# Auxiliary Space Preconditioners for Linear Virtual Element Method

Yunrong Zhu

## 1 Introduction

In this paper, we present the auxiliary space preconditioning techniques for solving the linear system arising from linear virtual element method (VEM) discretizations on polytopal meshes of second order elliptic problems in both 2D and 3D domains. The VEMs are generalizations of the classical finite element methods (FEMs), which permit the use of general polygonal and polyhedral meshes. Using polytopal meshes allows for more flexibility in dealing with complex computational domains or interfaces (cf. [12]). It also provides a unified treatment of different types of elements on the same mesh. In recent years, a lot of work has been devoted to the design and analysis of the discretization methods. Less attention has been paid to developing efficient solvers for the resulting linear systems. Only recently, have the balancing domain decomposition by constraint (BDDC) and the finite element tearing and interconnecting dual primal (FETI-DP) methods been studied in [6] for VEM methods. Some two-level overlapping domain decomposition preconditioners were developed and analyzed in [8, 9] for VEM in two dimensions. A $p$-version multigrid algorithm was proposed and analyzed in [1].

The auxiliary space preconditioners we consider here can be understood as two-level methods, with a standard smoother on the fine level and a "coarse space" correction. The fine level problem is the VEM discretization on polytopal mesh, and the coarse level problem is a standard conforming $\mathbb{P}_1$ finite element space defined on an auxiliary simplicial mesh. It is natural to choose the standard $\mathbb{P}_1$ finite element space as the coarse space for a couple of reasons: (1) the degrees of freedom of the coarse space are included in the VEM space – so asymptotically, the "coarse" space should provide a good approximation for the solution on the "fine" space; (2) there are a lot of works on developing efficient (and robust) solvers for the standard conforming

Yunrong Zhu
Department of Mathematics & Statistics, Idaho State University, 921 S. 8th Ave., Stop 8085 Pocatello, ID 83209, USA. e-mail: zhuyunr@isu.edu

$\mathbb{P}_1$ finite element discretization, so we can use any existing solvers/preconditioners as a coarse solver. One of the main benefits of these preconditioners is that they are easy to implement in practice. The procedure is the same as for the standard multigrid algorithms with the grid-transfer operators between the virtual element space and the conforming $\mathbb{P}_1$ finite element space. Since the same degrees of freedom are used, we can simply use the identity operator as the intergrid transfer operator between the coarse and fine spaces.

Due to page limitation, we only state the main result and provide some numerical experiments to support it. We refer to [20] for more detailed analysis and further discussion of the preconditioners. The rest of this paper is organized as follows. In Section 2, we give basic notation and the virtual element discretization. Then in Section 3, we present the auxiliary space preconditioners and discuss its convergence. Finally, in Section 4, we present several numerical experiments in both 2D and 3D to verify the theoretical result.

## 2 Virtual Element Methods

Let $\Omega \subset \mathbb{R}^d$ $(d = 2, 3)$ be a bounded open polygonal domain. Given $f \in L^2(\Omega)$, we consider the following model problem: Find $u \in V := H_0^1(\Omega)$ such that

$$a(u, v) := (\kappa \nabla u, \nabla v) = (f, v), \qquad \forall v \in V, \tag{1}$$

where $(\cdot, \cdot)$ denotes the $L^2(\Omega)$ inner product, $\kappa = \kappa(x) \in L^\infty(\Omega)$ is assumed to be piecewise positive constant with respect to the polytopal partition $\mathcal{T}_h$ of $\Omega$ but may have large jumps across the interface of the partition.

Let $\mathcal{T}_h$ be a partition of $\Omega$ into non-overlapping simple polytopal elements $K$. Here we use $h_K$ for the diameter of the element $K \in \mathcal{T}_h$ (the greatest distance between any two vertices of $K$), and define $h = \max_{K \in \mathcal{T}_h} h_K$, the maximum of the diameters. Following [11], we make the following assumption on the polytopal mesh:

(**A**) Each polytopal element $K \in \mathcal{T}_h$ has a triangulation $\mathcal{T}_K$ of $K$ such that $\mathcal{T}_K$ is uniformly shape regular and quasi-uniform. Each edge of $K$ is an edge of certain elements in $\mathcal{T}_K$.

