

autoVB User Guide

1. Introduction

autoVB(v0.1.4) is an intelligent automation tool for XMVB calculations that operates directly from molecular structures. It automatically calls Gaussian for NBO calculations, intelligently chooses the active space, supplies reliable initial guesses, generates XMVB input files, runs the calculations, and delivers visual result analysis—all without manual intervention, thereby greatly reducing the complexity of valence bond theory computations.

If you are not familiar with valence bond theory, refer to the [XMVB tutorial](#).

2. Input Files

2.1 Input File Examples

Two input formats are supported. The default format closely resembles a Gaussian .gjf file:

```
%mem=4GB
%nprocshared=4
#p vb3cf(4,4)/cc-pVDZ

autovb{}

0 1
C   -3.329219900    2.091826940    0.000000000
H   -2.839967430    1.120094350   -0.000000040
H   -4.414989510    2.089928580    0.000000000
C   -2.624068780    3.231943550    0.000000010
H   -3.149818500    4.187367240    0.000000030
C   -1.168018280    3.297558130    0.000000020
H   -0.642289190    2.342122530    0.000000040
C   -0.462840870    4.437656730    0.000000010
H   -0.952055180    5.409407090   -0.000000040
H    0.622929910    4.439514340   -0.000000010
```

The other format resembles an XMVB .xmi input file. Use the .xmi file extension to select this format. If the extension is incorrect, autoVB will also attempt to identify the format automatically. See the [XMVB documentation](#) for details:

```
C4H6
$ctrl
vbscf
nao=4
nae=4
basis=cc-pvdz
$end

$actorb
1
4
6
8
$end

$geo
C   -3.329219900    2.091826940    0.000000000
H   -2.839967430    1.120094350   -0.000000040
H   -4.414989510    2.089928580    0.000000000
C   -2.624068780    3.231943550    0.000000010
H   -3.149818500    4.187367240    0.000000030
C   -1.168018280    3.297558130    0.000000020
H   -0.642289190    2.342122530    0.000000040
C   -0.462840870    4.437656730    0.000000010
H   -0.952055180    5.409407090   -0.000000040
H    0.622929910    4.439514340   -0.000000010
$end
```

2.2 Route Section (the line beginning with #)

The route section must use the format # method(nae,nao)/basis-set or # method/basis-

set. For .xmi input files, these parameters are specified in the \$ctrl block instead.

2.2.1 Method

Currently supported methods include vbscf, vbpt2, lam-dfvb, blw, and tbvbscf. Active-space selection options are not available for blw.

2.2.2 Explicit Active-Space Specification

The numbers of active electrons and active orbitals can be declared explicitly in parentheses after the method name, for example, vbscf(4,4). If they are omitted, autoVB determines the active space automatically. Values specified here override the corresponding settings in autovb{ }.

2.2.3 Basis Set

The basis set used for the calculation. Any basis set supported by both Gaussian and XMVB can be used.

2.3 Title-Line Control Parameters

In the title line of the input file, use the autovb{...} syntax to control the valence bond calculation.

2.3.1 Active-Space Selection Parameters

During automatic active-space selection, autoVB reads the orbital types, occupation numbers, and atomic connectivity from the Gaussian NBO results. It then determines the active orbitals according to the following priority:

1. If aat is provided, related NBOs are searched for on the specified atoms. If aoa_bond is provided, bonding orbitals are searched for on the specified atom pairs.
2. If ao_i is provided, the specified NBO indices are used directly.
3. If only nae and/or nao are provided, occupied NBOs are selected in ascending order of NBO occupation number.
4. If threshold is provided, orbitals satisfying $1 < \text{occ} \leq \text{threshold}$ are selected first.
5. If none of the parameters above are provided, the default rules are used. These rules select low-occupation BD orbitals, bonding/antibonding pairs for which the BD occupation is low and the corresponding BD* occupation is significant, and low-occupation LP orbitals. The default thresholds are $\text{BD} < 1.96$, $\text{BD} < 1.99$ and $\text{BD}^* > 0.06$, and $\text{LP} < 1.96$. If nolp is enabled, LP orbitals are skipped.

The default rules automatically tighten the thresholds according to the resulting numbers of active electrons and orbitals to avoid an excessively large active space. If

the active space remains too large, the program issues a warning and recommends specifying `aat`, `aoi`, or `nae/nao` manually.

