An automatic generation of metro-like maps to display Flight Routes for Air Traffic Controllers: structure and color optimization

Christophe HURTER
DSNA/DTI R&D
IRIT
7, avenue Edouard. Belin
31055 Toulouse France
christophe.hurter@aviation-civile.gouv.fr

Mathieu SERRURIER
DSNA/DTI R&D
IRIT
7, avenue Edouard. Belin
31055 Toulouse France
serrurier@irit.fr

Roland ALONSO
DSNA/DTI R&D
IRIT
7, avenue Edouard. Belin
31055 Toulouse France
alonso@cena.fr

ABSTRACT

Aircraft must follow strict Air Traffic Control (ATC) rules. One of these rules is that aircraft have to fly over pre-defined Flight Routes (FR). Current ATC visualizations do not display FRs because they are numerous and run into each other, and thus spoil the visualization. The schematic views for metro maps are used to maximize the transmission of relevant information (lines, metro stops) of network visualization. In this paper, we will focus on two different issues. First, we show how we transposed mathematical constraints used to produce metro maps into the specific field of ATC. The view produced is a context compatible, 2D picture of a metro-like view for Air Traffic Control. Second, we propose to investigate the generation and placement of colors to be assigned to lines of the network. The first step is to find as many colors as lines of the network. These colors must be perceptually as distinct as possible, and available in the vocabulary of colors. The second step is to solve the NP-complete problem of the optimal assignment of these colors so that close lines have the most perceptively distant color. Finally, we assess the map produced through experimentation to validate its quality.

Author Keywords

Visualization, metro map, colors assignment, Air Traffic Controller.

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

I. INTRODUCTION

Fundamental research in visualization is concerned with the impact of presentation on visual perception and understanding [22] [23]. Current Air Traffic Control (ATC) environments employ complex visualization systems. These visualizations display large quantities of information that must be understandable with the minimum cognitive workload. As traffic increases together with safety levels, The ATC environment requires new kinds of visualizations.

The activity of Air Traffic Controllers (ATCos) consists in maintaining safe distances between aircraft by giving clearance to pilots (heading, speed, or altitude orders). The traffic is planned in advance: companies must request a flight plan from the regulatory authorities, which is translated into a mandatory Flight Route (FR) *i.e.* a sequence of beacons (named locations on the ground), with an associated time and altitude. A first task associated with the FRs is the a posteriori analysis of the traffic in order to determine the most conflicting FRs. Moreover, future ATC systems promote the use of FR as their main component. For instance, ATC regulators envisage a system where airline companies have to book time-slots along FRs. A natural visualization for this kind of task is the geographical view. However FRs are numerous and run into each other. A direct representation of the FRs is too complex and unusable. An efficient visualization of FRs would simplify their utilization.

In this context, metro map visualization appears to be a suitable solution. Metro maps are schematic drawings of the underlying geographical network that represents the different stations and lines of a metro system [6]. Automatic generation of metro maps is an active, ongoing research area [16] [20]. Most existing methods consist in optimizing the shape of the networks by considering the mathematical criteria the authors want to minimize (network density, straight lines...).

Our approach is divided into two parts: The geographical optimization and the color assignment of FRs.

In the first part of this paper, we study how to adapt existing metro map generation to the ATC context. Most existing metro maps are produced manually: while metro-map design is time-consuming, it is also limited by the number of metro lines, and by the fact that the metro network does not often change. On the contrary, a metro-like view of FRs must be generated automatically, as FRs are too numerous (more than 600 FRs in the French airspace) and change regularly (their life-time can be as low as 40 days, and adding a new flight route is cheaper and physically less challenging than creating a new metro line). In order to adapt the metro map generation, we compare differences between metro and FRs layouts. Then we define specific mathematical cost functions that measure the quality of a metro-like view of the FRs. These functions are finally used in an optimization algorithm to generate optimal configurations.

The second part of our paper concerns the color assignment of the lines of a metro map. The schematic views for metro maps are used to maximize the transmission of relevant information (lines, metro stops) of network visualizations. Automatic generation of metro maps focuses principally on the physical structure of the network, but less on the color choice which is, nevertheless, an accurate visual discrimination variable. The FR discrimination will increase if close FRs have distant perceptual colors. A random color assignment is not optimal for two reasons: all the colors do not have the same differenciation power and distant FRs are already visually separated and do not need the color differenciation. Hence, we use an algorithm of color generation and color assignment. Our goal is to produce a set of N discriminating colors to assign for N FRs. In order to facilitate the communication between users, we favor colors that can be named in the common color language. The color assignment is an NP complete problem; therefore we use simulated annealing to find an optimal solution [9].

The remainder of this paper is organized as follows. We first recall the state of the art of metro- like automatic generation. We present a description of a specific activity (Air Traffic Controllers with Flight Route management), and highlight differences with public transportation. We show how we adapted the algorithm of automatic metro-map generation by introducing new mathematical cost functions in an optimization process. Then we explain how we create a color set and how we assign it to FRs. Finally, we perform experimentation to assess the views produced.

