



ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΨΗΦΙΑΚΗΣ ΤΕΧΝΟΛΟΓΙΑΣ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

## Πτυχιακή εργασία

Τίτλος Πτυχιακής:

Υποστήριξη λήψης αποφάσεων σε προβλήματα πολυ-  
κριτηριακής βελτιστοποίησης με χρήση οπτικής  
ανάλυσης δεδομένων.

Του Μηλιώτη Μάριου

Επιβλέπων καθηγητής: Τσερπές Κωνσταντίνος, Επίκουρος  
Καθηγητής στο Τμήμα Πληροφορικής και Τηλεματικής του  
Χαροκοπείου Πανεπιστημίου

**Υπόλοιπα μέλη τριμελούς επιτροπής:**

**Βασιλοπούλου Κωνσταντίνα**, Λέκτορας στο Τμήμα Πληροφορικής και Τηλεματικής του  
Χαροκοπείου Πανεπιστημίου

**Δημητρακόπουλος Γεώργιος**, Επίκουρος Καθηγητής στο Τμήμα Πληροφορικής και  
Τηλεματικής του Χαροκοπείου Πανεπιστημίου

**Μηλιώτης Μάριος**  
**A.M.: 21147**  
**Αθήνα, Σεπτέμβριος 2016**

**Τίτλος Πτυχιακής:** Υποστήριξη λήψης αποφάσεων σε προβλήματα πολυ-κριτηριακής βελτιστοποίησης με χρήση οπτικής ανάλυσης δεδομένων.

# Περιεχόμενα

Εισαγωγή .....	4
Abstract .....	7
1. Data analytics .....	10
2. Βελτιστοποίηση πολυ-κριτηριακών προβλημάτων.....	13
2.1 State of the art – Multi Objective Optimization .....	16
3. Visual Analytics .....	23
3.1 State of the art – Visual Analytics .....	25
4. Αγορά επεξεργαστή και κάρτας γραφικών. Περιγραφή του προβλήματος .....	30
5. Μοντέλο αντιμετώπισης του προβλήματος.....	32
5.1 Συλλογή δεδομένων επεξεργαστών και καρτών γραφικών .....	32
5.1.1 Επεξεργαστές .....	34
5.1.2 Κάρτες γραφικών.....	36
5.1.3 Συνδυασμοί επεξεργαστών και καρτών γραφικών.....	37
5.2 Μελέτη και υλοποίηση αναζήτησης των βέλτιστων λύσεων .....	38
5.3 Βελτιστοποίηση επιλογής συνδυασμού επεξεργαστή και κάρτας γραφικών .....	41
6. Υλοποίηση ιστοτόπου .....	42
7. Συμπεράσματα.....	46
8. Βιβλιογραφία.....	47

# Εισαγωγή

Πολλές φορές στη ζωή μας, ερχόμαστε αντιμέτωποι με καταστάσεις, στις οποίες καλούμαστε να πάρουμε μία απόφαση και να διαλέξουμε μία επιλογή. Ιδανικά, γνωρίζουμε ποια από τις επιλογές υπερέχει απέναντι στις υπόλοιπες, οπότε τίποτα δεν μας εμποδίζει από να την επιλέξουμε, χωρίς δεύτερη σκέψη. Ειδικότερα, όταν το πρόβλημα μας εξαρτάται από ένα και μόνο κριτήριο (μονο-κριτηριακό), είναι εύκολο να συγκρίνουμε τα δεδομένα μας με βάση αυτό το κριτήριο και να διακρίνουμε την καλύτερη λύση. Ακόμη όμως και όταν έχουμε παραπάνω από ένα κριτήριο, αν το πρόβλημα μας είναι απλό και τα δεδομένα μας λίγα, μπορούμε πάλι με μερικούς υπολογισμούς να καταλήξουμε σε μία τελική επιλογή. Δυστυχώς όμως, τις περισσότερες φορές αυτό δεν συμβαίνει, αλλά αντιθέτως ερχόμαστε αντιμέτωποι με πολύπλοκα προβλήματα, των οποίων τα δεδομένα χαρακτηρίζονται από πολλά κριτήρια, τα οποία συνήθως ανταγωνίζονται το ένα το άλλο. Έτσι, καταλήγουμε σε δίλημμα, καθώς δεν μπορούμε να διακρίνουμε ποια είναι η καλύτερη επιλογή. Τα προβλήματα αυτά ονομάζονται πολυ-κριτηριακά. Ειδικότερα, σε περιπτώσεις που καλούμαστε να συγκρίνουμε καταστάσεις οι οποίες δεν είναι μετρήσιμες και η τελική απόφαση πρόκειται να έχει μεγάλες συνέπειες, όπως οι επιπτώσεις μίας κατασκευής στο περιβάλλον και στους ανθρώπους, ή καταστάσεις στις οποίες τα δεδομένα είναι πολλά, οι υπολογισμοί αυξάνονται σε αριθμό και πολυπλοκότητα και επομένως η κατάληξη σε μία απόφαση ή ένα συμπέρασμα καθίσταται μία δύσκολη και επίπονη διαδικασία, η οποία εμπεριέχει και μεγάλη πιθανότητα σφάλματος.

Παρουσιάζεται λοιπόν η ανάγκη για ανάλυση των επιλογών, ώστε να είμαστε σίγουροι ότι θα καταλήξουμε στη σωστή απόφαση. Ο επιστημονικός κλάδος ο οποίος ασχολείται με την ανάλυση αυτή ονομάζεται **multiple-criteria decision analysis (MCDA)** ή **multiple-criteria decision making (MCDM)** και όπως δηλώνει η ονομασία του, έχει ως στόχο την υποστήριξη λήψης αποφάσεων σε πολυ-κριτηριακά προβλήματα. [1]

Το πρόσφατο, μεγάλο ενδιαφέρον που έχει εκδηλωθεί για την Πληροφορική, χάρη στην ραγδαία εξέλιξη που υπήρξε και υπάρχει τα τελευταία χρόνια στο χώρο του λογισμικού και της τεχνολογίας του υλικού των συστημάτων, έχει ως αποτέλεσμα την διαρκή αύξηση των

απαιτήσεων για επίλυση όλο και μεγαλύτερων και δυσκολότερων προβλημάτων απόφασης, με αυξημένο βαθμό πολυπλοκότητας και πλήθος δεδομένων. Για να ανταπεξέλθουμε στις απαιτήσεις αυτές, έπρεπε να βρούμε τρόπους να συλλέγουμε και να επεξεργαζόμαστε τα δεδομένα μας σε μικρό χρόνο (ώστε να μπορούμε να κάνουμε συγκρίσεις και προβλέψεις μεταξύ διαφορετικών συνόλων δεδομένων) και να μπορούμε να τα ερμηνεύσουμε, ώστε να εξάγουμε εύληπτα και ουσιώδη συμπεράσματα και να καταλήγουμε εύκολα στην τελική απόφαση. Δόθηκε λοιπόν η ευκαιρία σε μερικά επιστημονικά πεδία να ερευνηθούν και να αναπτυχθούν, με σκοπό να προσφέρουν τα κατάλληλα εργαλεία για τη δουλειά αυτή:

- Στα **data analytics**, στα οποία στόχος είναι η ανάπτυξη τεχνικών οι οποίες θα μπορούν σε μικρό χρονικό διάστημα, να διεκπεραιώσουν το **data mining**, δηλαδή να συλλέξουν τη ζητούμενη πληροφορία, να την επεξεργαστούν και να την οργανώσουν σε μοντέλα, σύμφωνα με τις απαιτήσεις του προβλήματος. Πολλές είναι οι γλώσσες προγραμματισμού οι οποίες έχουν συμπεριλάβει στο δυναμικό τους λειτουργίες που διευκολύνουν την ανάπτυξη τέτοιων μοντέλων, όπως η R, η Python, η SQL κ.ά. Στη συνέχεια, τα δεδομένα αυτά αναλύονται με την εφαρμογή στατιστικών μεθόδων, υποθέσεων και άλλων εργαλείων πάνω στα αποτελέσματα database ερωτημάτων, ώστε να προκύψει χρήσιμη πληροφορία και συμπεράσματα..
- Στην **βελτιστοποίηση πολυ-κριτηριακών προβλημάτων**, η οποία μπορεί να προσφέρει πολύτιμη βοήθεια στην ανάλυση και στο decision-making, καθώς φιλτράρεται η περιττή πληροφορία και μένουν δεδομένα, τα οποία ικανοποιούν τις απαιτήσεις του προβλήματος.
- Στα **visual analytics**, με τον οποίων τη χρήση μπορούμε στα δεδομένα που έχουν προκύψει από τις παραπάνω διαδικασίες, να εφαρμόσουμε data visualization, το οποίο αφορά τους τρόπους που χρησιμοποιούνται, ώστε να προσφέρεται στο χρήστη μία γραφική αναπαράσταση των δεδομένων και η οποία, μέσω διαδραστικών βημάτων, μπορεί να τον φέρει στη θέση να εξάγει συμπεράσματα

για πολύπλοκα προβλήματα, με γρήγορο και αποτελεσματικό τρόπο και να καταλήξει σε αποφάσεις..

Όλες οι προαναφερθείσες τεχνικές, ορίζονται και αναλύονται εκτενέστερα στα συνέχεια της εργασίας.

Ενδεικτικά, μερικά παραδείγματα προβλημάτων στα οποία ενσωματώνονται τα ζητήματα και οι δυσκολίες που περιγράφηκαν παραπάνω είναι η πρόβλεψη της εξάπλωσης ενός ιού για την καταπολέμησή του (H1N1 2009, Google) ή η πρόβλεψη της αυξομείωσης της τιμής ενός προϊόντος, ώστε να βρεθεί η πιο κατάλληλη στιγμή για την αγορά του π.χ. αεροπορικά εισιτήρια, ξενοδοχεία. [2] Και στα δύο αυτά προβλήματα, ο όγκος δεδομένων που πρέπει να αντληθεί από τις διάφορες πηγές, η επεξεργασία που πρέπει να γίνει για να εξαχθούν συμπεράσματα που να έχουν λογική και νόημα, καθώς και η αναπαράσταση με έναν τρόπο που θα μπορεί να δίνει άμεσα την πληροφορία σε αυτόν στον οποίο προβάλλεται, αποτελούν δυσκολίες που απαιτούν, πέρα από την μεγάλη υπολογιστική ισχύ και το data analysis, προσπάθεια και έρευνα για να επιλυθούν.

# Abstract

Several times in our lives, we find ourselves in situations where we need to make decisions and choices. Ideally, we know which choice among them outmatches the others, so we can easily choose the right one. Especially when there is only a single criterion to compare (single-criterion problems), it's easy to compare our data and find the best solution. Even when there are more criteria than one, if our problem is simple and the number of data is low, we can come up with a solution just by making a few calculations. Unfortunately though, most of the time, that's not the case. Instead, we are facing complex problems, with lots of data and multiple and often conflicting criteria. Thus, we end up with a dilemma, as we cannot decide which choice is right for us. Such problems are called multi-criteria problems. Notably, in situations where we have to compare criteria that are uncountable and the final decision of which might have big consequences, such as the impact of a construction on the environment and the society, or situations where the number of data is huge, the calculations that need to be done increase in number and complexity, making the decision-making or the conclusion a difficult and cumbersome process, with a high probability of error in the final choice.

As a result, the need of analyzing our choices is created, so that we can be sure that we will make the best possible decision in order to solve our problem. The field of science that studies this analysis is called **multiple-criteria decision analysis** (MCDA) or **multiple-criteria decision making** (MCDM) and, as its name implies, aims at supporting the decision making in multiple-criteria problems. [1]

The recent, big interest in Informatics, thanks to the rapid growth of the electronic system software and hardware during the past years, has led us into willing to solve continuously bigger and more complex decision problems. To cope with the new requirements, we had to find ways to efficiently collect and process data in short times (in order to be able to make comparisons and estimations between different data sets) and interpret them in useful ways for us, so as to come to intuitive and substantial conclusions and reach easier to a final decision. This left the door open



for research and development in some other fields of science, which aimed in developing the right tools for solving such problems:

- In **data analytics**, the aim is to develop techniques which will enable us to complete in short times the **data mining** process; namely, the collection of the desired information, the processing of it and the structure of the data in models, according to the requirements of the problem. Many programming languages have built-in functions which facilitate the construction of such models, such as R, Python, and SQL. Afterwards, these data are analyzed with the use of statistical methods, hypothesis testing and other tools, which are applied to the results of database queries we make, so as to extract useful information and conclusions.
- **Multi-objective optimization**, which may provide valuable help in analyzing and decision-making, as it offers us the option to filter redundant data, leaving us only the information which will help us solve our problem.
- **Visual analytics**, which include the data visualization, which refers to the visualization of the data that occurred from the above-mentioned methods, in order to offer a graphical representation of them to the user, and the interactive methods which the user can follow, in order to solve complex problems and make decisions quickly and efficiently.

All of the above-mentioned techniques are defined and analyzed extensively during the next chapters.

Some problems that incorporate the issues and the difficulties that were described above and could serve as examples are the prediction of the epidemic spread of a virus in order to cure it (H1N1 2009, Google) or the prediction of the price fluctuation of a product, in order to find the best moment to buy it e.g. flight tickets, hotel bookings. [2] In both of these problems, the amount of data that needs to be extracted from the different sources, the processing that needs to be done in order to come into reasonable and meaningful conclusions and the visualization, which must give direct information to those that it's presented, constitute a bigger, more

complex problem that calls for not only increased computational power and efficient data analysis, but also hard work, effort and research, so it can be solved.

# 1. Data analytics

Ήδη από τα παλαιότερα χρόνια, ο άνθρωπος άρχισε να καταγράφει δεδομένα και γεγονότα (π.χ. με τα βιβλία), ώστε να μπορεί να ανακαλεί στοιχεία για μία δεδομένη στιγμή όποτε το θελήσει. Τα στοιχεία αυτά μπορεί να αφορούσαν γεγονότα, ποσότητες, αριθμούς, χαρακτηριστικά, με σκοπό την μεταλαμπάδευση της γνώσης, τη σύγκριση του παρόντος με το παρελθόν κ.ο.κ. Σαν συνέπεια, τα δεδομένα άρχισαν να συσσωρεύονται, οπότε έπρεπε να αναζητηθεί ένας τρόπος να αποθηκεύονται κάπου, ώστε να διαφυλάσσονται. Μία τέτοια κατασκευή π.χ. όταν μιλάμε για βιβλία, είναι η βιβλιοθήκη. Στη συνέχεια, όταν ο όγκος των στοιχείων αυτών μεγάλωσε κι άλλο, πέρα από το πρόβλημα της αποθήκευσης, προέκυψε και ένα άλλο θέμα: η δυσκολία στο να βρεθεί το δεδομένο το οποίο αναζητούνταν κάθε φορά, ανάμεσα στα τόσα άλλα. Έτσι, εισήχθη η ανάγκη για οργάνωση των αποθηκευμένων δεδομένων.

Η δίψα για περισσότερη γνώση και η καταγραφή φυσικά αυτής, αποτελούν γνωρίσματα του ανθρώπου στα οποία δεν επέρχεται κορεσμός. Το 1944, ο Fremont Ryder, βιβλιοθηκάριος στο πανεπιστήμιο του Wesleyan, δημοσίευσε ένα άρθρο με τίτλο *The Scholar and the Feature of the Research Library*, στο οποίο εκτίμησε ότι οι βιβλιοθήκες των Αμερικανικών πανεπιστημίων διπλασιάζονται σε μέγεθος κάθε 16 χρόνια και ότι με αυτόν τον ρυθμό, η βιβλιοθήκη του Yale, σε διάστημα 100 ετών, υπολόγισε ότι θα στεγάζει 200.000.000 τόμους και ότι θα χρειάζεται περισσότερο από 5.000 άτομα ως προσωπικό, για να λειτουργήσει αποτελεσματικά. [3, pp. 9, 13]

Η εφεύρεση του ηλεκτρονικού υπολογιστή, όπως μπορεί εύκολα και με ασφάλεια να υποθέσει κανείς, έφερε επανάσταση στον τομέα των δεδομένων. Η ψηφιοποίησή τους κατέστησε την αποθήκευσή τους εύκολη και φθηνή, ενώ η οργάνωση και ο υπολογισμός τους γινόταν πια αποτελεσματικότερα. Δεν άργησε όμως η στιγμή που το ίδιο πρόβλημα που δημιουργούνταν στις βιβλιοθήκες, με την αύξηση των δεδομένων, άρχισε να κάνει την εμφάνισή του και στα ηλεκτρονικά συστήματα.

