Stable Marriages to Globally-Satisfying and Equitable Image Matching

Nikom SUVONVORN, Bertrand ZAVIDOVIQUE

PARIS XI University I.E.F. Bat. 220, 91405 ORSAY, FRANCE

Abstract

This paper deals with designing algorithms based on the "stable marriages" paradigm, for them to take additional constraints into account. First algorithms scan the so-called "marriage table" to optimize "global satisfaction" and "equity" over all couples. Then, we introduce an hybrid version between the Gale-Shappley classical algorithm and ours. Results on thousands of populations, up to 200 person large, are systematically evaluated. These algorithms progressively improve both satisfaction and equity but they do not guarantee complete stability. Thus, the *BZ* algorithm is made to blow blockages out. It still produces about 5% instable results in average. After a case study that sorts blocking situations into 4 types, we explain how to resolve instability in forcing blocking pairs to marry wrt. their type. The resulting *S*-procedure applies after every previous algorithm, and results are systematically compared to *Gale-Shapley's* on 1000 instances of a 200-person-large population. In conclusion, comparative examples of respective matching results from the algorithms are given for illustration of possible applications to registration, stereo reconstruction or motion finding.

Key words: Stable marriages, Matching algorithm

1 Introduction

Matching is a crucial step towards accurate understanding of the world by machines. Over the years, many matching methods have been studied, ranging from various kinds of mere correlation up to relaxation or dynamic programming. When applying to robot vision, *efficient matching* methods are needed in all areas of Image Processing, from Segmentation − e.g. motion detection or 3D stereo-reconstruction − to actual Pattern Recognition − e.g. model-fitting

Email address: nikom.suvonvorn@ief.u-psud.fr (Nikom SUVONVORN).

Preprint submitted to Elsevier Science *21 March 2006*

or classification. *Efficiency* then gets multiple meanings and can address properties as different as easy data extraction and straight coding format, model simplicity, limited prior assumptions, robustness against ambiguities, conflict freedom etc., not to forget computability. Within that framework, general enough methods are still to be produced. We tested several, from Dynamic Warping on edge or region chain-codes [1][2], for instance, to more recently Hough Transform on level lines [3]. In the latter we use n-tuples of carefully coded level-line segments. We sort their set into several sub-populations according to the confidence in them, to build an efficient multi-pass voting process. Efficiency here is to improve scene reconstruction in the sense of "fighting ambiguities (e.g. repetitive patterns) at a limited enough computing expense", and this is achieved thanks to a reinforcement of stronger features by weaker ones along the passes. In a wider approach to segmentation that aimed at exhibiting nD image features in gathering (n-1)D ones [4] [5] − edges from points of interest, regions from edges, etc. − an interesting paradigm of optimization on bi-partite graphs was tested against Bayesian techniques and was shown to sustain comparison with them well. It is called the "stable marriage problem". The main interest is to guarantee no logical contradiction in the pairing process provided

- two populations are distinguished to be cross-matched item-to-item;
- each item sorts every member of the antagonist set in a so-called "preference list".

Efficiency in that case stresses disambiguation again, and we thought of trying the method to limit or skip multipass voting in the level line associating scheme. Here the targeted application is feature-pairing in robot vision, to stereo-reconstruction, motion-understanding, or object-recognition. The two populations are features extracted from respective images − left/right, previous/current, unknown/template − and there is an a-priori equal importance of items from both sides. Then, unlike in previously known algorithms where only stability matters, *fairness* (or *sex equality*) needed to be accounted for. Moreover, matching had to be satisfactory whichever the corresponding regions in an image pair, thus requiring a *global satisfaction.* That meant for the algorithm to guarantee the best balanced solution among the many possible ones. And of course *stability* of the matching had to be considered.

We take for a postulate that most matching in robot vision targets global fitting from local attraction. Thus, a reliable algorithm of stable marriages in such an application should fulfill three criteria: stability (i.e. no local questioning of more global associations), sex equality (i.e. local/global balance of the resemblance to matching), and global satisfaction (i.e. limited amount of local counter run).

The present paper deals with designing algorithms to produce marriages balancing between the three constraints without forgetting the complexity. Section 1 is devoted to a quick reminder on the stable marriages problem and the basic algorithm. Section 2 introduces a new representation of the instances of populations called the "marriage table" [5]. Section 3 explains various scans of the latter corresponding to different properties of the result. In section 4 we propose a novel implementation of an hybrid of our algorithms and the Gale-Shappley classics. Section 5 details the process to recover complete stability without loosing any satisfaction or fairness, but with a raise in complexity. The paper concludes with a sample and a qualitative comparison of results in image matching.

2 The classical algorithm

The algorithmic problem of stable marriages was first solved by Gale and Shapley [6]. In this problem, two finite sub-sets M and W of two respective populations, say men and women, have to match. Assume n is the number of elements, $M = \{m_1, m_2, ..., m_n\}$ and $W = \{w_1, w_2, ..., w_n\}$. Each element x creates its preference list $l(x)$ i.e. it sorts all members of the opposite sex from most to least preferred. A matching $\mathcal{M}$ is a one to one correspondence between men and women. If (m, w) is a matched pair in $\mathcal{M}$, we note $\mathcal{M}(m) = w$ and $\mathcal{M}(w) = m$ and ρ_m is the rank of m in the list of w (resp. ρ_w the rank of w in the list of m) . Man m and woman w form a blocking pair if (m, w) is not in $\mathcal{M}$ but m prefers w to $\mathcal{M}(m)$ and w prefers m to $\mathcal{M}(w)$. If there is no blocking pair, then the matching $\mathcal{M}$ is stable [7], [8]. Gale and Shapley proved that there is always at least one stable matching $\mathcal{M}$ whichever the instance $[M, W, \{l(m), l(w)\}]$. They proposed the algorithm of Gale-Shapley (GS) to find $\mathcal{M}$ with complexity $O(n^2)$: every man attempts to marry their best choice at each step; in case of competition over one woman, she is asked to choose, and the looser goes back to his list. Then women get their best choice among the men who selected them, which guarantees no blockage left behind if m prefers w to $\mathcal{M}(m)$ then w preferred $\mathcal{M}(w)$ to m, or they were never introduced i.e. $\mathcal{M}(m)$ was before w in $l(m)$.

Since then, this optimization problem was constantly popular in combinatorics from both theoretical [9], [10],[11], [12], [9] and practical points of view [13], [14], [15]. According to [10], the stable marriage problem was generalized in 4 directions: (i) stable marriage with complete list and total order, the case of Gale and Shapley; (ii) stable marriage with incomplete list and total order [16], (iii) stable marriage with complete list and indifference [17], and (iv) stable marriage with incomplete list and indifference [10]. Note that in machine vision applications case (i) is rare due to the local nature of segments to pair and the latent incompleteness of information from their features.

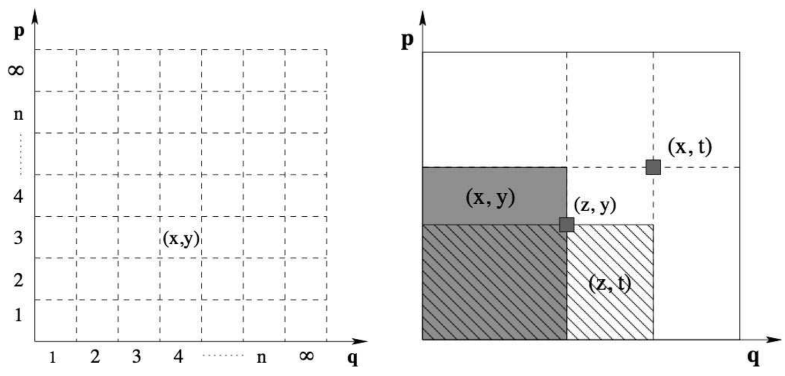
GS has usually two different solutions, *men-optimal* and *women-optimal* depending on who is asked to choose first. *Men-optimal* brings a stable matching in which men have the best possible partner, and women may have the worst and conversely. In many applications of such optimization on bi-partite graphs, as resource scheduling, there might be reasons why to favor one sub-population: for instance, between supply- and demand-constraints there is always one economically more important than the other, or teachers' constraints might be stricter than class-room ones. In many other such problems like segment-pairing in robot vision there is an a priori equal importance of both sets of segments respectively extracted from a couple of images or from the model [18], [19], [20], [21], [22] : then sex equality is likely worth accounting for, leading to a *fair* algorithm. Moreover, some global satisfaction from the matching may translate a better balanced solution among the many possible ones. Neither one is guaranteed by *GS* : the obtained stable matching can be such that every couple is unsatisfied.

3 A novel representation of the stable marriage problem

In order to build an algorithm that had a chance to meet the three criteria of stability, sex equality, and global satisfaction, we first change representation. The so-called *marriage table* translates and supplements the preference lists. Stable matchings are looked for by scanning this latter array, and suitable properties of the solution are associated to the type of scan. The *marriage table* is a table with $(N+1)$ lines and $(N+1)$ columns. Lines (resp. columns) frame the preference orders of men, $\{1 \cdots p \cdots N\ \infty\}$ (resp. women, $\{1 \cdots q \cdots N\ \infty\}$). The cell (p, q) contains pairs (m, w) such that w is the p^{th} choice of m, and m is the q^{th} choice of w. Cells can thus contain one pair, or more, or none. The cell (p, ∞) (resp. (∞, q)) contains the pairs where the woman is the p^{th} choice of the man (resp. the q^{th} choice of the woman) but the man does not exist in her preference list (resp. the woman is not in his preference list). A key feature of this table in the "complete list" case is that each line contains all men once, and each column contains all women once. The figure 1(a) shows a typical marriage table.

The table 1 is the example of an instance of three men and women. Every man or woman made their preference list. The figure 2 is the marriage table and stable matching established from the population 1.

One advantage of the marriage table is that satisfaction and equality of sex show concurrently in the same representation.