2.3.1.1 `nae=n`

Explicitly specifies the number of active electrons. If omitted, `autoVB` selects the active space automatically from the NBO occupation numbers.

2.3.1.2 `nao=n`

Explicitly specifies the number of active orbitals. If omitted, `autoVB` selects the active space automatically from the NBO occupation numbers.

2.3.1.3 `aat=(n1, n2, ...)`

Specifies the Active Atom sequence, i.e., the indices of the atoms involved in the active space. For example, `aat=(1,2,3,4)` automatically searches for NBOs on atoms 1 through 4 and uses them as active orbitals. Atom indices are one-based integers in the range [1, number of atoms]. For an `.xml` input file, this parameter uses the data in the `$actorb` block; see the [input file examples](#).

2.3.1.4 `aoi=(n1, n2, ...)`

Specifies the Active Orbital Indices. The one-based indices of the NBOs used in the active space are given directly. For example, `aoi=(1,2,3,4)` selects NBOs 1 through 4 as active orbitals.

2.3.1.5 `threshold=n`

Specifies the NBO occupation-number threshold used to select active orbitals automatically. The value must be greater than or equal to 1.

2.3.1.6 `rethre=n`

Selects orbitals according to rectified occupation numbers. For an NBO bonding/antibonding pair, the occupation numbers are rectified so that their sum is 2.

2.3.1.7 active_order=n

Controls how the active orbitals are reordered. The default is `rumer`; when [aat](#) is specified, the default is `aoa`. The available options are:

aoa: Follow the user-specified `aat` sequence.

rumer: Follow the topological order of the selected active-space atoms while attempting to minimize crossings.

seq: Sort in descending numerical order.

none: Do not reorder the active orbitals.

2.3.1.8 aoa_bond=(n1, n2, ...)

Specifies the atom sequence for bonding active orbitals. For example, `aoa_bond=(1,2,3,4)` indicates bonds between atoms 1 and 2 and between atoms 3 and 4; these two bonds are then selected for the active space. If an odd number of indices is supplied, the final atom is interpreted as a lone pair. Atom indices are one-based integers in the range [1, number of atoms]. This option cannot handle an intended active space containing more than one lone pair correctly, because two lone-pair atoms would be interpreted as a bond. Therefore, this option is not recommended in that case.

2.3.1.9 bond_first

A sub-option of [aoa](#). When enabled, bonding orbitals are searched before lone pairs. It can be enabled when the intended active space does not contain low-occupation lone pairs.

2.3.1.10 nolp

Excludes lone pairs from active-space selection. It is supported by both [aoa](#) selection and the parameter-free default selection. Unlike [bond first](#), which may still select a lone pair when no suitable bonding orbital is available, `nolp` always excludes lone pairs.

2.3.2 NBO Calculation Parameters

2.3.2.1 nbo=n

Specifies the method used to calculate the NBOs. The default is `hf`. Different methods may produce different active-space sizes and initial guesses. Methods other than HF and common DFT functionals such as B3LYP have not been validated and are not

recommended.

2.3.2.2 nbo_file=n

Reads an existing NBO calculation. Here, n is a file name including its extension, and the file must be in the same working directory as the input file. Molecular information is read from the .fch file, NBO output information from the .out file, NBO coefficients from the .37 file, and PNBO coefficients from the .36 file. autoVB does not run an NBO calculation when this option is used.

Note: autoVB currently ignores coordinates in the NBO output and uses the coordinates supplied in the autoVB input file. Make sure the coordinates used for the NBO calculation are identical to those in the autoVB input file.

2.3.3 XMVB Calculation Parameters

2.3.3.1 str=n/stru=n

Specifies the types of valence bond structures generated by XMVB. Available values include "full", "cov", "ion(a,b,c)", and "ion(a-b)". The default is "full". See the [XMVB documentation](#) for details.

2.3.3.2 int=n/inte=n

Specifies the integral method used by XMVB. Available values are libcint, ri, and cosx. The default is libcint. See the [XMVB documentation](#) for details.

2.3.3.3 iscf=n

Specifies the orbital-optimization method used by XMVB. Available values are 2, 5, and 6. The default is 5. See the [XMVB documentation](#) for details.