Αν και η ακριβής χρονολογία που πρωτοαναφέρθηκε σαν όρος δεν είναι εύκολο να προσδιοριστεί, το «Big Data» επινοήθηκε για να εκφράσει ακριβώς αυτό το πρόβλημα: ο όγκος

των δεδομένων είναι τόσο μεγάλος και πολύπλοκος, ώστε οι πόροι του υπολογιστή παύουν να επαρκούν (είτε από άποψη μνήμης, είτε από άποψη χώρου στο δίσκο, είτε από άποψη σχέσης χρόνου-υπολογισμού). Στις αρχές της χιλιετίας που διανύουμε, δημοσιεύτηκε μία έρευνα με τίτλο “How much information?”, στην οποία η πληροφορία που παράχθηκε από εκτυπώσεις, φιλμ και μαγνητικά και οπτικά μέσα για το έτος 2002 υπολογίστηκε και βρέθηκε ίση με περίπου 5 Exabyte - αυξημένη, βάση εκτίμησης, δηλαδή κατά 30% μέσα σε τρία χρόνια (1999-2002). [4] Με την εξάπλωση της τεχνολογίας σε ανθρώπους ανεξαρτήτως τεχνολογικής γνώσης και αντίληψης, καθώς και ηλικίας, τα τελευταία χρόνια, ο αριθμός αυτός, έχει εκτοξευτεί, με τα συνολικά δεδομένα που παράγονται καθημερινώς από οποιαδήποτε ηλεκτρονική πηγή, όπως αισθητήρες, κινητά τηλέφωνα, το internet κτλ. να φτάνουν σε ποσότητα ίση με 2.5 Exabyte ημερησίως. [5]

Η πρόκληση φυσικά ήταν η εκμετάλλευση του Big Data. Από τη στιγμή της εμφάνισης του, άρχισε να αποτελεί εμπόδιο σχεδόν σε κάθε ανάγκη οργανισμών ή επιχειρήσεων για εξαγωγή χρήσιμων συμπερασμάτων από τα δεδομένα που έχουν στην κατοχή τους. Όμως, ας μην ξεχνάμε ότι ο όγκος του προκύπτει από τη δική μας επιλογή να συλλέξουμε όλο το τεράστιο αυτό πλήθος των δεδομένων, τα οποία κουβαλάει όλη τη πληροφορία που χρειαζόμαστε για να καταλήξουμε σε κάποιο συμπέρασμα. Είτε μιλάμε για φυσικά ή ιατρικά πειράματα, είτε για ερμηνεία συμπεριφοράς ενός πληθυσμού ή ενός καταναλωτικού κοινού, είτε για τα οικονομικά στοιχεία μιας επιχείρησης, όπως και άλλα παρόμοια παραδείγματα, τα αποτελέσματα από την επεξεργασία του Big Data και η δυνατότητα να μπορούμε να λάβουμε τις σωστές αποφάσεις, οι οποίες θα μας οδηγήσουν στις σωστές κινήσεις, δίνουν γνώση και δύναμη σε αυτόν που μπορεί να το χειριστεί.

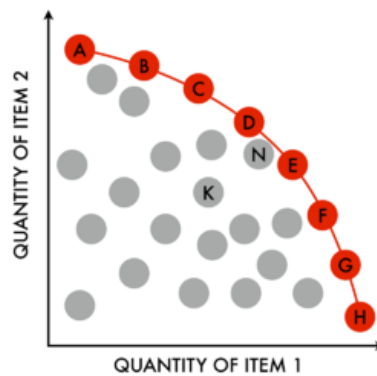
Καθότι το hardware από μόνο του δεν ήταν ικανό να ανταπεξέλθει στις νέες απαιτήσεις, ήταν λογικό να αναζητηθούν άλλες στρατηγικές για την επίλυση του προβλήματος που προέκυψε. Το βάρος, επομένως, έπεσε στην αποτελεσματικότερη ανάλυση των σετ δεδομένων που προέκυπταν. Για την επίτευξή της αναπτύχθηκαν εργαλεία, τα οποία στοχεύουν στην αυτόματη συλλογή δεδομένων και την εφαρμογή υποθέσεων και στατιστικών μεθόδων για την ανάλυση τους μέσω database ερωτημάτων, για την εξαγωγή συμπερασμάτων και τη λήψη αποφάσεων. Η συλλογή των εργαλείων αυτών ονομάζεται **data analytics**.

Η αποτελεσματικότητα της ανάλυσης εξαρτάται φυσικά από τα συστατικά της. Η συλλογή των δεδομένων, το **data mining**, πρέπει να γίνεται με τέτοιο τρόπο, ώστε να συλλέγεται, όσο γίνεται, μόνο η χρήσιμη πληροφορία και ο τρόπος με τον οποίο θα γίνεται και θα πρέπει να είναι “έξυπνος”, ώστε να μην χρειάζεται ανθρώπινη παρέμβαση, να μειώνονται τα σφάλματα και ο χρόνος εκτέλεσης της όλης διαδικασίας να ελαχιστοποιείται. Επίσης, θα πρέπει να δίνεται η δυνατότητα να εντοπίζονται patterns που ακολουθούν τα δεδομένα. Θα δούμε αργότερα, όταν μιλήσουμε για τα visual analytics, ότι και το machine learning μπορεί να βοηθήσει σημαντικά σε αυτή τη διαδικασία. Στη συνέχεια, τα δεδομένα που θα προκύψουν μπορεί να έχουν μεγάλο αριθμό, οπότε πρέπει να γίνουν τα κατάλληλα queries, εκμεταλλευόμενοι στο μέγιστο όλες τις δυνατότητες που μας δίνει η εκάστοτε βάση δεδομένων (MapReduce, in-database analytics, in-memory databases κ.ά.), ώστε να εξαχθούν μόνο τα σετ δεδομένων που χρειαζόμαστε, και έπειτα αναλαμβάνει η ανάλυση, μέσω στατιστικών μεθόδων [6]. Τα visual analytics είναι άλλος ένας πολύ χρήσιμος σύμμαχος στην ανάλυση των δεδομένων και όπως ειπώθηκε στην εισαγωγή, θα περιγραφτούν αναλυτικά στο κεφάλαιο 3 της εργασίας.

Σε μία έρευνα της TDWI, το 2009, το 38% των οργανισμών που συμμετείχαν στην έρευνα δήλωσαν ότι χρησιμοποιούν advanced analytics, ενώ το 85% δήλωσαν ότι πρόκειται να τα εφαρμόσουν μέσα στα επόμενα 3 χρόνια. [6] Παρατηρείται λοιπόν μία τάση προς εκείνη την κατεύθυνση και είναι λογικό, αφού οφείλει κάθε οργανισμός ή επιχείρηση να προσαρμοστεί στις νέες προκλήσεις που θέτουν οι νέες τεχνολογίες και στον υγιή ανταγωνισμό μεταξύ τους, ο οποίος θα ωθήσει την έρευνα να φτάσει σε ακόμη υψηλότερο επίπεδο. Άλλωστε, είναι ο τρόπος μας να ανακαλύψουμε πληροφορίες που δεν θα μπορούσε να συνδυάσει το ανθρώπινο μυαλό από μόνο του, πληροφορίες οι οποίες μπορεί είναι πολύτιμες σε καταστάσεις που απαιτούν σημαντικές αποφάσεις και συμπεράσματα..

## 2. Βελτιστοποίηση πολυ-κριτηριακών προβλημάτων

Η βελτιστοποίηση ενός προβλήματος αναφέρεται στο να βρεθεί ένας τρόπος να καταλήξουμε σε μία επιλογή, η οποία θα έχει πραγματική υπόσταση (χρειαζόμαστε πραγματικές λύσεις, οι οποίες να λύνουν πρακτικά το πρόβλημα μας) και να ικανοποιεί τους στόχους που έχουν τεθεί. Ανάλογα με το πρόβλημα, η αναζήτηση αυτής της επιλογής μπορεί να έχει ως στόχο ένα σύνολο από βέλτιστες λύσεις, ένα γράφημα το οποίο να αναπαριστά τις τιμές κάθε λύσης και τις διαφορές μεταξύ τους ή μία μοναδική λύση από το σύνολο των βέλτιστων, η οποία θα ικανοποιεί κάποιες προτιμήσεις του ατόμου που έχει κληθεί να λάβει την απόφαση (του decision maker – DM – όπως συνηθίζεται να ονομάζεται). Οι βέλτιστες αυτές λύσεις ονομάζονται αλλιώς non-dominated ή Pareto optimal. Το σύνολο των λύσεων, στο οποίο η κάθε μία θα προσφέρει μια εναλλακτική ισορροπία μεταξύ των κριτηρίων, στο βέλτιστο πάντα βαθμό σε σχέση με τις λύσεις εκτός συνόλου το έχουμε ονομάσει **Pareto frontier** ή **Pareto set** ή **Pareto front** ή **Non-dominated set**, ονομασία την οποία μπορούμε να κατανοήσουμε καλύτερα μέσω της **Εικόνας 1** παρακάτω. Στην εικόνα αυτή φαίνονται με κόκκινο οι βέλτιστοι συνδυασμοί ποσοτήτων 2 προϊόντων από τα οποία μπορεί να επιλέξει ένας παραγωγός, για να έχει βέλτιστες αποδόσεις στην παραγωγή του και με γκρι χρώμα οι υπόλοιποι συνδυασμοί. Παρατηρούμε ότι οι βέλτιστες επιλογές δημιουργούν ένα ευδιάκριτο “φράγμα” μπροστά από τις υπόλοιπες επιλογές, όταν το διάγραμμα είναι 2 διαστάσεων και σκοπός μας είναι να μεγιστοποιήσουμε τα κριτήριά μας. Αυτό είναι λογικό, αν αναλογιστεί κανείς ότι το μέγιστο υποδηλώνει τα ψηλότερα σημεία μιας καμπύλης. Σε προβλήματα περισσότερων κριτηρίων, και άρα περισσότερων διαστάσεων, η απεικόνιση δεν είναι τόσο εύκολη, όμως με ανάλογο τρόπο και εκεί, οι βέλτιστες λύσεις υπερσχύουν των μη βέλτιστων.



*Εικόνα 1.: Αναπαράσταση του Pareto front των ποσοτήτων 2 προϊόντων που πρόκειται να βγουν στην παραγωγή*

Το πρόβλημα της βελτιστοποίησης αποτελεί επίσης πρόβλημα το οποίο απασχολεί τον άνθρωπο από τα παλιά χρόνια. Ο πρώτος ο οποίος μίλησε για βέλτιστες επιλογές σε οικονομικά προβλήματα πολλών κριτηρίων ήταν ο Francis Ysidro Edgeworth, ο οποίος στο βιβλίο του με τίτλο “Mathematical Physics” του 1881 [7], εισήγαγε στον τομέα των οικονομικών τον όρο της καμπύλης αδιαφορίας, η οποία είναι ένα γράφημα με ένα σύνολο επιλογών, από το οποίο ο καταναλωτής αδιαφορεί για το τι θα επιλέξει [8]. Την καμπύλη αυτή χρησιμοποίησε αργότερα ο Vilfredo Pareto στο βιβλίο του “Manual of Political Economy” [9], και ήταν αυτός που περιέγραψε πρώτος την έννοια του Pareto optimality, μια κατάσταση στην οποία βρίσκεται ένα σύστημα, όταν καμία από τις μεταβλητές που το αποτελούν δεν μπορεί να γίνει καλύτερη από τις άλλες, χωρίς αυτές να γίνουν χειρότερες. Το σύνολο αυτό δεν περιγράφει τίποτα άλλο παρά το Pareto front, το οποίο αναφέραμε στην ακριβώς προηγούμενη παράγραφο, και το οποίο πήρε το όνομα από τον εμπνευστή του.

Στην διαδικασία αναζήτησης του Pareto front, οι περισσότεροι αλγόριθμοι χρησιμοποιούν την έννοια της κυριαρχίας. Γενικά ισχύει ο παρακάτω ορισμός:

*«Μία λύση A κυριαρχεί μία άλλη λύση B όταν 1) η A δεν είναι σε κανένα κριτήριο χειρότερη από την B και 2) όταν η A είναι τουλάχιστον σε ένα κριτήριο καλύτερη από την B» [10]*

Αξίζει να τονισθεί το εξής: το γεγονός ότι μια λύση δεν κυριαρχεί μία άλλη, δεν σημαίνει ότι η δεύτερη κυριαρχεί την πρώτη. Για παράδειγμα, ας υποθέσουμε ότι έχουμε 2 λύσεις, τις οποίες συγκρίνουμε ως προς 2 κριτήρια: η A έχει τις τιμές 5 και 3 και η B 7 και 2. Η A δεν κυριαρχεί τη

B, αφού είναι χειρότερη από αυτή στο πρώτο κριτήριο, αλλά ούτε η B κυριαρχεί την A, αφού είναι χειρότερη στο δεύτερο. Στην περίπτωση αυτή, λέμε ότι οι 2 λύσεις αλληλοεξουδετερώνονται μεταξύ τους.

Αναδιατυπώνοντας λοιπόν, αναζητούμε το σύνολο το οποίο θα περιέχει τις λύσεις εκείνες, οι οποίες θα κυριαρχούν στις λύσεις εκτός συνόλου και θα αλληλοεξουδετερώνονται μεταξύ τους (το Pareto front).

Για την εύρεση του Pareto front, αρκετοί είναι οι αλγόριθμοι που έχουν προταθεί, ο καθένας όμως παρουσιάζει διαφορετική πολυπλοκότητα, ανάλογα με το πρόβλημα που επιλύει. Μερικοί από αυτούς τους αλγόριθμους είναι: [10]

- Ο Naive and slow, στον οποίο κάθε λύση συγκρίνεται με όλες τις υπόλοιπες, μέχρι να βρεθεί κάποια η οποία θα την κυριαρχεί. Αν δεν κυριαρχείται από καμία, τότε ανήκει στο Pareto front.
- Ο Continuously Updated, ο οποίος αρχικοποιεί ένα σύνολο με τις pareto optimal λύσεις και προσθέτει σε αυτό την πρώτη λύση. Έστερα, συγκρίνει κάθε επόμενη με τις λύσεις στο σύνολο αυτό. Αν κάποια λύση κυριαρχεί κάποια από αυτό το σύνολο, τότε την αφαιρεί από αυτό. Αν κυριαρχεί όλες τις λύσεις του συνόλου ή αλληλοεξουδετερώνεται με κάποια, τότε προστίθεται και αυτή στο σύνολο. Αλλιώς, αν κυριαρχείται, σημαίνει ότι δεν είναι Pareto optimal λύση και συνεχίζουμε με την επόμενη. Οι λύσεις που θα μείνουν στο τέλος στο σύνολο, αποτελούν το Pareto front.
- Ο αλγόριθμος του Kung et Al, ο οποίος εκμεταλλεύεται την ταξινόμηση σε ένα κριτήριο για να μειώσει το χρόνο εκτέλεσης. [10, p. 38]

Εφόσον βρεθούν οι κυρίαρχες λύσεις, δηλαδή οι βέλτιστες, έχουμε ολοκληρώσει την πολυ-κριτηριακή μας ανάλυση και έχει γίνει ένα σημαντικό βήμα, ώστε να βρεθεί η λύση αυτή που ικανοποιεί περισσότερο από όλες τις απαιτήσεις του προβλήματος. Βασικός στόχος όμως είναι η βελτιστοποίηση και να βοηθήσουμε τον decision maker να καταλήξει σε μία τελική επιλογή, η οποία θα ταιριάζει στις προτιμήσεις του. Εργαλεία και τεχνικές που βοηθούν τη διαδικασία αυτή απόφασης του χρήστη, αναλύονται στο κεφάλαιο 2.1.



