchart of the Fuzzy C-Means clustering algorithm. [17]

As an meter for distance, Euclidean distance is used between a point and a sample. In order to select the center of the cluster, as the name of the algorithm finds, the mean value is used. To compute the center of the cluster, the sum of the degrees of the membership of each element is divided by the power of M in itself into the product of the power of the M degree of membership. [18] The problem with this algorithm is that the algorithm cannot identify clusters of different shapes, sizes and densities. To identify other shapes instead of the identity matrix, we can use other matrixes to determine the distance. Such as a diameter matrix to detect elliptical clusters. The benefits of this algorithm are ease, which reduces computational time. In practice, with little repetition, it can reach a near-final solution. [19]

4.4 Self-Organized Map (SOM)

In the self-organization map, the competitive learning method is used for training. This method is based on specific characteristics of the human brain. The cells in the human brain are organized in different regions in different sensory regions, with rigorous and meaningful computational maps. For example, sensory inputs for touch, hearing, etc. are associated with a significant geometric arrangement in different regions. [20]

In this method that is called SOM or sometimes called SOFM (Self-Organizing Feature Map) Processor units are placed in the nodes of a one-dimensional grid, two-dimensional or more. Units are organized in a competitive learning process rather than input patterns. The place of the units set in the network is organized in such a way as to create a meaningful coordinate system on the network for input characteristics. Therefore, a self-organized map forms a topographic map of the input patterns in which the location of the units corresponds to the inherent characteristics of the input patterns. [21,22]

The competitive learning used in this grid is that in each step of the learning, the units compete to engage with each other, At the end of a competition stage, only one unit wins, which weighs in a different way than the weights of other units. This type of learning is called "uncontrolled learning." [23]

Calculation flowchart of the self-organization map algorithm is shown in figure 7.

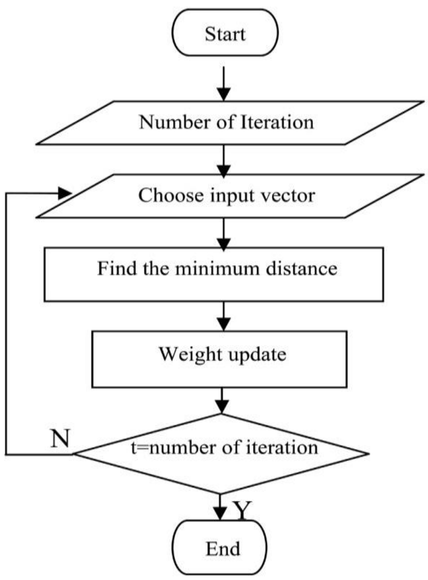


Figure. 7 Calculation flowchart of the Self-Organization Map (SOM) algorithm. [24]

5. General Map of Copper Concentration

In order to improve the understanding of the anomal areas, the main map of the dispersion of copper concentration in the region can be estimated as a preview. After analyzing the sample dispersion and copper concentration, a map was prepared with the estimation of other points by Kriging method [25,26].

In this map, we can see concentrations with respect to the color as well as the contours. (see Figure 5).

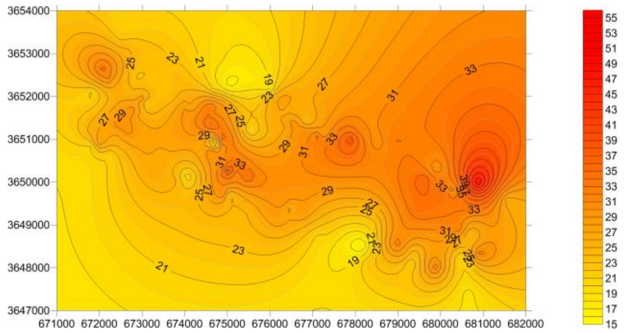


Figure. 5 Concentration map of copper element along the range with the conjugated estimator.

6. Maps of Anomaly Seperation Methods

In this section, the anomalies of the copper element in the study area, which have been obtained by different methods of anomaly seperation from the background, are mapped out.

6.1 Statistical Methods

As discussed in Section 4.1, the threshold can be obtained from three relative-like formulas. The copper anomaly map depicted using pure classical statistics (use of mean and standard deviation) is shown in Figure 6.

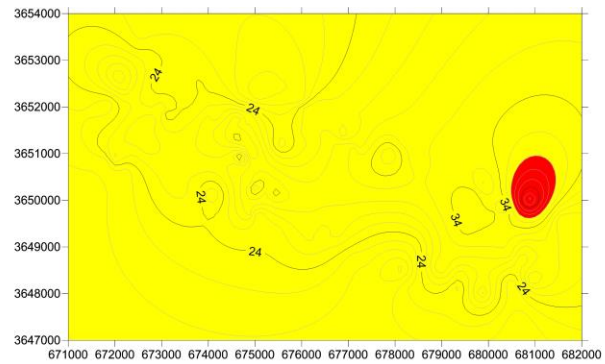


Figure. 6 Pure statistics (using mean and standard deviation) - The threshold value is 34.3 ppm.

The copper anomaly map depicted by using the updated statistical method of the first type (using MAD and mean) is shown in figure 7.

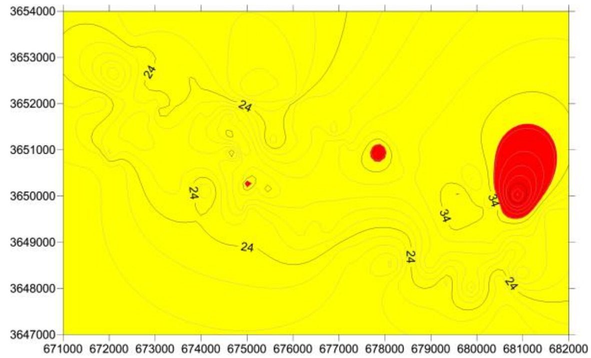


Figure. 7 The updated type I statistical method (using MAD and Mean) - The threshold value is 35.7 ppm.

In the following, the anomaly of the copper element derived from the statistical updated method of the second type (using MAD and Median) is shown in figure 8.

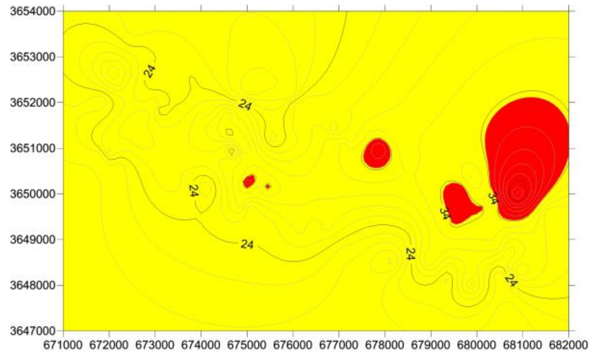


Figure. 8 The updated type I statistical method (using MAD and Mean) - The threshold value is 38.1 ppm.

As shown in the maps, The Copper threshold values were obtained for pure statistical methods, the updated type of statistical method, and the second type updated method were 34.3 ppm, 35.7 ppm and 38.1 ppm, respectively.

6.2 K-Means and K-Medoids

Due to the close approach of the two methods, the threshold value in both methods was estimated to be similar to each other. Figure 9 shows a map that confirms both methods. (Red color is anomaly and yellow color is background).

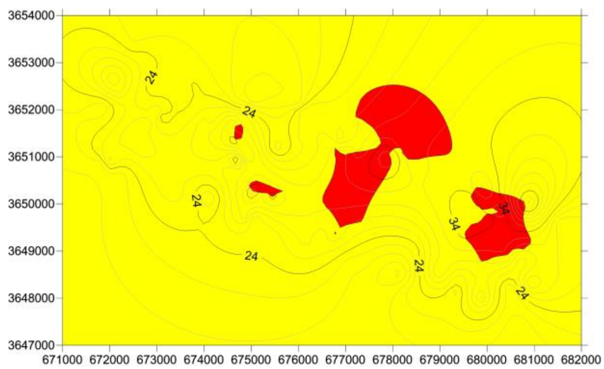


Figure. 9 Estimation of anomalies for the KaMeS and KaMoides method.

6.3 Fuzzy C-Means (FCM)

In the FCM method, due to the percentage probability of each sample being attributed to the anomaly, 80% was considered probable, since 80% could be a good possibility for diagnosis [27] and could reduce the wasting caution of much of the backgrounds. Figure 10 shows the map of the anomalous areas derived from the FCM method. (Red color is anomaly and yellow color is background).

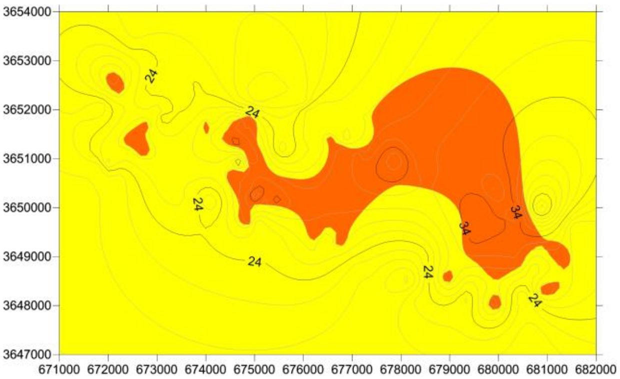


Figure. 10 Estimated anomalies by FCM method with consideration of 80% anomaly probability

6.4 Self-Organization Network Method

According to this method, the following map (Figure 11) emerges which represents two groups of anomalies (red) and a background (yellow).

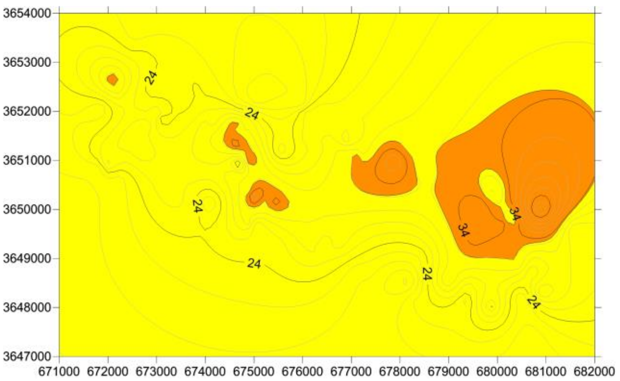


Figure. 11 Anomaly Estimation for SOM Method

7. Conclusion

The Siojan Polymetal Mineral Area is located in South Khorasan province, 30 km northwest of Birjand city. Considering the importance of recognizing the geochemical limit value for post-analysis studies, the limit value in the data of the lithogeochemical samples were determined using 5 statistical methods.

Methods such as classic statistics, K-Mean, K-Medoids, Fuzzy C-Mean clustering (FCM), Self-Organized Map (SOM) were used in this study. Anomalies were mapped in each method and the anomalies were seperated from the field. Each method showed different anomalies, but the K-Mean and K-Medoids methods had similar responses.

By matching all the maps, it can be seen that each method can be efficient and should not rely on a method.

It is recommended to use field studies and re-sampling by estimating the anomaly maps, the accuracy of each method for the region is estimated and the best method for further studies is to be selected.

4. REFERENCES

[1] Hasani Pak, A.A., Sharafodin, M. (2012). Exploration data analysis. University of Tehran Publication. Tehran.

[2] Govett, G. J. S., Goodfellow, W. D., Chapman, R. P., & Chork, C. Y. (1975). Exploration geochemistry-distribution of elements and recognition of anomalies. Journal of the International Association for Mathematical Geology, 7 (5-6), 415 - 446.

https://doi.org/10.1007/ BF02080498

[3] Emami, M. H. (1972). Geology and petrological investigation on Shahkuh volcanic rocks, south of Birjand, eastern Iran. University of Tehran, Tehran, Iran.

[4] Walker, R. T., & Khatib, M. M. (2006). Active faulting in the Birjand region of NE Iran. Tectonics, 25(4).

[5] Hoseinpoor, M. K., & Aryafar, A. (2016). Using robust staged R-mode factor analysis and logistic function to identify probable Cu-mineralization zones in Khusf 1: 100,000 sheets, east of Iran. Arabian Journal of Geosciences, 9(2), 157.

https://doi.org/10.1007/s12517-015-2266-9

[6] Aghanabati, A. (2004). Geology of Iran. Geological survey of Iran.

[7] Ghorbani, M. (2002). An introduction to economic geology of Iran. Tehran, Iran. Geological survey and mineral explorations of Iran (in Persian).

[8] Vahdati Daneshmand, F. (1989). 1: 100,000 Geology Map of Khusf, Geological survey of Iran.

[9] de Mulder, E. F., Cheng, Q., Agterberg, F., & Goncalves, M. (2016). New and game-changing developments in geochemical exploration. Episodes, 39(1), 70-71. http://dx.doi.org/10.22059/ijmge.2014.53107

[10] Journel, A. G., & Huijbregts, C. J. (1978). Mining geostatistics. Academic press.

[11] Howell, D. C. (2014). Median absolute deviation. Wiley StatsRef: Statistics Reference Online.