2.3.3.4 sort

Sorts the valence bond structures according to their structure coefficients.

2.3.3.5 guess=n

Specifies the initial guess used by XMVB and how the active space is selected. Available values are nbo, pnbo. The default is nbo.

nbo: Use NBOs both as the initial guess and as the basis for active-space selection.

pnbo: Use NBOs for active-space selection and PNBOs as the initial guess. PNBOs are nonorthogonal and strictly localized.

2.3.3.6 atom_slice

Truncates each molecular orbital to a single atom by setting basis-function coefficients on all other atoms to zero, thereby producing a localized initial guess. XMVB generally performs this truncation automatically, so this option is usually unnecessary.

2.3.3.7 novb

Skips the XMVB calculation and generates only the input file. This option is recommended when XMVB is unavailable or when the generated input file needs to be edited manually.

2.3.4 Plotting Parameters

2.3.4.1 draw_xmo

Invokes the draw_xmo tool to parse the .xmo file and generate an .svg image of the molecular valence bond structures. The generated molecular layout is provided for reference and may contain errors.

2.3.4.2 xmo2svg=n

Parses the .xmo file after the XMVB calculation and generates a valence bond structure .svg image with the same base name as the .xmo file. The following options are supported:

Not specified (default: None): Do not generate an .svg image.

xmo2svg=rdkit: Generate a standard two-dimensional layout with RDKit based on molecular connectivity. This option is suitable for ordinary single-fragment molecules and does not preserve features of the input three-dimensional geometry.

xmo2svg=pca: Project the input three-dimensional coordinates onto a two-dimensional plane using principal component analysis. This option attempts to preserve the overall orientation of the original geometry and is suitable when the drawing should reflect the input conformation.

xmo2svg=optimized3d: Search multiple viewing directions for a three-dimensional

view with less atom overlap, bond compression, and bond crossing, then orthographically project the molecule onto a two-dimensional plane. This option is suitable for nonplanar molecules and generally produces a clearer result than direct PCA projection.

xmo2svg=contact: Arrange separate molecular fragments primarily according to interfragment atom pairs occurring in the valence bond structures. This option is suitable for reactant complexes, transition states, and other systems containing multiple molecular fragments. For a single-fragment molecule, it uses the standard RDKit two-dimensional layout.

2.3.4.3 hide_svg_labels

Hide the atom numbers in an xmo2svg image and the bonded-atom labels that follow each structure weight. Use this parameter together with xmo2svg.

2.3.4.4 svgweight

Selects the structure weights displayed in the xmo2svg image. Supported values are cc, lowdin, and both; the default is both. With both, the CC and Lowdin weights are shown on separate lines, while structures are still selected and sorted by Lowdin weight.

2.3.4.5 draw_rumer

Generates a molecular .svg image when the selected active-space atoms are ordered topologically according to the Rumer rule.

2.3.5 Other Parameters

2.3.5.1 debug

Enables debug mode and prints additional output.

3. Examples

3.1 Automatic Calculation of Benzene

This example demonstrates how autoVB performs the NBO calculation, automatically selects the active space, generates the XMVB input, runs VBSCF, and plots the major valence bond structures without requiring a manually specified active space. The input

file is shown below:

```
# vbscf/6-31g*

autovb{str=cov}

0 1
C      0.6995000584   1.2115696411   0.0000000000
H      1.2460106991   2.1581538376   0.0000000000
C     -0.6995000584   1.2115696411   0.0000000000
H     -1.2460106991   2.1581538376   0.0000000000
C     -1.3990001169   0.0000000000   0.0000000000
H     -2.4920213982   0.0000000000   0.0000000000
C     -0.6995000584  -1.2115696411   0.0000000000
H     -1.2460106991  -2.1581538376   0.0000000000
C      0.6995000584  -1.2115696411   0.0000000000
H      1.2460106991  -2.1581538376   0.0000000000
C      1.3990001169   0.0000000000   0.0000000000
H      2.4920213982   0.0000000000   0.0000000000
```

This example performs a VBSCF/6-31G* calculation on neutral singlet benzene. Because nae, nao, aat, and aoi are not specified, autoVB automatically selects the active space from the NBO analysis.