Η βελτιστοποίηση προβλημάτων, πέρα από την εμφάνισή της σε απλές, καθημερινές πτυχές της ζωής του ανθρώπου σε πολύ απλή μορφή, χρησιμοποιείται κατά κόρον σε προβλήματα μηχανικής, βιομηχανικής παραγωγής προϊόντων, σχεδιασμού και κατασκευών, όπως είναι π.χ. ο σχεδιασμός συστημάτων άρδευσης, παροχής ηλεκτρικού ρεύματος, χρήσης υλικών και του κόστους τους στην παραγωγή ρούχων κ.ά. Επίσης χρησιμοποιείται στα οικονομικά και το marketing, όταν πρέπει να “ζυγιστούν” τα πλεονεκτήματα και τα μειονεκτήματα επενδυτικών αποφάσεων, στις τηλεπικοινωνίες και σε διάφορους άλλους τομείς.

## 2.1 State of the art – Multi Objective Optimization

Λόγω της συχνής εμφάνισης της πολύ-κριτηριακής βελτιστοποίησης ως ανάγκη για την επίλυση προβλημάτων, πολλές είναι οι τεχνικές και οι αλγόριθμοι που έχουν προταθεί για την υλοποίηση της, καθεμία από τις οποίες προσφέρει διαφορετικά πλεονεκτήματα και μειονεκτήματα.

Ανάλογα με την προεργασία που απαιτεί κάθε αλγόριθμος πριν τη διαδικασία της βελτιστοποίησης, χωρίζονται στις ακόλουθες μεθόδους: [11]

1. No-preference methods.
2. A priori methods.
3. A posteriori methods.
4. Interactive methods.

Στις **no-preference** μεθόδους, καμία υπόθεση δεν γίνεται για την βαρύτητα των κριτηρίων, ή με άλλα λόγια, ο decision maker δεν χρειάζεται να δώσει καμία πληροφορία για τις προτιμήσεις του σχετικά με τις λύσεις που θα αναζητηθούν. Τέτοια μέθοδος είναι η global criterion, με τη χρήση της οποίας λύνεται ένα πολυ-κριτηριακό πρόβλημα της μορφής

$$\begin{array}{ll} \min & \|f(x) - z^{ideal}\| \\ \text{s.t.} & x \in X \end{array}$$

το οποίο έχει μετατραπεί με τις κατάλληλες διαδικασίες σε μονο-κριτηριακό [12]. Στο παραπάνω πρόβλημα,  $\|\cdot\|$  μπορεί να είναι οποιαδήποτε  $\mathbf{L}_p$  νόρμα, με συνήθεις επιλογές τις  $\mathbf{L}_1$ ,  $\mathbf{L}_2$  και  $\mathbf{L}_\infty$  [13]. Αυτές οι μέθοδοι δεν προσπαθούν να βρουν πολλαπλές βέλτιστες λύσεις που ανήκουν στο Pareto front, παρά μόνο μία [10].

Στις **a-priori** μεθόδους, επαρκής πληροφορία για τις προτιμήσεις σχετικά με τις λύσεις που θα προκύψουν, πρέπει να γίνεται γνωστή πριν ξεκινήσει η διαδικασία της αναζήτησης των βέλτιστων λύσεων. Γνωστές a priori μέθοδοι είναι οι scalarizing μέθοδοι, στις οποίες διατυπώνονται μονο-κριτηριακά πρόβλήματα, των οποίων οι λύσεις είναι οι Pareto optimal του πολυ-κριτηριακού προβλήματος που θέλουμε να λύσουμε [14] (μέθοδοι όπως η linear scalarization και η  $\epsilon$ -constraint) και οι goal programming μέθοδοι, όπως η utility function και η lexicographic method: [7]

- Στη **linear scalarization μέθοδο** ή **weighted sum** υπολογίζεται το

$$\min_{x \in X} \sum_{i=1}^k w_i f_i(x),$$

όπου τα βάρη των κριτηρίων  $w_i > 0$  είναι είτε οι παράμετροι του scalarization είτε βάρη τα οποία έχει θέσει ο decision maker για να εκφράσει τις προτιμήσεις του.

- Η  $\epsilon$ -constraint μέθοδος

$$\begin{aligned} \min \quad & f_j(x) \\ \text{s.t.} \quad & x \in X \\ & f_i(x) \leq \epsilon_j \text{ for } i \in \{1, \dots, k\} \setminus \{j\}, \end{aligned}$$

όπου τα άνω όρια  $\epsilon_j$  είναι οι παράμετροι, όπως παραπάνω, και η  $f_j$  είναι το κριτήριο που πρέπει να ελαχιστοποιηθεί.

- Η **utility function μέθοδος** προϋποθέτει ότι έχει δημιουργηθεί πρώτα η utility function (συνάρτηση χρησιμότητας) του decision maker. Η συνάρτηση χρησιμότητας είναι μία συνάρτηση στην οποία έχει γίνει μία αντιστοιχία  $\mathbf{u} : \mathbf{Y} \rightarrow \mathbb{R}$  και για όλα τα  $\mathbf{y}^1, \mathbf{y}^2 \in \mathbf{Y}$  ισχύει ότι  $\mathbf{u}(\mathbf{y}^1) > \mathbf{u}(\mathbf{y}^2)$ , όταν ο decision maker προτιμάει το  $\mathbf{y}^1$  από το  $\mathbf{y}^2$  και  $\mathbf{u}(\mathbf{y}^1) = \mathbf{u}(\mathbf{y}^2)$ , όταν ο decision maker είναι

ουδέτερος ως προς τις λύσεις. Όταν βρεθεί το  $\mathbf{u}$ , αρκεί για να λύθει η  $\max \mathbf{u}(\mathbf{f}(\mathbf{x}))$  για τα  $\mathbf{x} \in \mathbf{X}$ , όμως στην πράξη είναι πολύ δύσκολο να κατασκευαστεί μία utility function η οποία θα εκφράζει με ακρίβεια τις προτιμήσεις του decision maker, καθώς το Pareto front είναι άγνωστο πριν ξεκινήσει η αναζήτηση για τις βέλτιστες λύσεις.[11], [10, p. 38]

- Στη **lexicographic μέθοδο**, τα κριτήρια κατατάσσονται σε σειρά προτιμήσεως. Χωρίς βλάβη της γενικότητας, θεωρείται ότι όσο πιο χαμηλά βρίσκεται ένα κριτήριο στην κατάταξη, τόσο περισσότερο προτιμάται από τον decision maker. Στη συνέχεια, αναζητούνται λύσεις που ικανοποιούν τους περιορισμούς του πρώτου κριτηρίου. Αν το σύνολο των λύσεων που θα προκύψει είναι μεγαλύτερο της μίας λύσης, τότε προστίθενται στους περιορισμούς του πρώτου κριτηρίου και οι απαιτήσεις του δεύτερου σε προτίμηση. Η διαδικασία σταματάει όταν βρεθεί μία μοναδική λύση, αφού θα έχουν ικανοποιηθεί όλοι οι περιορισμοί μέχρι εκείνο το σημείο και επομένως, η αναζήτηση με βάση τους περιορισμούς των υπολοίπων λιγότερο σημαντικών κριτηρίων, θα είναι περιττή. [10], [11]

Σε αυτές τις μεθόδους, το πλεονέκτημα είναι ότι οι λύσεις είναι κοντά στις απαιτήσεις του decision maker, όμως είναι δύσκολο για αυτόν να εκφράσει τις προτιμήσεις του πριν δει τις Pareto optimal λύσεις.

Στις **a posteriori** μεθόδους:

1. Υπολογίζεται πρώτα ένα σύνολο από λύσεις.
2. Στη συνέχεια, ο decision maker μελετάει το σύνολο που έχει προκύψει και επιλέγει από αυτό τη λύση ή τις λύσεις που τον ικανοποιούν περισσότερο.

Σε προβλήματα με 2 κριτήρια, το Pareto front (το οποίο συνηθίζεται να ονομάζεται trade-off curve σε αυτές τις περιπτώσεις), μπορεί να σχεδιαστεί σε ένα γράφημα κάθετων αξόνων, με κάθε κριτήριο να αντιστοιχίζεται σε έναν άξονα. Έτσι, ο decision maker πληροφορείται για τις τιμές κάθε κριτηρίου και τις διαφορές μεταξύ τους και μπορεί να δει άμεσα πως η βελτίωση του ενός κριτηρίου επηρεάζει το άλλο, μέσω της καμπύλης του Pareto front και να καταλήξει στη

βέλτιστη, για αυτόν, επιλογή. [15] Σε προβλήματα με περισσότερα κριτήρια, τα πράγματα δεν είναι τόσο απλά, επιβάλλοντας την χρήση άλλων τεχνικών, οι οποίες θα συζητηθούν στο κεφάλαιο των visual analytics.

Οι περισσότερες a posteriori μέθοδοι ανήκουν σε μία από τις παρακάτω κατηγορίες: [11]

1. Στις **mathematical programming**, οι οποίες λύνουν το πολυ-κριτηριακό πρόβλημα μετατρέποντάς το σε μονο-κριτηριακό. Για κάθε κριτήριο που μετατρέπεται, προκύπτει μία λύση, είτε βέλτιστη τοπικά είτε βέλτιστη ολικά. Αυτές οι μέθοδοι στοχεύουν στο να δημιουργηθεί ένα σύνολο το οποίο θα περιέχει ομοιόμορφα κατανεμημένα Pareto σημεία, τα οποία θα αποτελούν μία ομοιόμορφα κατανεμημένη προσέγγιση του πραγματικού Pareto front του πολυ-κριτηριακού προβλήματος. Παραδείγματα τέτοιων μεθόδων είναι η Normal Boundary Intersection (NBI) [16], η Modified Normal Boundary Intersection (NBIm) [17], η Normal Constraint (NC) [18], [19], η Successive Pareto Optimization (SPO) [20] και η Directed Search Domain (DSD) [21].
2. Στους **Evolutionary Algorithms**. Οι περισσότεροι evolutionary multi-objective optimization (EMO) αλγόριθμοι που χρησιμοποιούμε, εφαρμόζουν Pareto-based συστήματα κατάταξης. Το κυριότερο πλεονέκτημα τους, όταν εφαρμόζονται στην επίλυση πολυ-κριτηριακών προβλημάτων, είναι ότι συνήθως καταλήγουν σε ένα σετ από λύσεις, προσφέροντας έτσι τον υπολογισμό μίας προσέγγισης ολόκληρου του Pareto front. Από την άλλη, το κύριο μειονέκτημά τους είναι η μειωμένη ταχύτητα εκτέλεσης σε σχέση με άλλες μεθόδους και το γεγονός ότι δεν μπορούν να εγγυηθούν το ότι οι λύσεις που θα προκύψουν θα είναι pareto optimal, παρά μόνο ότι θα αλληλοεξουδετερώνονται μεταξύ τους. Μερικοί καθιερωμένοι evolutionary αλγόριθμοι είναι ο Non-dominated Sorting Genetic Algorithm – II (NSGA-II) [22], ο Strength Pareto Evolutionary Algorithm 2 (SPEA-2) [23], ενώ σημαντικές είναι και οι μέθοδοι που χρησιμοποιούν particle swarm optimization και simulated annealing [24].

Το 2015, παρουσιάστηκε ένα νέο παράδειγμα evolutionary αλγορίθμων πολυ-κριτηριακής βελτιστοποίησης βασισμένων στο novelty search [25]. Στο novelty search, “επιβραβεύονται” οι λύσεις οι οποίες αντί να τείνουν προς τον βασικό στόχο, παρουσιάζουν διαφορετική κατά κάποιο μέτρο, συμπεριφορά ως προς τις υπόλοιπες, είναι δηλαδή “καινοτόμες”. [26] Στο παράδειγμα που παρουσιάστηκε, πέρα από την αναζήτηση για τις non-dominated λύσεις, αναζητούνται και novel λύσεις, με αποτέλεσμα να καθίσταται ιδιαίτερα χρήσιμη τεχνική για την επίλυση κάποιων πολυ-κριτηριακών προβλημάτων. Ο πρώτος αλγόριθμος που στηρίζεται σε αυτό το παράδειγμα είναι ο Subpopulation Algorithm based on Novelty. [25]

Άλλοι γνωστές **a posteriori** μέθοδοι που χρησιμοποιούνται συχνά είναι οι εξής: [11]

- PGEN (Pareto surface generation for convex multi-objective instances). [27]
- IOSO (Indirect Optimization on the basis of Self-Organization).
- SMS-EMOA (S-metric selection evolutionary multi-objective algorithm). [28]
- Reactive Search Optimization (using machine learning for adapting strategies and objectives). [29], [30]
- ο αλγόριθμος του Benson για γραμμικά vector optimization προβλήματα. [10, pp. 64–65]

Στις **interactive methods** (διαδραστικές μέθοδοι), ο decision maker χρησιμοποιεί το διαδραστικό σύστημα που έχει υλοποιηθεί για να εκφράσει βηματικά τις απαιτήσεις του, ώστε να καταλήξει στις διαθέσιμες λύσεις που είναι κοντά στις προτιμήσεις του. [13], [31] Τα βήματα που ακολουθούνται συνήθως στις διαδραστικές μεθόδους είναι τα παρακάτω: [31]

1. Εκκίνηση μεθόδου.
2. Εύρεση και προβολή ενός Pareto front (χρησιμοποιώντας κάποια no-preference μέθοδο ή λύση που θα δοθεί από τον decision maker), από το οποίο ο decision maker θα επιλέξει την τελική λύση.
3. Εισαγωγή, από τον χρήστη, πληροφορίας για τις προτιμήσεις του.
4. Εύρεση του νέου Pareto front που προκύπτει από τις προτιμήσεις του decision maker και προβολή του, μαζί με οποιαδήποτε νέα πληροφορία για το πρόβλημα.

5. Αν το νέο Pareto front αποτελείται από πολλές λύσεις, αναμονή επιλογής μίας από αυτές από τον decision maker.
6. Αν ικανοποιηθεί ο decision maker, η διαδικασία σταματάει, αλλιώς πήγαινε στο βήμα 3.

Σε αντίθεση με τις mathematical optimization μεθόδους, στις οποίες το κριτήριο για να σταματήσει η διαδικασία είναι η σύγκλιση προς κάποιο ζητούμενο όριο που έχει οριστεί, στις διαδραστικές μεθόδους το κριτήριο διακοπής της διαδικασίας είναι η σύγκλιση προς την ψυχολογική κατάσταση στην οποία ο decision maker νιώθει ότι έχει βρει την βέλτιστη λύση.

Σε διαφορετικές διαδραστικές μεθόδους, ο τρόπος με τον οποίο εξάγεται η πληροφορία της προτίμησης από τον decision maker είναι διαφορετικός. Για παράδειγμα, 3 μέθοδοι θα μπορούσαν να αναγνωριστούν, ανάλογα με τον τρόπο με τον οποίο ενημερώνεται ο decision maker για το Pareto front:

1. **Trade-off information:** σε κάθε επανάληψη, προβάλλονται στον decision maker διαφορές μεταξύ κριτηρίων και ζητείται από εκείνον να δηλώσει αν ικανοποιείται κάποια προτίμηση του, αν όχι ή αν είναι αδιάφορος ως προς τη διαφορά που βλέπει. (μέθοδος των Zionts-Wallenius). [32]
2. **Reference points:** σε κάθε επανάληψη, ζητείται από τον χρήστη να ορίσει ένα reference point, το οποίο αποτελείται από τιμές που θα επιθυμούσε να έχει το κάθε κριτήριο και στη συνέχεια υπολογίζονται και προβάλλονται μία ή περισσότερες Pareto optimal λύσεις, τις οποίες καλείται να αναλύσει. [33], [34]
3. **Classification of objective functions:** στη μέθοδο αυτή, ο decision maker καλείται να εκφράσει τις προτιμήσεις του, υποδεικνύοντας πώς πρέπει να αλλάξουν οι τιμές (αν θα ήθελε να αυξηθεί η τιμή ενός κριτηρίου, να μειωθεί ή να παραμείνει στο σημείο που βρίσκεται ή αν πρέπει να τεθεί ένα κατώτατο ή ανώτατο όριο), ώστε να προσεγγιστεί μία πιο επιθυμητή λύση. Στη συνέχεια, η πληροφορία αυτή χρησιμοποιείται για να υπολογιστούν μία ή περισσότερες λύσεις, οι οποίες θα είναι πιο κοντά στις προτιμήσεις του decision maker. Τέτοιες μέθοδοι είναι η satisficing trade-off method (STOM) [35] και η NIMBUS [36], [37]

Τέλος, μπορούν να γίνουν και υλοποιήσεις οι οποίες συνδυάζουν μία ή περισσότερες από όλες τις μεθόδους που προαναφέρθηκαν στο κεφάλαιο αυτό, ώστε να επιτευχθεί το ζητούμενο αποτέλεσμα.