[12] Twain, M., & Weather, N. E. (2004). Mean Absolute Deviation. Dynamic Portfolio Theory and Management, 235. https://doi.org/10.1287/mnsc.37.5.519

[13] Likas, A., Vlassis, N., & Verbeek, J. J. (2003). The global k-means clustering algorithm. Pattern recognition, 36(2), 451-461.

https://doi.org/10.1016/S0031-3203(02)00060-2

[14] Hartigan, J. A., & Wong, M. A. (1979). Algorithm AS 136: A k-means clustering algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics), 28(1), 100-108. http://dx.doi.org/10.2307/2346830

[15] Park, H. S., & Jun, C. H. (2009). A simple and fast algorithm for K-medoids clustering. Expert systems with applications, 36(2), 3336-3341. https://doi.org/10.1016/j.eswa.2008.01.039

[16] Sohaib, M., & Mushtaq, Q. (2013). Dimensional Reduction of Hyperspectral Image DataUsing Band Clustering and Selection Based on Statistical Characteristics of Band Images. International Journal of Computer and Communication Engineering, 2(2), 101. http://dx.doi.org/10.7763/IJCCE.2013.V2.148

[17] Wang, Z. T., Zhao, N. B., Wang, W. Y., Tang, R., & Li, S. Y. (2015). A fault diagnosis approach for gas turbine exhaust gas temperature based on fuzzy c-means clustering and support vector machine. Mathematical Problems in Engineering, 2015. http://dx.doi.org/10.1155/2015/240267

[18] Gary, A. C., Wakefield, M. I., Johnson, G. W., & Ekart, D. D. (2009). Application of fuzzy c-means clustering to paleoenvironmental analysis: example from the Jurassic, central North Sea, UK. Geologic Problem Solving with Microfossils: A Volume in Honor of Garry D. Jones. SEPM Special Publication, 93, 9-20. http://dx.doi.org/10.2110/sepmsp.093.009

[19] Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The fuzzy c-means clustering algorithm. Computers & Geosciences, 10(2-3), 191-203. https://doi.org/10.1016/0098-3004(84)90020-7

[20] Carpenter, G. A., & Grossberg, S. (Eds.). (1991). Pattern recognition by self-organizing neural networks. MIT Press.

[21] Kohonen, T. (1998). The self-organizing map. Neurocomputing, 21(1-3), 1-6. http://dx.doi.org/10.1016/S0925-2312(98)00030-7

[22] Kohonen, T. (1990). The self-organizing map. Proceedings of the IEEE, 78(9), 1464-1480. https://doi.org/10.1109/5.58325

[23] Kohonen, T., Hynninen, J., Kangas, J., & Laaksonen, J. (1996). Som pak: The self-organizing map program package. Report A31, Helsinki University of Technology, Laboratory of Computer and Information Science.

[24] Anifah, L., Purnama, I. K. E., Hariadi, M., & Purnomo, M. H. (2013). Osteoarthritis classification using self-organizing map based on gabor kernel and contrast-limited adaptive histogram equalization. The open biomedical engineering journal, 7, 18. http://dx.doi.org/10.2174/1874120701307010018

[25] Oliver, M. A., & Webster, R. (1990). Kriging: a method of interpolation for geographical information systems. International Journal of Geographical Information System, 4(3), 313-332. http://dx.doi.org/10.1080/02693799008941549

[26] Adhikary, P. P., Dash, C. J., Bej, R., & Chandrasekharan, H. (2011). Indicator and probability kriging methods for delineating Cu, Fe, and Mn contamination in groundwater of Najafgarh Block, Delhi, India. Environmental monitoring and assessment, 176(1-4), 663-676.

https://doi.org/10.1007/s10661-010-1611-4

[27] Cheng, Q. (1999). Spatial and scaling modelling for geochemical anomaly separation. Journal of Geochemical exploration, 65(3), 175-194. https://doi.org/10.1016/S0375-6742(99)00028-X

International Journal of Science and Engineering Applications
Volume 7–Issue 04, 45-51, 2018, ISSN:-2319–7560

Remote Sensing Studies for Mapping of Iron Oxide Regions, South of Kerman, IRAN

Aref Shirazi
Amirkabir University of Technology (Tehran Polytechnic)
Mining and Metallurgical Engineering Department
Tehran, IRAN

Ardeshir Hezarkhani
Amirkabir University of Technology (Tehran Polytechnic)
Mining and Metallurgical Engineering Department
Tehran, IRAN

Adel Shirazy
Shahrood University of Technology
Faculty of Mining, Petroleum and Geophysics
Shahrood, IRAN

Abstract: Due to the economic and industrial importance of iron in the development of human societies, in recent decades, extensive explorations have been carried out on minerals containing this valuable metal. Remote sensing techniques are known as one of the most powerful tools for regional exploration of this mineral. In this study, various methods of remote sensing such as band ratios (BR), false color combinations (FCC), least square fitting (LS-Fit), spectral angle mapper (SAM), and finally principal components analysis (PCA) for mapping iron minerals in the Hana district, south of Kerman province, were used. The results of these methods were compared with each other as well as with the results of studies and field surveys. After reviewing and comparing the results, it was determined that in the studied region, spectral angle mapper (SAM) method has higher accuracy for mapping of oxidation regions and iron minerals.

Keywords: Remote Sensing; Image processing methods; Iron Altration; SAM, LS-Fit, FCC, PCA, BR

1. INTRODUCTION

Iron as a strategic element plays an important role in the development of industry and economics of countries [1]. The need to extract this element from the minerals in nature reveals the importance of the regional exploration of these minerals. On the other hand, the use of various remote sensing techniques will increase the speed of operation and significantly reduce the cost of finance [3, 4]. In this study, using remote sensing methods including band ratios (BR), false color combinations (FCC), least square fitting (LS-Fit), spectral angle mapper (SAM), and principal components analysis (PCA) and also considering the geological features of the study area, the detection of oxidation regions and iron minerals in the area of Hana, located in the south of Kerman province, is discussed.

2. DATA AND RESEARCH METHODS

2.1 Regional Geological Setting

The study area is located in the south of Kerman province and Kahnouj city, in the geological map of 1: 100,000 Hana (Figure 1) [2]. The area consists of four geological sequences which are described below :

- **Volcanic – Pyroclastic - Sedimentary sequence :**

  This sequence consists of the oldest rocks of the region with the Eocene age, which is characterized in the Fark River region with the units of green tuff and sandstone and limestone, including pyroclastic sediments, microdiorite dikes, tuff, andesite, conglomerate, sandstone and carbonate rocks.

- **Sedimentary - Sedimentary – Volcanic sequence :**

  Includes conglomerates, sandstone, marl , dacitic massive tuffs and fossil limestone and bioclasts.

- **Volcanic series :**

  This series introduces volcanics belonging to the after Oligomiocene and contains acidic up to the intermediate volcanic rocks, granite and granodiorite dikes, Porphyry and diabase rocks that dikes have penetrated into oligo-miocene rocks.

- **Sedimentary series :**

  The Neogene layers in the lower parts include red gypsum sandstones representing semi-arid conditions of sedimentary environments, and the upper layers are conglomerates.

Quaternary sediments in the form of sandy dunes and alluvial plains cover most of the southern and western parts of the area and are located on Neogene sediments [2].

The northern part forms the Jabal Barez river basin. This area includes Eocene-Quaternary sediments along with pyroclastic and granite sediments. The southern part of the area consists of a deep river basin, which is mainly covered with a thick layer of gravel, which forms a whole bulky desert.

The oldest observable rocks belong to the Eocene. In the "Fark" river, there are green tuff, sandstone and carbonate layers, and in the "Freezu" mountain range, agglomerate, rhyolite, rhyolitic tuff, conglomerate and dacitic tuff, the middle Eocene have been created.

In the north-eastern part of the region, there are acidic up to the intermediate volcanics, the broadest of which are green hornblend granites. The lower parts of it are often covered with tuff and conglomerate, which is the same process. In parts of the region there are rhyodactic masses and under them there are intrusive masses of quartz diorite to diorite, which are in some places outcrop.

In the northeastern and eastern part of the region, we also have a density of faults and fractures that there are lack of the trend and are often intersecting fractures [2].

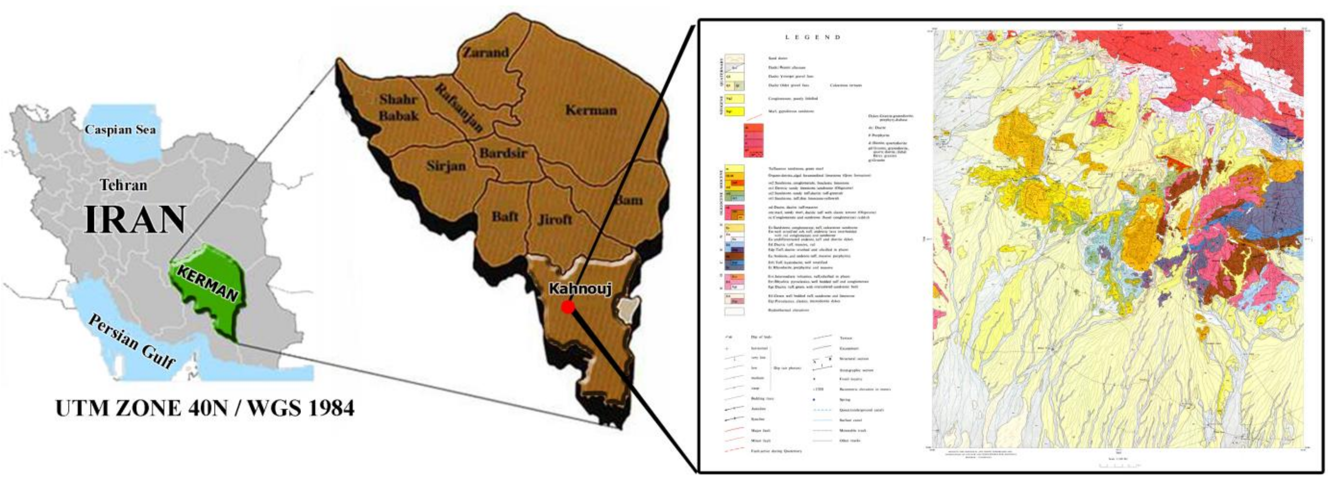


Figure 1. Location of the study area and its geological map [2].

2.2 Satellite Remote Sensing Data

In this study, the image of the ASTER Satellite Sensor was used. ASTER *(*Advanced Spaceborne Thermal Emission and Reflection Radiometer) is a high resolution imaging instrument that is flying on the Terra satellite [5]. *ASTER* will obtain detailed maps of land surface temperature, emissivity, reflectance and elevation of the Earth.

ASTER has three sensors to measure and record the reflected and emitted Electromagnetic Radiation (EMR). They are working in different wavelength regions the Visible and Near Infrared (VNIR) between 0.52 and 0.86 μm, Short Wave Infrared (SWIR) between 1.6 and 2.43 μm, and Thermal Infrared (TIR) between 8.125 and 11.65 μm. ASTER data consists of 14 spectral bands 3 VNIR, 6 SWIR, and 5 TIR with 15, 30, and 90 m spatial resolution, espectively [6]. The VNIR, SWIR and TIR wavelength regions provide complementary data for lithological mapping [7].
Geometric corrections were made using the satellite ETM^+ satellite image on the study area image. In order to ensure the results, IAR Reflectance radiometric corrections were applied to the region image and the results for applying different types of processes were introduced into ENVI software[8].

2.3 Band Ratios (BR) Method

In general, all materials are composed of atoms and molecules with a specific composition [9]. Therefore, various materials, depending on the structure, absorb or emit electromagnetic radiation at special wavelengths [10]. So that the wavelength curve and radiant energy for each object are unique and this is a clear feature of remote sensing science [11]. The result of dividing the values of the brightness of the pixels in a spectral band into another band is called the band ratio. And as a result, new lighting levels or, in other words, a new image are created. Band ratios method is used to detect complications that are not visible in the image of single bands [12].

This method is applicable to the recognition of the spectral reflection of various phenomena for the appearance of a particular phenomenon. Relative images that are based on the reflection characteristics of altered minerals and by dividing the digital values of a spectral band into another band are important in identifying altered areas [13].

In order to determine the alteration areas with respect to the spectral characteristics of the index minerals in any kind of alteration, the bands proportions can be defined. Many band ratios have been identified for the identification of various types of minerals in the case of ASTER data [14].

The results of applying band ratios method are gray-scale images that alone are not a valid criterion for determining the target areas in the study area. It only identifies the areas most likely to have the desired minerals or, in general, the objects to be searched for. Using false color combinations (RGB images) can be produced that make the interpretation and conclusions based on them more reliable and more practical [15].

2.3 False Color Combinations[1] Method

The importance of displaying the color combination of images in remote sensing can be considered due to their effectiveness in visual interpretation of various effects. One of the effective methods for identifying and separating various geological units is the false color combination (FCC) method [16].

The false color combination is a combination of three different bands combined in red, green, and blue (RGB) colors. If the combined bands of red, green, and blue wavelengths are the visible spectrum of electromagnetic spectrum, the resulting image will be a true color combination. If a different combination of red, green and blue bands or other bands of the electromagnetic spectrum is used, a false color image will be obtained that is not similar to the surface of the earth and its colors [17].

[1] FCC

In making false color combinations it is better to use bands that have less correlation. Since the interaction of different wavelengths of electromagnetic energy is different in dealing with rock units, the sensitivity of the human eye to minor changes in color is much greater than its sensitivity to changes in black and white images. Choosing the best band combination depends on the target [18].

Three images can be combined to make the images visible for viewing in three blue, green, and red wavelengths (original RGB color combinations). In this study, this combination has been used to display several images in a single image and simultaneously display different information from a single point [19].