3.1.1 Active-Space Selection

The program first identifies 42 electrons, 21 occupied orbitals, and 81 virtual orbitals. It then selects three low-occupation BD orbitals and their corresponding BD* orbitals according to the default thresholds:

NBO orbital	Atoms involved	BD occupation	BD* occupation
4	C1-C11	1.66476	0.33237
7	C3-C5	1.66476	0.33237
12	C7-C9	1.66476	0.33237

Each occupied BD orbital enters the active space together with its corresponding BD* orbital. The resulting active space therefore contains 6 active electrons in 6 active orbitals, i.e., (nae, nao)=(6,6). Of the 21 occupied NBOs, 3 are active occupied orbitals and the remaining 18 are inactive. The program then orders the active atoms according to the Rumer rule:

[1, 3, 5, 7, 9, 11]

These indices correspond to the six carbon atoms in the input file. The generated XMVB input writes nae=6 and nao=6 to the \$ctrl section and writes the atom sequence above to the active-orbital part of \$orb.

3.1.2 XMVB Results

The XMVB calculation converges successfully. The key results are:

Item	Result
Method	VBSCF
Basis set	6-31G*
Active space	(6,6)
Convergence	Converged
Iterations	51
VBSCF energy	-230.56875969 a.u.
Number of valence bond structures	5

The five most important structures by weight are:

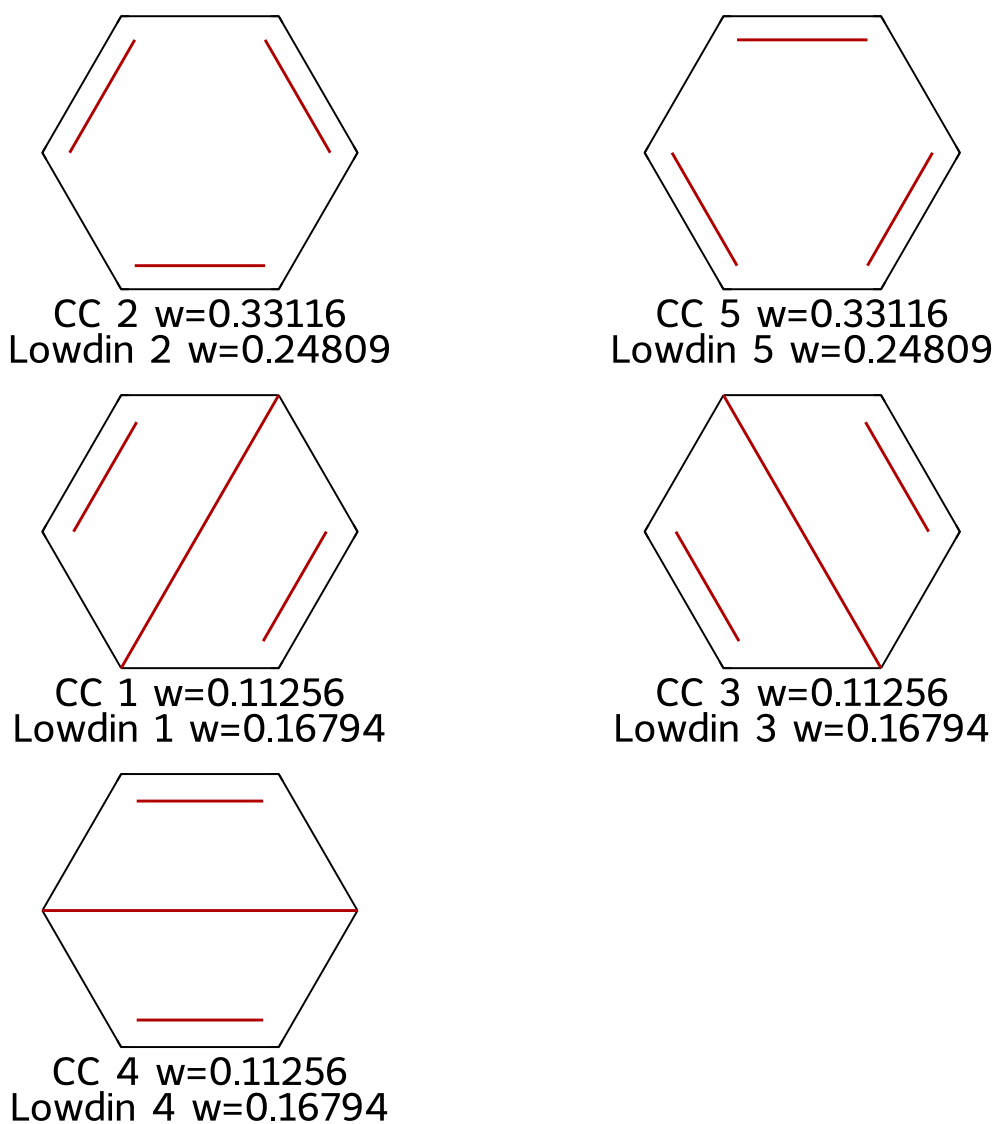
Structure	Bonded atom pairs (input atom indices)	CC weight	Lowdin weight
2	3-5, 7-9, 1-11	0.33116	0.24809
5	1-3, 5-7, 9-11	0.33116	0.24809
1	5-7, 3-9, 1-11	0.11256	0.16794