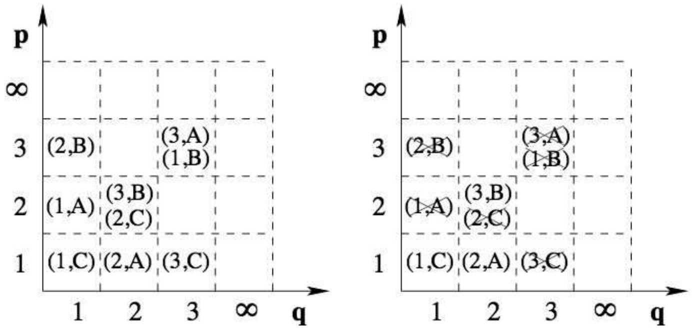


(a) (b)

Fig. 1. (a) Marriage table : the pair (x,y), y is the 3^{rd} choice of x and x is the 4^{th} choice of y, (b) Blocking situation in a marriage table.

Men	Women
1 : C, A, B	A : 1, 2, 3
2 : A, C, B	B : 2, 3, 1
3 : C, B, A	C : 1, 2, 3

Table 1
An instance of 3 men and women and their preference lists



(a) (b)

Fig. 2. (a) Marriage table established from the table 1, (b) Matching result : (1,C), (2,A) and (3,B).

For instance let us define a global satisfaction by

$$\overline{S} = \sum_{(m,w)\in M} (\rho_m + \rho_w) \tag{1}$$

Intuitively, the closer $\overline{S}$ to zero the greater global satisfaction: in average more people are satisfied. A solution with maximum global satisfaction would display matched pairs as close around the origin (table bottom-left) as mutual exclusion allows. More generally the table representation is indicative of a result global satisfaction through the lay out of the selected couples. Satisfaction is constant along antidiagonals (straight lines of equation $p+q=$ constant) and decreasing with the distance to the origin . And that provides some criteria to design scans of the marriage table that could favour better global solutions.

Conversely, sex equality tends to fit the diagonal of the marriage table. Let us define it as

$$\overline{E} = \sum_{(m,w)\in M} |\rho_m - \rho_w| \tag{2}$$

Intuitively the closer to the diagonal the more balanced the treatment. Elements of a pair in a cell close to the diagonal are equally satisfied or unsatisfied, depending on the distance to the origin. The smaller $\overline{E}$ the greater the equity. And again that provides some criteria to design scans of the marriage table that could favor more equitable solutions.

Stability gets a graphic translation too in the marriage table. In the case of complete preference lists, a blocking situation is represented figure 1(b). Assuming (x,t) and (z,y) were respectively paired, then (x,y) cannot be in the grey rectangle and (z,t) cannot be in the dashed one. These constraints will help building a new algorithm. In the case of incomplete preference lists an additional blocking situation is when m and w are not matched in $\mathcal{M}$, but m finds w acceptable, and w finds m acceptable.

4 Marriages and table scans

Algorithms can now be designed to find marriages that would globally guarantee one property or the other. Any selection process is a scan of the marriage table along which couples are stored or not, and then released or not depending on circumstances. After the analysis above, suitable scans likely to meet all constraints should be zig-zag ones that trade off between the first and second diagonal directions.

Considering a priori symmetries of the marriage table, scanning from left to right or conversely does not matter statistically, and likewise for scanning from origin to top or conversely. Indeed, given an instance, changing man's lists into woman's lists does it for the right/left invariance, and complementing preferences to N the population-size does it for the top-bottom invariance.

First, we study experimentally two strategies here (zigzag ZZ with man-optimal or woman-optimal and optimal (symmetric) zigzag OZ), and then we compare the results with *GS* in a systematic way: 200 instances are built at random for populations of 5, 10, 50, 100, 150, and 200 respectively. Each algorithm is run on the populations, and for each one the following plots are displayed and analyzed: global satisfaction vs. instances index, fairness vs. instances index and, in case there are, number of blocking pairs vs. populations. Indeed, it is obvious that no predesigned scan can avoid all blocking situations by itself in all circumstances.

4.1 A bounding result

Given a systematic scan i.e. a permutation of $\mathbb{N}^2$, $n = s(i, j)$, there exists an instance of population i.e. a set of preference lists $\{l(m), l(w)\}_{M\times W}$ such that (m_i, w_j), (m_k, w_l) and (m_k, w_j) are met along the scan in that order, then (m_i, w_j) and (m_k, w_l) are married while (m_k, w_j) is blocking them. It is trivially enough that

$$s(l_j(m_i), l_i(w_j)) \leq s(l_l(m_k), l_k(w_l)) \leq s(l_j(m_k), l_k(w_j))$$
$$\text{and} \quad l_k(w_j) \leq l_i(w_j) \qquad \text{and} \quad l_j(m_k) \leq l_l(m_k)$$

With $l_j(m_i)$ the ranking order of w_j in $l(m_i)$. Figure 3 shows an example of such a case for the anti-diagonal regular scan.

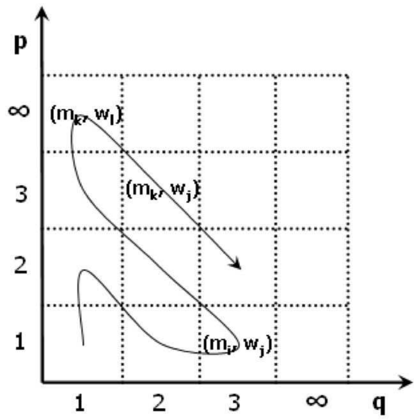


Fig. 3. Zigzag blocking situation

4.2 Zigzag with man-optimal or woman-optimal (ZZ)

The algorithm is straightforward: anti-diagonals are scanned by increasing $p+q$. In each cell, all pairs are accepted for marriage if their components are free. It appears from the global satisfaction graph (figure 4) that its patterns present some similarity (min/max, σ) by sample of 50. We pick up 40 consecutive instances at random for display to show results more in details. From the figure 5, one can see that the global satisfaction obtained by ZZ is better in average than the one by GS. Its variation is also smaller, meaning that results from ZZ are more consistent (hence more reliable to global matching) than by GS.

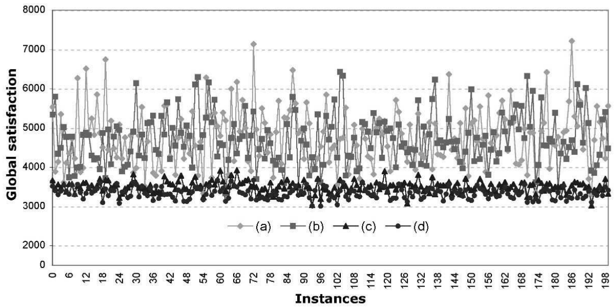


Fig. 4. Comparing global satisfaction between methods: (a) GS man-optimal (b) GS woman-optimal (c) Zigzag man-optimal and (d) Zigzag woman-optimal, in case of 150 large populations.

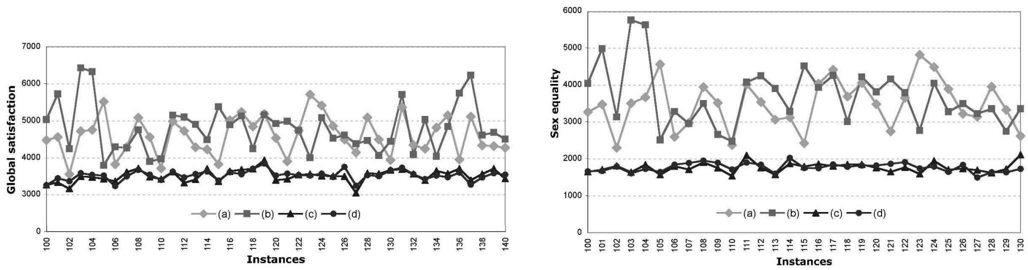


Fig. 5. Close up on global satisfaction(left)/sex equality(right) between methods: (a) GS man-optimal (b) GS woman-optimal (c) Zigzag man-optimal and (d) Zigzag woman-optimal

Figure 5(left) shows the same sample of 40 instances for sex equality. Trends of that type of plot are very similar to global satisfaction trends. The jittery pattern of GSs are similar for global satisfaction and sex equality, with an average standard deviation in the range of 40 to 50%. In both cases, strong variations from one instance to the next one show qualitative similar tendencies, contrasting with comparatively bounded variations by ZZ (15 to 20%). On average, ZZ performs twice as well as GS (3500 vs. 4800 and 1800 vs. 3800). However, the improvement is much more significant regarding sex equality that is genuinely denied by GS . Moreover, if one considers the number ζ of instances where ZZ is better than GS, defined as

$$\zeta = \sum_{\text{all instances}} \Upsilon_{[min(GS_m, GS_w) - max(ZZ_m, ZZ_w)]} \text{ with } \begin{cases} \Upsilon_x = 1 & \text{if } x > 0 \\ \Upsilon_x = 0 & \text{else} \end{cases} \quad (3)$$

$$\beta = \frac{100 \times \zeta}{\text{number of instances}}$$

$\beta = 96\%$ for global satisfaction and $\beta = 99\%$ for sex equality. This percentage depends directly on the population size. The larger the population, the greater the improvement. Experimentally, beyond 200 large populations ZZ becomes 100 percent better than GS for both global satisfaction and sex equality.

However, matching stability is not guaranteed by ZZ. As a gauge of instability, figure 6 displays the average numbers of blocking pairs with their standard deviations vs. the population size. Note that cases (a) and (b) − respectively ZZ man first and ZZ women first − are inseparable at that representation scale. It appears that the larger, the less stable.

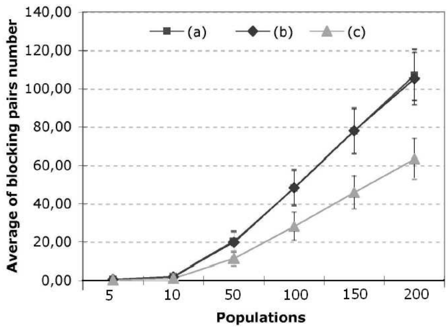


Fig. 6. The average number of blocking pairs using: (a) Zigzag man-optimal and (b) Zigzag woman-optimal (c) Optimal zigzag

4.3 Optimal (Symmetric) zigzag (OZ)

The primary implementation of a diagonal scan of the marriage table proves significantly better than GS for global satisfaction and fairness. But the scan start direction - men or women first - still matters in extreme cases, although its impact is less in general. We now propose an algorithm (algo.1) that targets optimal zigzag from bottom-left to top-right (forward). Here again, anti-diagonals of the table are scanned forward, from maximum to minimum global satisfaction but each one is read in swinging from center to sides meaning maximum to minimum sex equality. With this algorithm global satisfaction is considered first, and then sex equality. Figure 7(c) shows the global satisfaction/sex equality compared with GS figure 7(a)(b). Both global satisfaction and sex equality slightly worsen compared to ZZ, but the main result is blocking pair numbers decrease significantly by about 40% (figure 6 (c)). Still, the method does not guarantee any matching stability. The complexity for all these algorithms so far remains in $O(n^2)$, table building included.