Calculation of the optimum index factor amount (OIF) is required to obtain the best false color combination (OIF of the higher color combinative with more information). The formula below shows the OIF calculation method [20].

$$OIF = \frac{\sum_{k=1}^{3} S_k}{\sum_{j=1}^{3} r_j} \qquad (1)$$

In formula 1 : S_k is the standard deviation of the k band, r_j is the two-band correlation coefficient of the three-band combination [21]. Sometimes visually, the false color combinations containing major information are determined by the variety of colors [22].

2.4 Principal Components Analysis (PCA)

One of the methods used to reduce the correlation between multivariate data and increase the distinction is the main component analysis (PCA) method. The purpose of this method is to compress data and eliminate redundant data in order to save time and money. By using the PCA method, we can replace many independent and correlated variables with a limited number of new variables, which are called principal components and are not interconnected [23]. In this way, the dimensions of the problem are reduced. In general, the purpose of this method is to compress all the information contained in a main dataset composed of n channels into less than n channels or new components. Finally, the components are used instead of the original data [24].

In general, this approach reduces the compatibility between different bands data, and new information is obtained and sent to PC channels. By creating a combination of PC channels and dual-source bands, images can be created to illustrate the effects. This technique is a eigenvectors based method, using eigenvalues and eigenvectors, identifies directions with maximum variability and then decreases the dimensions of variables by defining new variables that are linear combinations of the initial variables [25]. New variables that are the product of the linear combination of initial variables do not show correlation between themselves [26].

To compute the main components, at first the variance, covariance, or matrix of correlation between the bands are formed and then eigenvalues and eigenvectors of this matrix are calculated. Because covariance is dependent on the unit of measurement of data and the bands of different bands do not have the same reflexion unit, it is better to use the correlation matrix [27].

For each principal component, an image is calculated from its eigenvectors. The numerical values of the principal component image are calculated using the values of numerical values in the initial images and the components of the eigenvectors as follows:

$$\alpha = \mathrm{Cos}^{-1}\left[\frac{\sum_{i=1}^{nb} x_i r_i}{(\sum_{i=1}^{nb} x_i^2)^{\frac{1}{2}}(\sum_{i=1}^{nb} r_i^2)^{\frac{1}{2}}}\right] \qquad (2)$$

In formula 2 : P_k is the numerical value of the desired pixel for the k th principal component, DN (i) The numeric value of the i-th band for the desired pixel, a_{ik} is the amount of the load obtained from the eigenvectors of the k component in the i-th band. Thus, for each principal component or eigenvectors, an image is obtained that represents the variability in its direction [28].

2.5 Least Square Fitting (LS-Fit) Method

In the regression least squares method, a band is estimated based on the linear combination of other bands using the least squares of errors [29]. In this method, the band of the mineral in question is high in adsorption with the rest of the bands, will be divided and the best areas will be detected with pixels containing those minerals [30, 31].

2.6 Spectral Angle Mapper (SAM) Method

Spectral angle mapping (SAM) method is an image classification method by calculating the similarity between the image spectrum and a reference spectrum (e.g., spectral libraries) [29]. The algorithm of this method calculates the similarity between two spectra by the spectral angle between them [32]. In fact, by transforming the spectra into a vector in a space in the number of dimensions of the bands, the angle between the two vectors is calculated (See figure 2) [33] .

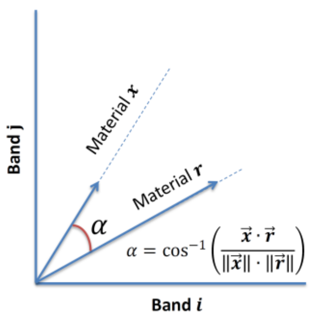


Figure. 1 Example of SAM classification in case of 2 spectral bands. Scalar product between unknown material x and library sample r [33].

In this method, the direction is important for calculating vectors, not length. Therefore, other factors are not considered in this method. In fact, the more the angle (between 0 and 1) is less, the more accurate it will be. If the value is 0, the whole image is identified as the desired phenomenon. To compare a pixel, the desired pixel spectrum is plotted from the examined area with the same pixel spectrum in the laboratory (library) on two bands in a coordinate axis. Then the points are connected to the coordinate center, and the angle between the two lines is used as the pixel identification angle. If the n bands are used to identify the phenomenon concerned, the following formula is used to obtain an angle [34].

$$\alpha = \mathrm{Cos}^{-1}\left[\frac{\sum_{i=1}^{nb} x_i r_i}{(\sum_{i=1}^{nb} x_i^{\,2})^{\frac{1}{2}} (\sum_{i=1}^{nb} r_i^{\,2})^{\frac{1}{2}}}\right] \quad (3)$$

In formula 3 : nb is the number of bands. unknown material x and library sample r.

3. RESULTS AND DISCUSSION

In this section, the results of each of the methods described in the previous section are presented.

3.1 Band Ratios (BR) Method

In the study area, a ratio of 2/1 to show Iron(II) oxides [35], 3/2 ratio to reveal vegetation coverings and a ratio of 5/7 to show hydroxylated minerals [36] as a false color combination RGB=(2/1 , 3/2, 5/7) was used [37]. The result is shown in Figure 3.

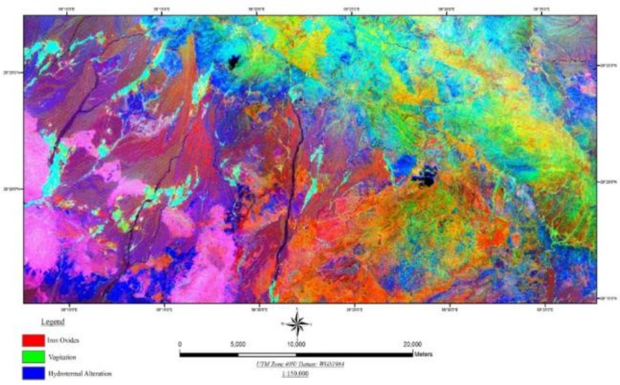


Figure. 3 false color combination image RGB=(2/1 , 3/2, 5/7)

In the resulting image, the pinky to red regions indicate the presence of iron oxides, the green to yellow zones indicate the presence of vegetation and eventually the blue zones indicate the presence of clay minerals.

3.2 False Color Combinations Method

In the study area, a false color combination (4,6,8) RGB was used [35]. and the result is shown in Figure 4.

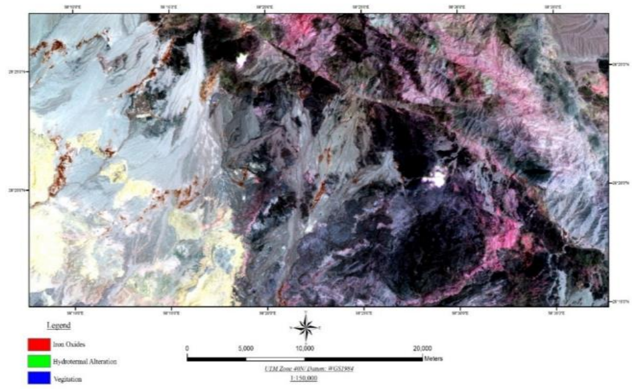


Figure. 3 false color combination image RGB=(4 , 6, 8)

3.3 Principal Components Analysis (PCA)

In the study area, the combination of bands 3,2,1 and 4 was used to show areas containing iron oxides. The statistical results and their PC coefficients are shown in Table 1.

Table 1. The statistical results and PC coefficients related to the ASTER bands composition.

Band 4	Band 3	Band 2	Band 1	
0.128783	0.456589	0.639967	0.60447	**PC1**
-0.335408	-0.794281	0.191027	0.469178	**PC2**
0.932605	-0.385483	0.007555	0.061825	**PC3**
-0.034116	-0.185138	0.744241	-0.640832	**PC4**

Looking at Table 1, it can be seen that the greatest difference between the absorption bands and the reflection of the iron oxide index appears in the third component.Therefore, the component can be used to show the probable areas containing iron oxides (Figure 4).

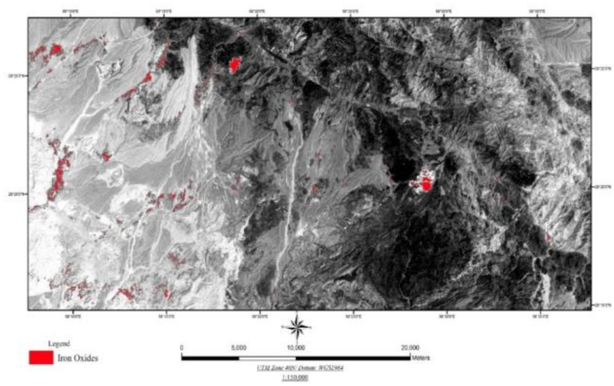


Figure. 4 The third component image (PC3) obtained after analyzing the principals components.

The red regions in the figure 4 shows regions containing iron oxides. Which are separated by PCA method.

3.4 Least Square Fitting (LS-Fit) Method

In this study, band 2, which has high adsorption index for iron oxide minerals, was selected as model band and other bands were selected as the predictor bands. The final image is shown in Figure 5. The blue regions that are separated on the ASTER image are regions containing iron oxide (see figure 5).

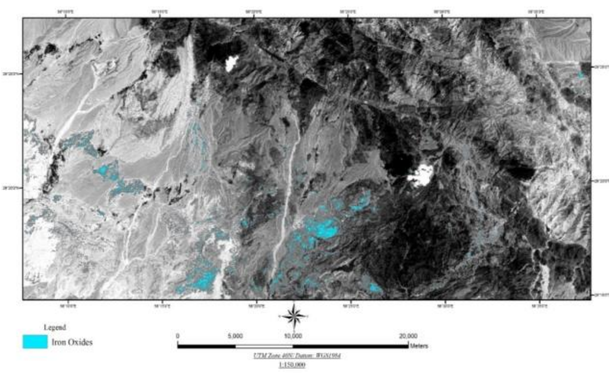


Figure. 5 The result of applying the Ls-Fit method.

3.5 Spectral Angle Mapper (SAM) Method

In the study area, using the spectral angle mapper base pixel method and using the spectral library, the hematite and limonite iron minerals were detected and shown in (Figures 6 and 7).

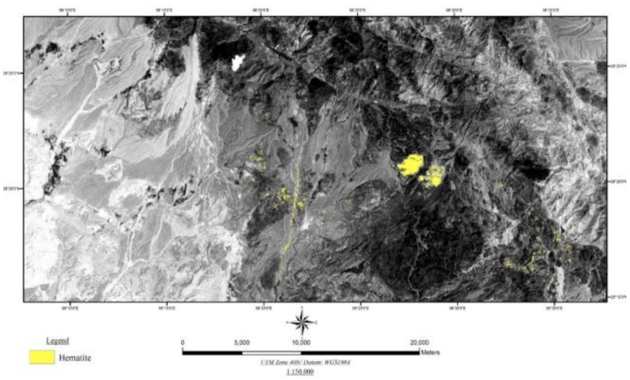


Figure 6. Hematite Separation Using Spectral Angle Mapper Method.

The yellow regions marked on the ASTER image (figure 6) are regions containing Hematite Cannabis, that are separated by Spectral Angle Mapper (SAM) Method.

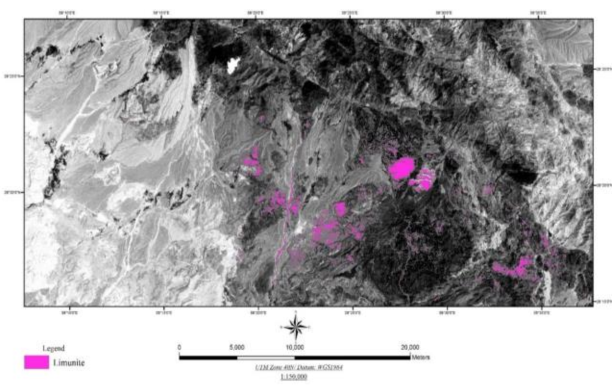


Figure 7. Limonite Separation Using Spectral Angle Mapper Method.

In figure 7, regions containing limonite, are indicated with purple color.

3.6 Field Studies and Control Point

After the remote sensing tests were carried out and the results were obtained. Regarding the determination of areas as iron oxides by various techniques. One point was determined as a control point and was referred to the position for checking the results. This point is a place designated by the Spectral Angle Mapper (SAM) Method as the region containing hematite and limonite minerals. While other methods used in this region, indicate the lack of iron oxide there. The location of the control point is shown in Figure 8.

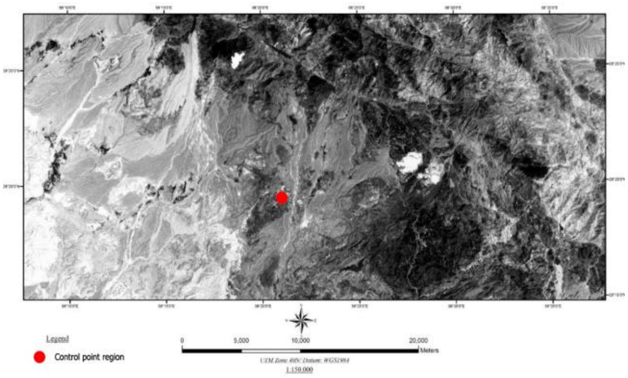


Figure 8. The location of the control point on the ASTER image.

After checking the control point, it was found that there is hematite and limonite mineralization in this region. The control point and mineralization of hematite and limonite is shown in Figure 9.