3	3-5, 1-7, 9-11	0.11256	0.16794
4	1-3, 7-9, 5-11	0.11256	0.16794

These five structures correspond to the two Kekule forms and three symmetry-equivalent Dewar forms of benzene; the two Kekule structures have the same maximum weight, consistent with the symmetry of the molecule.

3.1.3 Valence Bond Structure Plot

Plotting uses both CC and Lowdin weights by default. Red lines represent active bonds in each valence bond structure.



The figure shows five major covalent structures. Structures 2 and 5 are the two Kekule structures and have the highest weights, while structures 1, 3, and 4 are three Dewar-

type structures with lower weights. This result shows that, in the full five-structure model, benzene resonance is dominated by the two Kekule structures but also contains a smaller Dewar contribution.

3.2 Specified-Structure Valence Bond Calculation

If you want to specify particular valence bond structures, you typically need to use the XMVB .xmi input format rather than the standard Gaussian format. This example shows how to specify the two Kekule structures in the input file for benzene molecule.

The input file is:

```

$ctrl
autovb
vbscf
basis=6-31g*
$end

$str
1:18 19 20 21 22 23 24
1:18 19 24 20 21 22 23
$end

$geo
C      0.6995000584   1.2115696411   0.0000000000
H      1.2460106991   2.1581538376   0.0000000000
C     -0.6995000584   1.2115696411   0.0000000000
H     -1.2460106991   2.1581538376   0.0000000000
C     -1.3990001169   0.0000000000   0.0000000000
H     -2.4920213982   0.0000000000   0.0000000000
C     -0.6995000584  -1.2115696411   0.0000000000
H     -1.2460106991  -2.1581538376   0.0000000000
C      0.6995000584  -1.2115696411   0.0000000000
H      1.2460106991  -2.1581538376   0.0000000000
C      1.3990001169   0.0000000000   0.0000000000
H      2.4920213982   0.0000000000   0.0000000000
$end

```

This is an XMVB .xmi-format input file: the \$ctrl block controls the method, basis set, and image output, the \$str block gives the connection patterns for two valence bond structures, and the \$geo block gives the molecular coordinates.

3.2.1 Active-Space Selection

As in the automatic selection in [3.1.1](#), the program still selects three pairs of low-occupation BD/BD* orbitals in benzene, resulting in an active space of (nae, nao)=(6,6).

3.2.2 XMVB Results

The XMVB calculation converges successfully. The key results are:

Item	Result
Method	VBSCF
Basis set	6-31G*
Active space	(6,6)
Convergence	Converged
Iterations	49
VBSCF energy	-230.56408042 a.u.
Number of valence bond structures	2

The two valence bond structures have almost identical weights and correspond to the two Kekule bonding patterns of benzene:

Structure	Bonded atom pairs (input atom indices)	CC weight	Lowdin weight
1	1-3, 5-7, 9-11	0.49999	0.49999
2	1-11, 3-5, 7-9	0.50001	0.50001

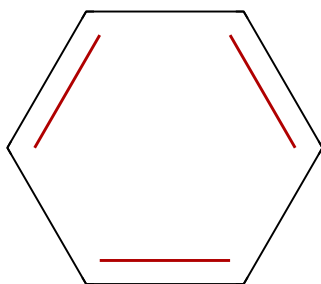
This shows that, under the two-structure restriction, the two Kekule structures contribute almost equally to the valence bond wavefunction. Further comparing with the five-structure result in [3.1.1](#):

$$\Delta E = E_{4.2} - E_{4.1} = (-230.56408042) - (-230.56875969) = 0.00467927 \text{ a.u.}$$

