POLITECNICO DI TORINO
Repository ISTITUZIONALE

Python-based calculation tool of wind-pressure coefficients on building envelopes

Original
Python-based calculation tool of wind-pressure coefficients on building envelopes / Chiesa, G., Grosso, M.. - In: JOURNAL OF PHYSICS. CONFERENCE SERIES. - ISSN 1742-6588. - ELETTRONICO. - 1343:1(2019), p. 6. [10.1088/1742-6596/1343/1/012132]

Availability:
This version is available at: 11583/2776899 since: 2023-10-10T11:12:35Z

Publisher:
Institute of Physics Publishing

Published
DOI:10.1088/1742-6596/1343/1/012132

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

28 November 2026

Journal of Physics: Conference Series

PAPER • **OPEN ACCESS**

Python-based calculation tool of wind-pressure coefficients on building envelopes

To cite this article: Giacomo Chiesa and Mario Grosso 2019 *J. Phys.: Conf. Ser.* **1343** 012132

View the article online for updates and enhancements.



IOP ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

This content was downloaded from IP address 151.32.35.72 on 20/11/2019 at 09:57

Python-based calculation tool of wind-pressure coefficients on building envelopes

Giacomo Chiesa[1*] and Mario Grosso

[1]Politecnico di Torino, DAD, Viale Pier Andrea Mattioli 39, 10125 Torino, Italy

*E-mail: giacomo.chiesa@polito.it

Abstract. Wind pressure distribution is an essential factor for calculation of airflow rates in controlled natural ventilation systems for both indoor air quality and cooling purposes. A pressure coefficient calculation model, CpCalc, with a modular parametric approach based on analysis of wind tunnel tests was developed by Mario Grosso within the COMIS workshop, held at Lawrence Berkeley Laboratory, UCB, CA, USA, in 1988-1992, and upgraded for the European Project PASCOOL in 1994-2001. This software allows for calculating Cp at any point of a building façade and roof slope as a function of various climate, environmental, and building geometry parameters. The present paper describes a further upgrading of this software using a Pyton script, which allows for applying CpCalc to hourly-based energy dynamic simulation codes as well as being connectable in future with parametric design software.

1. Introduction

Wind pressure distribution is an essential factor for calculation of airflow rates in controlled natural ventilation systems for both indoor air quality and cooling purposes. This calculation is necessarily to design these systems, as well as an input for dynamic energy simulation tools. Currently, pressure coefficients on building envelopes, characterising the wind pressure distribution, can be taken from the following ways: 1. wind tunnel tests; 2. computer fluid dynamic software (CFD), 3. tables of façade-averaged values; 4. simplified regression-based calculation tools. The previous two methods are very time consuming and cumbersome to be used by designers. The third one is too generic and not enough accurate for building higher than two storeys, especially if calculation points are not in the middle of façades. The last one can act as an intermediate sufficiently accurate solution for designers and energy specialists for common applications (e.g. residential and not complex office buildings).

Such as underlined by the numerous recent studies on the topic, few tools were developed to calculate Cp on façade points [1,2], while a continuous research interest on the topic is evident, including the definition of new equations – e.g. [3], or the adoption of neural network approaches – e.g. [4]. It is, in fact, evident the need to predict the expected Cp value, including horizontal and vertical variations, especially for inclusion in dynamic energy simulation tools. For example, EnergyPlus is using the ASHRAE database for high-rise buildings and the Swami and Chandra equation for low-rise ones being not able in including horizontal and vertical calculation point variations without manual inputting [1].

The work here presented represents the outcome of a first phase of the development of a new version of the software Cpcalc^{+}, an upgraded version of CpCalc, a wind pressure distribution calculation model developed by Mario Grosso. This model is still one of a few able to generate, automatically, Cp values at any points of a multi-storeys building envelope as a function of climate, environmental and building

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

geometry parameters. Even recently, this tool was mentioned in literature to be one of the very few model of such type an re-validated in comparison to monitored and CFD databases [2].