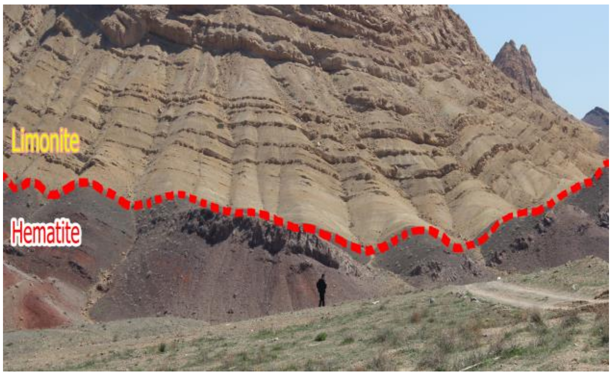


Figure 9. image of the control point in the field.
Hematite mineralization in lower layers and limonite mineralization in upper layers.

4. CONCLUSION

- The importance of exploration of iron ore is obvious because it is an economic and strategic element. In this regard, remote sensing has been used as one of the most important tools in the exploration of these minerals.
- In this study, using remote sensing methods including band ratios (BR), false color combinations (FCC), least square fitting (LS-Fit), spectral angle mapper (SAM), and principal components analysis (PCA) and also

considering the geological features of the study area, the detection of oxidation regions and iron minerals in the area of Hana, located in the south of Kerman province, was discussed.

- After examining the field evidence in the control point, it was determined that the Spectral Angle Mapper (SAM) Method can be cited and closer to the reality in this study area for iron remote sensing. In the results of this method, the regions were identified that contain mineralization of hematite and limonite. While these points were free from iron oxide in the images obtained from other methods.
- The overall results of this study, in addition to showing the high accuracy of the SAM method, show the importance of using remote sensing methods and techniques in exploration and prospecting of minerals, especially iron ore.

5. REFERENCES

[1] Singla, A., Ahuja, I., & Sethi, A. (2017). The effects of demand pull strategies on sustainable development in manufacturing industries. *International Journal of Innovations in Engineering and Technology, 8*(2), 27-34. http://dx.doi.org/10.21172/ijiet.82.005

[2] Afaghi, A., Sadjedi, T., Salek, M. M., & StamenKovic, D. (Cartographer). (1973). Geological Map of IRAN 1:100000 Sheets Series, Hana Sheet 7645

[3] Roonwal, G. S. (2018). *Mineral Exploration: Practical Application*: Springer.

[4] Shivakumar, V., Pokhrel, S., & Parekh, A. (2017). Block-1 Introduction to Remote Sensing: IGNOU.

[5] Arivazhagan, S., & Anbazhagan, S. (2017). ASTER Data Analyses for Lithological Discrimination of Sittampundi Anorthositic Complex, Southern India. https://dx.doi.org/10.22606/gr.2017.23005

[6] Obata, K., Tsuchida, S., Yamamoto, H., & Thome, K. (2017). Cross-Calibration between ASTER and MODIS Visible to Near-Infrared Bands for Improvement of ASTER Radiometric Calibration. *Sensors, 17*(8), 1793. https://dx.doi.org/10.3390/s17081793

[7] Amer, R., Kusky, T., & Ghulam, A. (2010). Lithological mapping in the Central Eastern Desert of Egypt using ASTER data. *Journal of African Earth Sciences, 56*(2-3), 75-82. https://doi.org/10.1016/j.jafrearsci.2009.06.004

[8] Padró, J.-C., Pons, X., Aragonés, D., Díaz-Delgado, R., García, D., Bustamante, J., . . . Cristóbal, J. (2017). Radiometric Correction of Simultaneously Acquired Landsat-7/Landsat-8 and Sentinel-2A Imagery Using Pseudoinvariant Areas (PIA): Contributing to the Landsat Time Series Legacy. *Remote Sensing, 9*(12), 1319. https://doi.org/10.3390/rs9121319

[9] Bransden, B. H., & Joachain, C. J. (2003). *Physics of atoms and molecules*: Pearson Education India.

[10] Zhao, B., Shao, G., Fan, B., Zhao, W., Xie, Y., & Zhang, R. (2015). Synthesis of flower-like CuS hollow microspheres based on nanoflakes self-assembly and their microwave absorption properties. *Journal of Materials Chemistry A, 3*(19), 10345-10352. https://doi.org/10.1039/c5ta00086f

[11] Gupta, R. P. (2017). *Remote sensing geology*: Springer.

[12] Dhara, M., Sengar, V. K., Chattoraj, S. L., & Bhattacharjee, S. (2017). Mapping of Alteration Zones in Mineral Rich Belt of South-East Rajasthan Using Remote Sensing Techniques. *World Academy of Science, Engineering and Technology, International Journal of Environmental, Chemical, Ecological, Geological and Geophysical Engineering, 11*(2), 154-158.

[13] Safari, M., Maghsoudi, A., & Pour, A. B. (2017). Application of Landsat-8 and ASTER satellite remote sensing data for porphyry copper exploration: a case study from Shahr-e-Babak, Kerman, south of Iran. *Geocarto International*, 1-16. http://dx.doi.org/10.1080/10106049.2017.1334834

[14] Gahlan, H., & Ghrefat, H. (2018). Detection of Gossan Zones in Arid Regions Using Landsat 8 OLI Data: Implication for Mineral Exploration in the Eastern Arabian Shield, Saudi Arabia. *Natural Resources Research, 27*(1), 109-124. http://dx.doi.org/10.1007/s11053-017-9341-8

[15] Garg, V., Kumar, A. S., Aggarwal, S., Kumar, V., Dhote, P. R., Thakur, P. K., . . . Muduli, P. R. (2017). Spectral similarity approach for mapping turbidity of an inland waterbody. *Journal of hydrology, 550*, 527-537. https://doi.org/10.1016/j.jhydrol.2017.05.039

[16] Abdelaziz, R., El-Rahman, Y. A., & Wilhelm, S. (2018). Landsat-8 data for chromite prospecting in the Logar Massif, Afghanistan. *Heliyon, 4*(2), e00542. https://doi.org/10.1016/j.heliyon.2018.e00542

[17] Masoumi, F., Eslamkish, T., Honarmand, M., & Abkar, A. A. (2017). A Comparative Study of Landsat-7 and Landsat-8 Data Using Image Processing Methods for Hydrothermal Alteration Mapping. *Resource Geology, 67*(1), 72-88. https://doi.org/10.1111/rge.12117

[18] Yao, K., Pradhan, B., & Idrees, M. O. (2017). Identification of rocks and their quartz content in gua musang goldfield using advanced spaceborne thermal emission and reflection radiometer imagery. *Journal of Sensors, 2017*. https://doi.org/10.1155/2017/6794095

[19] Hereher, M. E., & Abdullah, S. E. (2017). Lithologic mapping of Aja granitic batholiths, Ha'il, Saudi Arabia, using remote sensing. *Arabian Journal of Geosciences, 10*(14), 313. http://dx.doi.org/10.1007/s12517-017-3101-2

[20] Razmi, M., Asgari, H. M., Sohrab, A. D., Nazemosadat, S. M. J., & Khazaei, S. H. (2017). Monitoring oscillations coastline of Dayyer city during the El Niño and La Niño using OIF utility index.

[21] Abbaszadeh, M. (2010). Mapping hydrothermal alterations using ASTER images in Parkam area, Kerman. *GEOSCIENCES, 78*.

[22] Hooshyari, N. (2005). *Separation of alteration zones in relation to possible mineralization of gold and copper Zofre area using satellite data ASTER.* (BSc), Isfahan University of Technology.

[23] Manuel, R., Brito, M. d. G., Chichorro, M., & Rosa, C. (2017). Remote Sensing for Mineral Exploration in Central Portugal. *Minerals, 7*(10), 184. http://dx.doi.org/10.3390/min7100184

[24] Wang, G., Du, W., & Carranza, E. J. M. (2017). Remote sensing and GIS prospectivity mapping for magmatic-hydrothermal base-and precious-metal deposits in the Honghai district, China. *Journal of African Earth Sciences, 128*, 97-115. https://doi.org/10.1016/j.jafrearsci.2016.06.020

[25] Gómez-Palacios, D., Torres, M. A., & Reinoso, E. (2017). Flood mapping through principal component analysis of multitemporal satellite imagery considering the alteration of water spectral properties due to turbidity conditions. *Geomatics, Natural Hazards and Risk, 8*(2), 607-623. https://doi.org/10.1080/19475705.2016.1250115

[26] Hassanipak, A. A., & Sharafeddin, M. (2005). *Exploration Data Analysis* (Vol. 1). Tehran: Tehran university press.

[27] Soe, M., Kyaw, T. A., & Takashima, I. (2005). Application of remote sensing techniques on iron oxide detection from ASTER and Landsat images of Tanintharyi coastal area, Myanmar.

[28] Vincent, R. K. (1997). *Fundamentals of geological and environmental remote sensing* (Vol. 366): Prentice Hall Upper Saddle River, NJ.

[29] Asadzadeh, S., & de Souza Filho, C. R. (2016). A review on spectral processing methods for geological remote sensing. *International journal of applied earth observation and geoinformation, 47*, 69-90. https://doi.org/10.1016/j.jag.2015.12.004

[30] Ramezanali, A., Mansouri, E., & Feizi, F. (2017). Integration of aeromagnetic geophysical data with other exploration data layers based on fuzzy AHP and CA fractal model for Cu-porphyry potential mapping: a case study in the Fordo area, central Iran. *Bollettino di Geofisica Teorica ed Applicata, 58*(1). https://doi.org/10.4430/bgta0189

[31] Ezzati, A., Mehrnia, R., & Ajayebi, K. (2014). Detection of Hydrothermal potential zones using remote sensing satellite data in Ramand region, Qazvin Province, Iran. *Journal of Tethys, 2*(2), 93-100.

[32] Hasan, E., Fagin, T., El Alfy, Z., & Hong, Y. (2016). Spectral Angle Mapper and aeromagnetic data integration for gold-associated alteration zone mapping: a case study for the Central Eastern Desert Egypt. *International Journal of Remote Sensing, 37*(8), 1762-1776. https://doi.org/10.1080/01431161.2016.1165887

[33] Markovskiy, N. (2014). Drop-in Acceleration of GNU Octave.

[34] Esmaeeli, M., Tabaee, M., & Asadiharooni, H. (2012). *Remote Sensing Study (ASTER & TM) and Geology of Southwest Meyme Iron Ore Deposit, IRAN*. Paper presented at the 31st Symposium of Geosciences, Tehran, IRAN.

[35] Boloki, M., & Poormirzaee, M. Using ASTER image processing for hydrothermal alteration and key alteration minerals mapping.

[36] van der Werff, H., & van der Meer, F. (2015). Sentinel-2 for mapping iron absorption feature parameters. *Remote Sensing, 7*(10), 12635-12653. https://doi.org/10.3390/rs71012635

[37] Van Der Werff, H., & Van Der Meer, F. (2016). Sentinel-2A MSI and Landsat 8 OLI provide data continuity for geological remote sensing. *Remote Sensing, 8*(11), 883.

https://doi.org/10.3390/rs8110883

International Journal of Science and Engineering Applications
Volume 7–Issue 04, 52-53, 2018, ISSN:-2319–7560

Use of Treated Water from Sewage Treatment Plants in Irrigation by using Solar/Grid Powered Micro Irrigation Infrastructure

Neeraj Sharma
Executive Engineer
Command Area Development Authority
Kurukshetra, India

Rajiv Bansal
Chief Engineer
Command Area Development Authority
Panchkula, India

Amit Kumar Raghuvanshi
Superintending Engineer
Command Area Development Authority
Kaithal, India

Abstract: With a view of augmenting water for assured supply to the every field, a new intervention has been proposed for the reuse of treated waste water from the existing Sewage Treatment Plants for the use of water in the best alternative which will help in enhancing the irrigation in agriculture sector. Working on these lines, Installation of Solar/Grid Powered Micro Irrigation Infrastructure has been proposed by selecting the nearby area of the existing Sewage Treatment Plant by providing common infrastructure with components Retention tank near outlet head of STP, Pumping Unit (Grid/Solar Powered), Filtration units, HDPE pipe network/Hydrant/Outlet assembly, Valves etc with Drip/Sprinkler irrigation sets. The Solar Power System is proposed to be connected with the utility power grid so that the energy generated by the solar modules, whenever not required for operation of the pumping system or is in excess of requirement, can be sent to the Utility Grid through bidirectional meter.