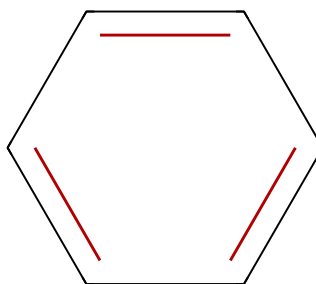
where $E_{4.1} = -230.56875969$ a.u. is the final energy of the five-structure model (two Kekule structures plus three Dewar structures), and $E_{4.2} = -230.56408042$ a.u. is the final energy of the two-Kekule model. This difference is about 2.94 kcal/mol, indicating that the three Dewar structures provide about 2.94 kcal/mol of resonance stabilization.

3.2.3 Valence Bond Structure Plot

This figure shows the two Kekule structures. Their weights are nearly identical.



CC 2 w=0.50002
Lowdin 2 w=0.50002



CC 1 w=0.49998
Lowdin 1 w=0.49998

3.3 Specifying the Active Space for a Complex System

This example demonstrates how atom indices can be used to specify the active space when the default automatic selection does not accurately represent the target reaction region. The input file is shown below:

```
#p vbscf/6-31g(d)

autovb{str=cov,nolp,aat=(5,4,2,1,6,7,8,9)}

0 1
C      -0.61461      1.64866      -0.14318
C       0.48763       0.94566      -0.65308
H       0.71579       0.95698      -1.71111
C       1.41105       0.27025       0.26981
O       1.10384       0.15207       1.45186
C      -2.15769       0.56469      -0.13681
C      -1.90990      -0.62090       0.58183
C      -0.95283      -1.51089       0.05185
O      -0.34442      -1.22918      -1.00876
C       2.69763      -0.28811      -0.29293
H      -2.89530       1.24987       0.27046
H      -0.66170      -2.39630       0.63398
H      -1.01367       2.43637      -0.77523
H      -2.26681      -0.75770       1.59602
H      -2.20092       0.45166      -1.21511
H      -0.55594       1.89304       0.91218
H       3.28877      -0.73227       0.50695
H       2.46925      -1.04416      -1.04890
H       3.27747       0.50325      -0.77898
```

This molecule is a Diels-Alder reaction transition state and is calculated at the VBSCF/6-31G(d) level. The control parameters have the following meanings:

- `aat=(5,4,2,1,6,7,8,9)`: Specify the atoms involved in the active space and define their active-orbital ordering.
- `nolp`: Exclude lone pairs during the NBO search so that the oxygen lone pairs are not included in the target active space.
- `str=cov`: Generate only covalent valence bond structures without expanding the ionic structures.

3.3.1 Active-Space Selection

The NBO calculation produces 68 electrons, 34 occupied orbitals, and 121 virtual

orbitals. According to the user-specified aat sequence and the nopl rule, the program selects 4 occupied active orbitals and adds the corresponding unoccupied orbital for each one. The resulting active space is (nae, nao)=(8,8):

Item	Result
Active electrons	8
Active orbitals	8
Active occupied orbitals	4
Inactive occupied orbitals	30
Active atom order	5, 4, 2, 1, 6, 7, 8, 9

3.3.2 XMVB Results

The XMVB calculation converges successfully. The key results are:

Item	Result
Method	VBSCF
Basis set	6-31G(d)
Active space	(8,8)
Structure type	Covalent structures (str=cov)
Convergence	Converged
Iterations	97
VBSCF energy	-420.26804610 a.u.
Number of valence bond structures	14

The five structures with the largest Lowdin weights are:

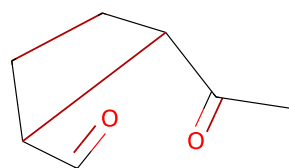
Structure	Bonded atom pairs	Lowdin weight	CC weight
-----------	-------------------	---------------	-----------

12	5-4, 1-6, 2-7, 8-9	0.27096	0.34829
11	5-4, 1-6, 7-8, 2-9	0.17858	0.23426
14	5-4, 2-1, 6-7, 8-9	0.12956	0.18634
7	4-2, 1-6, 5-7, 8-9	0.07724	0.07770
13	5-4, 2-1, 7-8, 6-9	0.07235	0.04798

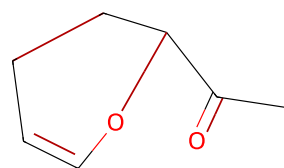
Structure 12 has the largest weight and is the dominant covalent structure in this calculation. Structures 11 and 14 also make substantial contributions.