II. *ATC/METRO MAP COMPARISON*

Metro maps are schematic drawings of the underlying geographical network that represents the different stations and metro lines of a metro system [6]. Ever since the first one was created in 1934 by Beck, the drawing rules have not changed: transport lines are straightened and restricted to horizontals, verticals and diagonals are depicted at 45 degrees (coined as *octilinear* in [16]). The scale in crowded downtown areas is larger than in the less dense suburbs in order to create more uniform distances between adjacent stations. In spite of all this distortion, the network topology and the relative position between metro stations must be retained. Metro maps have been tentatively applied to other classes of problems than metro traveling, such as understanding the structure of a thesis, or the sequence of teaching courses [15].

Automatic generation of metro maps is an active, ongoing research area, with no complete and general solution. The most frequently used methods are to optimize the shape of the networks by considering the mathematical criteria the authors want to maximize [16] [20]. The best results are performed by using meta-heuristic techniques such as simulated annealing. The criteria used are essentially based on the direction and bending of the line in order to generate metro maps where lines are as straight as possible and follow octilinear directions. Some methods are focused on specific criteria such as line crossing minimization [2], or path simplification [12]. A related problem is the automatic generation of directions [1]. Though not strictly a metro map, the authors have designed an activity-driven visualization, with multiple kinds of graphics and textual information.

ATCos manage air traffic in an area called a sector, (white area in Figure 1). Airways are the straight segments that planes have to follow (solid and dashed blue lines in Figure 1). Two beacons delimit an airway. A Flight Route is a sequence of airways (example: the route displayed as a thick red line is made up of three segments, and passes over 2 beacons). FRs span multiple sectors, but ATCos only need to know the section of FRs that cross their sector. Each flight has an associated FR. FR are often re-used by companies, either for regular flights (5 per day between Paris and Toulouse for example), or by companies that compete on the same route. With the increase of the traffic some new tasks arise. The first one is the *a posteriori* analysis of traffic in order to determine the most conflicting Flight Routes and the densest sections (e.g. Flight per hour). The second is a future task for companies. It has been envisaged to ask the companies to book time slots along flight routes in order to optimize the traffic. For these two tasks a geographical representation of the sectors is the most appropriate solution However, direct representation of FRs is note suitable due to their number and to their density in some areas. The exact location of beacons is not compulsory since this task does not require real-time traffic management. Users need an easy-to-read view of flight routes (with a low cognitive workload). In this way, the view produced can be used to understand the structure of the FRs, and to display the traffic density (e.g. by adding a size proportional to the traffic of each section of the FR).

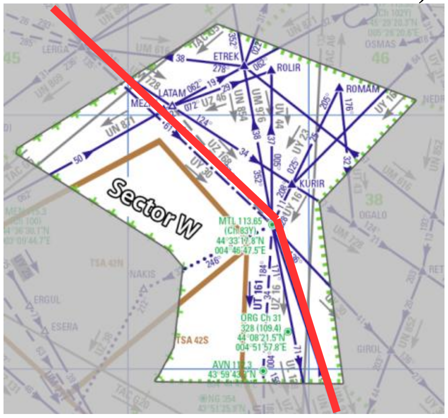

Figure 1 : a sector (white zone), and a route (red line)

By comparing problems and questions in public transportation and ATC, we can see how related they are. A common theory in the field of cognitive psychology is that people think of a path as a sequence of steps [1] [5]. For ATCo, these steps are the beacons [11]. We think that metro map-like visualization is a promising visualization for ATC. However, the metro map drawing rules used for public transportation cannot directly be applied to the ATC field. The specificity of ATC lies in the following aspects:

- Subway stations vs. beacons: stations are spread out to maximize the servicing of an area by public transportation, whereas beacons are very close together. Furthermore, beacon layout is much more heterogeneous than station layout.
- The regularity of station spacing: for effective service purposes, metro stations are inherently spaced regularly along a line. Thus, the constraint that equalizes distances between stations modifies only slightly the topology of the network. On the contrary

beacons are not spaced regularly along the FR. A constraint that equalizes distances between beacons is not desirable, as it would alter the topology of the network to an excessive extent.

- There are more “interchange” beacons than “intermediate” beacons, as a lot of FRs intersect at the same beacon. Metro stations seldom connect more than five lines.
- Flight Routes have a lot of crossings compared to subway lines. This is due to the general orientation of airways (East-West and South-North flows).
- Flight Routes are numerous and they often share the same sections. When FRs share the same altitude, it is as if metro lines shared the same rails.

These differences lead us to define new specific mathematical rules (cost functions) to produce a metro-like view of Flight Routes.

III. GRAPH OPTIMIZATION

Currently, the visualization of a sector and its FRs is produced manually. There is no significant previous work in the ATC field that tries to automatically produce visualizations of schematic FRs.

The graph of the representation of a sector is constructed from a set of beacons and FRs. A beacon has a name, a center position and a span value that describes the distance between each FR that connects with it. We also need the input and output location of the FR at each given beacon. The W sector (Figure 1) is composed of 19 beacons and 24 FRs. Figure 5 and Figure 7 are representations of FRs before the geographical optimization. As we can see, numerous beacons overlap, and FRs cross so much that it is difficult to follow a specific FR.