Keywords: Solar/Grid, Micro Irrigation, Sewage Treatment Plant (STP), Irrigation Efficiency, High Tension Line

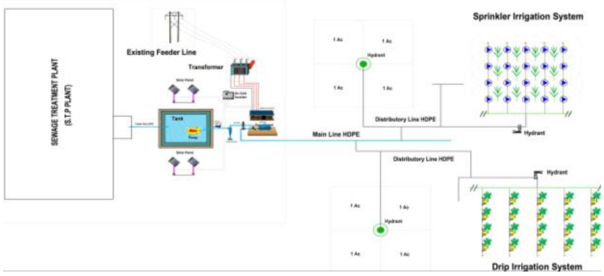
1. INTRODUCTION

Growing Economy, Urbanization, and Population along with rapid industrialization has increased the production of waste water many folds all over. The disposal of untreated waste water has therefore, become a serious problem as it affects the fresh water resources and human health. A large volume of waste water is produced in the cities & towns. In existing sewage system, domestic sewage and industrial effluents including washings from cattle sheds etc. flows through the sewerage system, which pollute land and water resources around towns and cities. Due to unorganized disposal of waste water, which is drained into rivers have eventually deteriorated the quality of freshwater resources besides creating serious problem of environmental hazard by polluting surface and sub-surface water, thereby also endangering the aquatic life as well as the human population living on the banks of these rivers. Due to over exploitation of ground water for agriculture, subsurface water is declining at an alarming rate. Water table is going down which creates endangered to the sustainability of the agriculture production system. Ground water along with existing water resources are proving inadequate, therefore there is need to tap and encourage alternate means which could provide irrigation to crops. In light of this treated sewage water is the best alternative which will help in irrigation. This paper is envisaging the implementation of a project on treated water from the STP's used in agricultural fields for irrigation purposes, which was otherwise going waste in to various rivers, besides polluting them. Irrigation to crops with secondary treated sewage water may also supply nutrients to the crops which mean less of inorganic fertilizers will be required along with saving on power and diesel consumption as supply of water from STP's will mean lesser dependence on tube wells for pumping out ground water, thus lowering production costs which in turn means higher profit for farmer community. Underground Pipe Line System (UGPS) will be installed in the farmer fields with command area, depending upon the discharge of each STP. Regular monitoring of quality of water is being assured at Sewage Treatment Plants, which has been found fit for irrigation purposes. Most of the toxic elements that are generally found in treated sewage water are below the permissible limits for use in agriculture for all the Sewage Treatment Plants.

2. METHODOLOGY

Solar/Grid Powered Micro Irrigation Infrastructure in the Sewage Treatment Plant Commands has been installed by providing Retention tank near outlet head, Pumping Unit (Grid/Solar Powered), Filtration units, HDPE pipe network/Hydrant/Outlet assembly, Valves etc. in the command area of Sewage Treatment Plant, as shown in layout plan Figure-1. Drip/Sprinkler irrigation sets will be installed by the individual farmers in their farm holdings by availing the benefits of subsidy. It is proposed to take water from Sewage Treatment Plant outlet through underground pipeline with gravity and to store the same in the retention tank of appropriate size within the Sewage Treatment Plant area. Solar/Grid powered pumping system connected through net metering has been installed nearby the tank with proper filtration systems to avoid any chocking. Water has been carried to entire area selected nearby the Sewage Treatment Plant through HDPE pipe line network under pressure. The entire pipe network has been buried under ground at 3 feet deep to avoid land acquisition. Water with the requisite pressure for running of the drip/sprinkler set has been made available to each shareholder at his farm holding through the common infrastructure to be operated & maintained by the Water User's Associations.

Figure 1



3. Design Parameters.

Modified penman method has been used to find out crop water requirement and computed the peak water requirement in rabi & kharif season. In this scheme average water requirement of 2mm/day has been considered. Considering this crop water requirement and capacity of Sewage Treatment Plant each component of this scheme shall be designed in such a manner that minimum operating pressure of 2.5Kg/cm^2 available to the farmers on their farm gate. Size of the retention tank has been designed by considering discharge of the outlet and volume of water accumulated as effective outflow in million litres per day. Further operation time of Sewage Treatment Plant per day has also been considered. A feeder pipe of required size in appropriate length has been provided from STP outlet to the storage tank by gravitational flow. Solar pumping system is a vital part of this scheme and in this scheme grid connected solar powered pump has been considered to reduce the cost of electricity of appropriate size. At least one pump is provided in a block of area 40 to 50 Hactare. Solar pumps of the capacity up to 10 to 20HP is preferred with average working of 14 hours/day. The HP of pump set required is based upon design discharge and total operating head. The total operating head is sum of total static head, friction loses worked out with hazen-williams equation in pipeline network and losses in filtration unit. Pipes in main line and sub-main shall not be below 110 mm (OD) and the size shall be decided based on the criteria to limit the friction loss in the main & sub main keeping the minimum flow velocity in the pipeline as 0.6m/sec.

$$\text{HP of pump set} = \frac{Q x H}{75e}$$

Q = discharge (in LPS)
H = head (in meter)
e = Pumping efficiency

Solar PV array of at least 1100wp capacities has been installed per HP rating of pumping sets and total capacity of the Solar pv array for operation of solar pumping sets has been worked out in such a manner that total annual solar energy generation from the PV power system in no case be lesser than the total energy requirement to run the Micro Irrigation System and there is no net import of energy from the utility grid on annual basis. Total Capacity of the Solar PV array to be installed on each STP outlet scheme for operation of the Solar Pumping Set(s) will be worked out in such a manner that total annual solar energy generation from the PV Power System is in no case lesser than the total annual energy requirements of the MI scheme including auxiliary requirements and there is no net import of energy from the Utility Grid on annual basis.

The output power of SPV would be fed to the inverters for conversion of the DC produced by SPV array to AC for operation of the motor pump sets and feeding the same into the nearest electricity grid through 11KV, 24 hours energized HT independent line after synchronization when in excess of requirement. A hydrant assembly has been provided with minimum 110 size for the land holding of every share holder with provision of at least one hydrant for every 04 acres or less.

4. Conclusion

Significant irrigation from tube wells are being done in various parts. Water use efficiency is very poor and ground water wastage in shape of flood irrigation is being over exploited. It also causes wastage of electricity. Use of micro irrigation infrastructure on Sewage Treatment Plant will reduce the use of tube wells by which ground water will be saved and treated water will be used which was otherwise going waste. More area can be brought under irrigation by using Sewage Treatment Plant treated water, which was otherwise either rain fed or irrigated by tube wells. Where there is no possibility of irrigation through canal commands and ground water is very low, the only solution is creating of Micro Irrigation infrastructure on Sewage Treatment Plant outlets. Where the ground water table is very high with brackish water, there are chances of creating the situation of water logging, which is harmful for soil properties, in these areas, it is essentially required to minimize the flood irrigation by replacing with micro irrigation. Hence, by installation of Solar/Grid Powered Micro Irrigation Infrastructure in the Sewage Treatment Plant Commands through integrated approach of supply management and demand management, yield & net sown area will increase, dependency of tube well & overexploitation of ground water will decrease, saving of highly subsidized electricity and above all change of the mindset of the farmers towards the use of available water judiciously.

5. References

1. Bucks, D.A. 1993. Micro Irrigation- Worldwide usage report. In Proceedings of Workshop on Micro Irrigation, Sept 2 1993. 15th Congress on Irrigation and Drainage.
2. Batchelor, C., Lovell, C.& Murata, Water User Efficiency of Simple Subsurface Irrigation Systems, In : Proceedings of 7th International Conference on Water and Irrigation, 13-16, May, 1996
3. Neeraj Sharma, Rajiv Bansal, for Installation of Solar/Grid Powered Micro Irrigation Infrastructure in the Canal Commands in International Journal of Engineering and Technology Volume-56- February 2018.
4. Design and Optimization of Irrigation Distribution Networks Food and Agriculture Organisation of the United Nations irrigation and Drainage Paper.
5. Howell, T.A. (2001), Enhancing water use efficiency in irrigated agriculture, Agron J 93 (2), 281-289.
6. Hsiao T.C., P. Steduto, and E. Fereres (2007), A Systematic and quantitative approach to improve water use efficiency in agriculture, Irrg. Sci., 25(3), 209-232, doi:10.1007/s00271-007-0063-2.

International Journal of Science and Engineering Applications
Volume 7–Issue 04, 54-63, 2018, ISSN:-2319–7560

EFFICIENT ALGORITHM TO TRANSFORM MINIMALIST SUBSET OF LTL FORMULA INTO FINITE STATE MODELS

Bilal Kanso
Lebanese University
Department of Computer Science, Beirut

Ali Kansou
Lebanese University
Department of Computer Science, Beirut

Abstract: The translation of LTL formula into equivalent *Büchi* automata plays an important role in many algorithms for LTL model checking, which consist in obtaining a *Büchi* automaton that is equivalent to the software system specification and another one that is equivalent to the negation of the property. The intersection of the two *Büchi* automata is empty if the model satisfies the property.

Generating the *Büchi* automaton corresponding to an LTL formula may, in the worst case, be exponential in the size of the formula, making the model checking effort exponential in the size of the original formula. There is no polynomial solution for checking emptiness of the intersection. That comes from the translation step of LTL formula into finite state models. This makes verification methods hard or even impossible to be implemented in practice. In this paper, we propose a subset of LTL formula which can be converted to *Büchi* automata whose the size is polynomial.

Keywords: Linear Temporal Logic, *Büchi* automata, Model checking, Compositional modelling

1. INTRODUCTION

Model checking becomes increasingly one of the most important tools to verify the correctness of computer-based control systems [1, 4, 12, 15]. It is a formal verification technique consisting in algorithmically verifying whether system properties such as the absence of deadlocks (described in some appropriate logical formalism such as temporal logic) are satisfied by the system (described as a suitable finite state model). The success of the model checking technique comes from the fact that it is completely automatic. Running a model checking on a given system model to verify a desired property leads automatically to fail state or successful state. In case the system model fails to satisfy the property, the model checking tool can offer a counterexample which can be used as an error trace provided for debugging purposes.

Model checking approaches vary according to the logic used to specify system properties [3, 12, 18]. One of the most used logics is the Linear Temporal Logic (LTL) [11]. The underlying idea consists in transforming the negation of the LTL expression into a Büchi automaton, and then computing the product between the Büchi automaton representing the system and the one representing the negation of the LTL expression. If the product is not empty, that means the property expressed by the negation of the LTL expression is not satisfied by the system, otherwise the property is well-satisfied. However, the decision problem for emptiness of the intersection is PSPACE-hard [2, 19]. That comes from the translation of LTL formula into Büchi automata. Indeed, the space complexity of this approach is linear in the size of Büchi automata and exponential in the length of the LTL formula: the Büchi automaton of a property (described as a LTL formula) is constructed in exponential space in the length of this property. This makes verification methods hard or even impossible to be implemented in practice and makes the scalability of the LTL model checking limited, which commonly referred to as the state explosion problem [8].

In this paper, we contribute to finding a subset of LTL properties that can be converted polynomially into Büchi automata. Finding such a subset of LTL logic will be viewed as one the most promising directions to bridge the gap between the increasing complexity of state models and actual model checking methods. We define a fragment that we call, *Flat LTL Logic* and we show how formula in this fragment can be transformed into Büchi automata whose the state space size is linear. Due to the structure of flat LTL formula, our algorithm can be compositional in the sense that the final finite state model associated to a given formula is obtained by developing a sub-automaton for each sub-formula of the principal formula. Hence, the basic idea for developing the final automaton for a flat LTL formula f is that f can be recursively decomposed into a set of sub-formula, arriving at sub-formula that can be completely handled. Composition is then used for assembling different sub-automata and then forming larger ones. Such a composition can be seen as an operation taking sub-automata for sub-formula as well as the flat LTL operator to provide a new more complex automaton.

In order to guide the construction of the final automaton for a flat LTL formula f from the sub-automata associated to the sub-formula $f_1, f_2, \dots, f_n$ of f, we build the finite syntax tree, *FST*(f) of the formula f. The nodes of a finite syntax tree are labeled, either by flat LTL operators or by propositional operators. The leaves are labeled only by atomic propositions.

Thus, the target Büchi automaton is obtained by exploring the tree in pre-order.

The rest of this article is organized as follows: Section 2 briefly describes Büchi automata. In Section 3, we describe our fragment of LTL logic and the reasons to choose it. In Section 4, we present for each formula in our fragment LTL, its equivalent Büchi automata and show the proof of this equivalence. Section 5 presents the finite syntax tree associated to a formula defined in our fragment LTL while Section 6 shows the final algorithm that generates to any formula in our fragment an equivalent Büchi automaton. Section 7 presents the conclusion and some future works.

2. Automata on infinite words

2.1 Büchi automata

Automata on infinite inputs were introduced by Büchi. A Büchi automaton is a non- deterministic finite-state automaton which takes infinite words as input [9, 10, 14]. A word is accepted if the automaton goes through some designated "good" states infinitely often while reading it. A **Büchi automaton** is a finite state automaton defined by a 5-tuple $A = (S, s_0, F, \sum, \delta)$ where:

- S is a finite set of states,
- $s_0 \in S$ is the initial state,
- Σ is a non-empty set of atomic propositions,
- $F \subseteq S$ is a finite set of accepting states,
- $\Delta : S \times \Sigma \rightarrow 2^S$ is a transition function.

In the following of this paper, the initial state of a Büchi automaton is pointed to by incoming arrows and the accepting states are marked by double circles.