The aim of this new development is to connect this tool with different software including parametrical and procedural design tools – such as BIMs and algorithmic scripting platforms (e.g. Grasshopper) – and dynamic energy simulation tools. This study implies the conversion of Cpcalc$^+$ from the programming languages on which is based the current version (VisualBasic) to Python 3.7 scripting one in order to improve flexibility and usability in several platforms.

This paper describes the first step of this upgrading developing including an introduction of the general approach and used models as well as sample results of beta-test running on a reference case study in comparison with other Cp calculation tools. In addition, an example of application of these Cp results on hourly-wind-driven airflow calculations is described considering an hourly-based-climatic file of a specific location. These databases can be used as potential input files in dynamic energy simulation software (e.g. TRNSYS or EnergyPlus).

1.1. CPCALC tool history

CpCalc was developed on FORTRAN 77 language, within the COMIS (Conjunction Of Multizone Infiltration Specialists) workshop, held at the Laurence Berkeley Laboratory 1988-90, and completed by 1992 [5].CpCalc$^+$ was developed on VISUAL BASIC language, within the EU PASCOOL Project (1994-96) [6,7] Both software have a modular structure with a parametrical approach based on regression-analyses of wind-tunnel test data. Although CpCalc was generated as a module of the Multizone Infiltration Model COMIS [8] and subsequently CpCalc$^+$ linked to AIOLOS [9] and to the energy dynamic simulation software ESP-r, both are standalone programmes.

The applied wind pressure distribution model is based on the regression analysis of an extensive data set of wind tunnel tests, compared and verified through a thorough accuracy and validation process in order to find the best reference wind pressure vertical profile on the centre of a façade as well as optimal fitting polynomial degrees with the lowest standard deviation values. The data sets used for the development of CpCalc were related to the vertical façades of rectangular-shaped models [10,11], while CpCalc$^+$ could rely on wind tunnel tests carried out on purpose on a cube-shaped model by INETI (Lisbon, Portugal, 1993).

2. Methodology and code structure

2.1. CpCalc 2.0 – general approach and used models

As CpCalc and CpCcalc$^+$, CpCalc 2.0 has a modular structure based on a parametrical approach, whereby each module represents a specific type of correction factor applied to a reference Cp profile in order to obtain Cp on a given point of a building envelope as a function of the relevant parameter.

The considered parameters can be grouped in the following categories.

a. Climate parameters: *wind incident angle (anw)* as a function of the wind direction; and *wind velocity profile exponent* (~) related to the terrain roughness of the land context surrounding the building under study.
b. Environmental parameters: *plan area density (pad)*, i.e., the ratio of building ground area to the total area of the territorial surface surrounding the building under study within a radius significant for the effect on pressure coefficient of changing flow regime; and *relative building height (rbh)*, i.e., the ratio of the height of the building under study to the average height of surrounding buildings.
c. Building geometry parameters: *azimuth angle of the building (anb)*, i.e., the angle formed by the longitudinal axis of the windward façade of the building under study with the North axis; *frontal aspect ratio (far)*, i.e., the ratio of its length to its height; *side aspect ratio (sar)*, i.e., the ratio of its width to its height; *relative vertical position (zh*, for vertical façade and *y*, for roof*)*, i.e., the ratio of the vertical, or longitudinal (roof), distance of the point where Cp is calculated on each building façade from the ground, or windward eave (roof), to the total height of the facade or projected depth of the roof; its *relative horizontal position (xl)*, i.e., the ratio of the horizontal distance of the point

where Cp is calculated on each building façade from the left edge of the facade or roof flap; *roof slope tilt angle* (¤), i.e., the angle formed by the roof single or double flap with the horizontal plane.

The fitting regression is made through polynomial curves whose exponents varies from the 3rd (windward side) to the 5th (leeward side) degree. The correlation factor was always higher than 0.95 and the relevant standard deviation lower than 0.024. Depending on the characteristics of the reference data sets, fitting coefficients are related to specific stepwise values for each parameters and linear interpolation is applied for values out of the considered range of each step. Application limits are shown for each parameter, based on the range of reference values used for the regression analyses. However, a certain extension out of those limits is allowed with a warning that the output Cp will be taken as the one of the boundary of each parameter application range.