Each of the above design constraints can be transposed into mathematical cost functions. Cost functions can be integrated into an optimization algorithm to obtain an optimal metro-like view with respect to these visual constraints. For ATC metro maps, we consider three major constraints:

1/ *Topological closeness*: it indicates the relative closeness of represented beacons to their original position,

2/ *Density*: it indicates the number of neighboring beacons,

3/ C*rossing:* it indicates the number of crossing FRs.

Topological closeness is inherited from the classical metro maps. *Density* and c*rossing* are more specific to the ATC field. *Linearity* (straight line) and *octolinearity* (rounded up angle values) are not very pertinent in the ATC context since the direction of a FR changes at each beacon.

A. Topological closeness: Cost$_{dist}$

The visual location of each beacon has to be as close as possible to the real beacon location. If the view produced significantly alters the topology, this may create a gap between the mental and physical representation of the ATC sector. In metro maps, metro stations are often inherently evenly spaced, and topological closeness is usually respected even if not constrained. Here, Cost$_{dist}$ exponentially increases with the distance between the real beacon location and its representation. Cost$_{dist}$ is defined as follows:

$$cost_{dist} = -\frac{\sum_{b\in B} erf\left(\frac{d(b,b')-MAX_{dist}}{C_{dist}}\right)}{|B|}$$

Where B is the set of beacons, b and b' are respectively the initial and the modified position of the beacon. D(b,b’) is the Euclidean distance between b and b’. |B| is the cardinal of B. Erf is the Gauss error function (see figure 4). C$_{dist}$ and MAX$_{dist}$ are constants respectively fixed at 200 and 30. Roughly speaking C$_{dist}$ corresponds to the displacement that is accepted without a penalty and MAX$_{dist}$ is the ratio of the penalty increase when the accepted displacement is reached.

B. Density: Cost$_{den}$

As pointed out in the introduction, one major ATC specificity is the high density of beacons in some areas. This density is incompatible with a clear representation. With a given beacon, Cost$_{den}$ increases with the proximity of its neighbors.

$$cost_{den} = \frac{\sum_{b_i,b_j\in B,i\neq j} erf\left(\frac{d(b_i,b_j)-MAX_{den}}{C_{den}}\right)}{|B|^2}$$

Where C$_{den}$ and MAX$_{den}$ are constants respectively fixed to 80 and 6. Roughly speaking MAX$_{den}$ is the minimum distance needed between the beacons. b_i, $b_j \in B, i \neq j$ is the set of all the different beacon couples.

C. Crossing: Cost$_{cross}$

Once again, a clear view requires a minimum of overlapping FRs. Cost$_{cross}$ is the rate of crossing segments. This issue has already been addressed in classical metro-map drawing, but in our context this criteria need a stronger optimization.

$$cost_{cross} = -\frac{\#crossingedges}{\#edges^2}$$

IV. ALGORITHM

According to the above considerations, an efficient view must minimize all the previously listed costs. This cost minimization is a complex problem since all these costs are interdependent. Thus this optimization problem is intractable using a deterministic algorithm. Nowadays, the best automatic metro maps are produced with meta-heuristic methods. In this paper, we use a method that is deeply rooted in optimization technique. Simulated Annealing (SA) is a powerful, general, optimization technique [9]. SA is computationally costly, but this remains acceptable since the time span to produce a metro-like view is considerably shorter than the lifetime of the visualized FR.

Cooling parameters of Simulated Annealing is tuned to 0.9997. The algorithm stops when it converges, i.e. 100 steps without increasing the function. The optimization of the sector W took approximately 3 hours on a Mac pro under Linux.

We minimize the following costs:

$$Cost_{phase1} = C*(Cost_{den}+Cost_{dist})+ Cost_{cross}$$

Where C is a constant fixed empirically at 5. This minimization is performed by the SA. The SA is based on the notion of neighbor which corresponds, roughly speaking, to a small variation of the current state. In our case, a neighbor of a current visualization is obtained by moving the location of a beacon, changing the span of a beacon or switching the FR order of a beacon. The visualization obtained is the best trade-off between $Cost_{dist}$ and $Cost_{den}$ which are antagonist. Parameters used in $Cost_{dist}$ and $Cost_{den}$ create an area where these costs do not change (e.g. when beacons are not close and not too far from their initial location). In this area, the configuration with the smaller number of crossings is computed.

The initial view of the W sector is presented in Figure 5 and in Figure 7. The geographically optimized view of the W sector is presented in Figure 6 and in Figure 8. We can see that the view is well spaced out. The number of crossings has been lowered. The remaining crossing segments would require too much modification of the topology. Having optimized the geometry of the metro maps, we will now choose and assign the colors associated to each FR.

V. *PLACEMENT AND MAP GENERATION ASSESSMENT*

Most metro maps are created by a graphic designer who chooses the appropriate design and color of each metro line. The designer creates a harmonious color set with graphic techniques [7] [18] [19] [24].