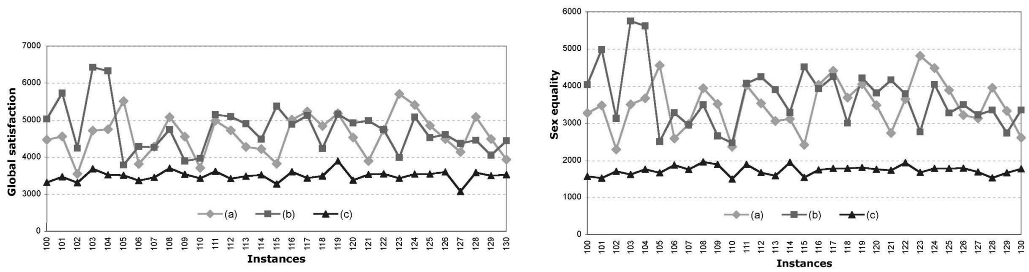


Fig. 7. Comparison of global satisfaction(left)/sex equality(right) between the methods: (a) GS man-optimal (b) GS woman-optimal (c) Optimal zigzag, with 200 instances of 150 large populations

Algorithm 1: Optimal zigzag algorithm

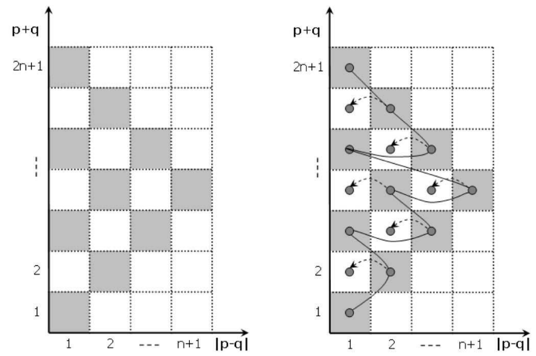
begin
  foreach *anti-diagonal, maximum to miminum global satisfaction* **do**
    foreach *diagonal, maximum to miminum sex equality in alternate directions* **do**
      foreach *pair* (m, w) **do**
        if *m et w are free* **then**
          Marry m with w
end

5 Table-turning to satisfaction and equity

Possibly driven by available computer architectures, one can think of another implementation likely to bring different results with different qualities too. As a functional [23][24] hybrid of GS and OZ, the present track may even lead to skip the marriage table.

5.1 RZ and RGS Algorithms

The algorithm, closer to *GS,* takes advantage from the coding of *satisfaction* and *sex equality* introduced through the *marriage table*. Indeed, if w is the p^{th} choice of m, and m is the q^{th} choice of w, then $p+q$ represents the satisfaction and $|p-q|$ shows sex equality between m and w. The primary idea is that scanning the marriage table in a diagonal way is (quasi) equivalent to rotating the same 45° before scanning horizontally or vertically. Doing so, results at least equivalent to ZZ will be obtained. Figure 8 (a) shows the rotated version of the marriage table. It is a table with $2N+1$ lines and $N+1$ columns. Lines represent the satisfaction and columns the sex equality. The grey area shows the projected area from the original table. It contains in each cell the couples which have the corresponding satisfaction and sex equality. Thanks to the empty cells after rotation in a sampled space, there is room for expanding couples from one cell over several cells if they concern a same person. A complete order is then recovered following additional constraints. By scanning the latter array left-right and bottom-up, and marrying the occurring free pairs (see figure 8(b)), similar results to the OZ-like algorithm ([25]) are obtained. Let us call RZ (Rotated Zigzag) the algorithm.



(a) (b)

Fig. 8. (a) rotated marriage table (b) scan and reorder.

Actually an horizontal (resp. vertical) scan of the marriage table is also the basis of *GS,* where the preference list is ordered following p for men and q for women. We should then improve the overall result by merely performing GS on lists transformed by $(p \hookrightarrow p+q; q \hookrightarrow p-q)$: Instead of transforming the preference lists into the *marriage table* and then finding the matching with maximum satisfaction and sex equality, we introduce the *satisfaction* and *sex equality* into the preference lists then finding the matching result by classical GS. Obviously the stability is no more guaranteed, although GS is run, since it would not be according to the original preferences. But one gets additional variants here as flexible stress can be put on satisfaction, equity, or stability in departing more or less from the initial lists. The latter flexibility amounts to the above-mentioned additional constraints supporting the cell expansion up to complete order in the marriage table. In the current version of our algorithm, the man's and woman's preference lists are first reordered by increasing $p+q$, then by increasing $|p-q|$ in case of equal $p+q$, and then by increasing p for man (respectively q for woman) in case of equal $p+q$ and $|p-q|$. Table 2 shows an instance of three men and women with their preference list ordered by p and q respectively.

Table 3 shows the value of satisfaction $p+q$ and sex equality $|p-q|$ for each element in their preference list.

Table 4 shows the same instance of men and women with their preference lists reordered by satisfaction, sex equality, and initial preference in the order. We

Men\p	1, 2, 3	Women\p	1, 2, 3
1	C, A, B	A	1, 2, 3
2	A, C, B	B	2, 3, 1
3	C, B, A	C	1, 2, 3

Table 2
An instance of 3 men and women and their preference lists

Men	Women
$(p+q, \|p-q\|)$	$(p+q, \|p-q\|)$
1: (2,0), (3,1), (6,0)	A: (3,1), (3,1), (6,0)
2: (3,1), (4,0), (4,2)	B: (4,2), (4,0), (6,0)
3: (4,2), (4,0), (6,0)	C: (2,0), (4,0), (4,2)

Table 3
Satisfaction and sex equality in the preference lists

can see that the preference list of 3 and B are reordered by $|p-q|$ since there is the conflict on $p+q$. Similarly, the preference list of A is reordered by q since there is a conflict on both $p+q$ and $|p-q|$.

Men	Women
1: C (2,0), A (3,1), B (6,0)	A: 1 (3,1), 2 (3,1), 3 (6,0)
2: A (3,1), C (4,0), B (4,2)	B: 3 (4,0), 2 (4,2), 1 (6,0)
3: B (4,0), C (4,2), A (6,0)	C: 1 (2,0), 2 (4,0), 3 (4,2)

Table 4
reordered preference lists

Then the marriages result from executing GS on the preference lists of table 4. As both man's and woman's lists are shuffled here in stressing the *satisfaction* first, GS with man-optimal and woman-optimal are likely to give the same solution.

The novel algorithm introduced here is then an intermediate version between GS and OZ, let us call RGS this algorithm. In the next section, we compare RGS with GS to better understand respective performances.

5.2 Algorithm performance

We study experimentally the global satisfaction, sex equality and stability obtained by *RGS*, and we compare with *GS*. About 3.000 instances are built at random for 200-large populations. Each algorithm is executed, and its results are displayed. Figure 9 zooms on 30 instances out of the 3.000 in order to display more in detail the global satisfaction and sex equality. Let ζ be here the number of instances where RGS is better than the better GS :

$$\zeta = \sum_{\text{all instances}} \Upsilon_{[GSm,w-RGS]} \tag{4}$$

We first notice that statistically RGS performs totally better than the better GS for both satisfaction and sex equality, $\beta = 100\%$. In terms of achieved *satisfaction* and *sex equality*, performances can be measured from the following average distances: $\frac{|S_{RGS}-S_{GS}|}{S_{GS}} = 22.64\%$ and $\frac{|E_{RGS}-E_{GS}|}{E_{GS}} = 46.96\%$.

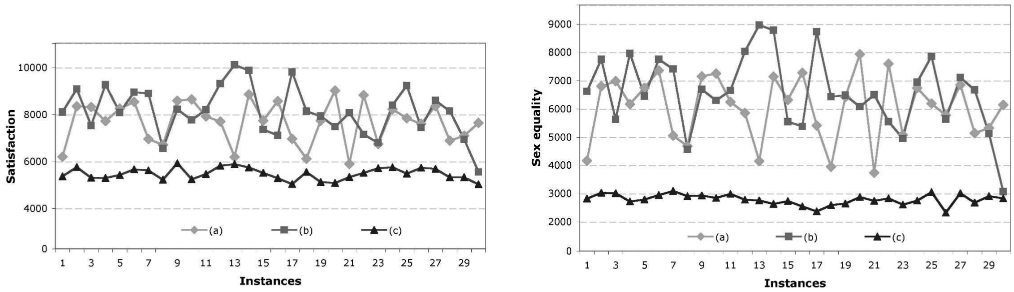


Fig. 9. Comparing global satisfaction(left)/sex equality(right) between methods : (a) GS man-optimal, (b) GS woman-optimal, (c) RGS

Stabilitywise, the matching results issued by GS are totally stable while around 87% of instances are unstable for RGS. But the average number of blocking pairs per unstable instance is 3.3 for RGS. Not only is the latter number negligeable in most applications, but, in case it is not, we will see in section 6 that the expectation of the number of iterations starting from RGS to overcome oscillations towards complete stability is significantly lower than with other proposed algorithms.