### 3. Visual Analytics

Αναφέρθηκε στην εισαγωγή η έννοια των visual analytics, ή αλλιώς οπτική ανάλυση δεδομένων. Παραθέσαμε μία μικρή περιγραφή, όμως ας δώσουμε εδώ έναν αυστηρότερο και πιο συγκεκριμένο ορισμό:

*Visual analytics ορίζουμε την επιστήμη η οποία ασχολείται με την ανάλυση δεδομένων, υποβοηθούμενη από αυτοματοποιημένες τεχνικές και διαδραστικές διεπιφάνειες γραφικών, με στόχο, μέσα από την κατανόηση, τη σύγκριση και τις αποφάσεις, με βάση τις απαιτήσεις που επιβάλλει η κατάσταση, να γίνει μία αποτελεσματική εξαγωγή συμπερασμάτων, πάνω σε μεγάλα και πολύπλοκα σύνολα δεδομένων. [38]*

Εδώ, πρέπει να διαχωρίσουμε 2 έννοιες, οι οποίες είναι κοντινές και είναι πολύ πιθανόν να συγχέονται μεταξύ τους, την οπτικοποίηση των δεδομένων και την οπτική ανάλυση δεδομένων. Στην οπτικοποίηση, ή αλλιώς data visualization, ο στόχος είναι η αναπαράσταση των δεδομένων που έχουμε, με έναν γραφικό τρόπο, με τον οποίο θα περιγράφονται επαρκώς και θα φαίνονται οι διαφορές μεταξύ τους. Τα visual analytics περιλαμβάνουν και αυτή τη διαδικασία, καθώς και την παροχή εργαλείων, με τα οποία ο χρήστης θα μπορεί, διαδραστικά, να συμμετέχει άμεσα στη διαμόρφωση του συνόλου των δεδομένων και να το σχηματίζει έτσι ώστε να απεικονίζει μόνο τα δεδομένα που ικανοποιούν τις απαιτήσεις του.

Ακόμη και σήμερα, που το hardware έχει εξελιχθεί αρκετά, το πρόβλημα του να εκμεταλλευόμαστε και να “αναλύουμε τις αναλύσεις μας”, παραμένει υπαρκτό. [38]. Σε περιπτώσεις που τα δεδομένα είναι πολλά και έχουν παρόμοιες τιμές, υπάρχει πιθανότητα να αλληλοκαλύπτονται σε ένα γράφημα, καθιστώντας το ιδιαίτερα πολύπλοκο στην κατανόηση, και στην εξαγωγή συμπερασμάτων. Επομένως, αν εφαρμόζαμε μόνο το visualization, δεν θα είχαμε καταφέρει και πολλά. Αντιθέτως, αν χρησιμοποιήσουμε τα εργαλεία που μας δίνουν τα visual analytics, μπορούμε να φιλτράρουμε τους συνδυασμούς πάνω στο διάγραμμα και να αφήσουμε



μόνο αυτούς που μας ενδιαφέρουν. Για να γίνει καλύτερα αντιληπτή η σημασία των visual analytics, για μία επιχείρηση, ως παράδειγμα, αυτό μπορεί να σημαίνει το φιλτράρισμα των πωλήσεων προϊόντων ανά περιοχή, ώστε να γίνεται ευκολότερα αντιληπτό ποιες είναι οι επικερδείς τοποθεσίες για την επιχείρηση και να οδηγήσει σε βελτίωση της οικονομικής της πορείας.

Είναι σημαντικό στα visual analytics να γίνεται σωστά η εξόρυξη και η οργάνωση των δεδομένων σε κατώτερο επίπεδο, καθώς αυτό αποτελεί κρίσιμο παράγοντα για την αποτελεσματικότητα της γραφικής απεικόνισης. Αν οι πληροφορίες που έχουν συλλεχθεί είναι λάθος, ή έχουν χωριστεί σε λάθος κατηγορίες, τότε το γράφημα θα είναι μάταιο, αφού και αυτό με τη σειρά του θα δίνει είτε λάθος ερμηνεία, είτε θα παρουσιάζει δυσκολίες στην ανάλυση. Πολλές φορές, μπορεί ακόμα να χρειαστεί να γίνει μετατροπή κάποιων τιμών των δεδομένων, ώστε να μπορούν να απεικονιστούν στο γράφημα, αν δεν έχουν εκ φύσεως αριθμητική τιμή, ή να κανονικοποιηθούν, αν το απαιτεί το διάγραμμα ώστε να έχει νόημα.

Επίσης, σημασία πρέπει να δοθεί και στον τρόπο με τον οποίο θα υλοποιείται η διαδραστικότητα της διεπιφάνειας που θα εμφανίζεται στον χρήστη, καθώς είναι αυτή που καθορίζει την ευκολία με την οποία θα μπορεί ο χρήστης να φτάσει στην τελική απόφαση. Όσο μεγαλύτερη ευελιξία του δίνεται, τόσο μεγαλύτερη ακρίβεια θα έχει στο να εκφράσει τις απαιτήσεις που έχει στο μυαλό του πάνω στο διάγραμμα, ώστε να προβάλλει τα αποτελέσματα που είναι πιο κοντά σε αυτές. Μία εύστοχη παρατήρηση είναι ότι με τα visual analytics, το ανθρώπινο μυαλό και η μηχανή συνδυάζουν τα πλεονεκτήματά τους μέσω του διαδραστικού γραφήματος, ώστε να πετύχουν τα μέγιστα αποτελέσματα. [38]

Ένας τομέας με αυξανόμενο ενδιαφέρον, ο οποίος μπορεί να βοηθήσει σημαντικά τα visual analytics είναι αυτός του machine learning. Όπως υποδηλώνει και το όνομά του, το machine learning αφορά το πεδίο εκείνο της Πληροφορικής το οποίο μελετά την κατασκευή αλγορίθμων, οι οποίοι μπορούν να “μάθουν” από τη συμπεριφορά των προτύπων των δεδομένων που λαμβάνουν και να κάνουν προβλέψεις για τα νέα δεδομένα που δέχονται, βάση αυτών. Στα visual analytics, μπορούν να βοηθήσουν σημαντικά, καθώς μπορούν να κατασκευάσουν εκ νέου μοντέλα δεδομένων, εντελώς αυτοματοποιημένα, ή να αυξήσουν την ακρίβεια που πετυχαίνει το εκάστοτε

εργαλείο στην ικανοποίηση των απαιτήσεων του χρήστη, εκμεταλλευόμενα το feedback από τις επιλογές του για να τον “μάθει” καλύτερα. Έτσι, το σύστημα γίνεται “εξυπνότερο” και αποτελεσματικότερο [39].

Τα πεδία τα οποία επωφελούνται από τη χρήση των visual analytics είναι πολλά και αυξάνονται, καθώς η τεχνολογία των υπολογιστών διεισδύει σε όλο και περισσότερους τομείς και επαγγέλματα. Ήδη χρησιμοποιούνται με επιτυχία σε οικονομικές στατιστικές αναλύσεις, σε αναλύσεις marketing, στην ιατρική (πειράματα φαρμάκων, ψυχολογία πλήθους κ.ά.), στην καταγραφή του web traffic (η ποσότητα δεδομένων που δέχεται και στέλνει μία ιστοσελίδα) κ.ά.

### 3.1 State of the art – Visual Analytics

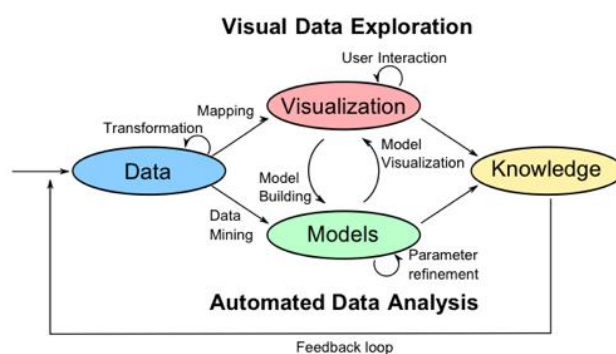
Το 1967, ο Jacques Bertin [40] ήταν ο πρώτος που έθεσε τις βάσεις για την οπτικοποίηση δεδομένων με χρήση συμβόλων και γραφημάτων διαφόρων τύπων [41]. Στη συνέχεια, ο Edward R. Tufte, στη δεκαετία του ‘80 αναφέρει τις αρχές που πρέπει να διέπουν την οπτική ανάλυση των δεδομένων σε 9 σημεία: [42]

1. Να προβάλλουν τα δεδομένα.
2. Να προτρέπουν τον θεατή να σκέφτεται την ουσία του γραφήματος και όχι την μεθοδολογία υπό την οποία κατασκευάστηκε, τον σχεδιασμό του, την τεχνολογία της παραγωγής του ή κάτι άλλο.
3. Να μην διαστρεβλώνουν, με οποιοδήποτε τρόπο, την πληροφορία που προβάλλει μέσω των δεδομένων.
4. Αν είναι πολλά τα νούμερα που πρέπει να προβληθούν, να είναι μικρά σε μέγεθος και να καταλαμβάνουν μικρό χώρο.
5. Τα μεγάλα σετ δεδομένων να έχουν συνοχή στην προβολή τους.
6. Να ενθαρρύνει τον θεατή να συγκρίνει τα διαφορετικά δεδομένα.
7. Να εμφανίζονται τα δεδομένα σε πολλαπλά επίπεδα λεπτομέρειας, από την ευρεία επισκόπηση τους μέχρι το οπτικό γράφημα.

8. Να υπηρετούν έναν ξεκάθαρο σκοπό, ο οποίος θα δηλώνεται είτε μέσω περιγραφής, είτε θα είναι ξεκάθαρος από τα στοιχεία που δίνει το γράφημα, είτε από κάποιο σύμβολο.
9. Να συνδέεται άμεσα με τη στατιστική ή την προφορική περιγραφή που έχει δοθεί για τα δεδομένα (να δίνεται η ευκαιρία στο χρήστη να μπορεί να δει τα δεδομένα από τα οποία προέκυψε το γράφημα).

Επίσης, σημαντική είναι και η προσφορά του John W. Turkey, ο οποίος το 1977 έγραψε το βιβλίο του με τίτλο “Exploratory Data Analysis” και προήγαγε την ομώνυμη στατιστική προσέγγιση, κατά την οποία περισσότερη έμφαση έπρεπε να δίνεται στην δημιουργία υποθέσεων μέσα από την εξερεύνηση των δεδομένων, παρά στη δοκιμή στατιστικών υποθέσεων τυχαίων μεταβλητών [43]. Η προσέγγιση αυτή ενθάρρυνε την ανάπτυξη της γλώσσας προγραμματισμού S, η οποία στη συνέχεια ενέπνευσε τη δημιουργία της R και της εφαρμογής S-Plus. [44]

Όπως ειπώθηκε στο κεφάλαιο 3, η διαδικασία των visual analytics συνδέεται άμεσα με την αυτοματοποιημένη συλλογή των δεδομένων. Στο Visual Analytics Pipeline (εικόνα 2), όπως ονομάζεται, φαίνεται η συνεχής και άμεση σχέση που πρέπει να έχει ο χρήστης με τις αυτοματοποιημένες διαδικασίες που χειρίζονται τα δεδομένα, ώστε να μπορεί να εισάγει τις παραμέτρους του και να ελέγχει τα ενδιαμέσως αποτελέσματα που προκύπτουν. Με τον τρόπο αυτό, αντί να ελέγχει απλά τα τελικά αποτελέσματα, ο χρήστης συμμετέχει ενεργά στην αξιολόγηση, τη διόρθωση και τη γρήγορη, ακριβέστερη βελτίωση των μοντέλων και των διαδικασιών που χρησιμοποιεί το εκάστοτε σύστημα, με αποτέλεσμα να αναλύονται μεγαλύτεροι όγκοι δεδομένων και να βελτιώνονται τα συμπεράσματα και οι αποφάσεις του. [38]



Εικόνα 2. Visual Analytics Pipeline

Με την ανάπτυξη των visual analytics, τα γραφήματα έχουν δείξει ιδιαίτερη ανάπτυξη. Παρακάτω, αναφέρονται μερικοί τύποι γραφημάτων οι οποίοι είναι γνωστοί και χρησιμοποιούνται συχνά.

### **Geospatial Data Visualization**

Έχει παρουσιαστεί μεγάλο ενδιαφέρον για τα γραφήματα με geospatial data [38]. Σε τέτοια γραφήματα, απεικονίζονται δεδομένα σχετικά με περιοχές του πραγματικού κόσμου, τα οποία αντιστοιχίζονται σε κάθε σημείο του χάρτη και δίνουν πληροφορίες σχετικά με εκείνο το μέρος π.χ. μπορεί να δηλώνουν τα σύνορα μεταξύ των χωρών, τις πόλεις που υπάρχουν εκεί, να δηλώνουν γεωγραφικές μετρήσεις, συντεταγμένες GPS, δεδομένα αισθητήρων κ.ά. Είναι συνηθισμένη ανάγκη να ζητούνται σχέσεις μεταξύ των σημείων αυτών ή να αναζητούνται συγκεκριμένα patterns, διαδικασίες οι οποίες μπορούν να παρουσιάσουν ιδιαίτερη υπολογιστική πολυπλοκότητα και δυσκολία στην ανάλυση, όταν τα δεδομένα είναι πολύπλοκα και πολλά [45]. Για το λόγο αυτό, απαιτείται καλή οργάνωση των δεδομένων και προηγμένες τεχνικές visualization, οι οποίες κατέχουν ρόλο-κλειδί σε αυτόν τον τύπο των γραφημάτων [46].

Ένα διαδραστικό visualization σύστημα που χρησιμοποιεί τέτοια γραφήματα, έχει παρουσιαστεί από τους Aidan Slingsby, Jason Dykes και Jo Wood. [47], ενώ οι ίδιοι, μαζί με την Donia Badawood, έχουν αναπτύξει και το BallotMaps, που κατατάσσεται στην ίδια κατηγορία [45], [48].

### **Multivariate Data Visualization**

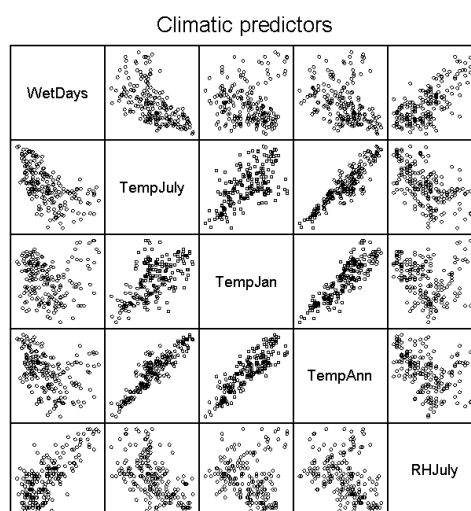
Τα visual analytics των δεδομένων πολλών διαστάσεων είναι μία ερευνητικά ενεργή περιοχή. Μέθοδοι οι οποίες χρησιμοποιούνται περιλαμβάνουν τις pixel-oriented μεθόδους, τα parallel coordinates plots (PCPs) και τα scatterplots matrices: [49]

- Στις pixel-oriented μεθόδους, κάθε τιμή των πολυδιάστατων δεδομένων αντιστοιχίζεται σε ένα pixel ή block, αποκτά οπτικά χαρακτηριστικά, όπως χρώμα, και η οθόνη χωρίζεται σε υποπαράθυρα ανάλογα με τις τιμές. Έτσι, ο χρήστης του συστήματος αποκτά γνώση από την οπτική πληροφορία που λαμβάνει, ώστε να κατευθύνει την απόφασή του.
- Στα parallel coordinates plots, κάθε διάσταση-κριτήριο αναπαρίσταται από έναν άξονα, ο οποίος βρίσκεται σε ίση απόσταση από τους άλλους και είναι παράλληλος ως προς

εκείνους. Στη συνέχεια, μία γραμμή ενώνει τα σημεία των τιμών των κριτηρίων για το εκάστοτε δεδομένο και ο χρήστης μπορεί να συγκρίνει τις διάφορες λύσεις από την πορεία των γραμμών τους..