In this section we propose a generic method to create color sets based on designers' techniques and previous works [8] [10] [13]. Colors must be as distinguishable as possible. The generation of distinguishable colors in the RGB space is not suitable since Euclidean distance in this space cannot be considered as a perceptual distance. Indeed, color differences in RGB space are not homogenous with the color difference perception of human beings. The only way to set up colors and assess their perceptive distances is to use the findings of the CIE (International Commission of Illumination) [4]. The CIE built, with empirical methods, models that describe the chromatic colors, taking into account human visual perception. The CIE L*a*b* model (CIELAB) is a color space that is homogeneous with human perception. The Euclidean distance between two colors, in the L*a*b* space, can be computed with de Deltae 2000 formula. The LCHab color space is the result of the transformation of Lab space in a cylindrical coordinate system. This color space uses three intuitive dimensions: Luminosity, Chroma (i.e., saturation) and Hue. L, C, H dimensions are orthogonal and perceptually uniform. However, the use of the CIELAB model is limited to the RGB gamut (actual displayable colors) of the output device.

With the CIE LCHab, we can easily compute any color set regularly spaced on the circular Hue axis. The division of the hue axis is the cornerstone of the generation of perceptually separated color classes. It is also possible to compute color distance with the Deltae method, but this is not sufficient. We also want colors with everyday language semantics (this constraint will help users to designate colors and then the routes orally). Constant value of luminosity and hue do not automatically produce colors with a correspondence in the language. To address this issue, we based the extraction of color with the "named colors" experimentations [3] [17]. Finally, we use the Munsell tables [14] to extract, with a given luminosity, the available Hue in the « named color » space.

```
createRGBColorset (numColors, colorModel, ICCprofil, startAngle,extend)
{
   colorModel = 'LCHab' if not defined
   ICCprofil = 'sRGB' if not defined
   startAngle = 0.0 if not defined
   extend = 360.0 if not defined

   stepAngle = extend / numColors
   RGBcolors = array of numColors color

   foreach  $i \in \{1 . . numColors\}$
   {
         newH = startAngle + stepAngle * i
         newL = namedColorLum(newH,colorModel)
         newC = maxChromaInRGBgamut(newH,newL,colorModel,ICCprofil)
         RGBcolors[i] =  LCHcolor2RGB([newL,newC,newH], colorModel,ICCprofil)
   }
   return RGBcolors
}
```

The pseudo code presents the extraction of an RGB color set with the number of colors to produce, the color model to use (by default LCHab), the ICC profile of the output device (by default sRGB), the starting angle of the Hue, and the available range. The function namedColorLum returns the best luminosity value for a given hue. The function maxChromaInRGBgamut computes the maximum Chroma (i.e., saturation) in the gamut. Finally, the function LCHcolor2RGB converts the RGB value of an LCH color with the ICC profile.

This algorithm could produce a countless number of colors, but in practice, a sector contains of maximum of 30 FRs and their corresponding color.

A. *Colors assignments*

With a metro map composed of N routes, we can generate, with our color generation algorithm, N visually separable colors. But N! available choices do exist to assign them to the N routes. We need to se up a numerical value that corresponds to a quality level of this color assignment. We want to assign to close routes the most distinguishable color. To compute the color distance we use the $Delta_e$ 2000 formula.

In order to compute the distance between lines, we consider lines as discreet points regularly spaced:

$$\text{i.e. } l_1 = \{ Pt_{11},\dots Pt_{1m}\}.$$

The standard distance between two lines is the minimum distance that separates two points of each line. This distance is not suitable regarding the visual perception of distance. For instance, two crossing lines have, with this distance computation, a distance equal to zero, whereas perceptually they are very distinct. Therefore, we define a new distance computation which is close to the perceptual distance between lines:

$$l_1 = \{ Pt_{11},\dots Pt_{1m}\} \text{ et } l_2 = \{ Pt_{21},\dots Pt_{2q}\} \text{ with } m< q$$

$$dist\ (l_1, l_2) = \frac{\left(\sum_{i=1}^{m} \min(d(\ Pt_{1i}\ ,\ \ Pt_{2j}), j = 1 \rightarrow q\right)}{m}$$

In this formula, d is the Euclidean distance between two points. Thus, the distance between the line l1 and l2 is the sum of the minimum distances between each point of l2 and each point of l1. Finally, we normalize this distance by dividing it by the biggest distance between the lines.

A color assignment is a couple of a line and color couple (l_i, C_i). We define the quality of a color assignment as following:

$$qual = \sum_{i,j,i<j}^{N} Delta_e\left(C_{i,}C_j\right) \times 2\ \times (1 - \frac{1}{1 + e^{-dist\ (l_i,l_j)}})$$

This value corresponds to a weighting mean of $Delta_e$.

The weight of each $Delta_e$ depends of the distance between lines. We apply a sigmoid curve to this distance in such a way that its weight tends towards 1 when the distance tends toward 0 (the case of two close lines), and towards 0 when the distance tends towards 1 (case of distant lines). Thus, the weight of the $Delta_e$ increases when the distance between lines decreases.

Finding the optimal assignment between the N! assignments is an NP complete problem. No deterministic methods can solve it in a reasonable time span. We will therefore use the simulated annealing method which is very efficient for this kind of problem. In our case, we try to optimize the color assignment. Colors are computed beforehand. The current state is a group a couples [color, line], a neighbor is computed by switching the color of two couples. Cooling parameters of simulated annealing are tuned to 0.9997. The algorithm stops when it converges, i.e. 100 steps without increasing the function.