2.2. CpCalc 2.0 code structure

Such as mentioned in the previous section, CpCalc 2.0 is based on the polynomial regressions elaborated for the early CpCalc version and fully reported in Ref. [5]. A totally new code was developed in order to overcome compatibility problems between the old CpCalc version (Fortran based) and current operative systems. Furthermore, this new code is conceived as fully implementable and adaptable to different platforms, including parametric CADs. Moreover, it is not based on a single wind-direction calculation at once, but is directly able in reading *.epw climate files and perform hourly calculations. CpCalc 2.0 was developed using Python v3.7 scripting language. This language was chosen as being dynamic, and flexible, and able to be directly coupled with IoT systems – e.g. Raspberry Pi server units for real time operation– and with CAD platforms – e.g. Rhinocersos.

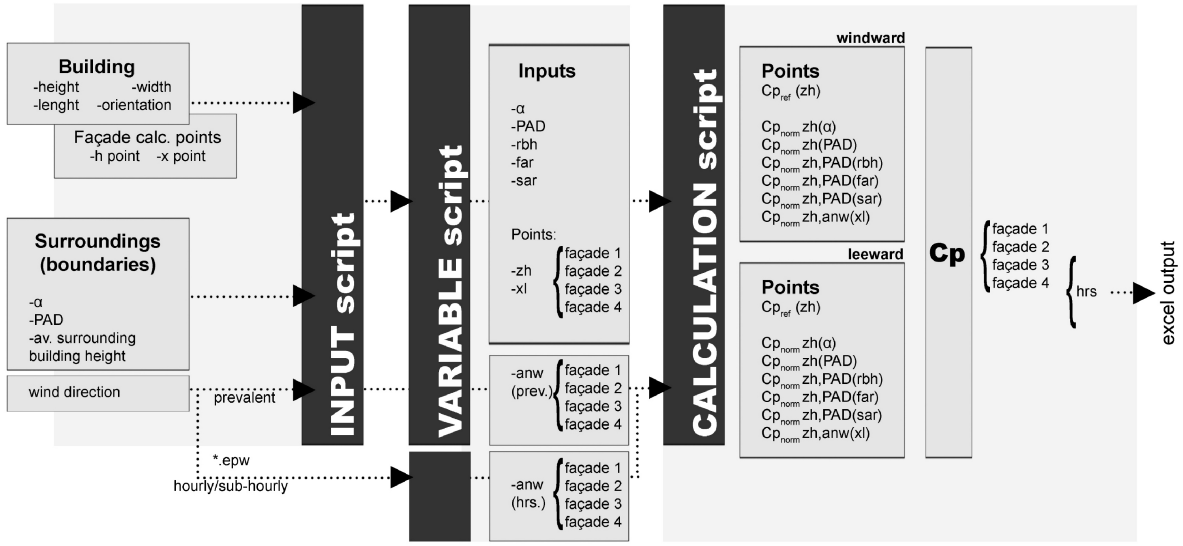

Figure 1. schematic restitution of the flowchart of the elaborated code

The CpCalc 2.0 code flowchart can be ideally subdivided into three main scripts in order to be easily adaptable to further reconfigurations. The first refers to input definition and allows users to input the required building and surrounding data. Inputs are automatically organised into an excel file which is conceived to be also operated by different approaches (e.g. it is open to additional manual modifications). The second script focuses on the elaboration of inputs to produce the required data to be used for the Cp calculation phase. For example, in this step the parameters *far*, *sar*, *rbh*, *zh*, *xl*, and *anw* are calculated for each façade. The calculation of this last variable is organized in a scalable way in order to be adapted to both a single prevalent wind direction – e.g. for early design phases – and multiple directions to perform hourly or sub-hourly calculations. Finally, the third script is devoted to the calculation of Cp for each of the inputted façade points and for the façade mid point. The calculated Cp

are hence reported in an output excel file. When hourly wind directions are considered, the output file automatically generates a Cp database. The script includes an *.epw analyser able to read wind direction and wind velocity for all hours included in the typical meteorological file. The used python wheels are openpyxl, os, bisect, and pyepw.