6 Matching instability

6.1 Brute force attempt to resolve instability: Blocked zigzag (BZ)

All previous algorithms may provide matching solutions that are globally satisfactory and fair but unstable: they might even be such that everybody would like to split! In this section an algorithm (algo.2) is designed to meet all three criteria concurrently, at the cost of a reasonable increase in complexity due to systematic added test. We scan anti-diagonals as before. In each cell, all pairs are accepted for marriage if their components are free. After all cells have been considered, the table is then scanned again, and up to complete removal of blocking situations as follows: potential blocking pairs are matched upon detection (test according to figure 1(b)) while both blocked couples involved are broken, and the abandonned mates are freed. The process repeats until there is no more blocking situation (case "stability"), or the iteration number is greater than the population size ("instability"). Indeed, to overcome cycles (see section 6.2) in the assignment, the number of rescanning is limited. Careful selection of the scan directions along with questioning all previous marriages when necessary affords ultimate end product. Global satisfaction and sex equality are comparable (figure 11) to former ones (figure 9), if not even increased. The main improvement is matching stability now obtained in a great majority of cases, with the number of instabilities significantly lowered otherwise (figure 10(right)). More precisely, the number of blocking pairs is null until 50 man-or-woman large populations. It is still 0 in an average 96% of the 200 instances for populations larger than 50, and its maximum ranges in the 15 blocking pairs for 100. However, the algorithm complexity is in the order $O(n^3)$.

In figure 10(left), the algorithm performances are compared wrt. population size. The improvement β from GS to BZ increases with the population size. Again, beyond 100 large populations BZ becomes 100 percent better than GS for both global satisfaction and sex equality.

Algorithm 2: Blocked zigzag algorithm

```
begin
    while there is a bloking pair and rescan number < population size do
        foreach anti-diagonal, maximum to miminum global satisfaction do
            foreach diagonal, maximum to miminum sex equality back and
            forth do
                foreach pair (m, w) do
                    if m and w are free then
                        Marry m with w
        foreach anti-diagonal, maximum to miminum global satisfaction do
            foreach diagonal, maximum to miminum sex equality back and
            forth do
                foreach pair (m, w) do
                    if (m, w) is blocking pair then
                        Free m and w and their spouse
                        Marry m with w
end
```

Comparing BZ with RGS on 3000 instances of 200 large populatios, $\beta =$

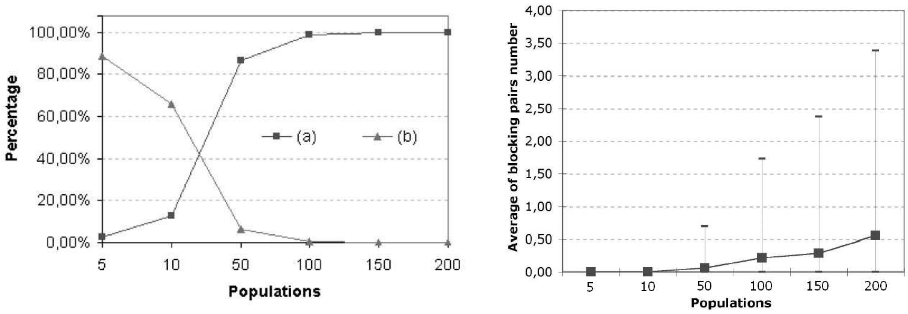


Fig. 10. Performance comparison between methods : (a) BZ and (b) GS (left)/Average of blocking pairs number of BZ (right)

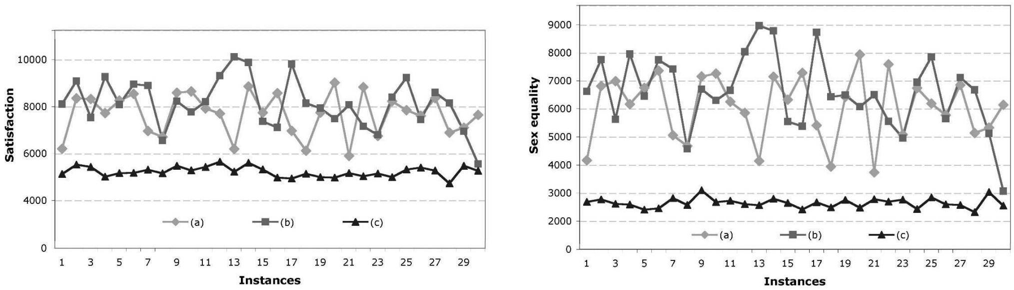


Fig. 11. Comparing global satisfaction(left)/sex equality(right) between methods : (a) GS man-optimal, (b) GS woman-optimal, (c) BZ

35.77% and $\beta = 8.13\%$ respectively for the global satisfaction and sex equality. Moreover, we note that RGS is comparable to BZ with respect to GS for the *satisfaction* and *sex equality* achieved. This is indicated again by the following distances: $\frac{|S_{RGS}-S_{BZ}|}{S_{BZ}} = 5.72\%$ and $\frac{|E_{RGS}-E_{BZ}|}{E_{BZ}} = 7.24\%$ on average. But, more importantly, the average number of blocking pairs per unstable instance is 10.8 for BZ, compared to 3.3 for RGS.

Stabilitywise, the matching results issued by GS are totally stable, while around 5% instances are unstable for BZ.

Table 5 shows a numeric comparison between RGS and BZ or GS for the stability (number of blocking pairs), global satisfaction and sex equality.

6.2 Analysis of instability after BZ

The instability of the marriages was studied by several authors [7] [9]. To understand the source of matching instability, let us unfold two matching examples on the same population (table 6). The table 7 shows an instance of the marrying process. For each repetition, free pairs are coupled and blocking situations removed. At step 3, a stable matching is output. The table 8 shows a second instance of the same: free pairs to marry and blocking pairs to be

	INB	NBP	Avg Distance	$\overline{S}$	$\overline{E}$
GS	0%	0	(BZ, RGS)	280	189
BZ	3.9%	11.7	(BZ, RZ_m)	1614	1310
RGS	87.9%	3.3	(BZ, RZ_w)	8276	2080
RZ_m	100%	23.3	(GS, RGS)	1733	2714
RZ_w	100%	73.1	(GS, RZ_m)	854	4108
			(GS, RZ_w)	6277	4881

Table 5
Comparison of stability, global satisfaction and sex equality between methods. INB = Instability, NBP = Number of blockages, Avg Distance = absolute value of difference of means.

removed are scrutinized in a different order. It can be noticed in this case that after marrying (m_1, w_1) to solve the blocking situation $-$ then releasing w_3 and m_3- and later on marrying (m_3, w_3), the marriage at step 4 turns again into the marriage at step 1. From then on the marriage-process would *cycle* indefinitely. Unlike the previous scan, the current process could not find a stable solution, and the matching will remain unstable for ever.

Man	Woman
$l(m_1) = w_2,\ w_1,\ w_3$	$l(w_1) = m_1,\ m_3,\ m_2$
$l(m_2) = w_1,\ w_3,\ w_2$	$l(w_1) = m_3,\ m_1,\ m_2$
$l(m_3) = w_1,\ w_2,\ w_3$	$l(w_1) = m_1,\ m_3,\ m_2$

Table 6
Preference list

Marriage	Blocking situation
$1 : (m_1, w_1),\ (m_2, w_2),\ (m_3, w_3)$	$\underline{(m_1, w_2)} : [(m_1, w_1),\ (m_2, w_2)]$
	$(m_3, w_2) : [(m_3, w_3),\ (m_2, w_2)]$
$2 : (m_1, w_2),\ (m_2, w_1),\ (m_3, w_3)$	$\underline{(m_3, w_1)} : [(m_3, w_3),\ (m_2, w_1)]$
	$(m_3, w_2) : [(m_3, w_3),\ (m_1, w_2)]$
$3 : (m_1, w_2),\ (m_2, w_3),\ (m_3, w_1)$	

Table 7
Example of a marriage without cycle

In the BZ algorithm, marrying free pairs and cleaning blocking situations depends explicitly on the marriage-table scan that obeys independent constraints. In the next section, we classify all possible blocking situations to appear in a marriage table. The case list is further used to determining stable optimal marriages.

Marriage	Blocking situation
$1 : (m_1, w_1),\ (m_2, w_2),\ (m_3, w_3)$	$\underline{(m_1, w_2)} : [(m_1, w_1),\ (m_2, w_2)]$
	$(m_3, w_2) : [(m_3, w_3),\ (m_2, w_2)]$
$2 : (m_1, w_2),\ (m_2, w_1),\ (m_3, w_3)$	$(m_3, w_1) : [(m_3, w_3),\ (m_2, w_1)]$
	$\underline{(m_3, w_2)} : [(m_3, w_3),\ (m_1, w_2)]$
$3 : (m_1, w_3),\ (m_2, w_1),\ (m_3, w_2)$	$\underline{(m_1, w_1)} : [(m_1, w_3),\ (m_2, w_1)]$
	$(m_3, w_1) : [(m_3, w_2),\ (m_2, w_1)]$
$4 : (m_1, w_3),\ (m_2, w_2),\ (m_3, w_1)$	$\underline{(m_1, w_1)} : [(m_1, w_3),\ (m_3, w_1)]$
	$(m_1, w_2) : [(m_1, w_3),\ (m_2, w_2)]$

Table 8
Exemple of a marriage with cycle

6.2.1 Blocking situations and oscillating behaviors of marriages

Not all blocking situations make cycles occur over the marriage table. We classify the blocking situations into four principal types:

Type 1, already explained for the basics, comes from one blocking pair (m, w) which relates to one blocked pair by its man $(m, w_?)$ and one blocked pair by its woman $(m_?, w)$. (m, w) is not matched in $\mathcal{M}$ but m prefers w to $w_?$ and w prefers m to $m_?$. So, $w_?$ $(m_?)$ is the current choice of m (resp. w) in $\mathcal{M}$. The situation in the marriage table is displayed in figure 12(a)

Type 2 is composed of M blocking pairs (m, w_j), $j = 1 \cdots M$ corresponding to one blocked pair by its man $(m, w_?)$ and M blocked pairs by their woman. $(m_?, w_j)$, $j = 1 \cdots M$. (m, w_j) are not matched in $\mathcal{M}$ but m prefers w_j to $w_?$ and w_j prefers m to $m_?$. The situation is shown in the figure 12(b)

Type 3 is a symmetric version of type 2. It shows as L blocking pairs (m_i, w), $i = 1 \cdots L$ relating to one blocked pair by its woman $(m_?, w)$ and L blocked pairs by the man $(m_i, w_?)$, $i = 1 \cdots L$. (m_i, w) are not matched in $\mathcal{M}$ but m_i prefers w to $w_?$ and w prefers m_i to $m_?$. The situation is sketched figure 12(c).