A run of A on $\sigma = \sigma(0)\sigma(1)\sigma(2) \ldots \in \Sigma^\omega$ is an infinite sequence of states $s_0s_1s_2 \ldots \in S^\omega$ starting with the initial state s_0 of A such that $\forall i, i \geq 0$, $s_{i+1} \in \delta(s_i, \sigma(i))$. A run $s_0s_1s_2 \ldots$ is **accepting** by an automaton A if A goes through accepting states (*i.e.* $\in F$) infinitely often while reading it. The *accepted language* of a Büchi automaton A, denoted by $\mathscr{L}_\omega(A)$, is then defined by:

$$\mathscr{L}_\omega(A) = \{ \sigma \text{ in } \Sigma^\omega \mid \text{there is an accepting run for } \sigma \text{ in } A \}$$

2.2 Operations on Büchi automata

The basic idea of the construction of the union of two Büchi automata A_1 and A_2 is to add a new initial (nonaccept) state s_{new} to the set of states union of A_1 and A_2. The transitions of the union of A_1 and A_2 are transitions of both A_1 and A_2 with the following two transitions:

a) A transition from s_{new} to a state s labeled with a proposition p if and only if there is transition from the initial state of A_1 to the state s labeled with the proposition p;

b) A transition from s_{new} to a state s labeled with a proposition p if and only if there is transition from the initial state of A_2 to the state s labeled with the proposition p

Definition 1 (Buchi automata union). *Let $A_1 = (S_1, s_{10}, F_1, \Sigma, \delta_1)$ and $A_2 = (S_2, s_{20}, F_2, \Sigma, \delta_2)$ be two büchi automata. The union $A_1 \cup A_2$ of A_1 and A_2 is the büchi automaton $A = (S, s_0, F, \Sigma, \delta)$ defined as follows:*

- $S = S_1 \cup S_2 \cup \{s_0\}$
- $s_0 \in S$ *is the initial state*
- $F = F_1 \cup F_2$
- *the transition relation δ is defined as follows:*

$$\delta(s,p) = \begin{cases} \delta_1(s,p) \text{ if } s \in S_1 \\ \delta_2(s,p) \text{ if } s \in S_2 \\ \delta_1(s_{10},p) \cup \delta_2(s_{20},p) \text{ if } s \text{ is the initial state } s_0 \end{cases}$$

The construction of the intersection automaton works a little differently from the finite state automata case. One needs to check whether both sets of accepting states are visited infinitely often. Consider two runs r_1 and r_2 and a word σ where r_1 goes through an accept state after $\sigma(0)$, $\sigma(2)$, $\cdots$ and r_2 enters accept state after $\sigma(0)$ $\sigma(3)$ $\cdots$. Thus, there is no guarantee that r_1 and r_2 will enter accept states simultaneously. To overcome this problem, we need to identify the accept states of the intersection of the two automata. To do so, we create two copies of the intersected state space. In the first copy, we check for occurrence of the first acceptance set. In the second copy, we check for occurrence of the second acceptance set. When a run enters a final state in the first copy, we wait for that run also enters in an accept state in the second copy. When this is encountered, we switch back to the first copy and so on. We repeat jumping back and forth between the two copies whenever we find an accepting state.

Definition 2 (Buchi automata intersection). *Let $A_1 = (S_1, s_{10}, F_1, \Sigma, \delta_1)$ and $A_2 = (S_2, s_{20}, F_2, \Sigma, \delta_2)$ be two büchi automata. The intersection $A_1 \cap A_2$ of A_1 and A_2 is the büchi automaton $A = (S, s_0, F, \Sigma, \delta)$ defined as follows:*

- $S = S_1 \times S_2 \times \{1, 2\}$
- $s_0 = (s_{10}, s_{20}, 1)$
- $F = S_1 \times F_2 \times \{2\}$
- *The transition function δ is defined as follows:*

$$\delta((s_1, s_1', 1), p) = \begin{cases} (s_2, s_2', 1) \text{ if } s_2 \in \delta(s_1, p), s_2' \in \delta(s_2, p) \text{ and } s_1 \notin F_1 \\ (s_2, s_2', 2) \text{ if } s_2 \in \delta(s_1, p), s_2' \in \delta(s_2, p) \text{ and } s_1 \in F_1 \end{cases}$$

$$\delta((s_1, s_1', 2), p) = \begin{cases} (s_2, s_2', 2) \text{ if } s_2 \in \delta(s_1, p), s_2' \in \delta(s_2, p) \text{ and } s_1' \notin F_2 \\ (s_2, s_2', 1) \text{ if } s_2 \in \delta(s_1, p), s_2' \in \delta(s_2, p) \text{ and } s_1' \in F_2 \end{cases}$$

Theorem 1. *Let $\psi = \varphi_1 \vee \varphi_2$ (resp. $\psi = \varphi_1 \wedge \varphi_2$) be a LTL formulæ and A_{φ_i} be the büchi automaton equivalent to φ_i for $i = 1, 2$. Let A_ψ be the LTL automaton built according to Definition 1 (resp. Definition 2). Then, $\mathsf{Words}(\psi) = \mathcal{L}_\omega(A_\psi)$*

3. Flat LTL Logic

In this section, we introduce our subset of LTL logic that we call *Flat LTL Logic*. This fragment will be used to express temporal properties and then translate them into Büchi automata in linear size. The syntax of our Flat LTL logic adds to usual boolean propositional operators ¬ (negation) and ∧ (conjunction), some modal operators that describe how the behaviour changes with time.

- **Next**: **X**φ requires that the formula φ be true in the next state;
- **Until:** φ_1 ***U*** φ_2 requires that the formula φ_1 be true *until* the formula φ_2 is true, which is required to happen;
- **Eventually:** $\Diamond\varphi$ requires that the formula φ be true at some point in the future (starting from the present) and it is equivalent to $\Diamond \varphi \equiv true\ U\ \varphi$;
- **Release:** φ_1 ***R*** φ_2 requires that its second argument φ_2 always be true, a requirement that is *released* as soon as its first argument φ_1 becomes true. It is equivalent to $\varphi_1\ R\ \varphi_2 \equiv \neg\ (\neg\ \varphi_1\ U\ \neg\ \varphi_2)$.

3.1 Our fragment LTL Logic

Definition 3 (Flat LTL formulæ). *The set of Flat LTL formulæ $\mathcal{L}_f$ is given by the following grammar:*

$$\varphi := \Theta \mid \Theta \mathsf{U} \varphi \mid \varphi \mathsf{R} \Theta \mid \mathsf{X}\varphi \mid \neg\Delta \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2$$

where Θ is a propositional formula defined by the following grammar:

$$\Theta := \mathsf{true} \mid p \mid \neg\Theta \mid \Theta_1 \wedge \Theta_2$$

and Δ is the temporal formula defined by the following grammar:

$$\Delta := \Delta \mathsf{U} \Theta \mid \Theta \mathsf{R} \Delta \mid \mathsf{X}\varphi \mid \neg\Delta \text{ with } p \in \Sigma$$

Example: the formula $X(a\ U \neg(d\ R\ (\neg\ b\ U\ X\ c)))$ is not in $\mathcal{L}_f$ since the sub-formula $(\neg b\ U\ X\ c)$ in $\neg(d\ R\ (\neg b\ U\ X\ c))$ should be of the form $\Delta\ U\ \theta$ that is not the case. But, the formula $X(a\ U \neg\ (d\ R\ (\neg\ b\ R\ X\ c)))$ is in $\mathcal{L}_f$.

For the sake of brevity and the lack of space, we only discuss here why the fragment $\theta\ U\ \varphi$ is included within our LTL fragment to the detriment of both formula $\varphi_1 U\ \varphi_2$ and $\varphi_1\ U\ \theta$. It is well-known the size of an Büchi automaton $\overline{A}$ that recognizes the complement language $\mathcal{L}_\omega(\overline{A})$ of the language accepted $\mathcal{L}_\omega(A)$ by an automaton A is exponential [13, 16]. Suppose we have separately built an automaton A_1 for φ_1 and an automaton A_2 for φ_2, and let us then try to compositionally obtain the resulting automaton A for φ. According to the until operator's semantics, it is required that φ holds at the current moment, if there is some future moment for which $\varphi 2$ holds and $\varphi 1$ holds at all moments until that future moment. That means constructing the automaton for $\varphi = \varphi 1\ U\ \varphi 2$ firstly requires constructing of the intersection of A_1 and $\overline{A_2}$. As stated previously, computing $\overline{A_2}$ is exponential and therefore, constructing the Büchi automaton for $\varphi_1\ U\ \varphi_2$ is exponential. To avoid this kind of formula, we choose the formula $\theta\ U\ \varphi$ to be a part of our LTL subset where the construction of the Büchi automaton associated to it, does not need to complement any Büchi automaton.

3.2 Flat Positive Normal Form (FPNF)

As LTL formula, flat LTL formula can be transformed into the so-called *Flat Positive Normal form (FPNF)*. This form is characterized by the fact that negations only occur adjacent to atomic propositions. All negation symbols of the given LTL formula have to be pushed inwards over the temporal operators.

Definition 4 (FPNF). *The set of Flat Positive Normal Form (FPNF) formulæ $\mathcal{L}_{FPNF}$ is given by the following grammar:*

$$\varphi := \mathsf{true} \mid \mathsf{false} \mid p \mid \neg p \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \Theta \mathsf{U} \varphi \mid \varphi \mathsf{R} \Theta \mid \mathsf{X}\varphi$$

Each formula $\varphi \in \mathscr{L}_f$ can be transformed into a formula $\varphi' \in \mathscr{L}_{FPNF}$. This is done by pushing negations inside, near to atomic propositions. To do this, we use the following transformation rules:

$\neg\, true \rightarrow\ false$	$\neg\,(\varphi\ U\ \theta) \rightarrow\ \neg\varphi\ R\ \neg\theta$
$\neg\,\neg\,\varphi \rightarrow \varphi$	$\neg\,(\varphi_1 \wedge\ \varphi_2) \rightarrow \neg\varphi_1 \vee\ \neg\,\varphi_2$
$\neg\, X\varphi \rightarrow X\,\neg\varphi$	$\neg\,(\theta\ R\ \varphi) \rightarrow \neg\theta\ U\ \neg\varphi$

This transformation is done in linear complexity as it is shown by the following theorem:

Theorem 2. *For any flat LTL formulæ $\varphi \in \mathcal{L}_f$, there exists an equivalent LTL formula $\varphi\prime \in \mathcal{L}_{FPNF}$ in flat positive normal form with $|\varphi\prime| = \mathcal{O}(|\varphi|)$.*

Example: the formula $X(a\ U\ \neg(d\ R\ (\neg\ b\ R\ Xc)))$ is in $\mathscr{L}_f$, but not in $\mathscr{L}_{FPNF}$. It can be transformed into $X\ (a\ U\ \ (\neg d\ U\ (b\ U\ X\ \neg c)))$ which is in $\mathscr{L}_{FPNF}$.

3.3 Semantics

The semantics of LTL formula is defined over infinite[1] sequences $\sigma : \mathbb{N} \rightarrow 2^{\Sigma}$. In other words, a model is an infinite sequence $A_0\ A_1\ \cdots$ of subsets of Σ. The function σ, called *interpretation function*, describes how the truth of atomic propositions changes as time progresses. For every sequence σ, we write $\sigma = (\sigma(0), \cdots, \sigma(n), \cdots)$. Thus, we have the following notations:

- **σ(i)** denotes the state at index i and **σ(i:j)** the part of σ containing the sequence of states between i and j;
- **σ(i…) =A_i A_{i+1} A_{i+2} ⋯** denotes the suffix of a sequence $\sigma = A_0\ A_1\ A_2 \cdots \in\ (2^{\Sigma})^{\omega}$ starting[2] in the (i+1)st symbol A_i.

We also write $\sigma(i) \vDash\ \ \varphi$ to denote that "*φ is true at time instant i in the model* σ". This notion is defined inductively, according to the structure of φ.

The LTL formula are interpreted over infinite sequences of states $\sigma: \mathbb{N} \rightarrow\ 2^{\Sigma}$ as follows:

Definition 5 (Semantics of FLat LTL). *Let $\sigma : \mathbb{N} \longrightarrow 2^{\Sigma}$ be an interpretation function and $\varphi \in \mathcal{L}$. σ* satisfies *φ, noted $\sigma \models \varphi$, is inductively defined over the construction of φ as follows:*

- $\varphi = \mathsf{true}$, *then* $\sigma \models \mathsf{true}$
- *if* $\varphi = p$, *then* $\sigma \models p$ *iff* $p \in \sigma(0)$
- *if* $\varphi = \mathsf{X}\varphi'$, *then* $\sigma \models \mathsf{X}\varphi'$ *iff* $\sigma(1) \models \varphi'$
- *if* $\varphi = \Theta \mathsf{U} \varphi$, *then* $\sigma \models \Theta \mathsf{U} \varphi$ *iff* $\exists i, i \geq 0, \sigma(i,\dots) \models \varphi$ *and* $\forall j, 0 \leq j < i, \sigma(j...) \models \Theta$
- *if* $\varphi = \varphi \mathsf{R} \Theta$, *then* $\sigma \models \varphi \mathsf{R} \Theta$ *iff* $\exists i, i \geq 0, \sigma(i,\dots) \models \varphi$ *and* $\forall j, j \geq 0, \sigma(j...) \models \Theta$ *or* $\exists i, i \geq 0\ (\sigma(i...) \models \varphi \wedge \forall k, k \leq i, \sigma(k...) \models \Theta)$
- *if* $\varphi = \neg\varphi'$, *then* $\sigma \models \neg\varphi'$ *iff* $\sigma \not\models \varphi'$
- *Propositional connectives are handled as usual*

The semantics of a LTL formula can be also seen as the language **Words**(φ) that contains all infinite words over the set of atomic propositions (*i.e.* alphabet) 2^{Σ} that satisfy φ. Thus, the language **Words**(φ) for a LTL formula φ is formally defined by **Words**(φ) $=\{\sigma \in\ (2^{\Sigma})^{\omega} \mid \sigma \vDash \varphi\}$.