- Στα scatterplot matrices, φτιάχνεται ένας πίνακας για κάθε συνδυασμό διαστάσεων ανά 2 και οπτικοποιεί τη σχέση μεταξύ τους, όπως στην **εικόνα 3**.

Στην περίπτωση που τα κριτήρια είναι 2, όπως είδαμε στο κεφάλαιο 2.1, το πρόβλημα λύνεται πολύ εύκολα με τη χρήση κάθετων αξόνων.



*Εικόνα 3. Scatterplot Matrix*

## Network Visualization

Network visualization χρησιμοποιούμε για να απεικονίσουμε συνδέσεις μεταξύ κόμβων-σημείων π.χ. απεικόνιση ατόμων που είναι φίλοι σε ένα κοινωνικό δίκτυο, απεικόνιση δικτύου αερομεταφορών μεταξύ πόλεων κ.ά. Οι συνδέσεις αυτές μπορεί να έχουν κατεύθυνση ή βάρος, για να χαρακτηρίσουν τη σχέση μεταξύ των 2 σημείων, δίνοντας έτσι πολύτιμη πληροφορία στην ανάλυση των επιλογών. Μερικές απεικονίσεις που χρησιμοποιούνται για την network οπτικοποίηση είναι η ορθογώνια, στη οποία κάθε κόμβος τοποθετείται στην ίδια γραμμή ή στήλη με άλλους κόμβους, ώστε να φαίνεται καλύτερα ποιος συνδέεται με ποιόν, η κυκλική, στην οποία οι κόμβοι συνδέονται με ακτίνες και η ιεραρχική, στην οποία, ανάλογα με τη θέση στη συνολική ιεραρχία, κάθε κόμβος τοποθετείται σε διαφορετικό ύψος. Ένα εργαλείο που μπορεί να σχεδιάσει

network διαγράμματα για να απεικονίσει τις δικτυακές σχέσεις και τη δομή της ιεραρχίας σε κοινωνικά δίκτυα είναι το TreeNet Viz. [50]

### Άλλα εργαλεία οπτικοποίησης δεδομένων

Μερικά visualization tools που έχουν αναπτυχθεί και κυκλοφορούν αυτή τη στιγμή στην αγορά, είναι:

- Το **Many Eyes** της **IBM** [51], το οποίο είναι ένα γενικής φύσεως εργαλείο οπτικοποίησης, εννοώντας ότι δεν επικεντρώνεται σε έναν συγκεκριμένο τύπο δεδομένων και γραφημάτων. Ανάμεσα στα γραφήματα που προσφέρει για την απεικόνιση των δεδομένων είναι τα pie charts [50], τα tag clouds [52], τα tree maps [53], τα Venn diagrams [54] και πολλά άλλα.
- Το **Fusion Tables** της **Google**, που είναι παρόμοιο εργαλείο με το προηγούμενο που αναφέρθηκε [55].
- Το **InstantAtlas**, το οποίο επικεντρώνεται σε δεδομένα που σχετίζονται με τον τομέα της Γεωγραφίας και τους χάρτες [56].
- Το **Exhibit**, το οποίο, πέρα από τους developer, αναφέρεται και σε δημοσιογράφους, στηριζόμενο στην ευκολία που προσφέρει στον χειρισμό του, ώστε να μπορούν να διεκπεραιώσουν τις εργασίες τους, χωρίς τη βοήθεια κάποιου έμπειρου προγραμματιστή. [57]

Επίσης, σε αρκετές γλώσσες προγραμματισμού είτε έχουν ενσωματωθεί λειτουργίες είτε έχουν αναπτυχθεί βιβλιοθήκες, οι οποίες προσφέρουν εύκολη και γρήγορη απεικόνιση στατιστικών αναλύσεων και γραφημάτων, όπως είναι η Python (π.χ. matplotlib, seaborn), η JavaScript (π.χ. D3, chart.js), η R κ.ά.

## 4. Αγορά επεξεργαστή και κάρτας γραφικών.

### Περιγραφή του προβλήματος

Το πρόβλημα στο οποίο επιλέχθηκε να γίνει η πρακτική εφαρμογή των τεχνικών που αναφέρθηκαν στα προηγούμενα κεφάλαια είναι η αγορά ενός ζευγαριού επεξεργαστή-κάρτας γραφικών για desktop. Ο λόγος για τον οποίο επιλέχτηκε αυτό το θέμα, ήταν η ευκαιρία να προσφερθεί ένα δωρεάν εργαλείο για την ανάλυση και την σύγκριση των χαρακτηριστικών συνδυασμών επεξεργαστών και καρτών γραφικών, ώστε να διευκολύνεται το decision-making του ενδιαφερόμενου αγοραστή, όπως επίσης και η πρόκληση που έθετε το πρόβλημα, όσων αφορά τη συλλογή των δεδομένων, την εύρεση του αλγορίθμου βελτιστοποίησης και την αναζήτηση τρόπων, ώστε να υποστηριχτεί ο χρήστης, για να μπορεί να λαμβάνει εύκολα αποφάσεις και να είναι αποτελεσματική η οπτική ανάλυση.

Όταν κάποιος αποφασίζει να αγοράσει έναν επεξεργαστή και μία κάρτα γραφικών για το desktop του, συνήθως κάνει έρευνα για να δει τι είναι αξιόλογο αυτήν την εποχή, αποφασίζει για τα χαρακτηριστικά του υπολογιστή του, βασιζόμενος στη χρήση που σκέφτεται να του κάνει μελλοντικά και ανάλογα με το χρηματικό ποσό που διαθέτει, κάνει μία έρευνα αγοράς στις διάφορες ιστοσελίδες με προϊόντα, βρίσκει ποια τον ικανοποιούν, ανάμεσα σε έναν αριθμό από άλλα που τα ανταγωνίζονται και τελικά αποφασίζει τι θα αγοράσει. Η αγορά όμως, προσφέρει πληθώρα προϊόντων προς απόκτηση, με το καθένα να έχει διαφορετικά χαρακτηριστικά και πλεονεκτήματα σε σχέση με τα υπόλοιπα, θέτονταν έτσι εμπόδια στην πραγματοποίηση της τελικής απόφασης. Σκοπός της πτυχιακής αυτής είναι, μέσα από μια αυτοματοποιημένη διαδικασία, να καταλήγει σε ένα μικρό, σε σχέση με τον αριθμό των διαθέσιμων συνδυασμών, σύνολο από λύσεις, οι οποίες υπερέχουν από τις υπόλοιπες σε ένα τουλάχιστον κριτήριο, και με τη χρήση ενός χρωματισμένου διαγράμματος, και διαφόρων φίλτρων, να τις απεικονίζει με τέτοιο τρόπο στην οθόνη, ώστε να μπορεί να αποφασίσει ο χρήστης για την επιλογή του καλύτερου συνδυασμού που λύνει το πρόβλημά του.

Για την ομαλότερη εκπόνηση και οργάνωσή της, η πτυχιακή χωρίστηκε σε τέσσερα υποπροβλήματα:

1. Συλλογή δεδομένων επεξεργαστών και καρτών γραφικών.
2. Μελέτη και υλοποίηση αναζήτησης των βέλτιστων λύσεων.
3. Βελτιστοποίηση επιλογής συνδυασμού επεξεργαστή – κάρτας γραφικών.
4. Δημιουργία ιστοτόπου που υλοποιεί την αναπαράσταση των βέλτιστων λύσεων στο πρόβλημα που θέτει ο χρήστης, κάνοντας οπτική ανάλυση των δεδομένων τους.

Για την καλύτερη δομή της πτυχιακής, επιλέχθηκε να αφιερωθεί ένα ξεχωριστό κεφάλαιο για κάθε υποπρόβλημα, ώστε να μπορεί ο αναγνώστης να ανατρέξει απευθείας σε οποιοδήποτε στάδιο της επιθυμεί και να μπορέσει να ακολουθήσει ευκολότερα τη διαδικασία της εκπόνησης της. Στα 2 τελευταία κεφάλαια της εργασίας, παρατίθενται τα συμπεράσματα που προέκυψαν μετά την μελέτη και την εκπόνηση της πτυχιακής και η βιβλιογραφία από την οποία αντλήθηκαν όλες οι πληροφορίες που χρησιμοποιήθηκαν για να συντάξουν το τρέχων κείμενο.

Οι γλώσσες προγραμματισμού που χρησιμοποιούνται στα εργαλεία της πτυχιακής είναι η **Python** και η **JavaScript**.

Ο **server** της ιστοσελίδας φιλοξενείται στο [oceanos.grnet.gr](https://oceanos.grnet.gr).

Το Web Framework που χρησιμοποιείται είναι το **Django**.

Για το data visualization χρησιμοποιήθηκε η JavaScript βιβλιοθήκη **chart.js**.

Ο κώδικας βρίσκεται στο [https://github.com/MnMes/compilations\\_visual](https://github.com/MnMes/compilations_visual).



## 5. Μοντέλο αντιμετώπισης του προβλήματος

Το μοντέλο αντιμετώπισης του προβλήματος, επιλέχθηκε να περιγραφεί μέσα από την ανασκόπηση των βασικών βημάτων που ακολουθήθηκαν για να αντιμετωπιστούν οι διάφορες δυσκολίες.

### 5.1 Συλλογή δεδομένων επεξεργαστών και καρτών γραφικών

Το πρώτο ζήτημα που έπρεπε να επιλυθεί στην εργασία, ήταν η αυτοματοποιημένη συλλογή των δεδομένων τα οποία χρησιμοποιούν τα μετέπειτα επίπεδα για να εξάγουν αποτελέσματα. Το ίδιο το υποπρόβλημα χωρίστηκε σε μικρότερα υποπροβλήματα, όπως η συλλογή πληροφοριών ξεχωριστά για τους επεξεργαστές, για τις κάρτες γραφικών και τέλος, για το συνδυασμό τους.

Ένα βασικό πρόγραμμα ελέγχει τη ροή, καλώντας κάθε διαδικασία που έχει γραφτεί με τη σειρά που πρέπει. Στην αρχή, πριν από οτιδήποτε άλλο, ελέγχει αν υπάρχει ήδη κάποιο αρχείο βάσης δεδομένων, και αν όχι, δημιουργεί ένα. Εν συνεχεία, επιλύει κάθε υποπρόβλημα σειριακά, μέχρι να συλλέξει όλα τα απαραίτητα στοιχεία για τη βάση δεδομένων. Γίνονται οι απαραίτητοι έλεγχοι για να αποφευχθούν τα διπλότυπα και επίσης, δημιουργούνται δύο λίστες, μία για τους επεξεργαστές και μία για τις κάρτες γραφικών, οι οποίες κρατάνε μόνο τους νέους συνδυασμούς που συλλέχθηκαν από το [skroutz.gr](http://skroutz.gr) τη στιγμή που έτρεξε το πρόγραμμα και χρειάζονται να περιγραφτούν και από τα υπόλοιπα στοιχεία. Όσων αφορά τον πίνακα των συνδυασμών, την πρώτη φορά που θα δημιουργηθεί το αρχείο της βάσης, θα αναζητηθούν συνδυασμοί για όλους τους επεξεργαστές ως προς τις κάρτες αλλά και όλων των καρτών ως προς τους επεξεργαστές, εφόσον οι δύο λίστες του βασικού προγράμματος θα είναι γεμάτες με τα νέα στοιχεία που έχουν εισαχθεί. Κάθε επόμενη όμως, μόνο τα καινούργια στοιχεία συνδυάζονται με τα παλαιότερα, για την αποφυγή διπλού φόρτου.

Τέλος, κάθε πληροφορία η οποία δεν συλλέχθηκε, αποθηκεύεται με την τιμή -1, για να εξεταστεί σε υψηλότερο επίπεδο και να συλλεχθεί χειρωνακτικά.

Το βασικό πρόγραμμα έχει ως εξής:

[https://github.com/MnMes/compilations\\_visual/blob/master/compilation\\_creator.py](https://github.com/MnMes/compilations_visual/blob/master/compilation_creator.py)

### 5.1.1 Επεξεργαστές

Αρχικά, αποφασίστηκε ότι η σελίδα η οποία θα καθορίσει το σύνολο των προϊόντων τύπου επεξεργαστή θα είναι το [skroutz.gr](https://www.skroutz.gr), καθώς από αυτήν εξάγονται και καθορίζονται και οι αξίες τους. Επίσης, πέρα από την **τιμή**, από την ίδια σελίδα εξάγεται και η πληροφορία για το **όνομα** του επεξεργαστή, τον **τύπο** του, το κωδικό όνομα της **μικροαρχιτεκτονικής** και το **part number** του. Αναλυτικά οι κατηγορίες που συλλέγονται από το skroutz.gr:

**Τιμή:** Ως τιμή επιλέγεται η μικρότερη χρηματική αξία που εξάγεται από τη σελίδα του skroutz.gr για το κάθε προϊόν.

**Όνομα:** Το όνομα που αναγράφεται για κάθε επεξεργαστή στη σελίδα του skroutz.gr.

**Τύπος:** Η «σειρά» επεξεργαστών στην οποία έχει κατατάξει ο κατασκευαστής τον κάθε επεξεργαστή. Το στοιχείο αυτό εξάγεται για να βοηθήσει στην επιλογή της σωστής σελίδας στη Wikipedia, η οποία περιλαμβάνει τις υπόλοιπες πληροφορίες που χρειαζόμαστε για τον κάθε επεξεργαστή.

**Όνομα μικροαρχιτεκτονικής:** Το κωδικό όνομα που χρησιμοποιείται για τη μικροαρχιτεκτονική κατασκευής του επεξεργαστή. Χρησιμοποιείται αργότερα για να αντιστοιχιστεί στην τιμή της λιθογραφικής μεθόδου ολοκλήρωσης.

**Part number:** Ο κωδικός προϊόντος του επεξεργαστή. Χρησιμοποιείται για να αντιστοιχιστούν οι επεξεργαστές από το skroutz.gr με τα στοιχεία τους στη Wikipedia, σε μία προσπάθεια να υπάρξουν όσο λιγότερα σφάλματα αντιστοιχίας είναι δυνατό.

Ο κώδικας για τη συλλογή των παραπάνω δεδομένων είναι ο ακόλουθος:

[https://github.com/MnMes/compilations\\_visual/blob/master/skroutz\\_cpu\\_list\\_parser.py](https://github.com/MnMes/compilations_visual/blob/master/skroutz_cpu_list_parser.py)

Στη συνέχεια, για κάθε επεξεργαστή που έχει συλλεχθεί μέχρι αυτό το σημείο, καλείται από τη Wikipedia η σελίδα που περιέχει πληροφορίες για τους επεξεργαστές του κατασκευαστή στον οποίο ανήκει και του τύπου του π.χ. αν ο επεξεργαστής μας είναι ο AMD A4-3300, τότε θα κληθεί η σελίδα:

[https://en.wikipedia.org/wiki/List\\_of\\_AMD\\_accelerated\\_processing\\_unit\\_microprocessors](https://en.wikipedia.org/wiki/List_of_AMD_accelerated_processing_unit_microprocessors),

η οποία περιέχει πληροφορίες για AMD επεξεργαστές τύπου APU, στον οποίο ανήκει και ο AMD A4-3300. Επιλέχθηκε η σελίδα της Wikipedia καθώς προσφέρει ολοκληρωμένο προφίλ για ένα μεγάλο πλήθος επεξεργαστών και δίνει για τον καθένα και το part number του, κάτι το οποίο διευκολύνει πολύ την αντιστοιχία με το skroutz.gr.