On each polytopal element $K \in \mathcal{T}_h$, we define the local virtual finite element space:

$$V_h^K := \{v \in H^1(K) \ : \ v|_{\partial K} \in \mathbb{B}_1(\partial K), \ \Delta v = 0\},$$

where $\mathbb{B}_1(\partial K) := \{v \in C^0(\partial K) : v|_e \in \mathbb{P}_1(e), \ \forall e \subset \partial K\}$. Note that $V_h^K \supset \mathbb{P}_1(K)$, and may contain implicitly some other non-polynomial functions. The global virtual element space $V_h$ is then defined as:

$$V_h := \{v \in V \ : \ v|_K \in V_h^K, \ \ \forall K \in \mathcal{T}_h\}.$$

The VEM discretization of (1) is given by a symmetric bilinear form $a_h : V_h \times V_h \to \mathbb{R}$ such that

$$a_h(u_h, v_h) = \sum_{K \in \mathcal{T}_h} a_h^K(u_h, v_h), \qquad \forall u_h, v_h \in V_h,$$

where $a_h^K(\cdot,\cdot)$ is a computable bilinear form defined on $V_h^K \times V_h^K$. So the VEM discretization of (1) reads: Find $u_h \in V_h$ such that

$$a_h(u_h, v_h) = (f, v_h), \quad \forall v_h \in V_h. \tag{2}$$

Further details on how to construct the computable bilinear form $a_h$, as well as a study of the convergence and stability properties of the VEM can be found in [2, 4, 5]. We refer to [3, 15] for detailed discussion on the implementation of the methods, and refer to [7, 11] for the error estimates of the methods.

Let $A$ be the operator induced by the bilinear form $a_h(\cdot,\cdot)$, that is,

$$(Av, w) = (v, w)_A := a_h(v, w), \qquad \forall v, w \in V_h.$$

Then solving (2) is equivalent to solving the linear system

$$Au_h = f. \tag{3}$$

It is clear that the operator $A$ is symmetric and positive definite, and we can show that the condition number satisfies $\mathcal{K}(A) \lesssim \mathcal{J}(\kappa)h^{-2}$, where $\mathcal{J}(\kappa) = \max_x \kappa(x)/\min_x \kappa(x)$ is the variation of the discontinuous coefficient (see for example [20, Lemma 2.2]). Thus the resulting linear system of the VEM discretization (2) can be very ill-conditioned with the condition number depending on both the mesh size and the variation in the discontinuous coefficient. It is difficult to solve using the classic iterative methods such as Jacobi, Gauss-Seidel or conjugate gradient method, without effective preconditioners. In the next section, we describe efficient auxiliary space preconditioners for (3) that are robust with respect to the variation in the discontinuous coefficient and the mesh size.

## 3 Auxiliary Space Preconditioners

To solve the discrete system (3) efficiently, we use the auxiliary space preconditioning technique (cf. [17]). For this purpose, we need an "auxiliary space". For each polytopal element $K \in \mathcal{T}_h$, we introduce an auxiliary triangulation $\mathcal{T}_K$ of it such that each edge of $K$ is an edge of some element in this triangulation. By Assumption (**A**), this is possible and can be done using a Delaunay triangulation. With this triangulation, we obtain a conforming quasi-uniform triangulation $\mathcal{T}_h^c := \bigcup_{K \in \mathcal{T}_h} \mathcal{T}_K$ of the whole domain $\Omega$. Let $V_h^c \subset V$ be the standard conforming $\mathbb{P}_1$ finite element space defined on this auxiliary triangulation $\mathcal{T}_h^c$. We introduce the auxiliary problem: find $u_h^c \in V_h^c$ such that

$$a(u_h^c, v_h) = (f, v_h), \qquad \forall v_h \in V_h^c. \tag{4}$$

Similarly, let $A_c$ be the operator induced by the bilinear form $a(\cdot,\cdot)$, that is,