Type 4 is a combination of 2 and 3. It results from $L + M$ blocking pairs (m_i, w), (m, w_j), where $i = 1 \cdots L$ and $j = 1 \cdots M$ generating L blocked pairs by the man $(m_i, w_?)$, $i = 1 \cdots L$, M blocked pairs by the woman $(m_?, w_j)$, $j = 1 \cdots M$ and one pair blocked from both sides (m, w). The situation is that (m_i, w) and (m, w_j), where $i = 1 \cdots L$ and $j = 1 \cdots M$ are not matched in $\mathcal{M}$ but m_i prefers w to $w_?$ and w prefers m_i to $m_?$. And conversely, m prefers w_j to $w_?$ and w_j prefers m to $m_?$. The corresponding situation of the marriage table appears in figure 12(d).

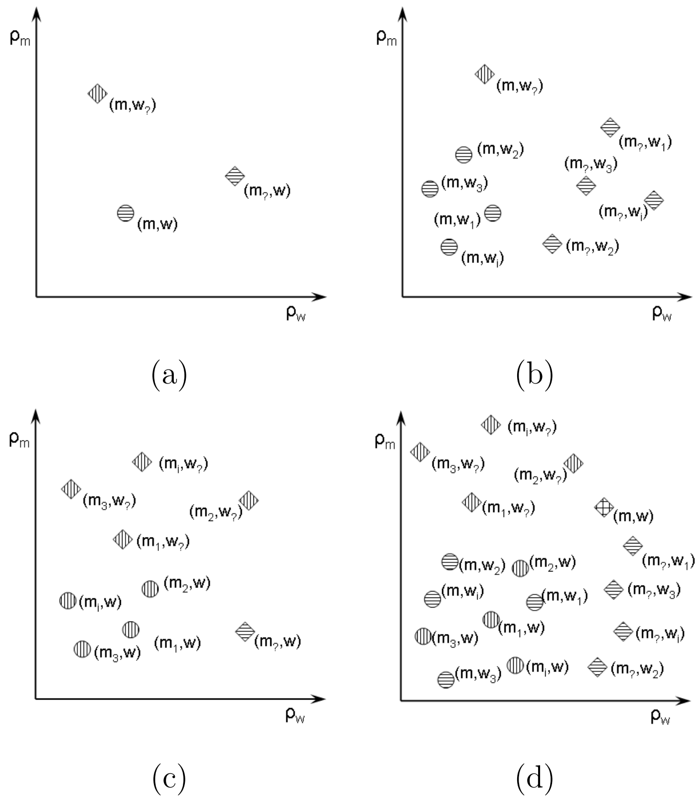


Fig. 12. Blocking situations in the marriage table : (a) Type 1 (b)Type 2, (c) Type 3 (d) Type 4. Circles represent blocking pairs and squares represent blocked pairs. Vertical and horizontal dashed-textures inside squares feature the blocking situation by men and women respectively. Textures inside circles indicate, likewise, the sex of the multiple blocker.

Cells of the marriage table are visited sequentially to marry free pairs; the scan generates dynamically blocking pairs to be eliminated afterwards. Other scans, other marriages, and then other blocking situations and pairs. So each instance of a blocking situation is independent in the sense that resolving it does not influence concurrent blocking situations at that step. Additionally, we found that over 40.000 tries, early blocking pairs tend to be mostly of type 1. As the process goes on, instances belong more and more to the three other types while decreasing in number. That is trivially explained by the logics of the situation:

$$[(m, w_?)\text{got married first}] \Longrightarrow [m \text{ and } w_?\text{were free}]$$
$$\textbf{then } [(m, w)\text{blocks } (m, w_?)] \textbf{ and } [(m_?, w)\text{got married}]$$
$$\Longrightarrow [m_?\text{was free } \textbf{and } w\text{was free because } m\text{was married}]$$
$$\textbf{then } [(m, w)\text{gets married }]$$
$$\textbf{and } [(m, w_?)\text{and } (m_?, w)\text{are released}]$$
$$\Longrightarrow w_? \text{ and } m_? \text{ will marry further in the scan}$$

The longer the process, the longer the distance between blocking and blocked pairs, and the more likely the occurence of multiple blocking pairs for the same blocked pairs.

Reaching stability, the process stops when there are no more blocking pairs in the table. Failing to do so (unstable marriage) corresponds to *oscillations* of blocking pairs and *cycles* of the process. Cycles cross the diagonal of the marriage table (equal satisfaction), possibly several times, corresponding to oscillations of blocking pairs from "man satisfaction side" to "woman satisfaction side", and back.

6.2.2 Oscillation and cycle elimination

For the matching solution to become stable, oscillations and cycles have to be eliminated. Although they are not independent phenomena we can get rid of the oscillation and the cycle separately.

Stopping oscillations that relate to sex equality can be done by forcing man-first or woman-first matching when marrying the free pairs. We test both and keep the best with respect to satisfaction and equity.

Eliminating cycles that relate to the stack of candidates for a same person could be performed by marrying an optimal blocking pair among the several possible ones. It applies for each situation type separately. In type 1 there is only one blocking pair for the optimal choice. In type 2, all blocking pairs contain the same man m with different partners w_j. Ideally we would choose the blocking pair (m, w^*) which contains the woman $w^* \in \{w_j\}$ that m prefers the most (that is man-first).

$$(m, w)^* = \Big\{(m, w^*) / \rho_{w^*} \ll_m \rho_{w_j},\, j = 1, \cdots M\Big\} \tag{5}$$

In reality choosing the blocking pair that features the woman w_j who prefers m (and the one preferred by m in case of equality, i.e. the first couple met along a vertical scan) would work as well. Since oscillation removal forbids switching from man-first to woman-first, or conversely during a same process, actually we keep the one retained for stopping oscillations.

Type 3 is symmetric of type 2, $(m, w)^*$ ideally contains the man $m^* \in \{m_i\}$ that w prefers the most.

$$(m, w)^* = \{(m^*, w) / \rho_{m^*} \ll_w \rho_{m_i}, i = 1, \cdots, L\} \tag{6}$$

Here again the selection policy depends on oscillation removal and is maintained all along the elimination process.

Type 4 combines types 2 and 3. There are two different groups of blocking pairs, then two possible optimal choices, respectively man-first (eq.5) and women-first (eq.6), figure 12(d). Again we keep the same option as for oscillation removal.

In the next section we propose a stable-marriages algorithm designed after BZ from the criteria introduced in this section.

6.3 New algorithm : Stable Blocked Zigzag (SBZ) with man-optimal (SBZ_m) or woman-optimal (SBZ_w)

The so-called *Stable Blocked Zigzag* algorithm summarizes the study above. It runs in two separate phases: it first applies the BZ procedure aiming to find the best solution for global satisfaction, sex equality, and then stability. As previously mentioned (section 6.1) that rules about 95% instances. The second phase deals with oscillation/cycle removal for marriages to get stable. It concerns the 5% unstable marriages output by BZ. Two options are considered in this case: *SBZ* with man-optimal (SBZ_m) and *SBZ* with woman-optimal (SBZ_w).

SBZ_m gives the scan priority to the man side: blocking situations are searched for and typified by scanning rows of the table bottom-up and left-right, then pairs will be married upon release iteratively according to the following:

- Type 1 marry the blocking pair (there is only one) and release related blocked pairs as shown figure 13(a).
- Type 2 marry the blocking pair which contains the woman $w^* \in \{w_j\}$, $j = 1, \cdots, M$ that m prefers among $\{w_j\}$and then release corresponding blocked pairs , see figure 13(b).
- Type 3 marry the blocking pair which contains the man $m^* \in \{m_i\}$, $i = 1, \cdots, N$ who prefers w (the first couple met along the present horizontal scan in case of equality of preferences) and then release the tied blocked pairs as in figure 13(c).
- Type 4 is strictly the same as type 2, figure 13(d).

SBZ_w is the exact symmetric version of the former, in giving the importance to the woman side: it marries the free pairs by scanning columns, and women chose first in a systematic manner.

Algo. 3 shows SBZ_w. To get the SBZ_m procedure it suffices to exchange m and w systematically, in keeping indexes and exponents.

Note that OZ is the simplest algorithm based on the marriage table. BZ and SBZ are improved version of it to target maximum satisfaction and sex equal-

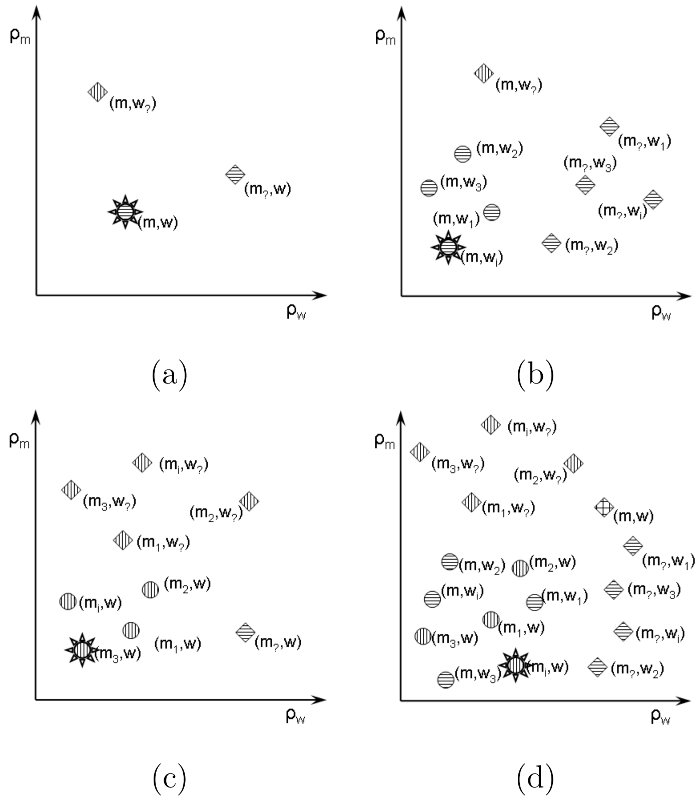


Fig. 13. Elimination of a blocking situation by *SBZ* with man-optimal : (a) type 1, (b) type 2, (c) type 3, (d) type 4. The star shows the selected blocking pair in every case.

ity, and then to meet the full stability constraint. It means that OZ, RZ or RGS can later be completed in a S-fashion to achieve full stability. Complexity would statistically remain in $O(n^3)$ compared to $O(n^4)$ for SBZ.