B. *Optimal or worst color assignment*

The optimal view is the Figure 8: it corresponds to optimized geographical positions and an optimized crossing with an optimized color assignment.

We also produced maps with the worst color assignment. Therefore we use the above optimization algorithm with the opposite of the quality function. Thus, the view produced displays close lines with close perceptual colors.

We observe, especially in Figure 8, that the opposite of the cost function produces assignments where close FRs have perceptually close colors. This creates a rainbow effect through FRs. On the contrary, with the original cost function, close FRs are discriminated by colors as expected.

In the next section we assess the views produced.

VI. Experiments

Firstly, we performed a qualitative evaluation by showing the produced metro-maps produced to Air Traffic Controllers. We assessed the extent to which the corresponding sector was easier to recognize and if the view produced was clearer than the original one. As a first result, the optimized geographical views are clearer than the original view and are easier to recognize.

Secondly, we performed a quantitative evaluation to assess the quality of the metro maps produced.

In this experiment, we selected nineteen participants without sight problems (color blind). They were seated facing an 18” video monitor located at a distance of 0.80 m from their head.

The protocol simulates a real use case with a FR metro-like map: users have to accurately select a specific section of a Flight Route as fast as possible.

Two different color assignments have been tested:

- Optimal color assignment: close FR have distant color,
- Worst color assignment: close FR have close color.

We justify our worst color choice (i.e. non discriminating color assignment) as compared to a random color placement, because random can, with serendipity, create a good color assignment (strong discriminating power).

Two different geographical topologies have been tested:

- No geographical optimization, which corresponds to the initial view without beacon spacing and with tangled FR,
- Geographical optimization, which corresponds to the optimized view, with beacon spacing, and FR crossing optimization (few crossing FR).

We created metro maps of five ATC sectors. Each sector produces four different metro maps:

- No geographical optimization with the worst color assignment.
- No geographical optimization with the best color assignment.
- Geographical optimization with the worst color assignment.
- Geographical optimization with the best color assignment.

We also defined three difficulties levels for each section of FR:

- Level 1 (easy): the section is located in a non dense area without crossing lines,
- Level 2 (medium): the section is located in a dense area with less than 2 crossing sections,
- Level 3 (hard): the section is located in a dense area with more than 2 crossing lines.

Then, for each map with extracted 3 sections of each difficulties level (easy, medium, and hard) which represents 9 sections per map.

Each participant had to select 45 sections (5 maps, 9 sections per map) with two kinds of metro map:

- The worst color assignment with, and without, geographical optimization,

Or

- The best color assignment with, and without, geographical optimization.

Each participant performed 90 randomized tries (45 sections with two conditions). The number of maps and the kinds of map (worst colors or optimized color assignment) reduced the learning effect. We recorded the time needed to select a section of a FR (this time record starts when the requested color and the blinking beacons appear) and the correctness of the answer (we counted the number of right or wrong answers).

A. *Procedure*

In the view produced, a horizontal straight line (at the top left corner of the view) indicates the requested color (Figure 2). Two blinking beacons indicate the requested section (Figure 2). When the user flies over a section of a FR with the mouse pointer, the fly-over section oscillates indicating that this section can be selected by pressing the mouse (Figure 2).

For each try, the mouse pointer is located over the requested color position (the horizontal straight line); hence the user can memorize the color. The two blinking beacons use the peattentive effect [21], so the user immediately sees which area must be investigated. Finally, the user has to select the correct section by choosing the requested color between the blinking beacons. The fly-over section is animated; other visual designs were envisaged but they created color distortions: wider line sections, blinking colors.

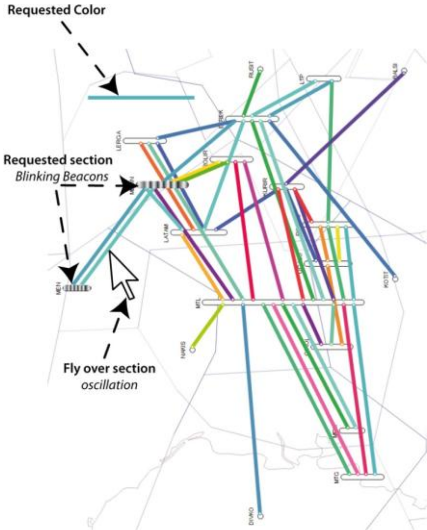

Figure 2 : The User's task is to select the section of a Flight Route between two blinking beacons with a requested color.

B. *Results*

All dependent measures were separately analyzed by performing an analysis of variance (ANOVA) with two groups of participants (no geographical optimization, and geographical optimization) and with repeated measurements for two factors: levels of difficulty (easy, medium, and hard) and color assignments (best and worst color).

Concerning the selection time, there was a significant effect regarding the level of difficulty F (2,28)= 45.77 p <.05; the greater the level of difficulty , the longer the time movement (Figure 3).