$$(A_c v, w) = (v, w)_{A_c} := a(v, w), \qquad \forall v, w \in V_h^c.$$

The auxiliary space preconditioners can be understood as a two-level algorithm involving a "fine level" and a "coarse level". In this setting, the fine level problem is the VEM discretization (2) on polytopal mesh $\mathcal{T}_h$, and the coarse level problem is the standard conforming $\mathbb{P}_1$ finite element space defined on the auxiliary simplicial mesh (4). Since $A_c$ is the standard conforming piecewise linear finite element discretization of (1) on the auxiliary quasi-uniform triangulation $\mathcal{T}_h^c$, the "coarse" problem in $V_h^c$ can be solved by many existing efficient solvers such as the standard multigrid methods or domain decomposition methods (see, for example [18, 19] and the references cited therein). It can be either an exact solver or an approximate solver. We denote $B_c : V_h^c \to V_h^c$ to be such a "coarse" solver, that is $B_c \approx A_c^{-1}$. Next, on the fine space $V_h$, we define a "smoother" $R: V_h \to V_h$, which is symmetric positive definite. For example, $R$ could be a Jacobi or symmetric Gauss-Seidel smoother. Finally, to connect the "coarse" space $V_h^c$ with the "fine" space $V_h$, we need a "prolongation" operator $\Pi : V_h^c \to V_h$. The restriction operator $\Pi^t : V_h \to V_h^c$ is then defined as

$$(\Pi^t v, w) = (v, \Pi w), \quad \text{for } v \in V_h \text{ and } w \in V_h^c.$$

Note that the auxiliary space defined in this way has a natural intergrid transfer operator because the degrees of freedom for the space $V_h^c$ are included among the degrees of freedom for the space $V_h$. Thus for each $v \in V_h$, we can define $\Pi^t v = v^c \in V_h^c$ such that $v^c(z_i) = v(z_i)$ for each vertex $z_i$ in the element $K \in \mathcal{T}_h$. We can view this as a linear interpolation of $v$ onto $V_h^c$. Then, the auxiliary space preconditioner $B: V_h \to V_h$ can be chosen as

$$\text{Additive} \qquad B_{\text{add}} = R + \Pi B_c \Pi^t, \tag{5}$$

$$\text{Multiplicative} \qquad I - B_{\text{mul}} A = (I - RA)(I - \Pi B_c \Pi^t)(I - RA). \tag{6}$$

For these preconditioners, we have the following theorem.

**Theorem 1** *The auxiliary space preconditioner $B = B_{\text{add}}$ defined by* (5) *or $B = B_{\text{mul}}$ defined by* (6) *satisfies:*

$$\mathcal{K}(BA) \le C,$$

*where the constant $C > 0$ depends only on the shape-regularity of the auxiliary triangulation, and is independent of the mesh size $h$ and the coefficients $\kappa$.*

The analysis is based on the auxiliary space framework [17], with some technical error estimates from [11]. Due to the page limitation, we refer to [20] for more detailed analysis and discussion.

*Remark 1* In the auxiliary space preconditioners defined in (5) and (6), if we ignore the smoother $R$, the resulting preconditioner is usually called the *fictitious space preconditioner* ((cf. [14]). In this case, we denote $B_{\text{fict}} := \Pi B_c \Pi^t$. In fact, the auxiliary space preconditioners can be viewed as a generalization of the fictitious space preconditioner by a special choice of the "fictitious space". In particular, the fictitious space is defined as the product space $V_h \times V_h^c$. Including $V_h$ as a component of the fictitious space makes it easier to construct the map from the fictitious space to the original space, which is required to be surjective. For example, there is no surjective mapping from the linear FEM space $V_h^c$ to higher order VEM space. In this case the smoother $R$ will play an important role in the auxiliary space preconditioners.

On the other hand, note that the operator $\Pi$ defined above is surjective for linear VEM discretization. If the mesh satisfies Assumption (A), one can show that the fictitious space preconditioner is also robust with respect to the problem size and the discontinuous coefficients. We refer to [20] for more detailed discussion. However, our numerical experiments indicate that $B_{\text{fict}}$ is more sensitive to the shape-regularity of the auxiliary triangulation, while $B_{\text{add}}$ and $B_{\text{mul}}$ are more stable with respect to the mesh quality.

## 4 Numerical Experiments

In this section, we present some numerical experiments in both 2D and 3D to verify the result in Theorem 1. In all these tests, we use 2-sweeps symmetric Gauss-Seidel smoother. The stopping criteria is $\|r_k\|/\|r_0\| < 10^{-12}$ for the PCG algorithm, where $r_k = f - Au_k$ is the residual. For the coarse solver, we use the AMG algorithm implemented in *i*FEM [10].