Proposition 1. *Two LTL formula φ_1 and φ_2 are* equivalent, *denoted $\varphi_1 \equiv \varphi_2$, if* Words*($\varphi_1$) =*Words*($\varphi_2$).*

4. Construction Of Buchi Automata For Flat LTL Logic

Our algorithm is a compositional algorithm. It constructs for each sub-formula in our fragment LTL logic, an equivalent Büchi automaton and in a compositional way regroup all resulting Büchi automata in order to get the target Büchi automaton of the target flat LTL formula.

In the sequel, we firstly explain for each sub-formula in our fragment LTL logic how its equivalent Büchi automaton can be obtained.

4.1 Büchi automata for θ formula

The Büchi automaton associated to a propositional formula θ is obtained by creating two states s_0 and s_1 and two transitions tr_1 and tr_2. s_0 is the only initial state while s_1 is the only final state. tr_1 is the transition from s_0 to s_1 labeling with θ while the transition tr_2 is a loop labeled with *true* over the state s_2.

[1] 2^{Σ} is the power set of the proposition set Σ.
[2] ω: is typically used to denote *infinity*.

Definition 6 (Θ automaton). *Let Θ be a propositional formulæ. The automaton $A_\Theta = (S_\Theta, s^0_\Theta, F_\Theta, \Sigma, \delta_\Theta)$ associated to Θ is defined as follows:*

– $S_\Theta = \{s_0, s_1\}$, $s^0_\Theta = s_0$, $F_\Theta = \{s_1\}$
– *The transition function δ is defined as follows:*

$$\delta_\Theta(s_0, \Theta) = \{s_1\} \text{ and } \delta_\Theta(s_1, \text{true}) = \{s_1\}$$

Figure 1 shows the Büchi automaton associated to the propositional formula θ = a ∧ ¬b.

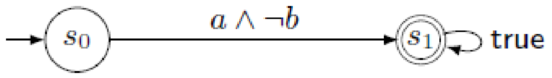


Figure 1: Example of automaton associated to θ

4.2 Büchi automata for θ U φ formula

The main idea behind the composition θ *U* φ is to add a new initial (nonaccept) state s_{new} to the set of states of the automaton A_φ associated to φ with the following transitions:

a) A loop over the added state s_{new} labelled with the propositional formula θ

b) Transitions s_{new} to a state s labelled with a proposition p if and only if there a transition from the initial state s^0 of A_φ to the state s labelled with the proposition p .

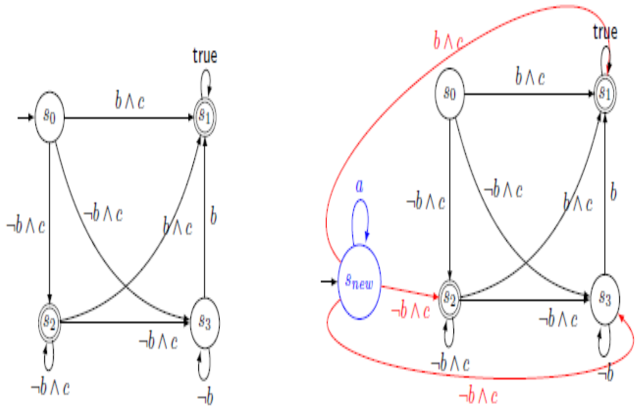
All other transitions of A_φ, as well as the accept states, remain unchanged. The state s_{new} is the single initial state of the resulting automaton, is not accept, and, clearly, has no incoming transitions except the loop one.

Definition 7 ($\Theta \cup \varphi$ automaton). *Let Θ be a propositional formula and φ be an LTL flat formulæ. Let $A_\varphi = (S_\varphi, s^0_\varphi, F_\varphi, \Sigma, \delta_\varphi)$ be the automaton associated to φ. The automaton $A_\psi = (S_\psi, s^0_\psi, F_\psi, \Sigma, \delta_\psi)$ associated to $\psi = \Theta \cup \varphi$ is defined as follows:*

– $S_\psi = \{s_{\text{new}}\} \cup S_\varphi$
– $s^0_\psi = s_{\text{new}}$, $F_\psi = F_\varphi$
– *The transition function δ_ψ is defined as follows:*

$$\delta_\psi(s,p) = \begin{cases} \delta_\varphi(s,p) \text{ if } s \in S_\varphi \ (A_\varphi \text{ transitions}) \\ \delta_\varphi(s^0_\varphi, p) \text{ if } s = s_{\text{new}} \ (\text{Connection initial state to } A_\varphi) \\ \{s_{\text{new}}\} \text{ if } s = s_{\text{new}} \text{ and } p = \Theta \ (\text{Loop over the new initial state}) \end{cases}$$

Example: Figure 2 illustrates the composition definition of θ *U* φ. **Figure 2a** shows the Büchi automaton associated to (◊ b) *R* c. To construct the Büchi automaton associated to (a *U* ((◊ b) *R* c)), we add a new state *snew* that we consider as initial state. Then, for each transition outgoing from snew with label l and goes to state s, we add a transition from s*new* to the state s with a label l. Finally, we then add a loop labeled with the atomic proposition a over the added state.



(a) (◊b) R c (b) a U (◊b R c)

Figure 2: Example of composition: θ *U* φ

Theorem 3. *Let $\psi = \Theta \cup \varphi$ be a flat LTL formulæ and A_φ be the büchi automaton equivalent to φ. Let A_ψ be the automaton built according to Definition 7. Then, $\text{Words}(\psi) = \mathcal{L}_\omega(A_\psi)$.*

4.3 Eventually operator ◊φ:

The Büchi automaton construction of the formula ◊φ is a particular case of the Büchi automaton construction of the formula θ *U* φ where θ is the *true* formula. Thus, the main idea behind the composition ◊φ is to add a new initial (nonaccept) state s_{new} to the automaton states set A_φ associated to φ with the same transitions defined for θ *U* φ where the loop over the added state s_{new} is labeled with *true* instead of the atomic formula θ.

4.4 Büchi automata for Xφ formula

The main idea behind the composition *X*φ consists in adding two new states s_{new} (neither initial state or accept state) and s_{init} (considered as the initial state) to the state set of the automaton A_φ with the following transitions:

a) Add for any transition in A_φ which starts from the initial state s^0 to a state s, a transition from s_{new} to s;

b) Add a transition from the initial state s_{init} to the s_{new} labeled with *true*.

All other transitions of A_φ remain unchanged and final states of A_φ become accept ones of A_ψ and initial state of A_ψ become the state s_{init}.

Definition 8 ($X\varphi$ **automaton**). *Let φ be an Flat LTL formulæ. Let $A_\varphi = (S_\varphi, s^0_\varphi, F_\varphi, \Sigma, \delta_\varphi)$ be the automaton equivalent to φ. The* **automaton** *$A_\psi = (S_\psi, s^0_\psi, F_\psi, \Sigma, \delta_\psi)$ equivalent to $\psi = X\varphi$ is defined as follows:*

- $S_\psi = S_\varphi \cup \{s_{\mathsf{new}}, s_{\mathsf{init}}\}$
- $s^0_\psi = s_{\mathsf{init}}$, $F_\psi = F_\varphi$
- *The transition function δ is defined as follows:*

$$\delta_\psi(s,p) = \begin{cases} \delta_\varphi(s,p) \text{ if } s \in S_\varphi \ (A_\varphi \text{ transitions}) \\ \delta_\varphi(s^0_\varphi, p) \text{ if } s = s_{\mathsf{new}} \ (\text{Connection } s_{\mathsf{new}} \text{ state to initial state of } A_\varphi) \\ \{s_{\mathsf{new}}\} \text{ if } s = s_{\mathsf{init}} \text{ and } p = \mathsf{true} \ (\text{Connection } s_{\mathsf{init}} \text{ to } s_{\mathsf{new}}) \end{cases}$$

Example: Figure 3 illustrates the definition of $X\varphi$. **Figure 3a** shows the Büchi automaton associated to the formula (a *U* (*X* b *R* c)). To construct the Büchi automaton equivalent to *X*(a *U* (*X*b *R* c)), we add a new state s_{new} and for each transition tr starting from the initial state s^0_φ to a state s, a transition from s_{new} to s with the same label. Finally, we add the state s_{init} that we consider as initial and we connect s_{init} to s_{new} with a transition labeled with the *true* label.

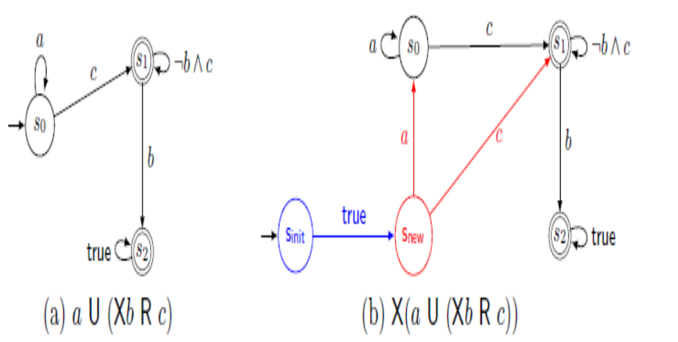


Figure 3: Example of composition: *X*φ

Theorem 4. *Let $\psi = X\varphi$ be a LTL formulæ and A_φ be the büchi automaton equivalent to φ. Let A_ψ be the LTL automaton built according to Definition 8. Then,* $\mathsf{Words}(X\varphi) = \mathcal{L}_\omega(A_\psi)$.

4.5 Büchi automata for φ R θ formula

The formula φ *R* θ informally means that θ is true until φ becomes true, or θ is true forever. Thus, the construction of a Büchi automaton for φ *R* θ can be done by construction the Büchi automaton associated to the fact that θ is true until φ becomes true and the construction of a Büchi automaton associated to the fact that θ is true forever. Finally, make the union between the two constructed Büchi automata. Consequently, to build the Büchi automaton for φ *R* θ, we need to add two new states s_i and s_f to the set of states of the automaton A_φ. s_i becomes the single initial state of the resulting automaton and s_f is added to set of final states of the resulting automaton. The following transitions are added to the set of transitions of the resulting automaton:

a) For any transition from the initial state s^0 of A_φ to a state s labeled with a proposition p, add a transition from the state s_i to s labeled with the proposition p ∧ θ;

b) A loop over the added state s_i labeled with the propositional formula θ;

c) A loop over the added state s_f labeled with the propositional formula θ;

d) A transition from the state s_i to the state s_f labeled with the proposition θ.

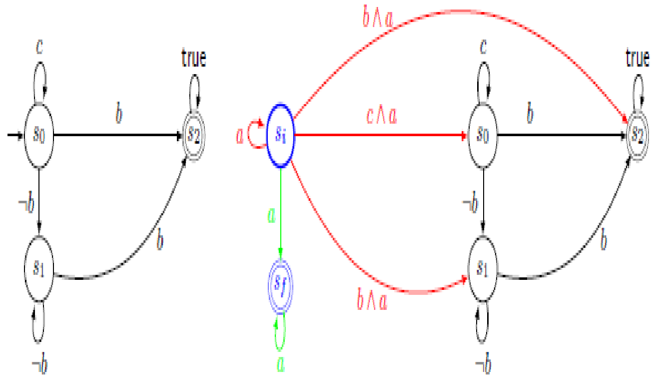
All other transitions of A_φ, as well as the accept states, remain unchanged.

Definition 9 (φ R Θ **automaton**). *Let Θ be a propositional formula and φ be an LTL flat formulæ. Let $A_\varphi = (S_\varphi, s^0_\varphi, F_\varphi, \Sigma, \delta_\varphi)$ be the automaton associated to φ. The* **automaton** *$A_\psi = (S_\psi, s^0_\psi, F_\psi, \Sigma, \delta_\psi)$ associated to $\psi = \varphi \mathrm{~R~} \Theta$ is defined as follows:*

- $S_\psi = \{s_i, s_f\} \cup S_\varphi$
- $s^0_\psi = s_i$, $F_\psi = F_\varphi \cup \{s_f\}$
- *The transition function δ is defined as follows:*

$$\delta_\psi(s,p) = \begin{cases} \delta_\varphi(s,p) \text{ if } s \in S_\varphi \ (A_\varphi \text{ transitions}) \\ \delta_\varphi(s^0_\varphi, p\prime) \text{ if } s = s_i \text{ and} \\ \qquad p = \Theta \wedge p\prime \ (\text{Connection } s_i \text{ to initial state of } A_\varphi) \\ \{s_i, s_f\} \text{ if } s = s_i \text{ and } p = \Theta \ (\text{Loop over } s_i \text{ or connection } s_i \text{ to } s_f) \\ \{s_f\} \text{ if } s = s_f \text{ and } p = \Theta \ (\text{Loop over } s_f) \end{cases}$$

Example: Figure 4 illustrates the composition definition of φ *R* θ. **Figure 4a** shows the Büchi automaton associated to the formula c *U* ◊b. To construct the Büchi automaton associated to the flat LTL formula ((c *U* ◊b) *R* a), we add a state s_i that we consider as the only initial state and a state s_f that we consider as a final state. We add a loop labelled with the atomic proposition a over the two added states. Finally, for each transition outgoing from the initial state of the automaton φ with label l and goes to state s, we add a transition from the added state s_i to the state s with a label (l ∧ a). We also add a transition labelled with a from the state s_i to the state s_f.



(a) c U ◊b (b) (c U ◊b) R a

Figure 4: Example of composition: φ *R θ*

Theorem 5. *Let* $\psi = \varphi$ R Θ *be a LTL formulæ and* A_φ *be the büchi automaton equivalent to* φ*. Let* A_ψ *be the LTL automaton built according to Definition 9. Then,* Words(φ R Θ) $= \mathcal{L}_\omega(A_\psi)$.