6.4 Algorithm performance

We studied experimentally the global satisfaction, sex equality, and stability issued by *SBZ* and compared with *GS* (Gale-Shapley) and *BZ*. More than 800,000 instances − couples of preference list sets − were built at random for a 200-persons-large population. Each algorithm is run, and results are displayed. From the experiments, we first note that the eventual results by *SBZ* are stable for all instances while 40.000 results by *BZ*, confirming around 5%, remained unstable. Here again it appears that displayed results show some similarity by subsets of 30 in average. Figure 14(left) zooms on 30 instances. Let us consider again the number ζ of instances where *SBZ* is better than *GS* and *BZ*, and then β same as in section 4.2

$$\zeta = \sum_{\text{all instances}} \Upsilon_{[GS_{m,w}(BZ) - SBZ_{m,w}]} \tag{7}$$

Algorithm 3: Stable Blocked Zigzag with woman-optimal

```
begin
  Run BZ algorithm
  while there is cycle, blocking situtation do
    foreach column : bottom -up do
      foreach pair (m, w) do
        if m and w are free then
          Marry m with w
    foreach column : bottom-up do
      foreach pair (m, w) do
        if (m, w) is blocking pair then
          Identify situation type
          Type 1 : (m, w)* = (m, w)
          Type 2 : (m, w)* = {(m, w*)⁄ρ_m ≪_{w*} (ρ_m ≪_{w_i})}
          Type 3 : (m, w)* = {(m*, w)⁄ρ_{m*} ≪_w ρ_{m_i}}
          Type 4 : (m, w)* = {(m*, w*)⁄ρ_{m*} ≪_w ρ_{m_i}}
    foreach colom : bottom-up do
      foreach pair (m, w) do
        if (m, w) is (m, w)* then
          Release the blocked pairs : (m, w_?) and (m_?, w)
          Marry m with w
end
```

Comparing $SBZ_{m,w}$ with BZ as for the global satisfaction, $\beta = 72\%$ and 72.8% for m and w respectively. Likewise, comparing $SBZ_{m,w}$ with $GS_{m,w}$, $\beta = 100\%$ in both cases. SBZ is always better than GS, figure 15(left).

For sex equality, comparing $SBZ_{m,w}$ with BZ, β equals 58.4% and 56% respectively, figure 14(right). And comparing $SBZ_{m,w}$ with $GS_{m,w}$, β equals 100% in both cases, figure 15(right).

Actually, the better SBZ performs better than BZ in many cases where SBZ_m and SBZ_w score very differently. Then BZ performs better than one SBZ , whether it is for satisfaction or sex equality, and the improvement from SBZ is not really significant ($\frac{S_{BZ}-S_{SBZ}}{S_{BZ}}$ =3.17% in average for the m option and 3.15% for w and $\frac{E_{BZ}-E_{SBZ}}{E_{BZ}}$ =3.41% in average for the m option and 3.48% for w). But in the cases where it is the contrary $-$ BZ performs better than both $SBZ-$, the distance between the better SBZ and BZ is important: $\frac{S_{SBZ}-S_{BZ}}{S_{SBZ}} = 18.81\%$ in average for the m option and 21.28% for w , and $\frac{E_{SBZ}-E_{BZ}}{E_{SBZ}} = 32.16\%$ in average for the m option and 33.17% for w . As a last index, one can compare the following means: average difference in satisfaction when BZ outperforms

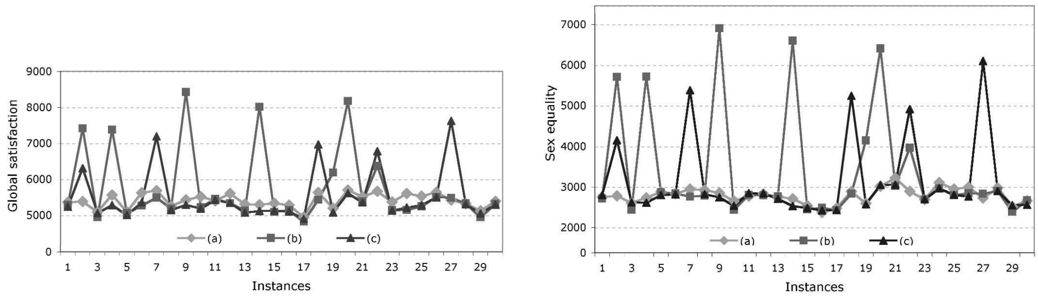


Fig. 14. Close up on global satisfaction(left)/sex equality(right) between methods (30 instances) : (a) BZ algorithm (b) SBZ_m (c) SBZ_w

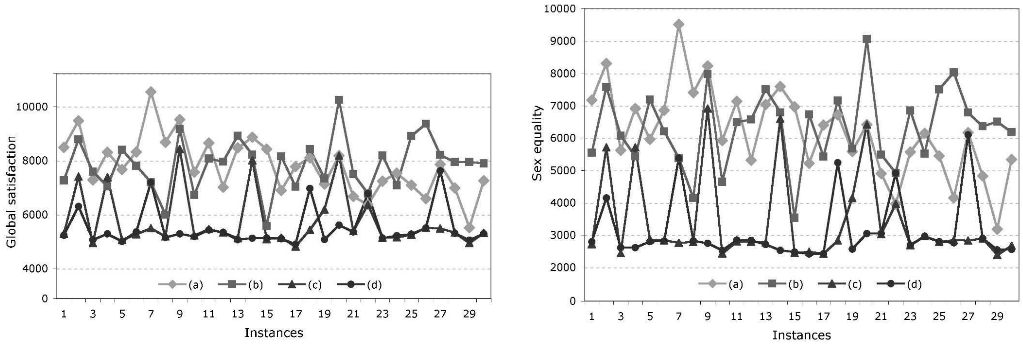


Fig. 15. Comparing global satisfaction(left)/sex equality(right) between methods : (a) GS_m (b) GS_w (c) SBZ_m (d) SBZ_w

$SBZ, \partial^S_{B,S} = 1258$, average difference in satisfaction when SBZ outperforms BZ, $\partial^S_{S,B}$= 166, average difference in equity when BZ outperforms $SBZ, \partial^E_{B,S}$ = 1298, average difference in equity when SBZ outperforms $BZ, \partial^E_{B,S} = 95$.

Eventually it is important to underline again and keep in mind that if SBZ is globally better than BZ and GS, its complexity in the $(O(n^4))$ is high compared to BZ ($O(n^3)$) and GS ($O(n^2)$) respectively. That is why in order to conclude with the performance evaluation we compare in a systematic manner OZ, BZ, RZ and RGS with GS for a reference basis, and then their respective completed versions in the $S-$fashion. The entire study is performed over *the identical set of 1000 instances* of 200 large populations, and results are shown in table 9. All the indexes introduced along the paper are displayed to be compared at a glance. For the rest, choosing among them is a matter of application requirements.

The figures 16 to 17 show the comparison of global satisfaction and sex equality between OZ, SOZ, BZ, SBZ, RZ_m, SRZ_m, RZ_w, SRZ_w, RGS, $SRGS$ with GS respectively.

GS/X	$\overline{S}_{(X>GS)}$	$\overline{S}_{(GS>X)}$	$\overline{E}_{(X>GS)}$	$\overline{E}_{(X>GS)}$	INB	NBP
OZ	98.80%	1.20%	100.00%	0%	100.00%	106
$\|GS-OZ\|$	1724	126	2666	0		
$\frac{\|GS-OZ\|}{GS}$	22.93%	2.26%	46.88%	0%		
SOZ	52.80%	47.20%	52.30%	47.70%	0%	0
$\|GS-SOZ\|$	33	1256	41	1557		
$\frac{\|GS-SOZ\|}{GS}$	0.41%	18.46%	0.65%	32.52		
BZ	100.00%	0%	100.00%	0%	3.20%	9.94
$\|GS-BZ\|$	1973	0	2746	0		
$\frac{\|GS-BZ\|}{GS}$	26.43%	0%	48.39%	0%		
SBZ	99.70%	0.30%	99.80%	0.20%	0%	0
$\|GS-SBZ\|$	1973	633	2739	1088		
$\frac{\|GS-SBZ\|}{GS}$	26.45%	10.51%	48.27%	26.70%		
RZ_m	63.30%	36.70%	100.00%	0%	100%	23.34
$\|GS-RZ_m\|$	953	680	4057	0		
$\frac{\|GS-RZ_m\|}{GS}$	12.04%	10.87%	73.56%	0%		
SRZ_m	91.40%	8.60%	100.00%	0%	0%	0
$\|GS-SRZ_m\|$	1383	281	2943	0		
$\frac{\|GS-SRZ_m\|}{GS}$	18.10%	4.91%	52.20%	0%		
RZ_w	0.20%	99.80%	100.00%	0%	100%	73.26
$\|GS-RZ_w\|$	875	6307	4831	0		
$\frac{\|GS-RZ_w\|}{GS}$	10.22%	90.34%	88.43%	0		
SRZ_w	73%	27%	88.10%	11.90%	0%	0
$\|GS-SRZ_w\|$	1035	595	1899	596		
$\frac{\|GS-SRZ_w\|}{GS}$	13.24%	9.68%	32.18%	15.74		
RGS	98.80%	1.20%	100.00%	0%	87.60%	3.37
$\|GS-RGS\|$	1723	133	2666	0		
$\frac{\|GS-RGS\|}{GS}$	22.93%	2.39%	46.88%	0		
$SRGS$	99.30%	0.70%	100.00%	0%	0%	0
$\|GS-SRGS\|$	1779	127	2588	0		
$\frac{\|GS-SRGS\|}{GS}$	23.73%	2.33%	45.42%	0		

Table 9
Comparison of performances between methods.