In addition, there is an interaction between the level of difficulty and the geographical condition (with or without geographical optimization), F (2,28)= 4.33, p <.05 (Figure 3). The time movement increase, due to the level of difficulty, is less important with the geographical optimized view than with the non geographical optimized one (Fig. 4). Therefore, the geographical optimized view is more efficient (smaller selection time) than the non geographical optimized one.

The analysis of the error number yielded a main effect with the level of difficulty F (2,28)= 8.79 p<.05: The number of errors increases when the level of difficulty increases.

In addition, there is an interaction between the level of difficulty and the color assignment conditions (best or worst), F (2,28)= 40.79 p<.05 (Figure 4). The error number increase, due to the level of difficulty, is less with the best color assignment than with the worst one (Figure 4). Therefore, the good color assigned view is more efficient (smaller errors number) than the worst color assigned one.

As expected, the two proposed enhancements (geographical optimization and best color assignments) improve performance in the select a section of a FR.

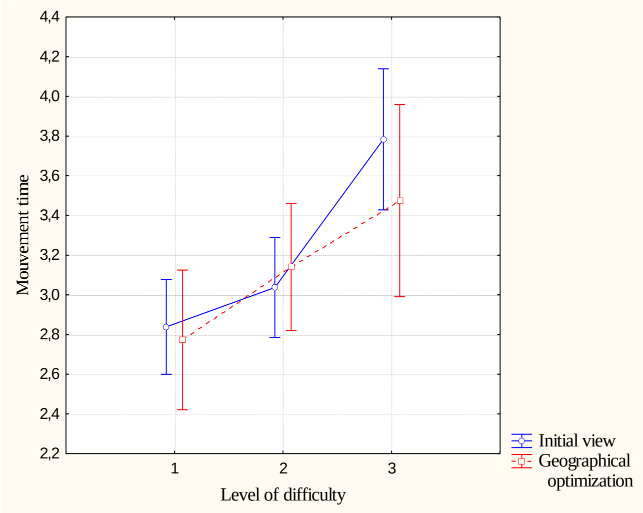

Figure 3 : Movement time as a function of the level of difficulty and geographical conditions.

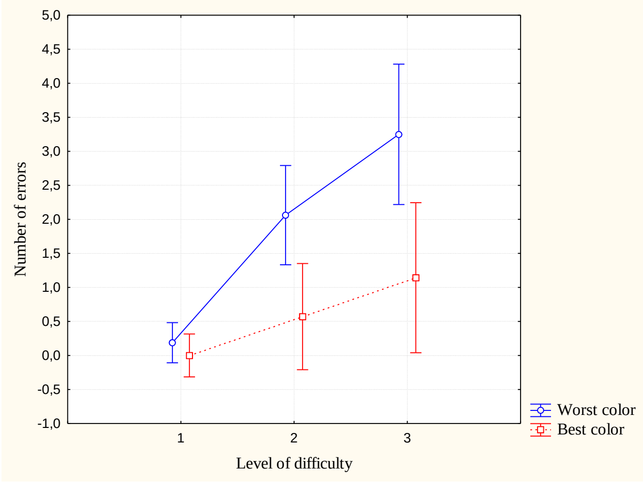

Figure 4 : Number of errors in terms of the level of difficulty and color conditions.

VII. CONCLUSION

In this paper, we address a new issue of metro map layout. We present a complete method to produce an efficient layout of aircraft Flight Routes in the ATC context. We turn ATC specific visual constraints into mathematical formulations. The simulated annealing algorithm, with these adapted cost functions, and optimizations, produces visualizations which fulfil the defined constraints. As we use an automatic process, the method allows the generation of metro-like views for any ATC sector. The automatic generation process is a great asset when sectors or FRs are updated.

We also studied the generation and the placement of colors on metro maps in order to enhance the readability of the view. This issue hasn't been extensively studied until now. We proposed a method to generate colors that takes into account their semantics and their perceptual distance with respect to other colors. We defined a cost function that measures the quality of a placement of color on a metro map. This cost function is integrated into an optimization process in order to obtain an optimal view.

We validated the produced views with qualitative and quantitative assessments. The geographical optimizations with optimized color views are clearer than the original view, and easier to recognize. The views produced improve the accuracy and speed up the selection of a specific section of a FR.

The proposed method is not specific to the ATC context and can be used for any metro- like visualization.

We plan to use the produced views produced with Air Traffic Controllers. Two main applications are envisaged. Firstly, we plan to add the traffic density of each FR the view. As a design choice, we plan to use the width of the line or its luminosity. Secondly, we will conduct a design study to use these maps to help companies to reserve time-slots on FR.