In the first example, we consider the model problem (1) in the unit square $\Omega = [0, 1]^2$ with constant coefficient $\kappa = 1$. Figure 1 is an example of the polytopal mesh of the unit square domain (with 100 elements) generated using PolyMesher [16], and Figure 2 is the corresponding Delaunay triangular mesh. The VEM discretization is defined on the polytopal mesh (cf. Figure 1), while the auxiliary space is the standard conforming $\mathbb{P}_1$ finite element discretization defined on the corresponding triangular mesh (cf. Figure 2).

Tables 1 shows the estimated condition number and the number of PCG iteration in parenthesis for the un-preconditioned and preconditioned systems with various preconditioners. Here and in the sequel, $B_{\text{sgs}}$ is the (2-sweep) symmetric Gauss-Seidel preconditioner; $B_{\text{fict}}$ is the fictitious space preconditioner defined in Remark 1; $B_{\text{add}}$ is the additive auxiliary space preconditioner defined in (5); and $B_{\text{mul}}$ is the multiplicative auxiliary space preconditioner defined in (6). As we can observe from

**Table 1:** Estimated condition number (number of PCG iteration) in 2D with constant coefficients.

| # Polytopal Elements | 10 | $10^2$ | $10^3$ | $10^4$ | $10^5$ |
|---|---|---|---|---|---|
| $\mathcal{K}(A)$ | 3.45 (9) | 3.86e01 (41) | 3.80e02 (117) | 3.88e03 (351) | 4.07e04 (1100) |
| $\mathcal{K}(B_{\text{sgs}}A)$ | 1.07(6) | 3.78 (15) | 3.20e01 (37) | 3.17e02 (104) | 3.17e03 (318) |
| $\mathcal{K}(B_{\text{fict}}A)$ | 2.92 (8) | 5.75 (26) | 7.53 (29) | 8.73 (32) | 9.67(36) |
| $\mathcal{K}(B_{\text{add}}A)$ | 1.53 (9) | 1.71 (14) | 1.94 (14) | 1.99 (14) | 2.00 (13) |
| $\mathcal{K}(B_{\text{mul}}A)$ | 1.06 (8) | 1.21 (10) | 1.04 (7) | 1.02 (6) | 1.02 (6) |

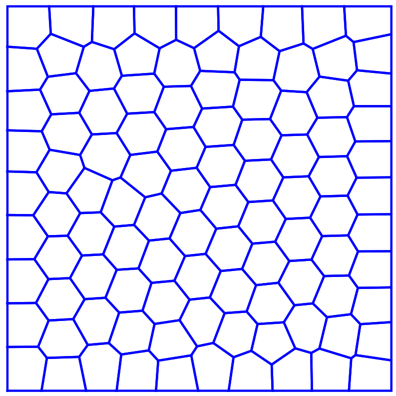

**Fig. 1:** Polygonal Mesh $\mathcal{T}_h$ of the Unit Square Domain (100 Elements)

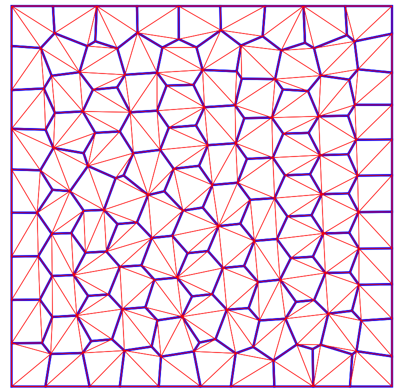

**Fig. 2:** The Corresponding Delaunay Triangle Mesh $\mathcal{T}_h^c$

this table, while the condition numbers $\mathcal{K}(A)$ and $\mathcal{K}(B_{\rm sgs}A)$ increase as the mesh refined, the condition numbers $\mathcal{K}(B_{\rm fict}A)$, $\mathcal{K}(B_{\rm add}A)$ and $\mathcal{K}(B_{\rm mul}A)$ are uniformly bounded.