5. Finite syntax tree of flat LTL formula

A flat LTL formula φ can be transformed into a tree containing all the information about the possible sub-formula of φ. It will form the cornerstone of the construction of Büchi automata from flat LTL formula. We assume that our flat LTL formula are fully parenthesized and we show how to build the finite syntax tree, named *FST*(φ), algorithmically for a flat LTL formula φ. This tree can be thought of as a data structure representing the sub-formula after a finite breaking up the formula into a list of tokens. We distinguish four kinds of tokens: left brackets "(", right brackets ")", FLTL operators and propositional variables. FLTL operators represent the internal nodes of our tree while the propositional variables represent the leaf nodes. Our algorithm to build *FST*(φ) is[3] inspired from [5] and uses a stack for operators and a stack for propositional variables, and consists of the following rules:

a) If the current token is a left bracket "(" (*i.e.* we are reading a new sub-formula), push it on the operator stack;

b) If the current token is a operator (*i.e.* in {'∧', '∨', '*X*', '*U*', '◊', '*R*', '¬' }), push it on the operator stack;

c) If the current token is a propositional variable p, create a tree with single node whose the value is p and push the created tree on the variable stack;

d) If the current token is a right bracket ")" (*i.e.* we have just finished reading a sub-formula), pop operators off the operator stack while this operator is not a left bracket. If the popped operator is an unary operator, pop one tree variable from variable stack and create new tree whose the root is the popped operator and it is only child is the popped tree. If the popped operator is a binary operator, pop two tree variables from variable stack and create new tree whose the root is the popped operator and its right child the first popped tree and its left child the second popped tree. If no left bracket is found during popping the variable stack, throw a mismatched bracket expression. Otherwise, pop found left bracket from the operator stack;

e) At the end of reading expression tokens, pop all operators off the operator stack and for each popped operator:

- If the popped operator is an unary operator, pop one tree variable from variable stack and create new tree whose the root is the popped operator and it is only child is the popped tree. Then, push the created tree on the variable stack;
- If the popped operator is a binary operator, pop two tree variables from variable stack and create new tree whose the root is the popped operator and its right child the first popped tree and its left child the second popped tree. Then, push the created tree on the variable stack;
- If the popped operator is left or right bracket, throw an unbalanced left brackets.

Hence, our mechanism of creating *FST*(φ) can be described by the algorithm illustrated in **Figure 5.**

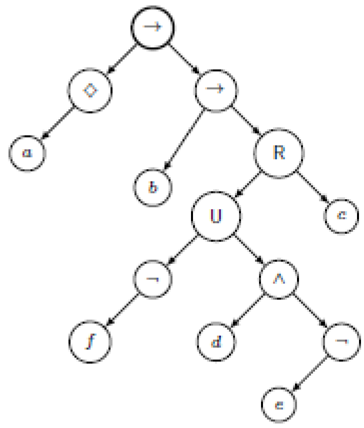
[3] Shunting-yard algorithm proposed by *Djikstra* and used to parse mathematical expressions specified in infix notation.

```
Input: a positive flat LTL formulæ φ  Output: the finite syntax tree
 FST(φ);
operatorStack ← createStack();
operandStack ← createTreeStack();
l ← split(φ);
for e ∈ l do
    if isSpace(e) then
        continue;
    else if leftBracket(e) or unary(e) or binary(e) then
        push(operatorStack, e);
    else if variable(e) then
        push (operandStack,createNode (e));
    else if rightBracket(e) then
        while !emptyStack (operatorStack) do
            popped ← pop (operatorStack);
            if unary (popped) then
                push (operandStack,addRight
                 (popped,pop(operandStack)));
            else if binary(popped) then
                push(stackOperand,addRightLeft (popped,pop
                 (stackOperand),pop (stackOperand),e);
            else
                break; //encountered a left bracket
        end
        if emptyStack (operatorStack) then
            throw Exception("Unbalanced right parentheses");
    else
        throw Exception(Unknown token);
end
while !emptyStack(operatorStack) do
    popped ← top (operatorStack);
    pop (operatorStack);
    if unary (popped) then
        push (operandStack,addRight (popped,pop(operandStack)));
    else if binary(popped) then
        push(stackOperand,addRightLeft (popped,pop
         (stackOperand),pop (stackOperand),e);
    else
        throw Exception("Unbalanced left parentheses");
end
if lenght (operandStack)=1 then
    return top (operandStack);
else
    throw Exception("Error in LTL expression");
```

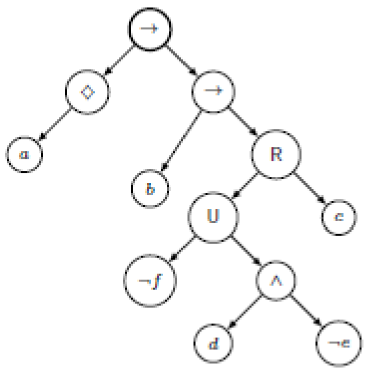
Figure 5: Building syntax tree for a FLTL formula

Example: Figure 6a shows the finite syntax tree *FST*(φ) generated for the FLTL expression:

$$\varphi = \Diamond a \rightarrow (b \rightarrow ((\neg f)\ U\ (d \wedge (\neg e))))\ R\ c.$$



(a) $FST(\varphi)$



(b) $FST(\varphi)$ with negations are pushed to leaves

Figure 6: Example of finite syntax tree

This finite syntax tree will be used to construct the Büchi automaton equivalent to a flat LTL formula φ in flat positive normal form. Since our algorithm takes as input a flat positive LTL formula, any node in the finite syntax tree labeled with the negation operator ¬ is certainly located directly before a leaf. For technical reasons, we merge the nodes labeled with ¬ with the leaf which directly follows in the finite syntax tree. **Figure 6b** illustrates the finite syntax tree presented in **Figure 6a** after pushing negations to leaves.

6. FROM FINITE SYNTAX TREE TO BUCHI AUTOMATA

Our algorithm to build Büchi automata from flat LTL formula is compositional in the sense that the final Büchi automaton is obtained by developing a sub-automaton for each sub-formula of the principal formula. Hence, the basic idea for developing the final automaton for a flat LTL formula φ is to explore *FST*(φ) in a pre-order traversal. That is to say, we visit the root node first, then recursively do a pre-order traversal of the left sub-tree, followed by a recursive pre-order traversal of the right sub-tree. The algorithm, illustrated in **Figure 7,** allows us to build a Büchi automaton from a finite syntax tree of a positive flat LTL formula T=*FST*(φ) and uses the following five functions:

a) **CreateBuchiProp(θ):** takes as input a propositional formula θ and returns the automaton as defined in Definition 6 (Section 4);

b) **CreateBuchiUnary(op, BA)**: takes as input an unary LTL formula (*i.e.* op ∈ {*X*, ◊}) and a Büchi automaton BA and returns a Büchi automaton defined according to definitions of ◊ and *X* given in Section 4;

c) **CreateBuchiBinary(op, BA_l,BA_r)**: that takes as input ∧ or ∨ operator and two Büchi automata BA_l and BA_r and returns a Büchi automaton defined according to definitions of ∧ and ∨ given in Section 2;

d) **BuchiUntil(θ, BA)**: that takes as input a propositional formula θ and a Büchi automaton BA and returns the automaton as defined in Definition 7 (Section 4);

e) **BuchiRelease(θ, BA):** that takes as input a propositional formula θ and a Büchi automaton BA and returns the automaton as defined in Definition 9 (Section 4).

```
Name : BuildBA
Input  : a finite syntax tree in which negations are pushed to
         leaves T = FST(φ)
Output: a büchi automaton A
A_φ ← CreateEmptyBA();
if IsEmpty(T) then
 | return CreateEmptyBA();
else if IsLeaf(T) then
 | return CreateBuchiProp(Root(T));
else
 | if Unary(Root(T)) then
 |  | return CreateBuchiUnary(Root(T), BuildBA(Left(T)));
 | else if Until(Root(T)) then
 |  | return BuchiUntil(Root(Left(T)), BuildBA(Right(T)));
 | else if Release(Root(T)) then
 |  | return BuchiRelease(Root(Right(T)), BuildBA(Left(T)));
 | else
 |  | return CreateBuchiBinary(Root(T),BuildBA(Left(T)),
 |  |  BuildBA(Right(T)));
end
```

Figure 7: building buchi automata: buildBA(T)

Theorem 6. *For any flat LTL formulæ $\varphi \in \mathcal{L}_f$, there exists an büchi automaton A_φ with $|A_\varphi| = O(|\varphi|)$.*

Theorem 7. *Let $FST(\psi)$ be the finite syntax tree of a flat LTL formulæ ψ and A_ψ is the büchi automaton generated by Algorithm 2, then: $Words(\psi) = \mathcal{L}_\omega(A_\psi)$*

7. CONCLUSION AND FUTURE WORK

This paper presents a compositional algorithm for generating Büchi automata from a fragment of LTL logic. We firstly proposed the grammar of this fragment and then built for each formula φ, its equivalent automata. We secondly showed how to compositionally build from Büchi automata associated to each sub-formula, the Büchi automaton of the target formula. We thirdly showed the complexity and the correctness of our Büchi automata generation method.

Future work: several research lines can be continued from the present work. First, some temporal operators such as always, precedes or since are not considered in this paper, as an immediate perspective, we will study how to include these operators in our LTL fragment. Second, in [6, 7], *Dwyer*'s presents a translational semantics for his pattern properties. Indeed, for each pattern property, he associates an equivalent LTL formula. In [17], the authors show how Büchi automata can be polynomially generated from pattern properties proposed by *Dwyer*. It will be interesting to study whether the translational semantics given by *Dwyer* is covered by our fragment. This will be done by comparing Büchi automata generated by the algorithm proposed in [17] with the Büchi automata generated by our algorithm.

REFERENCES

[1] C. Baier and J.P. Katoen. Principles of Model Checking (Representation and Mind Series). The MIT Press, 2008.

[2] E. Clarke, O. Grumberg, and K. Hamaguchi. Another look at LTL model checking. In Formal methods in system design, pages 415-427. Springer-Verlag, 1994.

[3] E. M. Clarke, E. A. Emerson, and A. P. Sistla. Automatic verification of finite state concurrent systems using temporal logic specifications. ACM Trans. Program. Lang. Syst., 8(2):244-263, April 1986.

[4] Edmund M. Clarke, Jr., Orna Grumberg, and Doron A. Peled. Model Checking. MIT Press, Cambridge, MA, USA, 1999.

[5] E.W. Dijkstra. An Algol 60 translator for the x1 and Making a translator for Algol 60. Technical Report 35, Mathematisch Centrum, Amsterdam, 1961.

[6] M.B. Dwyer, G.S. Avrunin, and J.C. Corbett. Property specification patterns for finite-state verification. In FMSP, pages 7- 15, 1998.

[7] M.B. Dwyer, G.S. Avrunin, and J.C. Corbett. Patterns in property specifications for finite-state verification. In Proceedings of the 21st International Conference on Software Programming, pages 411 - 420, 1999.

[8] P. Gastin and D. Oddoux. Fast LTL to Buchi automata translation. In Proceedings of the 13th International Conference on Computer Aided Verification (CAV'01), volume 2102 of LNCS, pages 53- 65, Paris, France, jully 2001. Springer.

[9] V. King, O. Kupferman, and M.Y. Vardi. On the Complexity of Parity Word Automata, pages 276 -286. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.

[10] M. Mukund. Finite-state automata on infinite inputs, 1996.

[11] A. Pnueli. The temporal logic of programs. In Proceedings of the 18th Annual Symposium on Foundations of Computer Science, SFCS '77, pages 46-57, Washington, DC, USA, 1977. IEEE Computer Society.

[12] J.P. Queille and J. Sifakis. Specification and verification of concurrent systems in cesar. In Proceedings of the 5th Colloquium on International Symposium on Programming, pages 337 - 351, London, UK, UK, 1982. Springer-Verlag.

[13] S. Safra. On the complexity of omega-automata. In 29th Annual Symposium on Foundations of Computer Science, White Plains, New York, USA, 24-26 October 1988, pages 319- 327, 1988.

[14] S. Safra. Complexity of Automata on Infinite Objects. PhD thesis, Weizmann Institute of Science, Rehovot, Israel, March 1989.

[15] A.P. Sistla and E.M. Clarke. The complexity of propositional linear temporal logics. J. ACM, 32(3):733- 749, july 1985.

[16] A.P Sistla, M.Y. Vardi, and P. Wolper. The complementation problem for Buchi automata with applications to temporal logic. In Automata, Languages and Programming, pages 465 - 474, Berlin, Heidelberg, 1985. Springer Berlin Heidelberg.

[17] S. Taha, J. Julliand, F. Dadeau, K. Castillos, and B. Kanso. A compositional automata-based semantics and preserving transformation rules for testing property patterns. Formal Asp. Comput., 27(4):641 - 664, 2015.

[18] M. Y. Vardi and P.Wolper. An automata-theoretic approach to automatic program verification. In Proc. 1st Symp. on Logic in Computer Science, pages 332 - 344, Cambridge, June 1986.

[19] M.Y. Vardi. Branching vs. linear time: Final showdown. In Proceedings of the 7th International Conference on Tools and Algorithms for the Construction and Analysis of Systems, TACAS 2001, pages 1 - 22, London, UK, 2001. Springer-Verlag.