Επειδή έχουμε επεξεργαστές δύο κατασκευαστών και πολλών τύπων, αναγκαστικά οι σελίδες που καλούνται από την Wikipedia είναι μία για κάθε κατασκευαστή και τύπο. Ακολουθεί ο κώδικας που γράφτηκε για την εκτέλεση της προαναφερθείσας διαδικασίας:

[https://github.com/MnMes/compilations\\_visual/blob/master/amd\\_apu\\_cpus\\_mining.py](https://github.com/MnMes/compilations_visual/blob/master/amd_apu_cpus_mining.py)

[https://github.com/MnMes/compilations\\_visual/blob/master/amd\\_fx\\_cpus\\_mining.py](https://github.com/MnMes/compilations_visual/blob/master/amd_fx_cpus_mining.py)

[https://github.com/MnMes/compilations\\_visual/blob/master/intel\\_cpus\\_mining.py](https://github.com/MnMes/compilations_visual/blob/master/intel_cpus_mining.py)

Τέλος, αντιστοιχίζεται το όνομά της μικροαρχιτεκτονικής κάθε επεξεργαστή με την τιμή της λιθογραφικής μεθόδου ολοκλήρωσης που του αντιστοιχεί, σύμφωνα με τον παρακάτω κώδικα:

[https://github.com/MnMes/compilations\\_visual/blob/master/cpu\\_process\\_parsing.py](https://github.com/MnMes/compilations_visual/blob/master/cpu_process_parsing.py)

Όπως φαίνεται στον κώδικα αυτό, οι τιμές δεν μπόρεσαν να εξαχθούν από κάποια πηγή με αξιόλογο πλήθος επεξεργαστών αντίστοιχο της Wikipedia και επομένως «καρφώνονται» απευθείας με χειρωνακτικό τρόπο.

### 5.1.2 Κάρτες γραφικών

Αντίστοιχη διαδικασία ακολουθείται και στην εξαγωγή πληροφοριών για τις κάρτες γραφικών. Η διαδικασία ξεκινάει ξανά με τη συλλογή των προϊόντων που βρίσκονται στο skrouzt.gr, και πιο συγκεκριμένα με τη συλλογή του **ονόματος**, της **τιμής**, του μεγέθους της **μνήμης**, της συχνότητας του ρολογιού και του memory bus. Πιο αναλυτικά:

**Όνομα:** Το όνομα της κάρτας γραφικών, όπως αυτό αναφέρεται στο skrouzt.gr. Στη συνέχεια, το όνομα αυτό, πέρα από την αποθήκευσή του στη βάση δεδομένων, φιλτράρεται ώστε να μπορεί να αντιστοιχηθεί με τις κάρτες γραφικών της σελίδας <https://www.techpowerup.com/>, από την οποία παίρνουμε τις υπόλοιπες πληροφορίες που χρειαζόμαστε, με όσο το δυνατόν λιγότερα σφάλματα αντιστοιχίας.

**Τιμή:** Ως τιμή επιλέγεται η μικρότερη χρηματική αξία που εξάγεται από τη σελίδα του skrouzt.gr για το κάθε προϊόν.

**Μέγεθος μνήμης:** Το μέγεθος της μνήμης που έχει η κάρτα γραφικών.

**Συχνότητα ρολογιού και Memory bus:** Τα δύο αυτά μεγέθη συνδυάζονται για να μας δώσουν το μέγεθος Memory bandwidth σύμφωνα με την εξίσωση

$$\text{Effective memory clock} * (\text{Memory bus} / 8) \text{ [58]}$$

Η ταχύτητα ρολογιού που δίνεται στο skrouzt.gr για τις κάρτες γραφικών δεν αντιστοιχούν στην πραγματική ταχύτητα αλλά στην ταχύτητα που επιτυγχάνεται σε συνδυασμό με τον τύπο της μνήμης της κάρτας (effective memory).

Το **memory bandwidth** αντιστοιχεί στην ποσότητα δεδομένων που μπορεί να διαβάσει ή να γράψει η κάρτα γραφικών στη μνήμη της στη μονάδα του χρόνου. Παραθέτω τον κώδικα που περιγράφει τις παραπάνω λειτουργίες:

[https://github.com/MnMes/compilations\\_visual/blob/master/skrouzt\\_gpu\\_list\\_parser.py](https://github.com/MnMes/compilations_visual/blob/master/skrouzt_gpu_list_parser.py)

Αφού συλλεχθούν οι κάρτες γραφικών από το skroutz.gr, καλείται η ιστοσελίδα του techpowerup. Να σημειωθεί σε αυτό το σημείο ότι προτιμήθηκε αυτή η σελίδα από τη Wikipedia, καθώς έχει συγκεντρωμένες όλες τις κάρτες, ανεξαρτήτως κατασκευαστή, σε μία σελίδα και δίνει με συγκεκριμένη σειρά (ασφαλές parsing) τις πληροφορίες που χρειαζόμαστε. Τα πράγματα είναι αρκετά ξεκάθαρα όσον αφορά τη συλλογή από το techpowerup, όπως φαίνεται και από τον κώδικα παρακάτω:

[https://github.com/MnMes/compilations\\_visual/blob/master/gpu\\_list\\_parser.py](https://github.com/MnMes/compilations_visual/blob/master/gpu_list_parser.py)

### 5.1.3 Συνδυασμοί επεξεργαστών και καρτών γραφικών

Αφού μαζευτούν τα επιμέρους στοιχεία που χρειαζόμαστε, συνδυάζονται μεταξύ τους για να μας δώσουν όλους τους πιθανούς συνδυασμούς που μπορούν να προκύψουν μεταξύ τους και που πρέπει να ληφθούν υπόψιν στη σύγκριση για να βρεθούν οι βέλτιστες λύσεις, ολοκληρώνοντας έτσι τη δημιουργία της βάσης δεδομένων για αυτό το πέρασμα.

Η διαδικασία που περιγράφηκε πρέπει να εκτελείται με τέτοια συχνότητα, ώστε να παραμένει ενημερωμένη η βάση με όλα τα καινούργια στοιχεία που εισέρχονται στο skroutz.gr.

## 5.2 Μελέτη και υλοποίηση αναζήτησης των βέλτιστων λύσεων

Επόμενο βήμα στην εργασία ήταν η μελέτη για το σχεδιασμό της στρατηγικής που θα ακολουθούνταν, ώστε να αναζητηθούν οι βέλτιστες λύσεις μεταξύ του συνόλου που θα ορίζει ο χρήστης με την εισαγωγή του χρηματικού ποσού που επιθυμεί να διαθέσει για την αγορά επεξεργαστή και κάρτας γραφικών.

Έχοντας δώσει τους απαραίτητους ορισμούς στο κεφάλαιο 1, μπορούμε να περιγράψουμε τα βήματα του αλγορίθμου αναζήτησης του Pareto front που χρησιμοποιήθηκε στην πτυχιακή:

**Βήμα 1:** Δημιουργία λίστας, έστω  $P$ , των έγκυρων συνδυασμών (συνδυασμοί των οποίων το κόστος δεν ξεπερνάει το ποσό που έχει θέσει ο χρήστης) και ταξινόμηση της σε φθίνουσα σειρά. Αυτό, γιατί οι συνδυασμοί που έχουν υψηλότερο κόστος, τείνουν να έχουν καλύτερα χαρακτηριστικά από τους πιο φθηνούς, επομένως έχουν και καλύτερη πιθανότητα, με λιγότερες συγκρίσεις να κυριαρχήσουν σε αυτούς. Επίσης, δημιουργία λίστας, έστω  $P'$ , για τις βέλτιστες λύσεις. Πέρασμα του πρώτου στοιχείου της  $P$  στην  $P'$ .

**Βήμα 2:** Για κάθε συνδυασμό στην  $P$ , σύγκριση με τις βέλτιστες λύσεις στην  $P'$ . Διαχωρίζουμε 2 περιπτώσεις:

- 1) Ο συνδυασμός κυριαρχεί την βέλτιστη, οπότε αφαιρούμε την βέλτιστη από το  $P'$ . Αν η  $P'$  δεν έχει άλλα στοιχεία προς σύγκριση, προσθέτουμε τον συνδυασμό στις βέλτιστες λύσεις. Αλλιώς, εξετάζουμε τον συνδυασμό ως προς τα υπόλοιπα στοιχεία της  $P'$ .
- 2) Ο συνδυασμός δεν κυριαρχεί την βέλτιστη λύση, οπότε εξετάζουμε αν η βέλτιστη λύση κυριαρχεί τον συνδυασμό (αναφέρθηκε πιο πάνω ότι αν δεν ισχύει το ένα, δεν εννοείται αυτόματα το άλλο). Αν συμβαίνει αυτό, τότε απορρίπτουμε τον συνδυασμό ως μη βέλτιστο και εξετάζουμε τον επόμενο. Αλλιώς, τα 2 στοιχεία αλληλοεξουδετερώνονται μεταξύ τους, οπότε προσθέτουμε τον συνδυασμό στη λίστα των βέλτιστων και πάμε στον επόμενο.

**Βήμα 3:** Αν εξετάσαμε όλους τους έγκυρους συνδυασμούς, επιστρέφουμε το σύνολο  $P'$  ως το Pareto front.

Να σημειωθεί στο σημείο αυτό ότι για το συγκεκριμένο πρόβλημα και μόνο, θεωρούμε ως βέλτιστη λύση και τον συνδυασμό που ισούται σε όλα τα κριτήρια με έναν βέλτιστο, εφόσον προσφέρει εναλλακτική επιλογή κατασκευαστή στον χρήστη (αυτό αντίκειται στον ορισμό της κυριαρχίας, αφού ισχύει αυστηρά και μόνο όταν μια λύση δεν είναι χειρότερη σε κανένα κριτήριο, αλλά είναι και καλύτερη τουλάχιστον σε ένα).

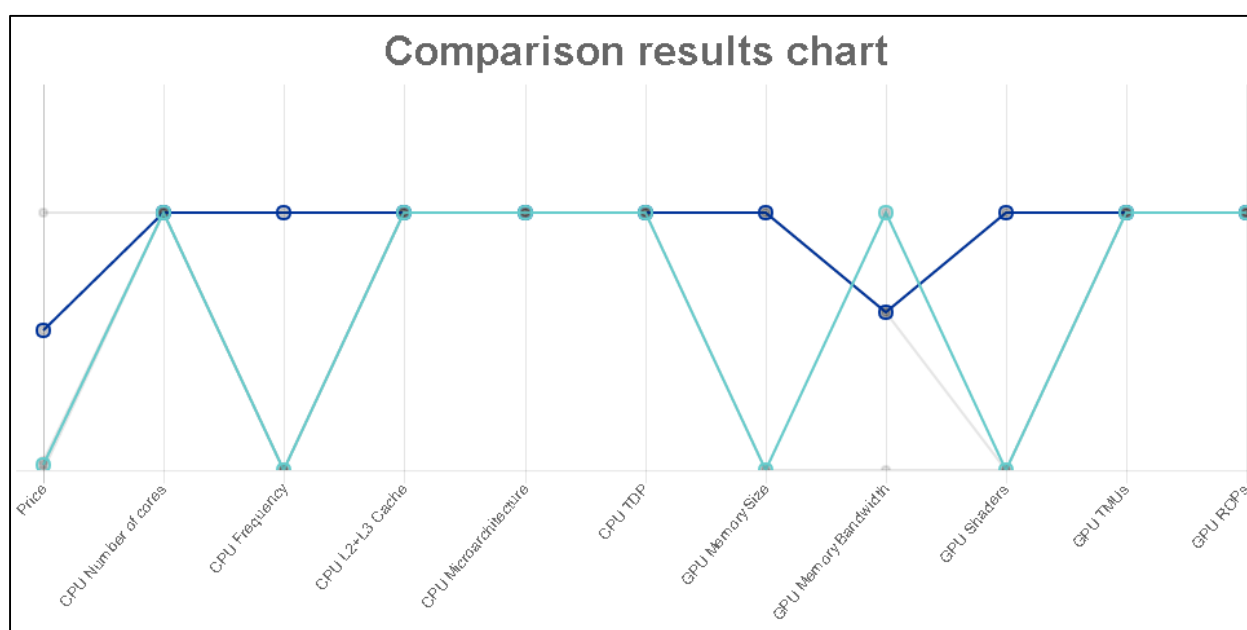
Ο αλγόριθμος που χρησιμοποιήθηκε, μοιάζει με την προσέγγιση που περιγράφεται στη σελίδα 36 του βιβλίου του Kalyanmoy Deb [10, p. 36], μόνο που έχει επεκταθεί με το στοιχείο της ταξινόμησης της τιμής, ώστε να ελαχιστοποιηθεί ο αριθμός των συγκρίσεων μεταξύ του συνδυασμού που εξετάζεται κάθε στιγμή με τις βέλτιστες λύσεις μέχρι εκείνη την ώρα, προσεγγίζοντας την λογική της μεθόδου του Kung et al.. Θεωρήθηκε ότι η ταξινόμηση με βάση την τιμή, αρκεί για να βελτιώσει την ταχύτητα του continuously updated αλγορίθμου, όμως σαν μελλοντική πρόκληση για την εργασία, θα μπορούσε να μελετηθεί η σύγκριση με τη μέθοδο του Kung et al. και να ελεγχθεί η διαφορά στον χρόνο εκτέλεσης των δύο αλγορίθμων. Επίσης, μελετήθηκε και ο αλγόριθμος που περιγράφεται σε ένα άρθρο με τίτλο **A Fast Algorithm for Finding the Non Dominated Set in Multi objective Optimization**, το οποίο δημοσιεύτηκε στο **International Journal of Computer Applications**, το 2010, και ο οποίος αποτελεί αντικείμενο για περαιτέρω μελέτη επίσης [59].

Τέλος, για να γίνει καλύτερα αντιληπτή η έννοια του Pareto front στο παράδειγμα της πτυχιακής, στην **Εικόνα 4** παρακάτω, υπάρχει ένα διάγραμμα, στο οποίο φαίνονται με χρώματα οι 2 βέλτιστες λύσεις που προέκυψαν από την εκτέλεση του προαναφερθέντος αλγορίθμου και με γκρι οι λύσεις που αποτελούσαν έγκυρους συνδυασμούς στα χρήματα αυτά, αλλά αποδείχτηκαν μη βέλτιστες:

Όπως φαίνεται στο διάγραμμα, μπορεί ο γκρι συνδυασμός να τα πηγαίνει καλύτερα σε θέμα κόστους, όμως με λίγα παραπάνω χρήματα που έχει επιλέξει ο χρήστης να δώσει, μπορεί να αγοράσει αυτόν με το σκούρο μπλε και να κερδίσει στον αριθμό των shaders της κάρτας γραφικών,



που μπορεί να τον ενδιαφέρουν. Επίσης, αν τον ενδιαφέρει το memory bandwidth της κάρτας γραφικών, μπορεί με σχεδόν το μέγιστο επιτρεπτό ποσό που έχει εισάγει, να αγοράσει τον συνδυασμό με το γαλάζιο χρώμα. Στο διάγραμμα δεν φαίνεται καλά, αλλά υπάρχει συνδυασμός με γκρι χρώμα που έχει τα ίδια ακριβώς χαρακτηριστικά με τον γαλάζιο, όμως έχει απορριφθεί γιατί είναι ακριβότερος. Όλο το σύνολο λοιπόν των υποψήφιων λύσεων φαίνεται με όλα τα χρώματα στο διάγραμμα, ενώ το Pareto front, που ενδιαφέρει τον χρήστη, το αποτελούν μόνο αυτές που είναι χρωματισμένες.



**Εικόνα 4.** στιγμιότυπο προβολής μη βέλτιστων και βέλτιστων συνδυασμών ταυτόχρονα στο διάγραμμα του ιστοτόπου, για την κατανόηση του Pareto front.