3.3.3 Valence Bond Structure Plot

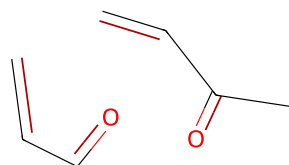
After the calculation, the image is generated by ordering all 14 covalent structures with Lowdin weights and searching for a clear 2D projection from the input 3D coordinates. Red lines represent active-space electron pairing, black lines represent the underlying molecular connectivity, and each structure is labeled with its index, weight, and bonded atom pairs.



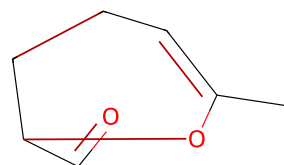
CC 12 $w=0.34829$
Lowdin 12 $w=0.27096$



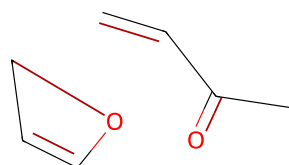
CC 11 $w=0.23426$
Lowdin 11 $w=0.17858$



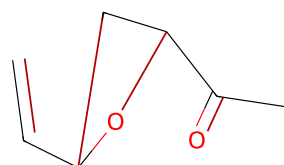
CC 14 $w=0.18634$
Lowdin 14 $w=0.12956$



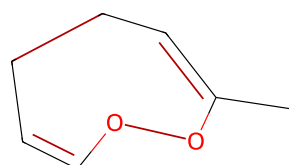
CC 7 $w=0.07770$
Lowdin 7 $w=0.07724$



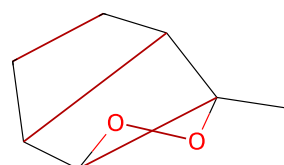
CC 13 $w=0.04798$
Lowdin 13 $w=0.07235$



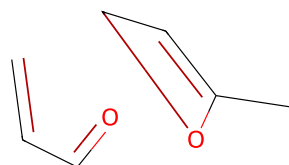
CC 10 $w=0.01641$
Lowdin 10 $w=0.05972$



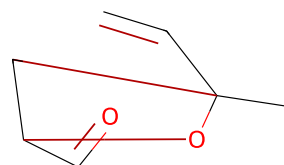
CC 6 $w=0.05437$
Lowdin 6 $w=0.04404$



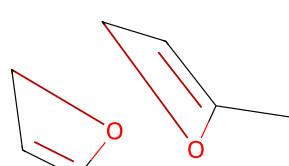
CC 1 $w=-0.00020$
Lowdin 1 $w=0.03884$



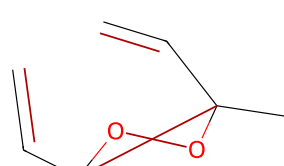
CC 9 $w=0.01815$
Lowdin 9 $w=0.03589$



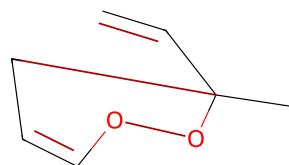
CC 4 $w=0.00542$
Lowdin 4 $w=0.03192$



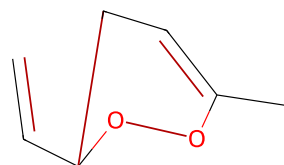
CC 8 $w=0.00304$
Lowdin 8 $w=0.01747$



CC 2 $w=0.00045$
Lowdin 2 $w=0.01690$



CC 3 $w=0.00391$
Lowdin 3 $w=0.01379$



CC 5 $w=0.00387$
Lowdin 5 $w=0.01273$

Structure 12 in the upper-left corner has the largest weight, with active electron pairing 5-4, 1-6, 2-7, and 8-9. The remaining structures show alternative electron-pairing patterns among the same active atoms, providing a direct view of the relative contribution of each covalent structure to the total wavefunction.