VIII. References

1. Agrawala, M. and Stolte, C. 2001. Rendering effective route maps: improving usability through generalization. In Proceedings of the 28th Annual Conference on Computer Graphics and interactive Techniques SIGGRAPH '01. ACM, New York, NY, 241-249.
2. Bekos M., Kaufmann M., Potika K., Symvonis A. : Line Crossing Minimization on Metro Maps. Proc. of Graph Drawing 2007. 231-242.
3. Berlin, B., Kay, P., Basic color terms: their uni-versality and evolution. Berkeley; Oxford: Univer-sity of California Press (1969) pg. 196.
4. A Color Appearance Model for Colour Manage-ment Systems: CIE CAM 2002, CIE 159, (2004)
5. Denis M.: The description of routes: A cognitive approach to the production of spatial discourse. Cahiers de Psychologie Cognitive, 1997.
6. Garland K.: Mr Beck's Underground Map, Capital Transportation Publishing, 1994.
7. Ishizaki, S. 1995. Color adaptive graphics: what you see in your color palette isn't what you get! In Conference Companion on Human Factors in Computing Systems (Denver, Colorado, United States, May 07 - 11, 1995). I. Katz, R. Mack, and L. Marks, Eds. CHI '95. ACM, New York, NY, 300-301.
8. Harrower, M. and Brewer, C.A., ColorBrewer.org: an online tool for selecting colour schemes for maps. Cartogr. J. v40 i1. 27-37.
9. Kirkpatrick S., Gelatt C. D., Vecchi M. P.: Optimization by simulated annealing. Science 220, 671-680, 1983.
10. Levkowitz, H. and Herman, G. T. 1992. Color Scales for Image Data. IEEE Comput. Graph. Appl. 12, 1 (Jan. 1992), 72-80.
11. Mackay W., Fayard A-L, Frobert, L., Médini L. : Reinventing the Familiar: Exploring an Augmented Reality Design Space for Air Traffic Control. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems 1998. 558-565.
12. Merrick D., Gudmundsson J.: Path Simplification for Metro Map Layout. Proc in Graph Drawing 2006. 258-269.
13. Moretti, G.S. and Lyons, P.J. Controlling the complexity of grouped items in colour interfaces. Proceedings of the 6th ACM SIGCHI New Zealand chapter's international conference on Computer-human interaction: making CHI natural, ACM (2005), 19-23.
14. Munsell, A. H. (1912). “A Pigment color System and Notation”. The American Journal of Psychol-ogy.
15. Nesbitt, K.V.: Getting to more abstract places using the metro map metaphor. Proceedings of the Information Visualisation International Conference, IV 2004, IEEE Computer Society, pp488–493, 2004.
16. Nöllenburg, N. and Wolf, A.: A mixed-integer program for drawing high-quality metro maps. Proc. 13th Internat. Sympos. Graph Drawing (GD'05), volume 3843 of Lecture Notes in Computer Science, pages 321-333. Springer-Verlag, 2006.
17. Párraga, A. Benavente, R Vanrell, M. and Baldrich, R. Modelling inter-colour regions of Colour Naming Space. In Proceedings on Color in Graphics, Imaging and Vision (CGIV'08).
18. Rheingans, P., Task-based Color Scale Design. Proceedings of Applied Image and Pattern Recognition '99, SPIE, pp. 35-43.
19. Robertson, P. and O'Callaghan, J. 1986. The Generation of Color Sequences for Univariate and Bivariate Mapping. IEEE Comput. Graph. Appl. 6, 2 (Feb. 1986), 24-32.
20. Stott, J. M. and Rodgers, P.: Metro Map Layout Using Multicriteria Optimization. In Proceedings of the Information Visualisation International Conference. IV. IEEE Computer Society, pp355-362, 2004.
21. Treisman, A., Preattentive Processing in Vision, Computer Vision, Graphics, and Image Processing, 31(2):156-177, August 1985.
22. Tufte, E. R. Visual Explanation. Graphics press, Cheshire 1997.
23. Ware C., Information Visualization, perception for design, Morgan Kaufmann, 2002.
24. Ware, C. 1988. Color Sequences for Univariate Maps: Theory, Experiments and Principles. IEEE Comput. Graph. Appl. 8, 5 (Sep. 1988), 41-49.

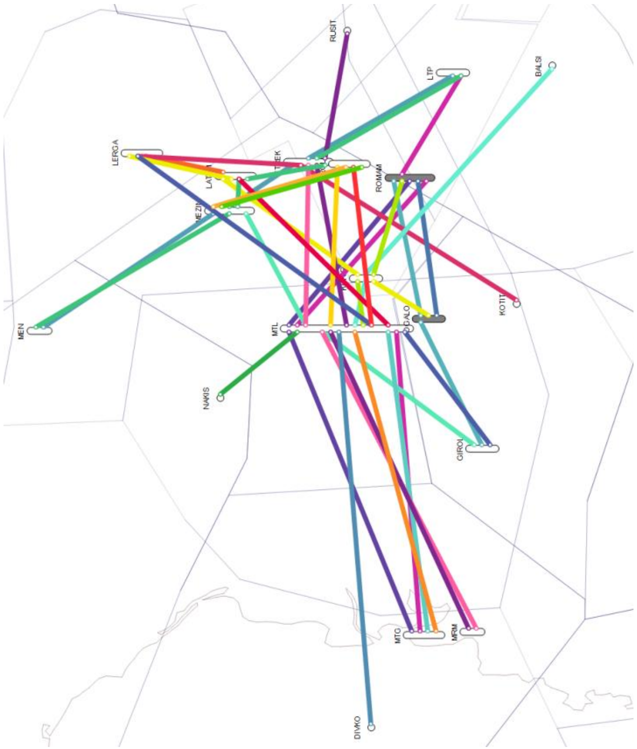

Figure 5 : View with no geographical optimization and with the best color assignment.