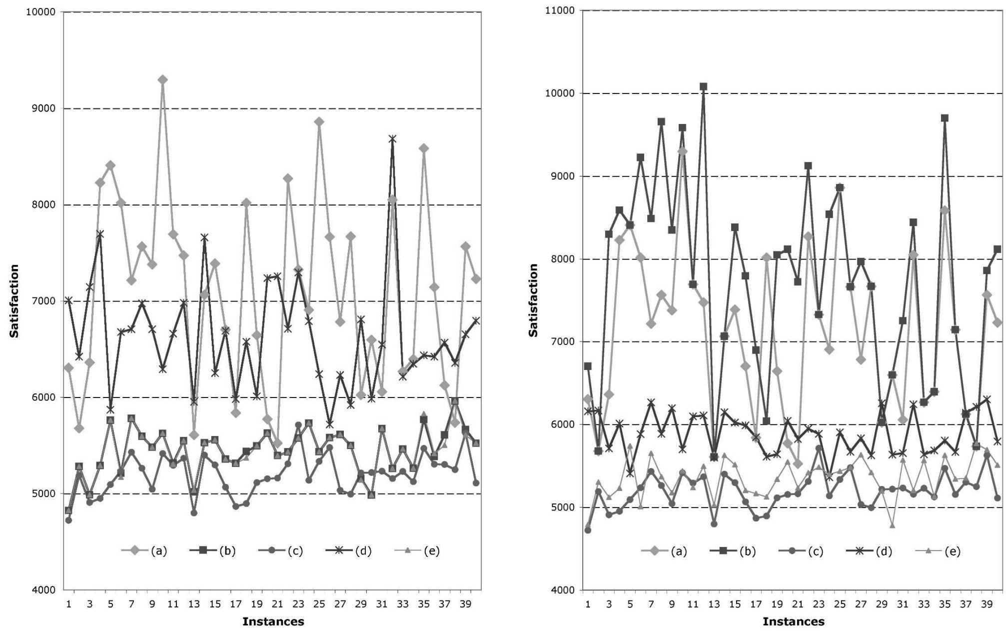


Fig. 16. Comparing global satisfaction (before S left/after S right) between methods : (a) the better GS (b) OZ/SOZ (c) BZ/the better SBZ (d) the better RZ/the better SRZ (e) RGS/SRGS

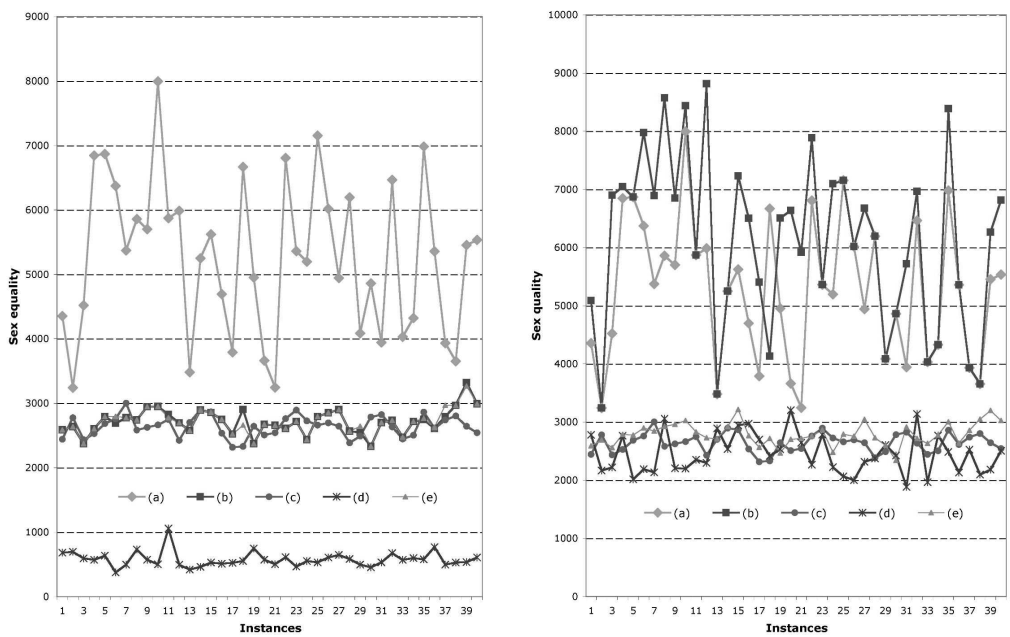


Fig. 17. Comparing sex equality (before S left/after S right) between methods : (a) the better GS (b) OZ/SOZ (c) BZ/the better SBZ (d) the better RZ/the better SRZ (e) RGS/SRGS

7 Conclusion

In this paper, we proposed and evaluated "stable marriages" algorithms based on a novel representation, called marriage table, of which the definition and features were presented. Algorithms follow different scan styles of this table, defining result properties accordingly. We first introduce three different ones aimed at progressively meeting the three criteria of global satisfaction, sex equality, and stability in this order. The last among these three offers the choice to dealing with satisfaction and equity directly from the preference list representation. To recover stability fully, a direct blocking-pair-removal procedure is designed and tested. The three criteria are conjointly satisfied in 95% cases, requiring a raise in complexity from $O(n^2)$ to $O(n^3)$. For the 5% remaining instances, a careful case analysis is completed leading to four types of instability, and to the SBZ algorithm. It starts from the best previous one in terms of stability, BZ, and removes blocking pairs if necessary. While stability of the matching is fully supported $-$ the important progress from the BZ algorithm $-$ both satisfaction and equity are increased in 60 to 70% of the processed cases. However, when satisfaction and equity drop it is in a proportion of about half the BZ score. For stabilizing BZ results with this procedure, the complexity increases from $O(n^3)$ to $O(n^4)$, but the same blocking-pair-removal process applies after any previous algorithm OZ, RZ or RGS, amounting to a raise from $O(n^2)$ to $O(n^3)$ again. Systematic testing on the same population set confirms a performance even better than $SBZ's$ in most cases regarding satisfaction and equity.

For the sake of illustration, we outline here some image-processing applications in stereovision, registration, and motion analysis. Matching relies on level-lines. Features as simple as junctions [26] or sequences of line segments are extracted from each image separately and then selected into primitives. Each primitive is given its preference list containing primitives of the other image (see for instance figure 18). The preference list is incomplete and sorted by features' similarity (e.g. contrast, length, relative orientation, relative position, etc.). BZ is then run. Figure 19(a)(b)(c)(d) show the original stereo images and the features extracted from them respectively. The result of feature matching by BZ shows as an optical flow in figure 19(e). And the figure 19(f) supports comparison with GS. The results to the naked eye are quite comparable due to the "incomplete list" nature of the implementation.

The same process can work for motion. Figures 20 show the image sequence and the matching result respectively.

As an opening for future work, we display comparative results (figure 21) of matching M and M' to compute the transform T (affine + projections) between left L and right R images from "http://www.gravitram.com/stereoscopic_photography.html".

Features to be matched are level line junctions again. We display images of the kind $sup[|R - \mathbb{T}_M(L)| - |R - \mathbb{T}_{M'}(L)|, 0]$ with M being equal to BZ or RGS and M' to GS. Regions where M is obviously worse than M' appear as darker as they are worse. Although several causes − such as extraction of features and parameters, or transform-computation − exist to explain local imperfections, it can be thought that stability of GS serves more a local goodness on details while satisfaction and equity would improve results in a wider fashion. Our forthcoming work will deal with this issue, testing such conjectures systematically for every proposed algorithm, and drawing consequences in vision applications.

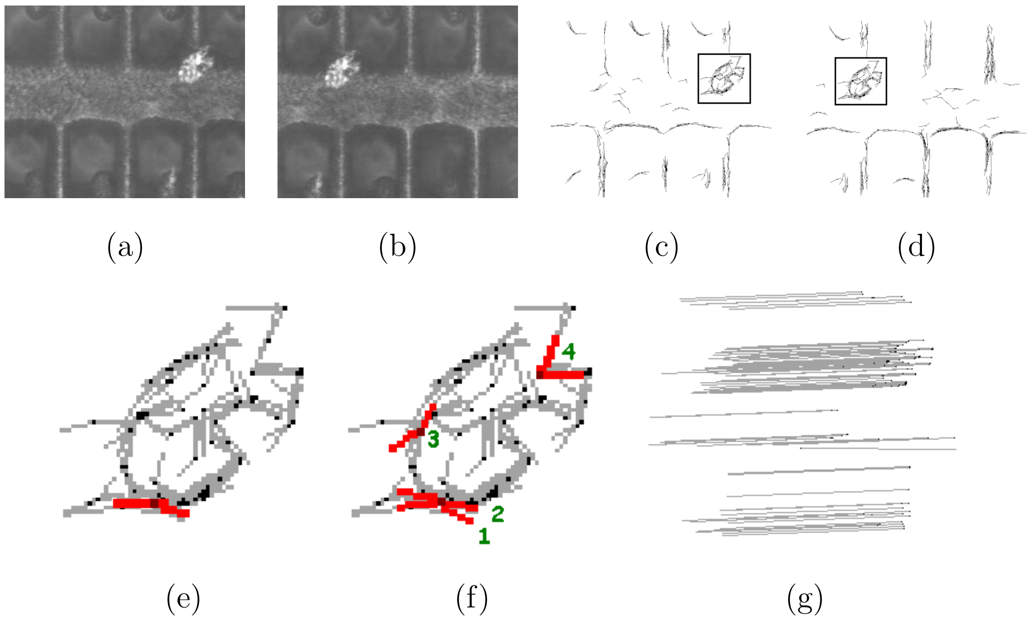


Fig. 18. Stable marriages for MEMS images registration in electron beam microscopy, (a) 1^{st} field-part scanned, (b) 2^{nd} field-part scanned to be super imposed into a larger field. Let us underline the VLSI implementation artefact : this defect will eventually support the perfect match between (a) and (b), despite the cumb ambiquity from periodicity, (c)(d) the primitives extracted from (a) and (b) with primitives from the defect underlined in the frame, (e) primitives in the defect that supports the perfect piecing, (f) potential mates of the primitive underlined in bold in (e) with their rank, (g) final matching results by BZ : they are in perfect agreement with the known reality

ACKNOWLEDGEMENTS:

Thanks to Guna Seetharaman for initial work together on the marriage table and to Maxime Zavidovique for the idea and first tries of rotating the table. We are indebted to Florence Brodkey (floindex@aol.com) for her editing of this final version.