Figure 1 schematically shows the structure of the python code. Each script can be individually run allowing for further planned expansions. Cp calculation input may be also re-classified into three categories: fixed variables related to building and surrounding data (e.g. *far*, *sar*, *PAD*, *α*, …); façade-related data including *anw*, to define *windward* or *leeward* list of calculation procedures, changing with wind direction; and façade-calculation points.

3. Results and discussions

3.1. Comparison with other existing tools

The adopted fitting polynomial curves were already validated in previous works [5,6]. Nevertheless, being CpCalc 2.0 based on a new code, a comparison with other calculation tools was also performed. Considering the comparison study reported in Ref.[1], the same testing cubic configuration was assumed defining a 10m X 10m X 10m building in order to compare the new CpCalc 2.0 with the same tools analysed in that reference, when different wind incident angles are defined. In particular, they are considered the analytical model Cp generator (TNO) [12], the model developed by Swami and Chandra [13], and the AIVC database [14]. Furthermore, results by the CpCalc+ VisualBasic version, based on a different set of regression equations, were also calculated.

The wind velocity profile exponent *α* was assumed to be 0.22 (suburban environment) in CpCalc 2.0 and CpCalc+ and compared with an aerodynamic roughness length of 0.5 in Cp generator [1], while the other considered tools do not include effects of surrounding buildlings. Three points are used for comparison: bottom-left point *zh,xl*=0.1, central façade point *zh,xl*=0.5, and upper-right point *zh,xl*=0.9 allowing for a direct comparison with the mentioned references. Figure 2 shows the obtained results for (a) bottom-left point, (b) mid-façade point, and (c) upper-right point in a wind incident angle domain {0-180°} considering a variation step of 15°. All models, including the new CpCalc 2.0 show similar behaviours especially for the windward case. Variations can be underlined especially for points farer from the mid façade ones, being Swami and Chandra and AIVC models not able to implement horizontal and vertical variations. Furthermore, CpCalc 2.0 responds similarly to the CpCalc+ version for the windward case, but differs from it for bottom and upper points in the leeward case, since CpCalc+ shows differences in the fitting regressions with respect to the original CpCalc – e.g., for the roof effect [6].

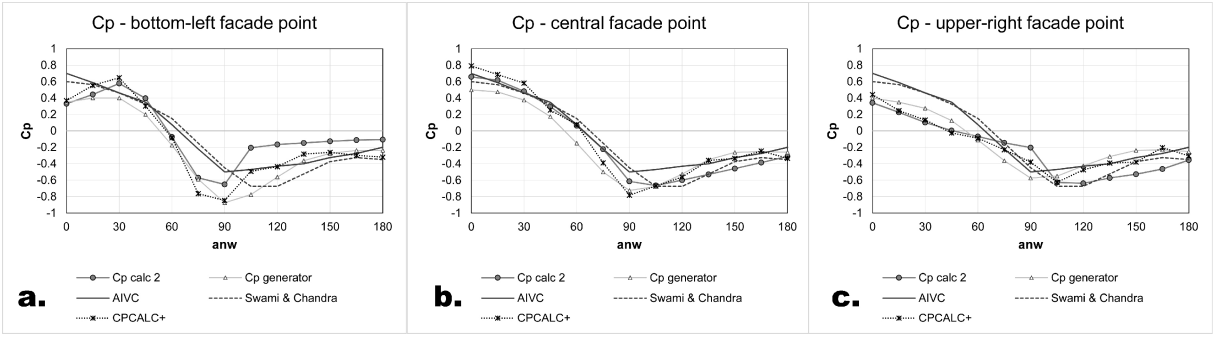

Figure 2. Comparison between the new CpCalc 2.0 and other existing Cp calculation tool – Cp generator, Swami and Chandra, and AIVC database [1], considering (a) a bottom-left point; (b) the mid façade point; and (c) a upper-right point.