## 5.3 Βελτιστοποίηση επιλογής συνδυασμού επεξεργαστή και κάρτας γραφικών

Το πρόβλημα που πραγματεύεται η πτυχιακή κατατάσσεται στα πολυ-κριτηριακά προβλήματα βελτιστοποίησης, καθώς κάθε κριτήριο του εκάστοτε συνδυασμού αποτελεί και έναν στόχο προς βελτιστοποίηση (θα θέλαμε ιδανικά, για κάθε συνδυασμό, κάποια κριτήρια να έχουν τη μέγιστη τιμή τους και κάποια άλλα την ελάχιστη).

Λόγω της φύσης του προβλήματος που επιλέχθηκε να αναλυθεί και της μη ξεκάθαρης υπεροχής, σε βαρύτητα, ενός κριτηρίου, επιλέχθηκε η διαδραστική μέθοδος βελτιστοποίησης με visualization (interactive method, όπως την είδαμε στο κεφάλαιο 2.1). Άλλωστε, λαμβάνοντας υπόψιν τα δεδομένα μιας πραγματικής κατάστασης, η απόφαση για την αγορά ενός επεξεργαστή και μίας κάρτας γραφικών εξαρτάται εξ 'ολοκλήρου από τη χρήση που σκέφτεται κανείς να τους κάνει και δεν σχετίζεται με τα συνολικά χαρακτηριστικά τους. Δόθηκε ιδιαίτερη προσοχή στα εργαλεία που υλοποιούν τη διαδραστικότητα και την ευελιξία που προσφέρουν, καθώς αποτελούν σημείο-κλειδί στην αποτελεσματικότητα του συστήματος. Για την απεικόνιση των λύσεων, χρησιμοποιήθηκε το μοντέλο του parallel coordinates plot (PCP), για 2 βασικούς λόγους:

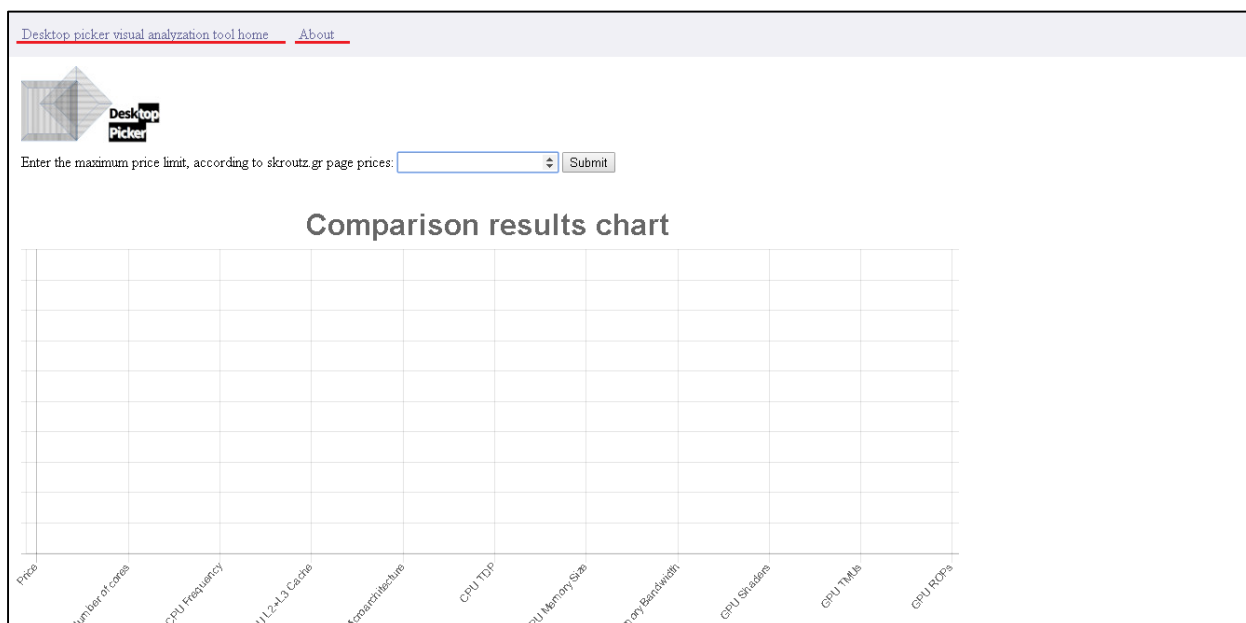
1. Σαφώς έχουμε να κάνουμε με πολυδιάστατα δεδομένα, αφού κάθε συνδυασμός χαρακτηρίζεται από πολλαπλές τιμές-κριτήρια. Βρισκόμαστε, λοιπόν, στην κατηγορία των multivariate data, τα οποία περιγράψαμε στο κεφάλαιο 3. Λόγω του μεγάλου πλήθους των συνδυασμών, θεωρήθηκε ιδανική επιλογή, σε συνδυασμό με τα interactive tools που υλοποιεί το σύστημα, ώστε ο χρήστης να μπορεί να κάνει άμεση σύγκριση και να ενημερωθεί για τον κάθε συνδυασμό.
2. Μην ξεχνάμε ότι, παρόλο που ο χρήστης θα πρέπει να έχει μία εμπειρία στην ανάλυση, ώστε να μπορέσει να εξάγει συμπεράσματα, δεν παύει να είναι ένας απλός χρήστης. Επομένως, επιλέχθηκε μία κοινή αναπαράσταση, την οποία όλοι γνωρίζουν και ο καθένας μπορεί, ακόμη και με μία πρόχειρη παρατήρηση, να εξάγει πληροφορίες.

## 6. Υλοποίηση ιστοτόπου

Η ιστοσελίδα που δημιουργήθηκε για να χρησιμοποιηθεί ως εργαλείο για την επίλυση του προβλήματος που περιγράφηκε στα προηγούμενα κεφάλαια, βρίσκεται στην διεύθυνση <http://83.212.124.246:8000>.

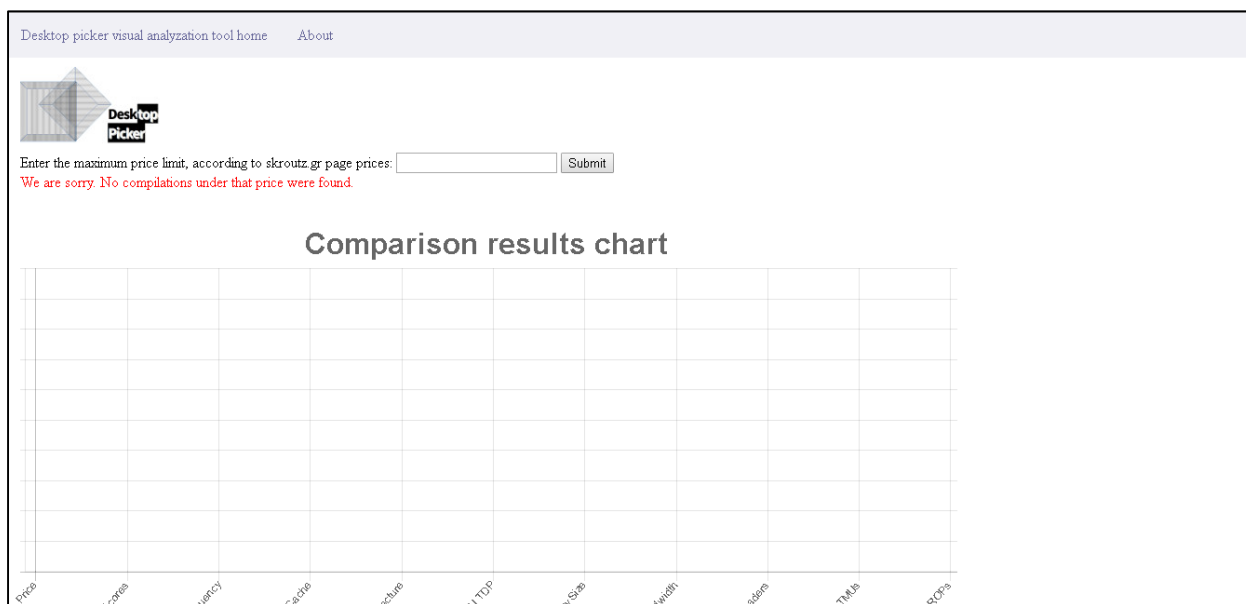
Πρέπει να σημειωθεί ότι αναφέρεται σε χρήστες ικανούς να πραγματοποιήσουν ανάλυση δεδομένων και να εξάγουν συμπεράσματα, επομένως κάποιος άπειρος χρήστης μπορεί να συναντήσει δυσκολίες στην ερμηνεία των αποτελεσμάτων.

Έγινε προσπάθεια ο αριθμός των ιστοσελίδων να είναι ο μικρότερος δυνατός, καθώς ο σκοπός του ιστοτόπου είναι αρκετά ξεκάθαρος. Στην αρχική ιστοσελίδα (**Εικόνα 5**), ο χρήστης καλείται να εισάγει το ποσό το οποίο διατίθεται να ξοδέψει για την αγορά ενός συνδυασμού επεξεργαστή και κάρτας γραφικών.

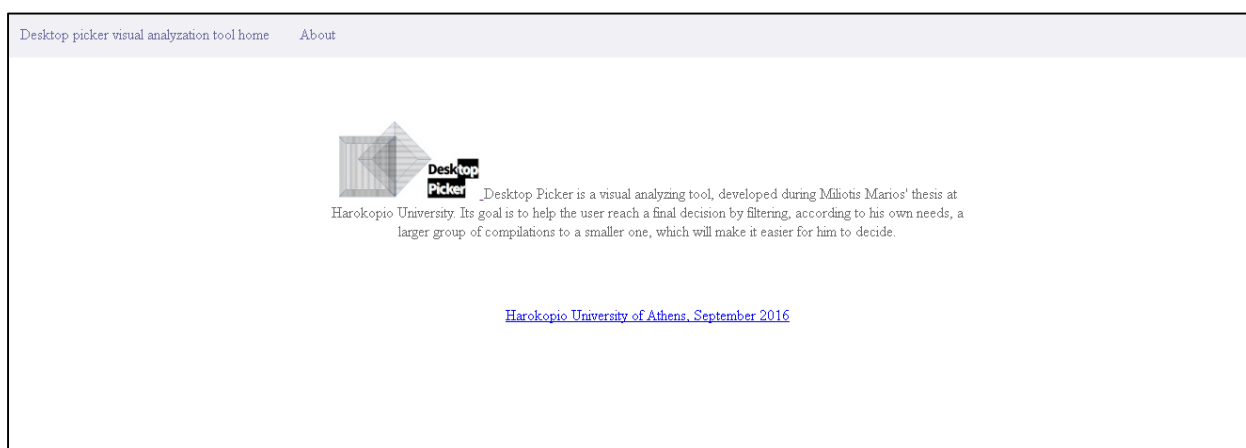


*Εικόνα 5, αρχική σελίδα ιστοτόπου.*

Σε περίπτωση που δεν βρεθεί κάποιο σύνολο συνδυασμών που να ικανοποιούν το χρηματικό ποσό που εισήχθη, ο χρήστης ειδοποιείται με σχετικό μήνυμα και καλείται να εισάγει ένα διαφορετικό ποσό (Εικόνα 6). Στο πάνω μέρος της σελίδας (Εικόνα 5, κόκκινη υπογράμμιση), όπως και σε όλες τις σελίδες, εμφανίζεται επιλογή για επιστροφή στην αρχική σελίδα και άλλη μία επιλογή, η οποία εμφανίζει μία σελίδα με πληροφορίες σχετικά με τον ιστότοπο (Εικόνα 7).

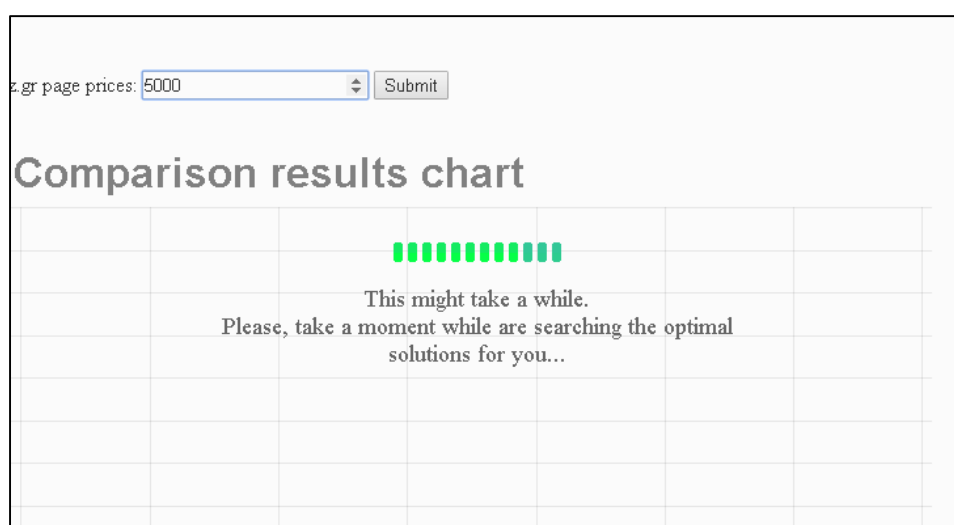


Εικόνα 6, στιγμιότυπο εσφαλμένης εισόδου στον ιστότοπο.

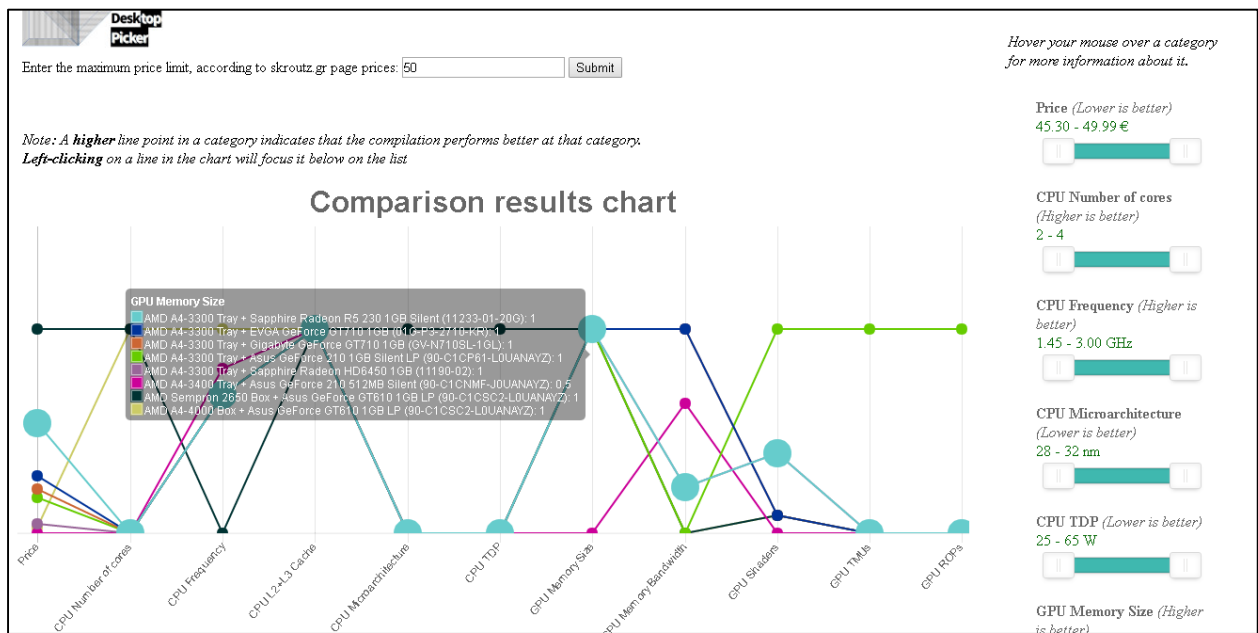


Εικόνα 7, σελίδα πληροφοριών του ιστοτόπου.