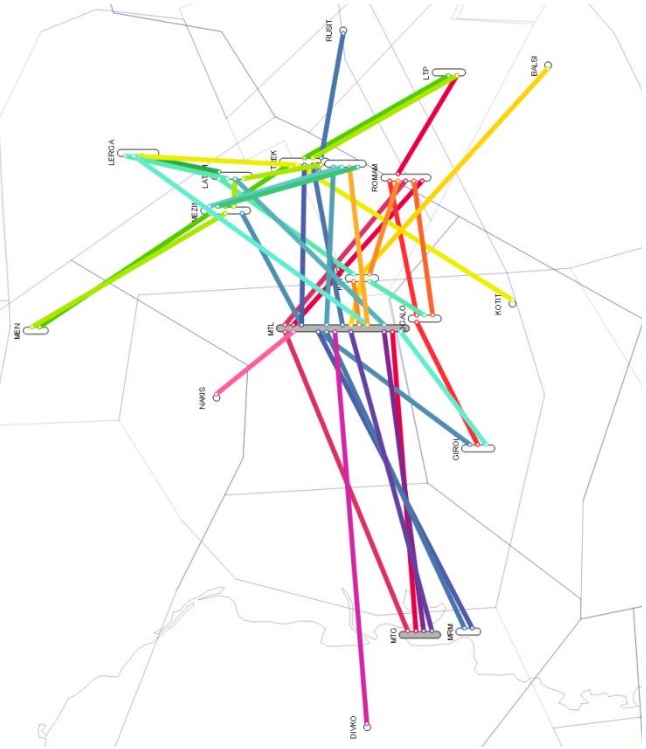

Figure 7 : View with no geographical optimization and with the worst color assignment. This view is the less efficient one.

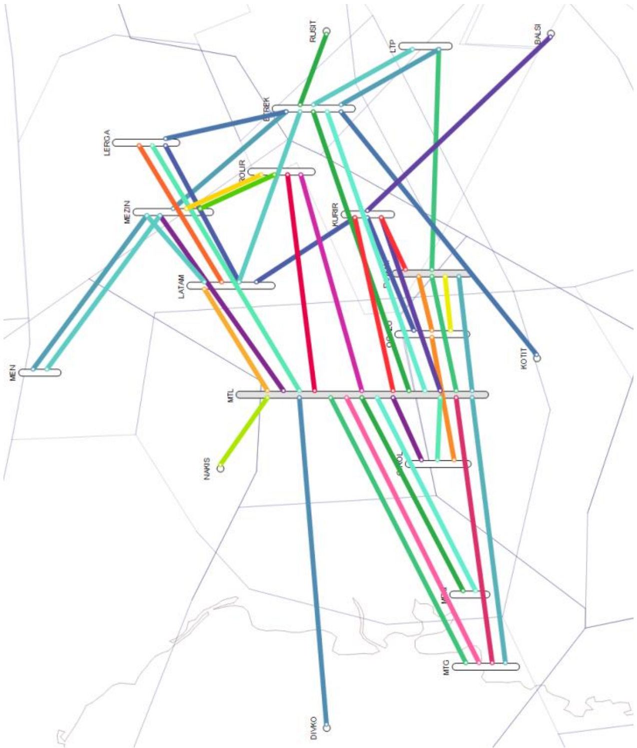

Figure 6 : View with geographical optimization and with the best color assignment. This view is the most efficient one.

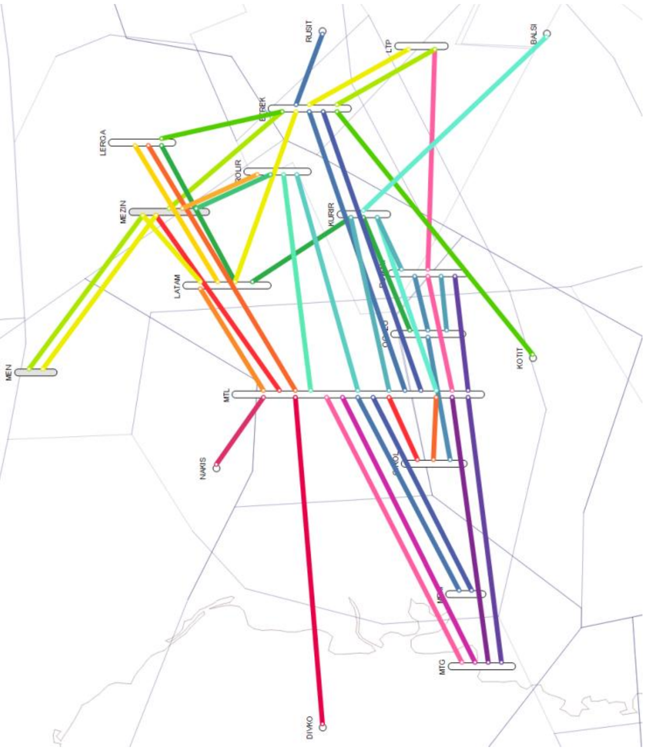

Figure 8 : View with geographical optimization and with the worst color assignment.