3.2. Sample application to define expected wind-driven ACH

A sample rectangular building (15m X 10m X 9m) facing south was assumed to test the applicability of CpCalc 2.0 for calculating the expected wind-driven ACH. This module was divided into 9 space units

(5m X 10m X 3m) with openings in opposite facades. Three units were selected, positioned in the bottom-left, central, and upper-right portions of the building. The openings have a net area of 0.25 m^2 and are positioned in the centre of each unit main facade. PAD was assumed to be 12.5, α 0.32 (suburban area), and surrounding building heights 12 meters. A sample location was assumed by adopting the typical meteorological year of Rome-Fiumicino airport produced by METEONORM (*.epw format). Fig. 3 shows the python script interfaces (a), together with (b) the calculated hourly ACH for the 3 considered units for the month of June. The volumetric airflow (q_n) was calculated using the following equation [15]:

$$q_n = v_{w,n,h} * c_d * A * \sqrt{\Delta c_p} \quad [m^3/s] \tag{1}$$

Where $v_{w,n,h}$ is the wind velocity of hours n assumed by the *.epw file and reported to opening heights using the power low equation [16], c_d is the discharge coefficient, here set to 0.6 [17], and A the window net opening area. Hence, ACH is derived by dividing q_n * 3600 [s/h] by the unit volume.

Fig. 3 show the simplicity of the early-interface of CpCalc 2.0 and an extraction of obtained results in terms of ACH. It is possible to see the variation of this value according to local wind velocity and direction and in respect to the opening points. Variations in between the 3 units are in fact related to the possibility to define Cp values for specific points on a façade and only refer to the façade average values. The graph of Fig. 3(b) can constitute a base point for optimizing controlled opening areas, opening positioning on a façade and further design and operation choices connected to wind-driven ventilative cooling aspects.

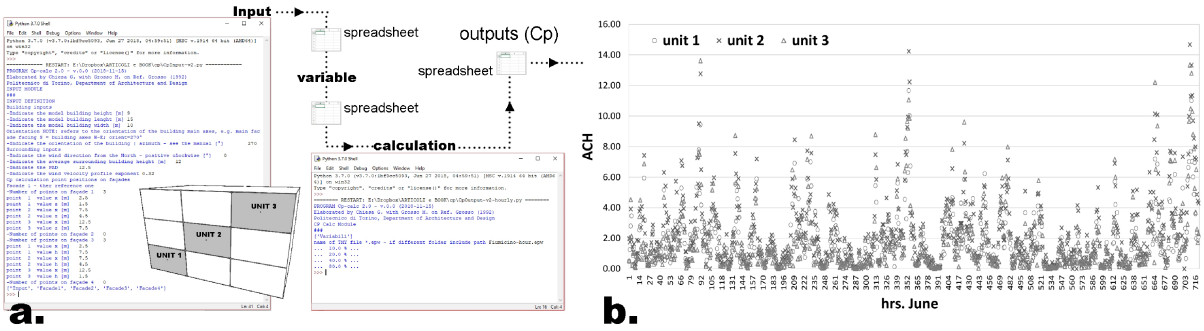

Figure 3. (a) the python code interfaced including a sketch of the sample building; (b) hourly calculated ACH for the 3 considered units (month of June).

3.3. Next development steps

CpCalc 2.0 development is part of a larger research action. The code elaborated for this paper constitutes, in fact, the first step of a larger project including the definition of inputs (first script) directly by a CAD environment – e.g. Rhinoceros with Grasshopper – and the expansion of the script by including also the calculation of wind-driven ventilation airflows in buildings. This last step allows to produce data that can be coupled with energy simulation tools to define a design methodology able to check the fulfilment of IAQ and cooling requirements by controlled natural ventilation (including stack-effect calculation). The first version of this tool is expected to be prepared for the Grasshopper environment in order to implement an optimisation process to opening positioning and dimensioning. Following an approach introduced in [18-20], a pure python code version coupled with dynamic energy simulations is also under development.