In the second test, we consider the problem with discontinuous coefficients. The coefficient $\kappa$ is generated randomly on each polygon element (see Figure 3 for an example of the coefficient distribution with 100 elements). Note that the

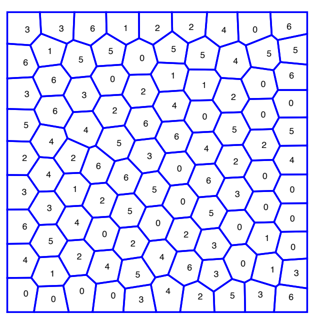

**Fig. 3:** Random Discontinuous Coefficients $10^k$ (100 Elements)

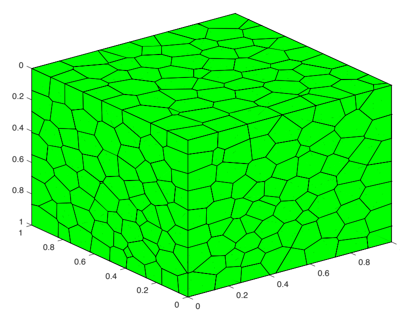


**Fig. 4:** Polyhedral mesh generated by CVT ($9^3$ Elements)

coefficient settings are different in different polytopal mesh. Tables 2 shows the estimated condition number and the number of PCG iteration in parenthesis. Here - denotes that the PCG algorithm fail to converge after 1200 iterations. As we can see from this table, while $\mathcal{K}(A)$ and $\mathcal{K}(B_{\rm sgs}A)$ increase dramatically, the condition numbers $\mathcal{K}(B_{\rm fict}A)$, $\mathcal{K}(B_{\rm add}A)$ and $\mathcal{K}(B_{\rm mul}A)$ are nearly uniformly bounded. These observations verify the conclusions given in Theorem 1 and Remark 1.

Finally, we consider the model problem on a 3D cubic domain $\Omega = [0, 1]^3$. We create a polyhedral mesh using Centroidal Voronoi tessellations (CVT, cf.[13]), see Fig 4 for an example. The VEM discretization is defined on the polyhedral mesh.

**Table 2:** Estimated condition number (number of PCG iteration) in 2D with discontinuous coefficients.

| # Polytopal Elements | 10 | $10^2$ | $10^3$ | $10^4$ | $10^5$ |
|---|---|---|---|---|---|
| $\mathcal{K}(A)$ | 2.44 (11) | 2.73e06 (578) | - | - | - |
| $\mathcal{K}(B_{\text{sgs}}A)$ | 1.18(5) | 3.90e02 (26) | 3.93e03 (409) | - | - |
| $\mathcal{K}(B_{\text{fict}}A)$ | 3.27 (8) | 6.94 (33) | 6.42 (36) | 11.6 (44) | 13.6 (53) |
| $\mathcal{K}(B_{\text{add}}A)$ | 1.54 (9) | 3.51 (20) | 3.60 (25) | 3.67 (25) | 3.80 (26) |
| $\mathcal{K}(B_{\text{mul}}A)$ | 1.06 (6) | 1.74 (15) | 1.82 (16) | 1.84 (16) | 1.88 (17) |

Then we subdivide each polyhedron into tetrahedrons using Delaunay triangulation to define the $\mathbb{P}_1$ conforming finite element discretization on this auxiliary mesh.

**Table 3:** Estimated condition number (number of PCG iteration) in 3D with $\kappa \equiv 1$.

| PolyElem | $3^3$ | $9^3$ | $15^3$ | $21^3$ | $27^3$ |
|---|---|---|---|---|---|
| TetQuality | 7.08e-06 | 4.09e-08 | 1.11e-09 | 2.98e-11 | 1.28e-11 |
| $\mathcal{K}(A)$ | 7.91 (25) | 5.94e+01 (60) | 1.35e+02 (77) | 3.20e+02 (104) | 6.66e+02 (139) |
| $\mathcal{K}(B_{\text{add}}A)$ | 2.36 (16) | 4.29 (21) | 3.38 (21) | 4.35 (24) | 5.36 (27) |
| $\mathcal{K}(B_{\text{mul}}A)$ | 1.00 (5) | 1.10 (8) | 1.13 (8) | 1.14 (8) | 1.18 (9) |

Table 3 shows the performance of the $B_{\text{add}}$ and $B_{\text{mul}}$. We do not present $B_{\text{fict}}$ here because the PCG algorithm does not converge within 200 iterations. To understand the reason, we have calculated the mesh quality of the auxiliary triangulation. Here the TetQuality is the minimum value: $\min_T \frac{r_i}{r_c}$ for all tetrahedral elements $T$, where $r_i$ and $r_c$ are the radii of the inscribed and circumscribed spheres of $T$, respectively. From this table, we notice that both $B_{\text{add}}$ and $B_{\text{mul}}$ are still robust, even in the case of poor TetQuality (which violates Assumption (A)). On the other hand, $B_{\text{fict}}$ is sensitive to the shape-regularity of the auxiliary tetrahedral mesh.