Σε περίπτωση που το ποσό αναφέρεται σε κάποιο σύνολο συνδυασμών, τότε ενεργοποιείται το script για να βρεθεί το σύνολο των κυρίαρχων λύσεων μεταξύ τους και ο χρήστης μεταφέρεται στη σελίδα του διαγράμματος. Καθώς το σύνολο των έγκυρων συνδυασμών μπορεί να είναι μεγάλο και επομένως και ο αριθμός των συγκρίσεων να είναι και αυτός αρκετά μεγάλος, η εκτέλεση του script και ο σχεδιασμός του διαγράμματος μπορεί να διαρκέσει κάποια δευτερόλεπτα, επομένως ζητείται από τον χρήστη να περιμένει υπομονετικά μέχρι να ολοκληρωθεί η λειτουργία (**Εικόνα 8**). Επίσης, επειδή και το σύνολο των κυρίαρχων λύσεων μπορεί να περιλαμβάνει αρκετούς συνδυασμούς, έχει γίνει προσπάθεια να δοθούν εναλλακτικές επιλογές στον χρήστη, ώστε να μπορεί να διακρίνει ποιες λύσεις υπάρχουν σε κάθε τιμή του διαγράμματος, καθώς και να μπορεί να ενημερωθεί για το σύνολο των πληροφοριών κάθε λύσης. Το πρώτο επιτυγχάνεται με την επιλογή των φίλτρων, στο δεξί μέρος της σελίδας (**Εικόνα 9, δεξί μέρος σελίδας**), ενώ το δεύτερο με τη δημιουργία σχετικής λίστας, η οποία εμφανίζεται είτε όταν πατήσει στο σχετικό κείμενο ο χρήστης, είτε όταν πατήσει πάνω σε μία λύση στο διάγραμμα, στην οποία περίπτωση, μεταφέρεται αυτόματα στη θέση που έχει στη λίστα (**Εικόνα 10**). Σε κάθε σημείο που κρίθηκε απαραίτητο, υπάρχουν διακριτικά μηνύματα, τα οποία ενημερώνουν τον χρήστη για την ορθή χρήση και ερμηνεία του διαγράμματος (**Εικόνα 9, πλάγια γράμματα**). Τέλος, η εφαρμογή των φίλτρων δεν ανανεώνει την σελίδα, για την ομαλότερη εμπειρία του χρήστη.



**Εικόνα 8**, στιγμιότυπο αναμονής για προβολή μεγάλου σετ δεδομένων στον ιστότοπο



Εικόνα 9, στιγμιότυπο από το γράφημα που δημιουργείται με μία έγκυρη είσοδο στον ιστότοπο

Name	Price	CPU Number of cores	CPU Frequency	CPU L2+L3 Cache	CPU Microarchitecture	CPU TDP	GPU Memory Size	GPU Memory Bandwidth	GPU Shaders	GPU TMUs	GPU ROPs	Skutoutz
AMD A4-3300 Tray + Sapphire Radeon R5 230 1GB Silent (11233-01-20G)	47.46	2	2.5	1024	32	65	1	10672	160	8	4	<a href="#">CPU</a> <a href="#">GPU</a>
AMD A4-3300 Tray + EVGA GeForce GT710 1GB (01G-P3-2710-KR)	48.69	2	2.5	1024	32	65	1	14400	48	8	4	<a href="#">CPU</a> <a href="#">GPU</a>
AMD A4-3300 Tray + Gigabyte GeForce GT710 1GB (GV-N710SL-1GL)	48.98	2	2.5	1024	32	65	1	14400	48	8	4	<a href="#">CPU</a> <a href="#">GPU</a>

GPU Shaders (Higher is better)  
16 - 384

GPU TMUs (Higher is better)  
8 - 16

GPU ROPs (Higher is better)  
4 - 8

Reset filters

Εικόνα 10, στιγμιότυπο από την λίστα των αποτελεσμάτων που εμφανίζεται κάτω από το γράφημα των βέλτιστων συνδυασμών, για τη λεπτομερέστερη περιγραφή τους.

## 7. Συμπεράσματα

Στην εποχή που διανύουμε, οι περισσότερες πτυχές της ζωής των ανθρώπων είναι άμεσα συνδεδεμένες με την Πληροφορική. Είτε αυτή είναι η επικοινωνία, είτε η υγεία ή η εκπαίδευση, είτε η ενημέρωση, είτε άλλα πράγματα, χρησιμοποιούμε συσκευές, ανταλλάσσουμε πληροφορίες και δεδομένα. Αυτό, σε συνδυασμό με την εξάπλωση σε όλο και μεγαλύτερο μέρος του πληθυσμού, της τεχνολογίας, και με το hardware που κατέχουμε σε αυτό το χρονικό σημείο, κάνει το φαινόμενο του «Big Data» ένα πρόβλημα μείζονος σημασίας. Επίσης, όπως είδαμε, οι συνέπειες του Big Data κλιμακώνουν έντονα ένα άλλο πρόβλημα το οποίο συναντάται συχνά σε αρκετά επιστημονικά πεδία, αυτό του να ληφθεί η απόφαση για την επιλογή της βέλτιστης λύσης ανάμεσα σε ένα πλήθος άλλων, βάση πολλών κριτηρίων. Σε τέτοιες περιπτώσεις, όταν οι υπολογισμοί είναι πολλοί και πολύπλοκοι, επιβάλλεται η διαδικασία του multiple-criteria decision analysis. Το πρόβλημα του να καταλήξουμε μετά την ανάλυση στην επιλογή μίας λύσης, το ονομάσαμε multi-objective optimization problem. Σε επιστημονικά πεδία, όπως τα οικονομικά, η υγεία, το marketing και η διαχείριση φυσικών πόρων, γίνεται επιτακτική η ανάγκη εύρεσης τρόπων που θα μας δίνουν αποτελεσματικές λύσεις και κατανοητές ερμηνείες. Τα visual analytics αποτελούν ένα από τα εργαλεία που αναπτύξαμε ώστε να αντιστρέψουμε το πρόβλημα του Big Data, να κάνουμε ευκολότερες τις πολύ-κριτηριακές αναλύσεις μας, και μέσω αυτών, να αποκτήσουμε ένα μεγάλο γνωστικό κέρδος από την εκμετάλλευσή τους.

Στην εργασία αυτή, επιλέχθηκε να ερευνηθεί το πρόβλημα της λήψης απόφασης στην αγορά ενός ζευγαριού επεξεργαστή και κάρτας γραφικών, καθώς οι επιλογές που υπάρχουν διαθέσιμες είναι πολλές και η ανάλυση που πρέπει να γίνει για να βρεθεί η ιδανική λύση, είναι υπολογιστικά μεγάλη και επίπονη. Για το λόγο αυτό, έγινε η προσπάθεια να αναπτυχθεί ένα εργαλείο το οποίο θα πραγματοποιεί αυτόματα την συλλογή των δεδομένων και την ανάλυση και, προσφέροντας στο χρήστη όλα τα απαραίτητα μέσα για να μπορεί να παρέμβει στη διαδικασία και να θέσει με ακρίβεια τις απαιτήσεις του, να οδηγηθεί με ευκολία και σιγουριά στην τελική επιλογή, μέσω του data visualization.

## 8. Βιβλιογραφία

- [1] “Multiple-criteria decision analysis,” *Wikipedia, the free encyclopedia*. 14-Sep-2016.
- [2] V. Mayer-Schönberger and K. Cukier, *Big Data: A Revolution That Will Transform How We Live, Work, and Think*. Houghton Mifflin Harcourt, 2013.
- [3] F. Rider, *The scholar and the future of the research library, a problem and its solution*,. New York city, 1944.
- [4] “Executive Summary.” [Online]. Available: <http://groups.ischool.berkeley.edu/archive/how-much-info-2003/execsum.htm>. [Accessed: 27-Sep-2016].
- [5] “IBM - What is big data?,” 07-Sep-2016. [Online]. Available: <https://www-01.ibm.com/software/data/bigdata/what-is-big-data.html>. [Accessed: 27-Sep-2016].
- [6] “big-data-analytics-105425.pdf.” .
- [7] F. Y. Edgeworth, *Mathematical Physics: An Essay on the Application of Mathematics to the Moral Sciences*. A. M. Kelley, 1881.
- [8] “Indifference curve,” *Wikipedia, the free encyclopedia*. 02-Aug-2016.
- [9] V. Pareto, *Manual of Political Economy: A Critical and Variorum Edition*. OUP Oxford, 2014.
- [10] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, 2001.
- [11] “Multi-objective optimization,” *Wikipedia, the free encyclopedia*. 25-Sep-2016.
- [12] Zeleny, M, *Compromise Programming*. .
- [13] K. Miettinen, *Nonlinear Multiobjective Optimization*. Springer Science & Business Media, 1999.
- [14] C.-L. Hwang and A. S. M. Masud, “Methods for Multiple Objective Decision Making,” in *Multiple Objective Decision Making — Methods and Applications*, Springer Berlin Heidelberg, 1979, pp. 21–283.
- [15] S. Gass and T. Saaty, “The computational algorithm for the parametric objective function,” *Nav. Res. Logist. Q.*, vol. 2, no. 1–2, pp. 39–45, Mar. 1955.
- [16] I. Das and J. Dennis, “Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems,” *SIAM J. Optim.*, vol. 8, no. 3, pp. 631–657, Aug. 1998.
- [17] “A modified NBI and NC method for the solution of N-multiobjective optimization problems.” [Online]. Available: [http://www.academia.edu/6185628/A\\_modified\\_NBI\\_and\\_NC\\_method\\_for\\_the\\_solution\\_of\\_N-multiobjective\\_optimization\\_problems](http://www.academia.edu/6185628/A_modified_NBI_and_NC_method_for_the_solution_of_N-multiobjective_optimization_problems). [Accessed: 28-Sep-2016].
- [18] “Messac\_2003\_SMO\_NC.pdf.” .
- [19] “Messac\_2004\_SDM\_NC.pdf.” .
- [20] D. Mueller-Gritschneider, H. Graeb, and U. Schlichtmann, “A Successive Approach to Compute the Bounded Pareto Front of Practical Multiobjective Optimization Problems,” *SIAM J. Optim.*, vol. 20, no. 2, pp. 915–934, Jan. 2009.
- [21] “DSDreprint.pdf.” .
- [22] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
- [23] E. Zitzler, M. Laumanns, and L. Thiele, “SPEA2: Improving the Strength Pareto Evolutionary Algorithm,” 2001.
- [24] B. Suman and P. Kumar, “A survey of simulated annealing as a tool for single and multiobjective optimization,” *J. Oper. Res. Soc.*, vol. 57, no. 10, pp. 1143–1160, Oct. 2005.
- [25] “2728849.pdf.” .



- [26] "Novelty Search Users Page." [Online]. Available: <http://eplex.cs.ucf.edu/noveltysearch/userspage/>. [Accessed: 28-Sep-2016].
- [27] "oo4\_pgen.pdf." .
- [28] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *Eur. J. Oper. Res.*, vol. 181, no. 3, pp. 1653–1669, Sep. 2007.
- [29] R. Battiti, M. Brunato, and F. Mascia, *Reactive Search and Intelligent Optimization*, vol. 45. Boston, MA: Springer US, 2009.
- [30] "Reactive Business Intelligence. From Data to Models to Insight." [Online]. Available: <http://www.kdnuggets.com/2011/03/book-reactive-business-intelligence-battiti-brunato.html>. [Accessed: 28-Sep-2016].
- [31] K. Miettinen, F. Ruiz, and A. P. Wierzbicki, "Introduction to Multiobjective Optimization: Interactive Approaches," in *Multiobjective Optimization*, J. Branke, K. Deb, K. Miettinen, and R. Słowiński, Eds. Springer Berlin Heidelberg, 2008, pp. 27–57.
- [32] S. Zionts and J. Wallenius, "An Interactive Programming Method for Solving the Multiple Criteria Problem," *Manage Sci*, vol. 22, no. 6, pp. 652–663, Feb. 1976.
- [33] A. P. Wierzbicki, "On the completeness and constructiveness of parametric characterizations to vector optimization problems," *Oper.-Res.-Spektrum*, vol. 8, no. 2, pp. 73–87.
- [34] A. P. Wierzbicki, M. Makowski, and J. Wessels, *Model-Based Decision Support Methodology with Environmental Applications*. Springer Netherlands, 2000.
- [35] H. Nakayama and Y. Sawaragi, "Satisficing Trade-off Method for Multiobjective Programming," in *Interactive Decision Analysis*, D. M. Grauer and P. A. P. Wierzbicki, Eds. Springer Berlin Heidelberg, 1984, pp. 113–122.
- [36] K. Miettinen and M. M. Mäkelä, "Interactive bundle-based method for nondifferentiable multiobjective optimization: nimbus," *Optimization*, vol. 34, no. 3, pp. 231–246, Jan. 1995.
- [37] K. Miettinen and M. M. Mäkelä, "Synchronous approach in interactive multiobjective optimization," *Eur. J. Oper. Res.*, vol. 170, no. 3, pp. 909–922, May 2006.
- [38] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon, "Visual Analytics: Definition, Process, and Challenges," in *Information Visualization*, A. Kerren, J. T. Stasko, J.-D. Fekete, and C. North, Eds. Springer Berlin Heidelberg, 2008, pp. 154–175.
- [39] "Machine Learning: What it is and why it matters." [Online]. Available: [http://www.sas.com/en\\_sg/insights/analytics/machine-learning.html](http://www.sas.com/en_sg/insights/analytics/machine-learning.html). [Accessed: 28-Sep-2016].
- [40] J. Bertin, *Semiology of Graphics: Diagrams, Networks, Maps*, 1 edition. Redlands, Calif: Esri Press, 2010.
- [41] "Data visualization," *Wikipedia, the free encyclopedia*. 14-Sep-2016.
- [42] "Edward Tufte: Books - The Visual Display of Quantitative Information." [Online]. Available: [https://www.edwardtufte.com/tufte/books\\_vdqi](https://www.edwardtufte.com/tufte/books_vdqi). [Accessed: 28-Sep-2016].
- [43] J. W. Tukey, *Exploratory Data Analysis*, 1 edition. Reading, Mass: Pearson, 1977.
- [44] "Exploratory data analysis," *Wikipedia, the free encyclopedia*. 29-Jul-2016.
- [45] "VisMaster-book-lowres.pdf." .
- [46] *Exploratory Analysis of Spatial and Temporal Data*. Berlin/Heidelberg: Springer-Verlag, 2006.
- [47] A. Slingsby, J. Dykes, and J. Wood, "Exploring Uncertainty in Geodemographics with Interactive Graphics," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2545–2554, Dec. 2011.
- [48] "Lo7-BallotMaps.pdf." .
- [49] "va\_survey.pdf." .
- [50] "Pie chart," *Wikipedia, the free encyclopedia*. 26-Sep-2016.
- [51] "Many Eyes: IBM's Free Online Data Visualization Tool," *Boost Labs*, 08-Jan-2015. .
- [52] "Tag cloud," *Wikipedia, the free encyclopedia*. 20-Sep-2016.

- [53] "Treemapping," *Wikipedia, the free encyclopedia*. 05-Sep-2016.
- [54] "Venn diagram," *Wikipedia, the free encyclopedia*. 09-Sep-2016.
- [55] "About Fusion Tables - Fusion Tables Help." [Online]. Available: <https://support.google.com/fusiontables/answer/2571232?hl=en>. [Accessed: 29-Sep-2016].
- [56] "Interactive mapping software and reporting services." [Online]. Available: <http://www.instantatlas.com/>. [Accessed: 29-Sep-2016].
- [57] "Exhibit: Data Visualization for Journalists." [Online]. Available: <http://people.csail.mit.edu/karger/Exhibit/CAR/>. [Accessed: 29-Sep-2016].
- [58] "Memory bandwidth," *Wikipedia, the free encyclopedia*. 09-Jun-2016.
- [59] K. K. Mishra and S. Harit, "A Fast Algorithm for Finding the non Dominated Set in Multiobjective Optimization," *Int. J. Comput. Appl.*, vol. 1, no. 25, pp. 46-54, Feb. 2010.