Furthermore, the code will be implemented with the adoption of regression fittings for differently tilted roofs – see CpCalc+ version –, and the expansion of the fitting boundary conditions including, for the first time, the effect of single obstacles and not only homogeneous conditions thanks to an already acquired database.

4. Conclusions

In this paper the new version of CpCalc tool was introduced. This tool is based on equations developed for the first CpCalc version, but is scripted in Python language allowing for higher integration and adaptation in comparison to the other CpCalc versions that in some cases are even not able to be operated simply on new operative systems. Furthermore, this new code is conceived to directly extract wind direction by typical meteorological years, such as *.epw files, in order to perform hourly calculations. Early comparison with other existing software show a good agreement, in line with the ones reported for other tools in literature.

The paper constitutes a first step in a larger research project aiming at developing a CAD compatible plug-in to optimize since early-design phases CNV potential.

Acknowledgment

Code development was co-funded by the University Grant 59_ATEN_RSG16CHG and 59_ATEN_RBA_17GRM.

References

[1] Cóstola D, Blocken B, Hensen JLM 2009 *Building and Environment* **44** 2027-36
[2] Ramponi R, Angelotti A, Blocken B 2014 *Applied Energy* **123** 185-95
[3] Muehleisena RT, Patrizi S 2013 *Energy and Buildings* **57** 245-9
[4] Bre F, Gimenez JM, Facchiotti VD 2018 *Energy and Buildings* **158** 1429-41.
[5] Grosso M 1992 *Energ Buildings* **18** 101-31
[6] Grosso M, Marino D, Parisi E 1994 *Proc. European Conf. on Energy Performance and Indoor Climate in Buildings*, 24-26 November, Lyon, France (Vaulx: Ecole National des Travaux Publics de l'Etat) pp 167-72
[7] Grosso M 2001 *CpCalc^{+} Final Report.* PASCOOL, JOU2-CT79-0013, EU, DG XII, Joule II–Non Nuclear Energy programme, Sub-programme: Energy Conservation and Utilisation.
[8] Feustel H, Grosso M, et al. 1990 *Fundamentals of the Multizone Air Flow Model - COMIS*. vol. 29, p. 1-114, Coventry: IEA-AIVC.
[9] Allard F (eds) 1998 *Natural Ventilation in Buldings*, James&James, London
[10] Akins RE, Cermak JE 1976 Wind Pressure on Buildings, CER76-77REA-JEC15. Fluid Dynamic and Diffusion Laboratory, Colorado State University, CO, USA.
[11] Hussain M, Lee BE 1980 An Investigation of Wind Forces on Three Dimensional Roughness Elements in a Simulated Atmospheric Boundary Layer, BS 55, Part I and BS 56, Part II. Department of Building Science, University of Sheffield, UK.
[12] Knoll B, Phaff JC, de GidsWF 1995 Pressure simulation program. In: *Proceedings of the conference on implementing the results of ventilation research*. AIVC
[13] Swami MV, Chandra S 1988 Correlations for pressure distribution on buildings and calculation of natural-ventilation airflow. ASHRAE Transactions **94** 243–66
[14] AIVC 1984 Wind pressure workshop proceedings AIC-TN-13, Brussels
[15] Grosso M 2017 *Il raffrescamento passivo degli edifici,* 4 edn., Maggioli, Rimini
[16] Brunt D 1952 *Physical and dynamical meteorology*, 2 end., Cambridge Un. Press, Cambridge
[17] Chiesa G, Grosso M 2015 *Energy Build.* **107** 376-91
[18] Chiesa G, et al. 2018 *Sustainable Mediterranean Construction* **8** 36-40
[19] Chiesa G 2015 *Paradigmi, tecnologie ed ere digitali* aAccademia University Press, Turin
[20] Chiesa G et al. 2019 *MedGreen Forum 2019*, 3-5 September, Florence. *Accepted paper*