**Acknowledgements** This work was partially supported by NSF DMS-1319110. The author would also like to thank the anonymous referee for carefully proofreading this manuscript and his suggestions, which greatly improved the presentation of this paper.

## References

1. P. F. Antonietti, L. Mascotto, and M. Verani. A multigrid algorithm for the $p$-version of the virtual element method. *arXiv preprint arXiv:1703.02285*, 2017.
2. L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. D. Marini, and A. Russo. Basic principles of virtual element methods. *Math. Models Methods Appl. Sci.*, 23(01):199–214, 2013.
3. L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo. The hitchhiker's guide to the virtual element method. *Math. Models Methods Appl. Sci.*, 24(08):1541–1573, 2014.
4. L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo. Virtual element method for general second-order elliptic problems on polygonal meshes. *Math. Models Methods Appl. Sci.*, 26(04):729–750, 2016.

5. L. Beirão da Veiga, C. Lovadina, and A. Russo. Stability analysis for the virtual element method. *Math. Models Methods Appl. Sci.*, 27(13):2557–2594, 2017.
6. S. Bertoluzza, M. Pennacchio, and D. Prada. BDDC and FETI-DP for the virtual element method. *Calcolo*, 54(4):1565–1593, Dec 2017.
7. S. C. Brenner, Q. Guan, and L.-Y. Sung. Some estimates for virtual element methods. *Comput. Methods Appl. Math.*, 17(4):553–574, 2017.
8. J. G. Calvo. On the approximation of a virtual coarse space for domain decomposition methods in two dimensions. *Math. Models Methods Appl. Sci.*, 28(07):1267–1289, Mar. 2018.
9. J. G. Calvo. An overlapping Schwarz method for virtual element discretizations in two dimensions. *Comput. Math. Appl.*, Nov. 2018.
10. L. Chen. *i*FEM: an integrate finite element methods package in MATLAB. Technical report, University of California at Irvine, 2009.
11. L. Chen and J. Huang. Some error analysis on virtual element methods. *Calcolo*, 55(1):5, Feb 2018.
12. L. Chen, H. Wei, and M. Wen. An interface-fitted mesh generator and virtual element methods for elliptic interface problems. *J. Comput. Phys.*, 334:327 – 348, 2017.
13. Q. Du, V. Faber, and M. Gunzburger. Centroidal voronoi tessellations: Applications and algorithms. *SIAM Rev.*, 41(4):637–676, 1999.
14. S. V. Nepomnyaschikh. Decomposition and fictitious domains methods for elliptic boundary value problems. In *Fifth International Symposium on Domain Decomposition Methods for Partial Differential Equations (Norfolk, VA, 1991)*, pages 62–72. SIAM, Philadelphia, PA, 1992.
15. O. J. Sutton. The virtual element method in 50 lines of MATLAB. *Numer. Algorithms*, 75(4):1141–1159, Aug 2017.
16. C. Talischi, G. H. Paulino, A. Pereira, and I. F. M. Menezes. PolyMesher: a general-purpose mesh generator for polygonal elements written in Matlab. *Struct. Multidiscip. Optim.*, 45(3):309–328, Mar 2012.
17. J. Xu. The auxiliary space method and optimal multigrid preconditioning techniques for unstructured meshes. *Computing*, 56:215–235, 1996.
18. J. Xu and Y. Zhu. Uniform convergent multigrid methods for elliptic problems with strongly discontinuous coefficients. *Math. Models Methods Appl. Sci.*, 18(1):77 –105, 2008.
19. Y. Zhu. Domain decomposition preconditioners for elliptic equations with jump coefficients. *Numer. Linear Algebra Appl.*, 15(2-3):271–289, 2008.
20. Y. Zhu. Auxiliary space preconditioners for virtual element methods. *Submitted*, 2018. Available as *http://arxiv.org/abs/1812.04423*