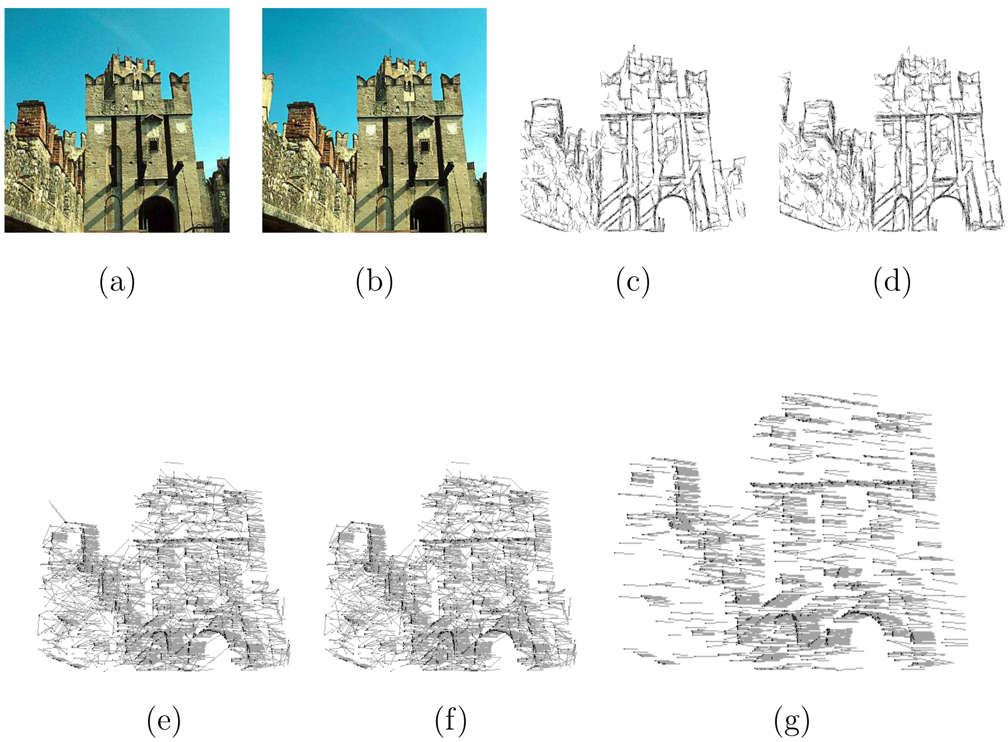


Fig. 19. Stable marriages matching for stereovision, (a)(b) stereo images, (c)(d) features extracted from (a)(b) respectively, (e) matching results by BZ, (f) matching results by GS, (g) final matching results by BZ

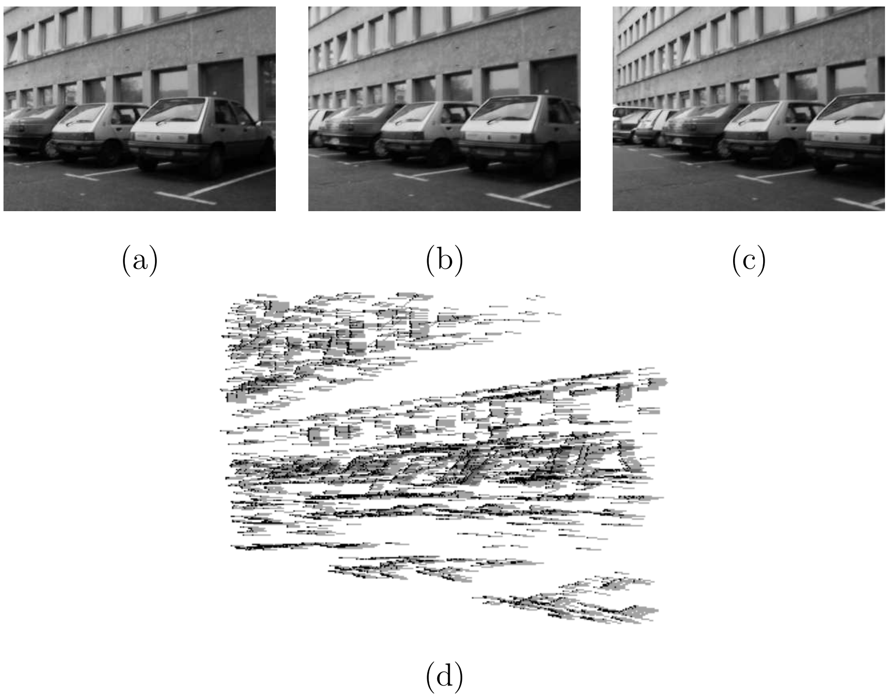
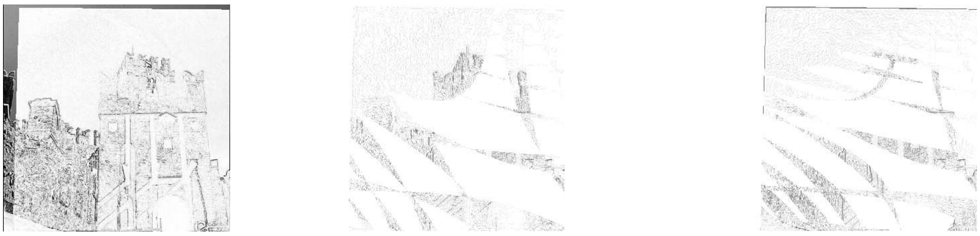


Fig. 20. (a)(b)(c) Image sequence for motion detection, (d) Matching result by using the stable marriages algorithm.

(a) L (b) R (c) $|R - \mathbb{T}_{RGS}(L)|$



(d) $|R - \mathbb{T}_{BZ}(L)|$ (e) $sup[(c) - |R - \mathbb{T}_{GS}(L)|, 0]$ (f) $sup[(d) - |R - \mathbb{T}_{GS}(L)|, 0]$

Fig. 21. Comparing reconstruction errors : BZ to GS and RGS to GS

References

[1] B. Burg, B. Zavidovique, Pattern recognitions and image compression by means of a time warping algorithm, in: ICPR'86, 1986.

[2] P. Adam, B.Burg, B. Zavidovique, Dynamic programming for region based pattern recognition, in: ICASSP/IEEE-ASJ, Tokyo, 1985.

[3] S. Bouchafa, B. Zavidovique, Efficient cumulative matching for image registration, IVC 1 (1) (2006) 70–79.

[4] K. Zemirli, G. Seetharamann, B. Zavidovique, Stable matching for selective junction points grouping, in: IEEE JCIS 2000 (CVPR-IP), 2000.

[5] K. Zemirli, Contrôle de segmentation pour l'extraction de formes, Ph.D. thesis, Université de Paris-Sud, phD dissertation (in French) (December 2000).

[6] D. Gale, L. Shapley, College admissions and the stability of marriage, American Mathematical Monthly 69 (1962) 9–15.

[7] H. Abeledo, U. G. Rothblum, Paths to marriage stability, Elsevier : Discrete Applied Mathematics 63 (1-12).

[8] E. Diamantoudi, E. Miyagama, L. Xue, Random paths to stability in the roommate problem, Elsevier : Games and Economic Behavior 48 (18-28).

[9] D. G. McVitie, L. B. Wilson, Three procedures for the stable marriage problem, Communications of the ACM 14,7 (1971) 491–492.

[10] K. Iwama, D. Manlove, S. Miyazaki, Y. Morita, Stable marriage with incomplete lists and ties, Proceedings of ICALP '99: the 26th International Colloquium on Automata, Languages and Programming 1644 (1999) 443–452.

[11] D. Manlove, R. Irving, K. Iwama, S. Miyazaki, Y. Morita, Hard variants of stable marriage, Theoretical Computer Science 276 (1-2) (2002) 261–279.

[12] I. Gent, P. Prosser, An empirical study of the stable marriage problem with ties and incomplete lists, Tech. Rep. APES-40-2002, APES Research Group (January 2002).

[13] D. Bianco, S. Hartke, A. Larimer, Stable matchings in the couples problem, Morehead Electronic Journal of Applicable Mathematics 2.

[14] C. Teo, J. Sethuraman, W. Tan, Gale-shapley stable marriage problem revisited: Strategic issues and applications, IPCO'99 (1999) 429–438.

[15] T. Kavitha, K. Mehlhorn, D. Michail, K. Paluch, Strongly stable matching in time o(nm) and extension to the hospitals-residents problem, STACS'04 (2004) 222–233.

[16] D. Manlove, Stable marriage with ties and unacceptable partners, Tech. Rep. TR-1999-29, Computing Science Department of Glasgow University (January 1999).

[17] R. Irving, Stable marriage and indifference, Discrete Applied Mathematics 48 (1994) 261–272.

[18] S. Bouchafa, B. Zavidovique, Stratégie de vote pour la mise en correspondance de lignes de niveaux, in: RFIA'04, Toulouse, 2004.

[19] J.L.Lisani, P. Monasse, L. Rudin, Fast shape extraction and applications, Tech. rep., Cahiers de l'ENS Cachan: C.M.L.A (2001).
URL `citeseer.nj.nec.com/557245.html`

[20] P. Monasse, F. Guichard, Fast computation of a contrast-invariant image representation, IEEE Trans. on Image Proc. 9 (5) (1998) 860–872.

[21] C. Ballester, E. Castan, M. Gonzalez, J. Morel, Contrast invariant image intersection, Tech. rep., Cahiers de l'ENS Cachan: C.M.L.A (1998).

[22] V. Caselles, B. Coll, J. Morel, Topographic maps and local contrast changes in natural images, International Journal of Computer Vision 33 (1) (1999) 5–27.

[23] B. Zavidovique, V. Serfaty, Image processing: Cooking or chemistry?, in: Keynote speech - 6th Int'l Conf. on Image Analysis and Processing COMO, 1991.

[24] B. Zavidovique, V.Serfaty, C.Fortunel, Image processing: from infancy to maturity, Journal of IEEE Software 8 (6) (1991) 37–50.

[25] B. Zavidovique, N. Suvonvorn, G. S. Seetharaman, A novel representation and algorithms for (quasi) stable marriages, in: ICINCO'05, 2005.

[26] N. Suvonvorn, B. Zavidovique, Eflam : A model to level-line junction extraction, in: VISAPP'06, Setubal Portugal, 2006.